Spatial Intelligence

Foundations, Architectures, Methodologies, Applications and Challenges

A Handbook

Explore the project: https://spatial-agi.github.io

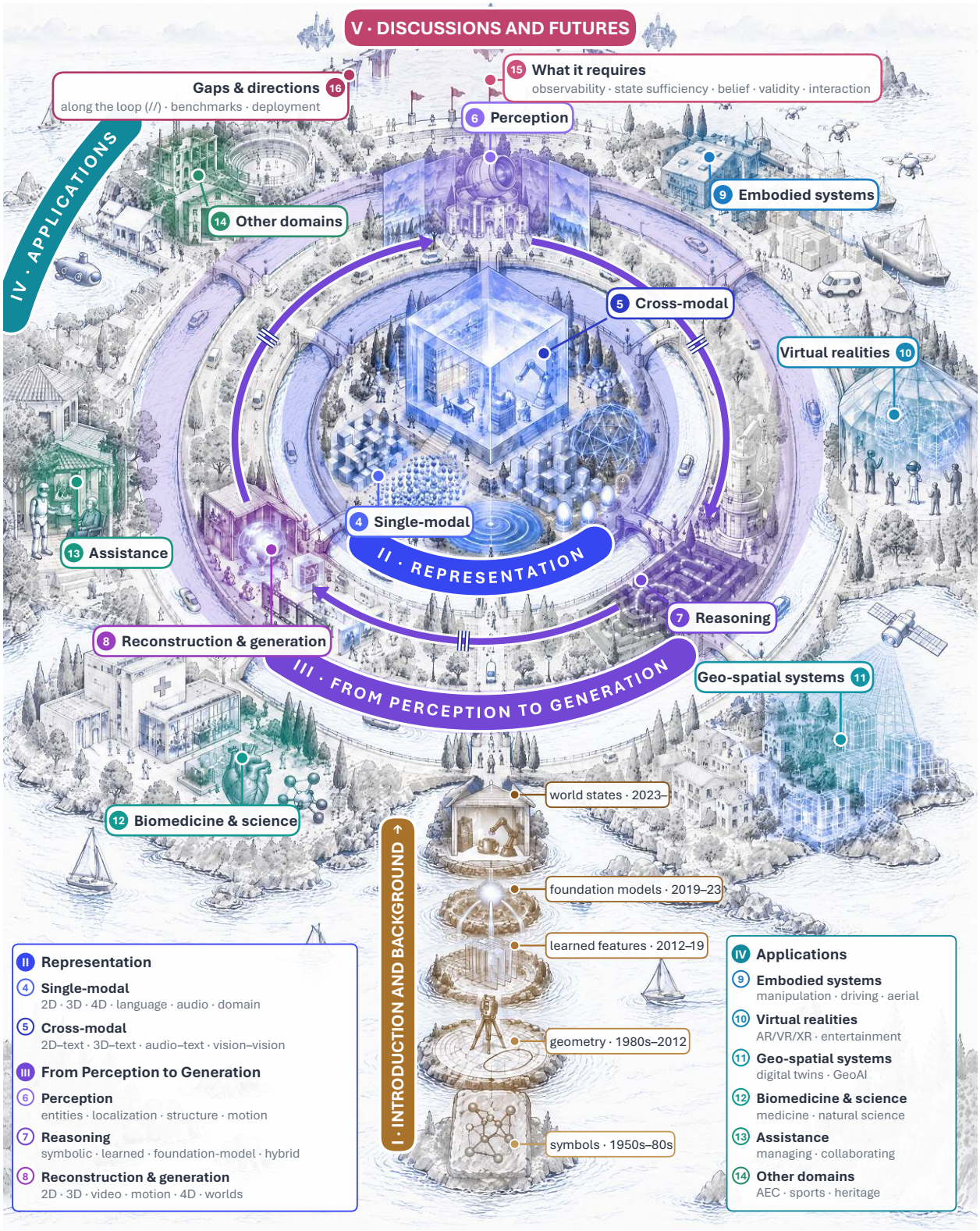
Shengqiong Wu[1,†], **Yu Zhao**[2], **Tianjie Ju**[2], **Jundong Xu**[2], **Bobo Li**[2], **Jingkang Yang**[3], **Shulin Tian**[3], **Daoan Zhang**[4], **Yanguang Zhao**[2], **Xuan Ju**[5], **Haoyu Zhao**[5], **Yuchen Zhou**[2], **Xiangtai Li**[3], **Zhedong Zheng**[6], **Joy Hsu**[7], **Xin Kong**[8], **Ruining Li**[1], **Cheng Zhang**[9], **Chengzu Li**[10], **Jiafei Duan**[2], **Yu (Bryan) Zhou**[11], **Kenneth Li**[12], **Chuanxia Zheng**[3], **Lei Cui**[13], **Furu Wei**[13], **Ziwei Liu**[3], **Steven K. Feiner**[14], **Jiebo Luo**[4], **Jianfei Cai**[9], **Shuicheng Yan**[2], **Ming-Hsuan Yang**[15], **Hao Fei**[1,*]

[1]University of Oxford, [2]National University of Singapore, [3]Nanyang Technological University
[4]University of Rochester, [5]The Chinese University of Hong Kong, [6]University of Macau
[7]Stanford University, [8]Imperial College London, [9]Monash University, [10]University of Cambridge
[11]University of California, Los Angeles, [12]Harvard University, [13]Microsoft Research
[14]Columbia University, [15]University of California, Merced

Abstract

The pursuit of spatial intelligence, i.e., enabling agents to perceive, represent, reason about, and act within complex environments, stands as a foundational pillar in the journey toward Artificial General Intelligence (AGI). From embodied perception to multimodal understanding and interactive world modeling, spatial intelligence spans a broad spectrum of research domains and practical applications. This handbook offers a systematic and forward-looking overview of spatial intelligence, bridging insights across computer vision, natural language processing, 3D/4D modeling, robotics, and cognitive reasoning. We organize our exploration into five parts. Part I motivates why spatial intelligence matters now, traces how the representation of space has evolved, and gives a working definition. Part II lays the groundwork in *Spatial-aware Representation*, where we study representations across single modalities (2D, 3D, 4D, language, audio, etc.) and their cross-modal alignment. Part III follows the full pipeline *From Perception to Generation*, including perception (e.g., detection, segmentation), reasoning (e.g., symbolic, geometric, LLM-driven), and generation (e.g., scene, video, world synthesis). Part IV turns to *Applications*, spanning embodied agents, virtual/augmented realities, geo-spatial systems, biomedical domains, and intelligent assistants. Finally, Part V, *Discussions and Futures*, examines what spatial intelligence requires and discusses research gaps, benchmarking, and directions for real-world deployment. By synthesizing principles from AI, cognitive science, and spatial computing, this handbook aims to serve as both a comprehensive reference for researchers and a practical guide for system developers. We hope it will inspire innovative directions toward building agents and systems that are spatially aware, robustly generalizable, and deeply grounded in real-world interactions.

[†]Contact: shengqiongwu@gmail.com.
[*]Hao Fei leads the project. Contact: haofei7419@gmail.com.



The handbook as one world. The map is laid out like the book: representation (Part II) at the center, the loop from perception through reasoning to generation (Part III) on the inner ring, and the application domains (Part IV) along the coast. The stepping stones below trace the five eras of Part I, the horizon above holds the open questions of Part V, and each chapter is pinned where it lives in this world.

Preface

1902 · MATHEMATICS

"To localise an object simply means that we represent to ourselves the movements that must take place to reach it."

Henri Poincaré
Science and Hypothesis

1983 · PSYCHOLOGY

"The capacities to perceive the visual world accurately, to perform transformations and modifications upon one's initial perceptions, and to be able to re-create aspects of one's visual experience, even in the absence of relevant physical stimuli."

Howard Gardner
Frames of Mind

2025 · ARTIFICIAL INTELLIGENCE

"Spatial intelligence is the scaffolding upon which our cognition is built."

Fei-Fei Li
From Words to Worlds

Spatial intelligence lies at the core of human cognition. We perceive our surroundings, remember places, plan routes, manipulate objects, and coordinate with one another in shared physical and virtual spaces, and we do all of this by building and continuously revising internal models of the world. Machines are only now acquiring the rudiments of this ability. Today's AI systems write fluent prose and reason impressively over text, yet they still struggle to judge whether a sofa will fit through a doorway, to keep a scene coherent while moving through it, or to anticipate what happens when an object is pushed. That gap has become one of the central open problems of contemporary AI. Advances in vision-language models, 3D and 4D perception, generative modeling, and interactive world models are now converging from different directions on the same goal: machines that perceive, reason, and act in space. This handbook is our attempt to give that convergence a coherent shape.

The immediate motivation is a long-standing fragmentation. Research on space in AI has been carried out in largely separate communities, spanning SLAM and state estimation, 3D vision and neural rendering, spatial language grounding, geographic information science, embodied navigation and manipulation, and the psychology and neuroscience of spatial cognition, each with its own representations, benchmarks, and vocabulary. Progress in one thread has rarely been legible to the others. We set out to organize the field as a whole. The organizing commitment of this handbook is *representation-centric*: spatial intelligence is the ability of a system to construct, maintain, and manipulate spatial representations of the world in support of perception, reasoning, prediction, and embodied action. Reading the field through this single lens, from the symbolic maps of early AI, through probabilistic geometry and learned features, to today's foundation models and executable world states, is what allows historically disjoint literatures to be placed on common ground.

A handful of questions shaped every chapter. What makes a system spatially intelligent, beyond performing well on any particular task? How should geometric, semantic, and commonsense structure be integrated within one representation, and aligned across modalities as different as pixels, point clouds, language, and audio? When does spatial knowledge learned in one environment transfer to another? How do planning and

action emerge from grounded spatial understanding, and how should the field measure genuine progress rather than benchmark saturation? None of them has a final answer yet. Each now admits systematic treatment, and the handbook is organized to provide it.

The volume proceeds in five parts. Part I lays the foundation: why spatial intelligence has risen to prominence now, how the idea evolved across five representational eras from symbolic spatial computing to world-centered intelligence, and how we define and formalize the concept itself. Part II examines spatial-aware representation, covering single modalities (2D, 3D, 4D, language, and audio), domain-specific representations, and the cross-modal alignment and fusion that bind them. Part III follows the full functional pipeline from perception, through reasoning in its rule-based, learning-based, foundation-model-driven, and hybrid forms, to spatial reconstruction and generation. Part IV turns to applications, where these capabilities are assembled into working systems: embodied agents and robots, virtual and extended realities, geo-spatial systems, biomedicine and the natural sciences, intelligent assistants, and a range of further domains. Part V closes with open discussions, unresolved challenges, and the directions we believe will matter most in the coming years. Chapter 3 offers tailored reading paths through this material for students, researchers, engineers, and cross-disciplinary explorers.

We wrote this handbook for several audiences at once: researchers and graduate students in computer vision, robotics, and multimodal learning; engineers building embodied agents, simulation platforms, and spatial computing systems; and scholars in cognitive science, neuroscience, and geography who want to see how their fields' insights are being absorbed into AI. Wherever possible we favor conceptual clarity over exhaustive enumeration, and we connect each technical thread back to the shared framework so that the whole remains visible through the parts.

A handbook about a fast-moving field is necessarily a snapshot. Some of the systems discussed here will be surpassed before long, and we regard that as a feature of the field: the framework is designed to outlast its examples. We call Chapter 2 "A Definition in Motion" deliberately. Spatial intelligence is still being defined by the community that builds it, and our hope is that this volume gives that community a common map, drawn honestly, mid-journey.

Contents

I Introduction and Background **22**

1 Spatial Intelligence: A Rising AI Paradigm **24**

1.1 Why Spatial Intelligence? Why Now? . . . 24

1.1.1 AI Paradigm Shift . . . 25

1.1.2 AI Methodological Shift . . . 26

1.1.3 Modality Shift . . . 27

1.1.4 Disciplinary Convergence . . . 28

1.2 Historical Evolution and Milestones . . . 28

1.2.1 Era 1: Symbolic and Cognitive Spatial Computing (1950s–1980s) . . . 29

1.2.2 Era 2: Geometric and Probabilistic Spatial Modeling (1980s–2012) . . . 32

1.2.3 Era 3: Learning-based Spatial Modeling (2012–2019) . . . 33

1.2.4 Era 4: Large Foundation Spatial Modeling (2019–2023) . . . 35

1.2.5 Era 5: World-Centered Spatial Intelligence (2023–Present) . . . 37

2 A Definition in Motion: Revisiting Spatial Intelligence **40**

2.1 The Landscape of Spatial Intelligence . . . 40

2.1.1 Beyond A Single Dimension . . . 40

2.1.2 Disciplinary Perspectives . . . 42

2.1.3 Toward A Unified View . . . 45

2.2 Defining A Spatial Intelligence Framework . . . 46

2.2.1 Core Elements . . . 47

2.2.2 Formal Characterization . . . 48

2.2.3 Compatibility with Existing Research and Paradigms . . . 50

3 Scope and Structure of This Handbook **51**

3.1 The Research Pillars of Spatial Intelligence . . . 51

3.2 Goal of the Handbook: Charting a Research Ecosystem . . . 52

3.3 How to Read This Handbook . . . 53

II Spatial-aware Representation: Building Blocks of Space 54

4 Single-Modal Spatial Representation 56

4.1 2D . . . 56
4.1.1 Image and Video Representations . . . 56
4.1.2 Segmentation Maps . . . 57
4.1.3 Depth Maps . . . 58
4.1.4 Surface Normal Maps . . . 58
4.1.5 Edge Maps . . . 59
4.1.6 Optical Flow Maps . . . 59
4.1.7 2D Keypoints . . . 59
4.2 3D . . . 60
4.2.1 RGB-D Images . . . 61
4.2.2 Voxel Grids . . . 62
4.2.3 Point Clouds . . . 62
4.2.4 Meshes . . . 62
4.2.5 Implicit Representations . . . 63
4.2.6 Neural Radiance Fields . . . 63
4.2.7 3D Gaussian Splatting . . . 64
4.2.8 Hybrid Methods . . . 65
4.3 4D . . . 65
4.3.1 RGBD Video . . . 65
4.3.2 Point Cloud . . . 67
4.3.3 Mesh . . . 67
4.3.4 NeRF . . . 67
4.3.5 Deformable 3D Gaussian Splatting . . . 67
4.4 Language . . . 68
4.4.1 The Evolution of Language Representation Learning . . . 68
4.4.2 Language Representation Learning Induces Spatial Intelligence . . . 69
4.5 Audio . . . 69
4.6 Domain-Specific Representations . . . 70

5 Cross-Modal Spatial Representation 73

5.1 Overview . . . 73
5.2 2D-text . . . 74
5.3 3D-text . . . 75

5.4 Audio-text . . . 76
5.5 Vision-vision . . . 77

III Spatial Intelligence: From Perception to Generation 78

6 Spatial Perception 80
6.1 Spatial Entity Recognition . . . 80
6.1.1 Object Detection . . . 80
6.1.2 Segmentation . . . 84
6.1.3 Keypoint and Landmark Detection . . . 89
6.1.4 Scene Classification . . . 93
6.2 Spatial Localization and Position Estimation . . . 96
6.2.1 Depth Estimation . . . 96
6.2.2 Surface Normal Estimation . . . 101
6.2.3 Height and Distance Estimation . . . 105
6.2.4 Visual Odometry and Camera Pose Estimation . . . 110
6.2.5 Pose Estimation . . . 114
6.3 Spatial Structure Construction . . . 119
6.3.1 Spatial Relation Extraction . . . 120
6.3.2 Scene Graph Generation . . . 122
6.3.3 Layout Estimation and Room Parsing . . . 126
6.3.4 Action Detection . . . 130
6.3.5 Mapping and SLAM . . . 134
6.4 Spatio-Temporal Perception . . . 137
6.4.1 Scene Flow Estimation . . . 137
6.4.2 Object Tracking . . . 141
6.4.3 Temporal Localization . . . 146
6.5 Multimodal Spatial Perception . . . 150
6.5.1 Audio-Visual Spatial Localization . . . 150
6.5.2 Vision-Language Grounding . . . 154

7 Spatial Reasoning 160
7.1 Rule-based Symbolic Spatial Reasoning . . . 160
7.1.1 Geometric and Algebraic Reasoning . . . 161
7.1.2 Physical Reasoning . . . 163

7.1.3 Topological and Graph-based Reasoning . . . 166
7.1.4 Constraint-based Spatial Solving . . . 168
7.1.5 Rule-based and Programmatic Reasoning . . . 171
7.1.6 Causal Reasoning . . . 174
7.2 Learning-based Spatial Reasoning . . . 176
7.2.1 Implicit Neural Spatial Reasoning . . . 177
7.2.2 Explicit Structured Neural Reasoning . . . 179
7.2.3 Geometry-aware Neural Spatial Reasoning . . . 183
7.2.4 Differentiable Optimization Layers for Spatial Reasoning . . . 187
7.2.5 Reinforcement Learning with Spatial Inductive Biases . . . 190
7.2.6 Multi-agent Spatial Reasoning . . . 195
7.3 Foundation Model-driven Spatial Reasoning . . . 199
7.3.1 LLM/VLM-centric Spatial Reasoning . . . 200
7.3.2 VLA-based Spatial Reasoning . . . 202
7.3.3 Chain-of-Thought for Spatial Reasoning . . . 204
7.3.4 Code/DSL Generation for Spatial Reasoning . . . 207
7.3.5 Tool-augmented Reasoning . . . 209
7.3.6 Retrieval- & Memory-augmented Spatial Reasoning . . . 212
7.3.7 Agentic Spatial Reasoning . . . 215
7.3.8 World-Model-based Spatial Reasoning . . . 219
7.4 Hybrid Spatial Reasoning . . . 222
7.4.1 Task Definition . . . 222
7.4.2 Methodology . . . 223
7.4.3 Benchmarks . . . 225
7.4.4 Challenges and Trends . . . 226
7.5 Spatial Reasoning Benchmarks . . . 226
7.5.1 Comparative Analysis . . . 227
7.5.2 Future Directions . . . 227

8 Spatial Reconstruction and Generation 229
8.1 2D Spatial Generation . . . 229
8.1.1 Text-to-image Generation . . . 230
8.1.2 2D Spatial-aware Image Generation . . . 235
8.1.3 2D Spatial Inpainting & Editing . . . 239
8.2 3D Spatial Reconstruction and Generation . . . 244

8.2.1 3D Object Synthesis . . . 245
8.2.2 3D Scene Synthesis . . . 251
8.2.3 3D Scene Reconstruction . . . 255
8.2.4 3D Panorama Synthesis . . . 260
8.2.5 3D Scene Editing . . . 268
8.3 Video Spatial Generation . . . 273
8.3.1 Text-to-Video Generation . . . 274
8.3.2 Spatial Controlled Video Generation . . . 278
8.3.3 Temporal Controlled Video Generation . . . 281
8.3.4 Other Controlled Video Generation . . . 284
8.3.5 Video Editing . . . 286
8.3.6 Summary and Critical Challenges and Future Directions . . . 289
8.4 Motion Generation . . . 291
8.4.1 Human Motion Generation . . . 291
8.4.2 Animal Motion Generation . . . 297
8.4.3 Vehicle Motion Planning . . . 299
8.4.4 Motion Editing . . . 301
8.5 4D Spatial Generation . . . 303
8.5.1 Text-to-4D Generation . . . 304
8.5.2 Image-to-4D Generation . . . 307
8.5.3 Video-to-4D Generation . . . 308
8.5.4 3D-to-4D Generation . . . 309
8.5.5 Multi-conditional 4D Generation . . . 310
8.5.6 Summary, Overarching Challenges and Future Directions . . . 311
8.6 World Modeling . . . 312
8.6.1 Representation World Model . . . 313
8.6.2 Generative World Model . . . 315
8.6.3 Mental World Modeling . . . 318
8.6.4 Discussions and Challenges . . . 321
8.7 Programmatic and Symbolic Spatial Generation . . . 322
8.7.1 Task Definition . . . 322
8.7.2 Methodology . . . 323
8.7.3 Benchmarks . . . 324
8.7.4 Challenges and Trends . . . 325
8.8 Summary and Outlook . . . 325

IV Applications **327**

9 Spatial Intelligence for Embodied Systems **329**

9.1 Robotic Manipulation . . . 329
9.1.1 Introduction . . . 329
9.1.2 System Overview . . . 330
9.1.3 Pivotal Task-1: Grounding Action in Spatial Anchors . . . 330
9.1.4 Pivotal Task-2: Inferring Functional Action Possibilities . . . 331
9.1.5 Pivotal Task-3: Encoding Spatial Reasoning in Generalist Models . . . 333
9.1.6 Pivotal Task-4: Satisfying Geometric and Logical Constraints . . . 335
9.1.7 Established Tools and Systems . . . 336
9.1.8 Challenges . . . 337
9.2 Autonomous Driving . . . 339
9.2.1 Introduction . . . 339
9.2.2 System Overview . . . 340
9.2.3 Pivotal Task-1: Perception and Detection . . . 340
9.2.4 Pivotal Task-2: Localization and Mapping . . . 342
9.2.5 Pivotal Task-3: Decision Making and Control . . . 344
9.2.6 Established Tools and Systems . . . 345
9.2.7 Challenges . . . 347
9.3 Wearable AI System . . . 348
9.3.1 Introduction . . . 348
9.3.2 Wearable Devices . . . 348
9.3.3 System Overview . . . 349
9.3.4 Pivotal Task-1: Embodied Motion Sensing and Pose Reconstruction . . . 350
9.3.5 Pivotal Task-2: Behavioral Understanding and Intention Modeling . . . 352
9.3.6 Pivotal Task-3: Safe Collaboration and Embodied Interaction . . . 354
9.3.7 Established Tools and Systems . . . 355
9.3.8 Challenges . . . 357
9.4 Aerial Agents . . . 358
9.4.1 Introduction . . . 358
9.4.2 System Overview . . . 359
9.4.3 Pivotal Task-1: Ground-Air Collaborative Perception and Visual Geolocalization . . . 359
9.4.4 Pivotal Task-2: Remote Sensing Detection and Environmental Understanding . . . 362
9.4.5 Pivotal Task-3: Autonomous Aerial Navigation and Decision Making . . . 365

9.4.6 Established Tools and Systems . . . 366
9.4.7 Challenges . . . 367
9.5 Warehouse and Industrial Logistics . . . 368
9.5.1 Introduction . . . 368
9.5.2 System Overview . . . 369
9.5.3 Pivotal Task-1: Perception and Recognition . . . 369
9.5.4 Pivotal Task-2: Localization and Mapping . . . 369
9.5.5 Pivotal Task-3: Reasoning and Planning . . . 371
9.5.6 Pivotal Task-4: Manipulation and Control . . . 373
9.5.7 Pivotal Task-5: Human-Robot Collaboration and Safety . . . 373
9.5.8 Established Tools and Systems . . . 374
9.5.9 Challenges . . . 376

10 Spatial Intelligence for Virtual Realities 378
10.1 AR/VR/XR . . . 378
10.1.1 Introduction . . . 378
10.1.2 System Overview . . . 379
10.1.3 Pivotal Task-1: Spatial Tracking, Mapping, and Registration . . . 380
10.1.4 Pivotal Task-2: Egocentric Human Sensing and Spatial Interaction . . . 381
10.1.5 Pivotal Task-3: Spatial Display and Real-Time Rendering . . . 382
10.1.6 Established Tools and Systems . . . 384
10.1.7 Challenges . . . 385
10.2 Entertainment . . . 386
10.2.1 Introduction . . . 386
10.2.2 System Overview . . . 386
10.2.3 Pivotal Task-1: World Reconstruction and Spatial Mapping . . . 387
10.2.4 Pivotal Task-2: Embodied Performance Capture and Avatar Control . . . 388
10.2.5 Pivotal Task-3: Spatially-Aware Content Authoring and Virtual Production . . . 388
10.2.6 Established Tools and Systems . . . 389
10.2.7 Challenges . . . 391

11 Spatial Intelligence for Geo-spatial Systems 392
11.1 Digital Twins . . . 392
11.1.1 Introduction . . . 392
11.1.2 System Overview . . . 393
11.1.3 Pivotal Task-1: High-Fidelity Twin Construction . . . 393

11.1.4 Pivotal Task-2: Physics-Based Modeling and Behavioral Simulation 395
11.1.5 Pivotal Task-3: Real-Time Synchronization and Predictive Operation 397
11.1.6 Established Tools and Systems 398
11.1.7 Challenges 399
11.2 GeoAI 400
11.2.1 Introduction 400
11.2.2 System Overview 400
11.2.3 Pivotal Task-1: Precision Agriculture 401
11.2.4 Pivotal Task-2: Urban Planning, Logistics, and Transport 402
11.2.5 Pivotal Task-3: Environmental Monitoring 403
11.2.6 Pivotal Task-4: Water Resources Management and Precipitation Forecasting 404
11.2.7 Pivotal Task-5: Disaster Management 405
11.2.8 Pivotal Task-6: Healthcare 407
11.2.9 Established Tools and Systems 408
11.2.10 Toward Geospatial Foundation Models and GeoLLMs 408
11.2.11 Challenges 409
11.3 Aerospace Engineering 410
11.3.1 Introduction 410
11.3.2 System Overview 410
11.3.3 Pivotal Task-1: Satellite Pose Estimation 410
11.3.4 Pivotal Task-2: Spacecraft Detection, Segmentation, and Parts Recognition 412
11.3.5 Pivotal Task-3: Operator Agents 413
11.3.6 Pivotal Task-4: Planetary Robotics 414
11.3.7 Established Tools and Systems 415
11.3.8 Challenges 416

12 Spatial Intelligence for Biomedicine and Natural Sciences 417

12.1 Medicine and Clinical Healthcare 417
12.1.1 Introduction 417
12.1.2 System Overview 417
12.1.3 Pivotal Task-1: Volumetric Medical Image Perception 418
12.1.4 Pivotal Task-2: Anatomical Reconstruction, Modeling, and Surgical Planning 420
12.1.5 Pivotal Task-3: Image-Guided Intervention and Surgical Navigation 421
12.1.6 Established Tools and Systems 423
12.1.7 Challenges 423

12.2 Natural Science . . . 425
12.2.1 Introduction . . . 425
12.2.2 Physics . . . 425
12.2.3 Chemistry . . . 428
12.2.4 Life Science . . . 430
12.2.5 Challenges . . . 432

13 Spatial Intelligence for Intelligent Assistance 434
13.1 Assistant . . . 434
13.1.1 Introduction . . . 434
13.1.2 System Overview . . . 435
13.1.3 Pivotal Task-1: Managing the Physical Environment . . . 436
13.1.4 Pivotal Task-2: Co-habiting and Collaborating . . . 437
13.1.5 Pivotal Task-3: Empowering Vulnerable Users . . . 439
13.1.6 Established Tools and Systems . . . 441
13.1.7 Challenges . . . 442

14 Spatial Intelligence for Others 443
14.1 Architecture, Engineering, and Construction . . . 443
14.2 Sports Analytics and Performance . . . 444
14.3 Cultural Heritage and Archaeology . . . 445
14.4 Marine and Underwater Systems . . . 447
14.5 Retail and Smart Spaces . . . 448
14.6 Outlook . . . 449

V Discussions and Futures 450

15 Foundations: What Spatial Intelligence Requires 452
15.1 Spatial Observability and Identifiability . . . 452
15.2 Representation and State Sufficiency . . . 453
15.3 Persistent World States and Belief Updating . . . 454
15.4 Predictive Validity and Physical Grounding . . . 456
15.5 Situated and Interactive Spatial Intelligence . . . 457

16 Research Gaps and Future Directions 459
16.1 Research Gaps along the Loop . . . 459

16.2 Benchmarking Gaps 461
16.3 Deployment Bottlenecks 462
16.4 Future Directions 463

17 Conclusion **465**

Notation

Symbols are listed in chapter order, with the meanings used in the referenced sections.

Symbol	Description	Reference
$\mathcal{E}$	Physical or simulated environment.	Sec. 2.2.2
$\mathcal{S}$	AI system or agent.	Sec. 2.2.2
$\mathcal{T},\ t \in \mathcal{T}$	Task space and a task in that space.	Sec. 2.2.2
$\mathcal{R}_{\mathcal{S}}$	Internal spatial representation space of the system.	Sec. 2.2.2
$\Phi_{\mathcal{S}}$	Mapping from the environment to the internal spatial representation space.	Sec. 2.2.2
$\mathcal{F}_{\mathcal{S}}$	Set of operators for perception, reasoning, prediction, and action.	Sec. 2.2.2
$\mathcal{O},\ \mathcal{A}$	Observation space and action space, respectively.	Sec. 2.2.2
$f^{\mathcal{S}}_{\text{perc}}$	Operator that maps observations to spatial representations.	Sec. 2.2.2
$f^{\mathcal{S}}_{\text{reason}}$	Reasoning operator inferring relations and constraints in the current representation.	Sec. 2.2.2
$f^{\mathcal{S}}_{\text{pred}}$	Prediction operator mapping a representation and an action to a future representation.	Sec. 2.2.2
$f^{\mathcal{S}}_{\text{act}}$	Action operator mapping a spatial representation to an action.	Sec. 2.2.2
$\pi_{\mathcal{S}}$	Task-conditioned policy for selecting actions.	Sec. 2.2.2
$\mathcal{R}^{\mathcal{S}}_t$	Internal spatial representation state at time t.	Sec. 2.2.2
$a^{\mathcal{S}}_t,\ o_{t+1}$	Selected action and subsequent observation.	Sec. 2.2.2
$\Psi_{\mathcal{S}}$	Update function for the internal spatial representation.	Sec. 2.2.2
Ω	Image grid or domain of pixel coordinates.	Sec. 4.1.1
$I,\ V$	Image and video, respectively.	Sec. 4.1.1
$S,\ \mathcal{L}$	Segmentation map and its set of labels.	Sec. 4.1.2
$D(u, v)$	Depth value at pixel (u, v).	Sec. 4.1.3
$\mathbf{N}(u, v)$	Unit surface normal at pixel (u, v).	Sec. 4.1.4
$\mathbf{F}(u, v)$	Optical flow giving horizontal and vertical pixel displacement.	Sec. 4.1.6
$\mathcal{K},\ H_k$	Set of keypoint locations and the heatmap for keypoint k.	Sec. 4.1.7
$P = \{p_i\}$	Point cloud with 3D coordinates and optional attributes.	Sec. 4.2.3
$M = (V, F)$	Triangle mesh with vertex set V and face set F.	Sec. 4.2.4
$f(\mathbf{x})$	Implicit scalar field whose level set defines a surface; a signed distance function in the SDF case.	Sec. 4.2.5
F_Θ	Neural radiance field mapping position and viewing direction to color and volume density.	Sec. 4.2.6
$\mathbf{x},\ \mathbf{d}$	3D query position and viewing direction in a radiance field.	Sec. 4.2.6
$\sigma,\ \mathbf{c}$	Volume density and emitted RGB color in a radiance field.	Sec. 4.2.6

Continued on next page

Symbol	Description	Reference
$\boldsymbol{\mu},\ \boldsymbol{\Sigma}$	Center and covariance matrix of a 3D Gaussian primitive.	Sec. 4.2.7
$\alpha_i,\ c_i$	Opacity and color of the i-th Gaussian primitive.	Sec. 4.2.7
$\Delta\mathbf{x}(t)$	Time-dependent displacement from a reference 3D position.	Sec. 4.3.4
$\mathcal{G}(t)$	Time-dependent collection of Gaussian primitives in a dynamic scene.	Sec. 4.3.5
$b_i,\ c_i,\ s_i$	Bounding box, class label, and confidence score of detection i.	Sec. 6.1.1
$f_\theta,\ \theta$	Parameterized prediction model and its learned parameters.	Sec. 6.1.3
$\mathbb{I}(\cdot)$	Indicator function, equal to one when its condition holds and zero otherwise.	Sec. 6.1.3
$Z,\ f,\ B,\ d$	Stereo depth, focal length, camera baseline, and disparity, with $Z = fB/d$.	Sec. 6.2.1
K	Camera intrinsic matrix.	Sec. 6.2.3
$R,\ \mathbf{t}$	Rotation and translation mapping world coordinates into the camera frame.	Sec. 6.2.3
$\mathbf{T}_t$	Camera pose at time t, represented by a rigid transformation.	Sec. 6.2.4
$\Delta\mathbf{T}_{i\to j}$	Relative camera transformation from frame i to frame j.	Sec. 6.2.4
$\mathcal{G}$	Scene graph of subject–predicate–object triplets.	Sec. 6.3.2
$\mathbf{v}(\mathbf{x})$	Scene flow: 3D displacement of the surface point observed at pixel $\mathbf{x}$.	Sec. 6.4.1
$\mathcal{T}_k$	Time-indexed trajectory of tracked object k.	Sec. 6.4.2
$\mathrm{SO}(3),\ \mathrm{SE}(3)$	Groups of 3D rotations and rigid transformations, respectively.	Sec. 7.1.1
$\mathbf{F}$	Fundamental matrix relating corresponding image points in two views.	Sec. 7.1.1
$\mathbf{q},\ \mathbf{J}(\mathbf{q})$	Joint configuration and the Jacobian mapping joint velocities to task-space velocities.	Sec. 7.1.1
$\mathcal{C}_{\text{free}}$	Feasible configuration space used in reachability and motion planning.	Sec. 7.1.1
$\mathcal{W}$	Set of contact wrenches used to test grasp force closure.	Sec. 7.1.1
$\mathbf{s}_t,\ \mathbf{u}_t$	Physical system state and applied action in the dynamics model.	Sec. 7.1.2
$M(\mathbf{q})$	Configuration-dependent mass matrix.	Sec. 7.1.2
$G = (V, E, w)$	Planning graph with vertices, edges, and edge weights.	Sec. 7.1.3
$\mathrm{do}(X = x)$	Intervention that sets variable X to x in a causal model.	Sec. 7.1.6
$z^\star$	Optimal solution returned by a differentiable optimization layer.	Sec. 7.2.4
π_θ	Learned policy for selecting actions.	Sec. 7.2.5
$r_t,\ \gamma$	Reward at time t and the discount factor.	Sec. 7.2.5
$Q^*(s, a)$	Maximum expected discounted return from taking action a in state s.	Sec. 7.2.5

Continued on next page

Symbol	Description	Reference
R_t	Discounted return from time t onward.	Sec. 7.2.5
$V_\phi,\ A_\phi$	State-value function and advantage function.	Sec. 7.2.5
$a_{t:t+H}$	Action sequence predicted over horizon H by a vision-language-action policy.	Sec. 7.3.2
$\tau = (t_1, \ldots, t_L)$	Sequence of intermediate reasoning steps in a chain of thought.	Sec. 7.3.3
$\mathcal{M}_t$	External spatial memory after the update at time t.	Sec. 7.3.6
$\mathcal{X},\ \mathbf{x}$	Space of tokenized text prompts and an input prompt.	Sec. 8.1.1
$\mathcal{Y},\ \mathbf{y}$	Pixel-space image domain and an image in that domain.	Sec. 8.1.1
$\mathcal{Z},\ \mathbf{z}$	Latent representation space and a latent code.	Sec. 8.1.1
$E,\ D$	Image encoder and decoder of the latent autoencoder.	Sec. 8.1.1
$\phi,\ \mathcal{E}$	Text encoder and its conditioning embedding space.	Sec. 8.1.1
G_θ	Conditional generator or sampler.	Sec. 8.1.1
$p_\theta(\cdot \mid \cdot)$	Learned conditional distribution over generated outputs or latent codes.	Sec. 8.1.1
ϵ_θ	Noise predictor used in diffusion generation.	Sec. 8.1.1
$\mathbf{z}_t$	Latent code at diffusion or flow time t.	Sec. 8.1.1
v_θ	Learned velocity field used in flow matching.	Sec. 8.1.1
$\mathbf{c},\ \mathcal{C}$	Structural conditioning signal and its input space.	Sec. 8.1.2
$\kappa,\ \mathcal{C}'$	Condition encoder and the space of encoded conditions.	Sec. 8.1.2
$\mathcal{M}$	Mask identifying the region to be edited.	Sec. 8.1.3
$\mathcal{R},\ \mathbf{r}$	Space of 3D representations and a 3D asset in that space.	Sec. 8.2.1
$\pi,\ \pi_v$	Camera pose used for rendering, with v indexing a view.	Sec. 8.2.1
Render	Renderer mapping a 3D representation and a camera pose to an image.	Sec. 8.2.1
$\mathcal{L}_{\mathrm{SDS}}$	Score-distillation objective for optimizing a representation with a frozen diffusion prior.	Sec. 8.2.1
$\mathcal{L}_{\mathrm{rec}}$	Loss measuring agreement between a reconstruction and its observations.	Sec. 8.2.3
$\lambda,\ \Omega$	Weight and regularizer used in scene reconstruction.	Sec. 8.2.3
Λ	Operator lifting a panorama or a set of panoramic views into a 3D representation.	Sec. 8.2.4
E_θ	Editing operator mapping a 3D representation and a condition to an edited representation.	Sec. 8.2.5
$\mathbf{m}^{1:T}$	Sequence of body poses describing a motion.	Sec. 8.4.1
$\mathbf{r}^{1:T}$	Time-indexed sequence of 3D representations for a dynamic scene.	Sec. 8.5
W_θ	Learned world model predicting the next state from state, action, and observation.	Sec. 8.6

Continued on next page

Symbol	Description	Reference
$S(t),\ A(t),\ O(t)$	World state, action, and observation, respectively.	Sec. 8.6
$E_\phi,\ P_\theta$	Observation encoder and latent-state predictor in a representation world model.	Sec. 8.6.1
$\mathrm{sg}(\cdot)$	Stop-gradient operator used in representation learning.	Sec. 8.6.1
$M(t)$	Latent mental state of another agent, including beliefs, desires, and intentions.	Sec. 8.6.3
$s_t^{\mathrm{phy}},\ s_t^{\mathrm{ment}}$	Physical and mental components of a joint world state.	Sec. 8.6.3
$\rho,\ \mathcal{L}$	Symbolic scene program and its language.	Sec. 8.7.1
$R(\rho)$	Execution or rendering of symbolic scene program ρ.	Sec. 8.7.1
M_t	Semantic environment map maintained during warehouse localization and mapping.	Sec. 9.5.4

Glossary

Terms and abbreviations are listed in the order of their first appearance in the chapters.

Term	Description	Reference
LLMs	**Large Language Models.** Models trained on large text corpora to process and generate language.	Sec. 1.1.1
RL	**Reinforcement Learning.** Learning to select actions through interaction and reward feedback.	Sec. 1.1.2
GIS	**Geographic Information Systems.** Systems for storing, managing, and analyzing geographically referenced data.	Sec. 1.1.4
VQA	**Visual Question Answering.** Answering natural-language questions about visual content.	Sec. 1.1.4
VLN	**Vision-and-Language Navigation.** Navigation using visual observations and natural-language instructions.	Sec. 1.1.4
GIScience	**Geographic Information Science.** The study of geographic information, its representation, and its analysis.	Sec. 1.2.2
SLAM	**Simultaneous Localization and Mapping.** Joint estimation of an agent's pose and a map of its environment.	Sec. 1.2.2
TSDF	**Truncated Signed Distance Function.** A signed distance function whose values are limited to a fixed range around a surface.	Sec. 1.2.2
SDF	**Signed Distance Function.** A function giving the distance to a surface, with a sign distinguishing its two sides.	Sec. 1.2.2
VO	**Visual Odometry.** Estimation of camera motion from a sequence of images.	Sec. 1.2.3
LiDAR	**Light Detection and Ranging.** Sensing that uses laser light to measure distances.	Sec. 1.2.3
CT	**Computed Tomography.** Imaging that reconstructs cross-sectional structure from X-ray measurements.	Sec. 1.2.3
MRI	**Magnetic Resonance Imaging.** Imaging that uses magnetic fields and radiofrequency signals to depict internal anatomy.	Sec. 1.2.3
CNNs	**Convolutional Neural Networks.** Neural networks that use convolutional filters to extract local features.	Sec. 1.2.3
BEV	**Bird's-Eye View.** A top-down representation of a scene.	Sec. 1.2.3
NeRF	**Neural Radiance Field.** A continuous representation of scene density and view-dependent radiance for novel-view synthesis.	Sec. 1.2.4
CoT	**Chain-of-Thought.** A sequence of intermediate reasoning steps used to solve a problem.	Sec. 1.2.4
VLA	**Vision-Language-Action Model.** A model that maps visual observations and language instructions to actions.	Sec. 1.2.4

Continued on next page

Term	Description	Reference
VLMs	**Vision-Language Models.** Models that jointly process visual and linguistic information.	Sec. 1.2.4
3DGS	**3D Gaussian Splatting.** A method for representing and rendering scenes using optimized 3D Gaussians.	Sec. 1.2.4
JEPA	**Joint Embedding Predictive Architecture.** An architecture that predicts target representations in an embedding space.	Sec. 1.2.5
XR	**Extended Reality.** An umbrella term for virtual, augmented, and mixed reality.	Sec. 2.2.3
RGB-D	**RGB and Depth.** Image data combining color channels with depth measurements.	Sec. 4.2
MLP	**Multilayer Perceptron.** A feedforward neural network with fully connected layers.	Sec. 4.2.6
SfM	**Structure from Motion.** Recovery of camera poses and 3D structure from overlapping images.	Sec. 4.2.7
DSL	**Domain-Specific Language.** A programming language designed for a particular class of tasks.	Sec. 5.1
IMU	**Inertial Measurement Unit.** A sensor unit that measures acceleration and angular velocity.	Sec. 5.5
IoU	**Intersection over Union.** The size of the overlap between two regions divided by the size of their union.	Sec. 6.1.1
mAP	**Mean Average Precision.** Average precision averaged across classes or evaluation categories.	Sec. 6.1.1
PnP	**Perspective-n-Point.** Estimation of camera pose from corresponding 3D points and 2D image locations.	Sec. 6.1.3
MVS	**Multi-View Stereo.** Dense reconstruction of 3D geometry from images with known camera poses.	Sec. 6.2.1
UAV	**Unmanned Aerial Vehicle.** An aircraft operated without a pilot on board.	Sec. 6.2.3
RANSAC	**Random Sample Consensus.** Robust model fitting by testing hypotheses from randomly sampled subsets of observations.	Sec. 6.2.4
LPIPS	**Learned Perceptual Image Patch Similarity.** A perceptual image distance computed from deep visual features.	Sec. 6.3.5
PSNR	**Peak Signal-to-Noise Ratio.** An image quality measure derived from mean squared reconstruction error.	Sec. 6.3.5
SSIM	**Structural Similarity Index.** An image similarity measure based on local luminance, contrast, and structure.	Sec. 6.3.5
VAE	**Variational Autoencoder.** A generative model that learns a probabilistic latent representation of data.	Sec. 6.5.2
IK	**Inverse Kinematics.** Finding joint configurations that achieve a desired pose of an end effector or body part.	Sec. 7.1

Continued on next page

Term	Description	Reference
TAMP	**Task and Motion Planning.** Joint planning of discrete task actions and feasible continuous motions.	Sec. 7.1.4
MPC	**Model Predictive Control.** Control that repeatedly optimizes a finite-horizon plan and executes its first action.	Sec. 7.1.4
GNNs	**Graph Neural Networks.** Neural networks that learn representations by aggregating information over graph connections.	Sec. 7.2.1
POMDP	**Partially Observable Markov Decision Process.** A sequential decision model in which the agent observes the environment state only indirectly.	Sec. 7.2.5
RAG	**Retrieval-Augmented Generation.** Generation conditioned on information retrieved from an external collection or memory.	Sec. 7.3.6
ToM	**Theory of Mind.** Reasoning about the beliefs, intentions, and knowledge of other agents.	Sec. 7.3.7
FVD	**Fréchet Video Distance.** A distance between feature distributions of generated and reference videos.	Sec. 7.3.8
FID	**Fréchet Inception Distance.** A distance between Inception feature distributions of generated and reference images.	Sec. 7.3.8
GANs	**Generative Adversarial Networks.** Generative models trained through competition between a generator and a discriminator.	Sec. 8.1.1
DDIM	**Denoising Diffusion Implicit Model.** A diffusion sampling formulation that supports deterministic sampling and fewer sampling steps.	Sec. 8.1.3
SDS	**Score Distillation Sampling.** Optimization of a representation using guidance from a pretrained diffusion model.	Sec. 8.2.1
GNSS	**Global Navigation Satellite System.** A satellite system providing positioning, navigation, and timing information.	Sec. 9.2.4
BIM	**Building Information Modeling.** Creation and management of digital models describing a building's physical and functional properties.	Sec. 14.1

Part I

Introduction and Background

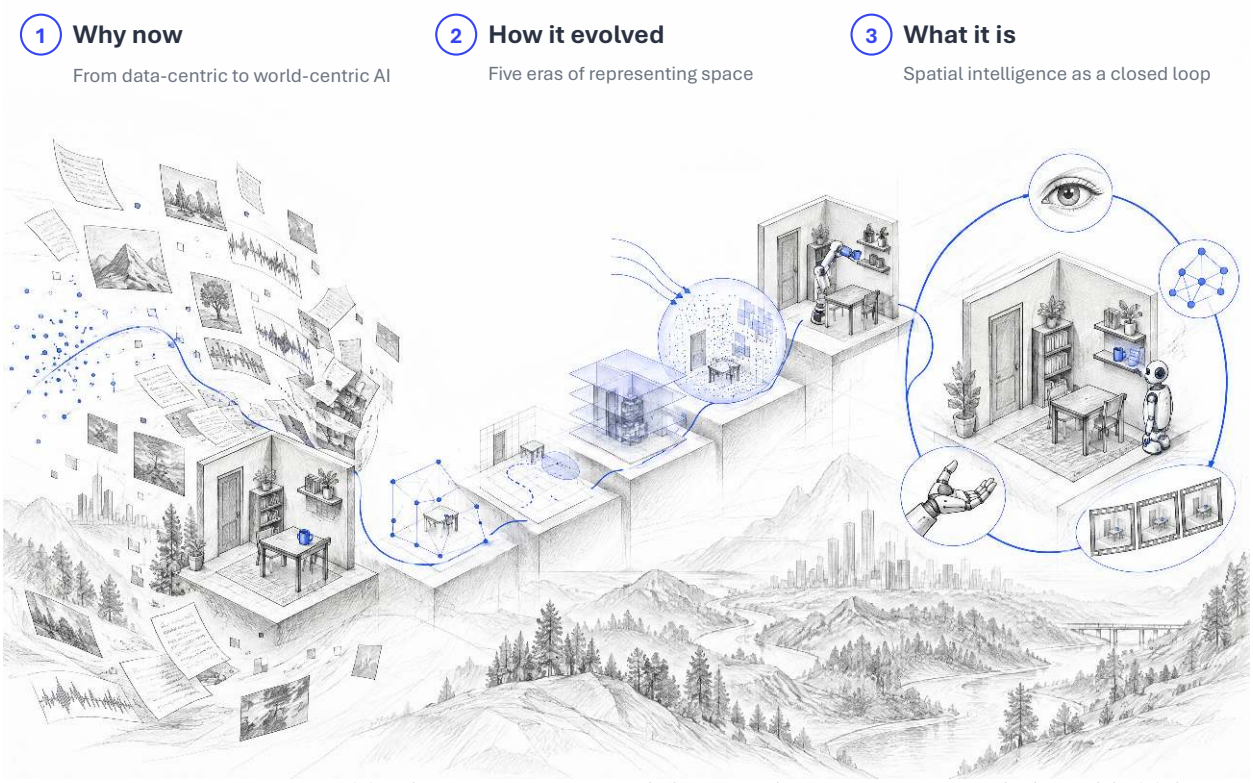


Figure I: Part I in one view: (1) why now, as scattered data condense into a grounded world; (2) how it evolved, one room in each of five eras; (3) what it is, a loop of perceiving, reasoning, predicting, and acting.

For decades, AI has advanced by mastering well-bounded tasks: recognizing patterns, answering questions, generating outputs from static inputs. Intelligence in the natural world unfolds in space, shaped by structure, constraint, motion, and interaction. As AI systems move from passive prediction to active participation in physical and simulated environments, models trained only on tokens, pixels, or isolated observations struggle to stay coherent across space, time, and action, because they have no explicit notion of where they are, how the world is arranged, or how their interventions change what comes next. Spatial intelligence supplies that notion, and robust, adaptive, and embodied intelligence depends on it. It anchors intelligence in world-centered representations that encode geometry, topology, semantics, dynamics, and affordances, and that let a system perceive beyond its current view, reason about regions it has not seen, predict how a configuration will evolve, and act on it with purpose. Navigation, manipulation, simulation, and long-horizon planning all draw on this one substrate.

This part lays the conceptual ground for treating spatial intelligence as a paradigm. Chapter 1 explains why it is rising now, through four converging shifts in what AI is expected to do, how it learns, the modalities it handles, and the disciplines that study it, and reads the field's history as five eras, each defined by what it assumed space to be. Chapter 2 reconciles the views of computer vision, robotics, geographic information science, psychology, cognitive science, and linguistics in a working definition, an AI system and its environment coupled through a world-centered representation, and gives it a light formalization. Chapter 3 sets out the research pillars that organize the rest of the handbook.

Chapter 1

Spatial Intelligence: A Rising AI Paradigm

AI has entered a stage where scale and capability alone are no longer sufficient. Despite remarkable advances in perception, language, and generation, modern AI systems continue to struggle with a fundamental challenge: understanding, representing, and acting within the structured physical world. As intelligent systems increasingly move beyond static inputs and isolated tasks, the question shifts from how well a model predicts tokens or labels to how it forms internal models of space, time, action, and causality. This shift marks the emergence of *spatial intelligence* as a central organizing paradigm in artificial intelligence.

Spatial intelligence concerns the ability of an intelligent system to build and manipulate internal representations of the world that support perception, reasoning, prediction, and embodied interaction. It is inherently world-centered, grounded in geometry, motion, and physical constraints, and inseparable from action and experience. Rather than treating space as an auxiliary signal or downstream output, spatial intelligence positions it as a core substrate through which intelligence is structured. This perspective cuts across traditional boundaries, unifying problems in computer vision, robotics, autonomous systems, simulation, cognitive science, spatial computing, AR/VR, and geospatial analysis under a shared representational question: how should an intelligent system model the world it inhabits?

This chapter establishes spatial intelligence as a rising AI paradigm by addressing two foundational questions. Section 1.1 examines why spatial intelligence has become critical *now*, tracing its emergence to converging shifts in AI methodology, data modalities, system scale, and problem scope. Section 1.2 then situates this paradigm historically, organizing decades of fragmented progress into a coherent sequence of representational eras. Together, these perspectives provide the conceptual grounding for the rest of the handbook, framing spatial intelligence not as a collection of tasks or techniques, but as a unifying foundation for world-centered artificial intelligence.

1.1 Why Spatial Intelligence? Why Now?

As AI continues to advance, a growing consensus has emerged that data alone is not enough; true intelligence must be grounded in the physical world [1, 2, 3]. This realization marks a fundamental shift from purely *data-centric* paradigms, centered around language or perception, to *world-centric* systems capable of modeling, reasoning, and acting within spatially structured environments. In this section,

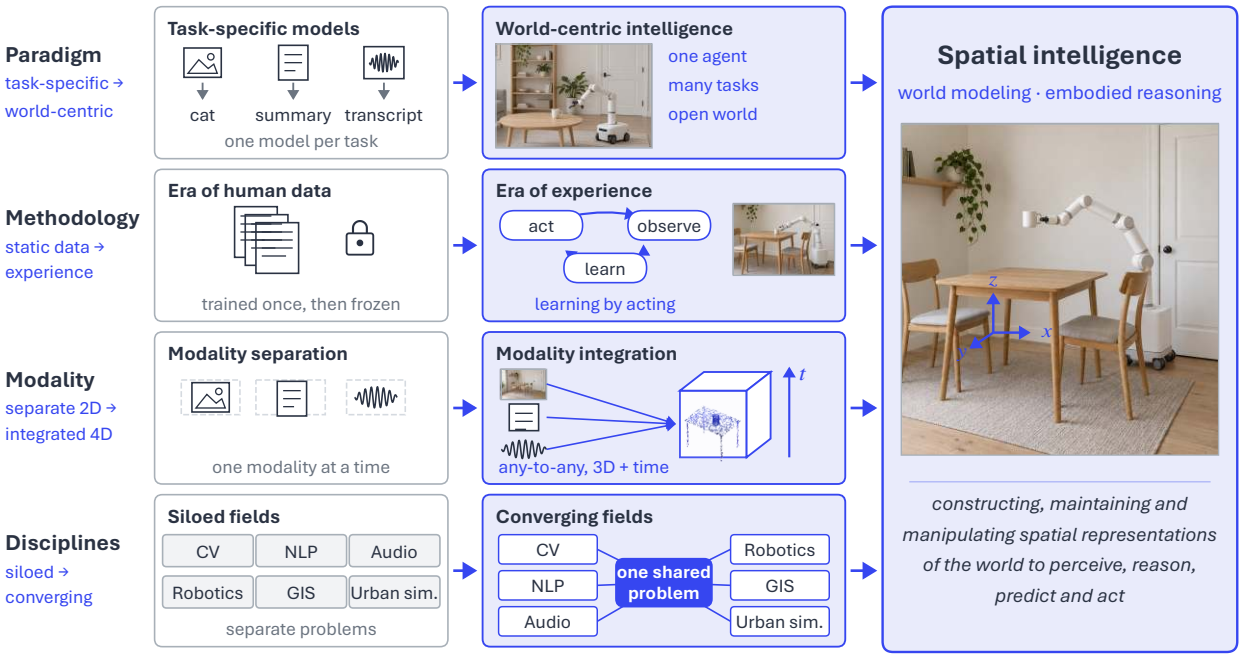


Figure 1.1: Spatial intelligence emerges from converging shifts in AI paradigms, methodologies, modalities, and disciplines, collectively pushing AI toward world modeling and embodied reasoning.

we explore the motivations behind this transition, drawing from the AI paradigm, methodology, and modality to disciplines.

1.1.1 AI Paradigm Shift

AI is undergoing a foundational paradigm shift, *i.e.*, from task-specific models toward world-centric general intelligence. Traditional AI systems were designed to solve narrowly defined problems, such as image classification, text generation, or speech recognition, each trained and optimized within a well-constrained input-output space. However, the emergence of foundation models and large language models (LLMs) has reshaped expectations: AI is now envisioned to generalize across domains, modalities, and tasks in open-world settings. This ambition hinges on a deeper capability, the ability to understand and model the world itself, often described as building a "world model" [4].

Spatial Intelligence emerges as a central pillar enabling this capability. It encompasses the perception and representation of objects, scenes, spatial layouts, motion, and physical interactions, and it serves as a carrier of world-generalizable knowledge that supports reasoning and transfer across tasks and modalities. Fei-Fei Li has made this stake explicit: "*Our dreams of truly intelligent machines will not be complete without spatial intelligence*" [5]. On her account, any AI system that aims to serve in physical environments must possess a rich understanding of spatial structure and dynamics.

Moreover, this transformation also signals a shift in the boundary of context: traditional AI focused on self-contained intelligence, where the model reasons primarily over its own input context. Today's systems, in contrast, must be understood as *individuals embedded in the world*[1], agents whose perception and action are grounded in broader spatial, temporal, and social environments. Beyond the technical evolution, this amounts to a philosophical deepening of AI's cognitive scope. Li captures the underlying continuity in

[1]In Context Engineering [6], the authors frame the former paradigm as *Context as Instruction*, and cast the latter as *Context as World*.

evolutionary terms: spatial intelligence is, in her words, what "*links perception with action*", the capacity that turned seeing into surviving and, eventually, into understanding and doing [7].

Without accurate spatial modeling of the world, therefore, there can be no robust reasoning, decision-making, or interaction. A wide range of emerging tasks, including robotic manipulation, embodied navigation, video understanding, and virtual human interaction, demand spatial grounding as a prerequisite, which makes spatial intelligence a load-bearing requirement for any AI system expected to act in, adapt to, and help within the real world.

1.1.2 AI Methodological Shift

Historically, from the era of classical machine learning [8, 9] to the deep learning revolution [10], and now to the LLM wave [11, 12], progress in AI has largely followed a data-centric trajectory: larger models trained on increasingly vast static corpora. While this paradigm has delivered remarkable achievements, it also imposes an inherent limitation, in that AI systems are still framed as static input–output function approximators rather than as agents that can perceive, reason, and act within the real world. The recent surge of LLMs, such as GPT [11], PaLM [13], and LLaMA [14], has only amplified this trajectory. Although these models exhibit stunning linguistic fluency, symbolic manipulation, and emergent generalization capabilities [15, 16], they remain intrinsically disembodied and spatially ungrounded. They can generate poetic descriptions of a cup on the table, yet they cannot perceive the cup, interact with it, or build a physical model to explain its behavior. This exposes the core limitation of the current paradigm: *LLMs might operate on surface-level linguistic statistics without accessing the structured constraints, causal mechanisms, and physical affordances of the real world*.

Fei-Fei Li has repeatedly emphasized this disconnect. Linguistic competence, she argues, must not be mistaken for understanding: for all their capability, today's language models *"remain wordsmiths in the dark; eloquent but inexperienced, knowledgeable but ungrounded"* [5]. Without spatial intelligence, encompassing physics, motion, and the transformative nature of actions, AI systems cannot develop a grounded, interactive model of reality. Her insight exposes a deeper methodological gap: true intelligence must ground its internal representations in embodied space, action, and physical experience, and text alone cannot supply that grounding.

Yann LeCun extends this critique at the architectural level, famously dismissing LLMs as "*an off-ramp, a distraction, a dead end*" on the path toward human-level intelligence. In his view, autoregressive language models do not, and cannot, construct models of how the world works; they continue text through pattern completion rather than through prediction grounded in the dynamics of an external world [17]. His proposed alternative, an architecture organized around a learned predictive world model, points squarely toward systems capable of predictive modeling, causal reasoning, and dynamic world interaction.

Rich Sutton, from a reinforcement learning (RL) perspective, presses the same point on the learning process itself. Sutton's "bitter lesson" famously argued that general methods which scale with computation win out over hand-engineered knowledge [18]; notably, he does not regard today's LLMs as the vindication of that argument. In his recent view, systems trained once on static human data and then frozen are cut off from the very thing that drives intelligence, namely continual learning from one's own experience [19]. Silver and Sutton argue that the field must move from the "era of human data" into an "era of experience", in which agents learn predominantly from acting in the world and observing consequences. Learning from static text captures correlations in what humans have already written; understanding the world requires action, consequence prediction, and causal discovery.

Taken together, these critiques, arising independently from visual cognition, neural architectures, and reinforcement learning, signal a profound methodological shift in progress: AI must evolve from a purely LLM-centric paradigm to one of world-integrated intelligence. It must move beyond static, offline training toward continual, interactive, and adaptive learning, and it must outgrow shallow pattern matching in text

to engage with the physical, spatial, and causal structures that govern real environments. More than an algorithmic upgrade, this demands a redefinition of what intelligence means: agents capable of perceiving the world, predicting its dynamics, interacting within it, and learning continuously from such experiences [20, 21, 22].

1.1.3 Modality Shift

Human intelligence is inherently multimodal. Any meaningful pursuit of AGI must therefore embrace multimodality as a foundational principle. The trajectory of multimodal AI is undergoing a significant shift, from early-stage *modality separation* toward deep *modality integration*. In earlier AI systems, perception and modeling are often confined to a single modality [23], such as text-only language models, speech recognition engines, or visual classification systems. However, with the rise of foundation models, the research paradigm is shifting towards hybrid-modality [24, 25] and even any-to-any cross-modal understanding and generation [26]. That is, both input and output can now be flexibly composed across arbitrary modalities (*e.g.*, audio-based question answering, video synthesis, image-conditioned dialogue) [27, 28, 29]. Simultaneously, the very nature of modality is expanding in dimensionality and diversity. Traditional sensory modalities, *e.g.*, vision, language, and audition, represent only a subset of how humans perceive the world. With the rapid advancement of sensor technology, future AI systems are expected to be equipped with a broader range of high-dimensional and multi-channel sensory inputs [30, 31, 32, 33], including spectrograms, millimeter-wave radar, infrared thermography, tactile feedback, gustatory signals, and olfactory landscapes. These emerging modalities are especially vital in domains such as Medicine [34], Robotics [35], Environmental Monitoring [36], and AR [37], offering a more human-aligned perceptual basis for multimodal AI systems. This multimodal expansion necessitates a unified modeling capability that operates at the level of holistic world understanding.

Among all modalities, vision remains the most central to human perception. One of the most critical trends in modality development is the transition from static imagery to video, and from 2D representations to richer 3D and even 4D (spatiotemporal) structures [38, 39]. Traditional image-based recognition tasks operate on single-frame static inputs. However, real-world environments demand continuous understanding, prediction, and generation across space and time. This drives the growing importance of tasks such as 3D reconstruction, temporal modeling, trajectory prediction, video-based question answering, and dynamic world state tracking [40, 41]. The increasing richness of modalities goes well beyond a technological extension; it is a necessary foundation for building AGI. To rival or surpass human intelligence, AGI must be equipped with a comparably rich sensorium, enabling it to perceive, reason, and act effectively in dynamic, multimodal environments. Thus, the integration and extension of modalities are not ends in themselves, but serve the larger ambition of world modeling, *i.e.*, the central focus of this work.

This point has been underscored by Fei-Fei Li in her discussion of spatial intelligence. Tracing the evolutionary arc from the first light-sensing organisms to primates, she observes that *"nature has created this virtuous cycle of seeing and doing powered by spatial intelligence"* [7]. Such capability acts as a bridge between cognition and the physical world, a necessary scaffold for grounding AI in embodied reality; without it, meaningful interaction with the real world is out of reach. Where traditional intelligence centers on language or symbolic manipulation, spatial intelligence is tightly coupled with physical structure, three-dimensional relationships, and dynamic evolution, making it the cornerstone of embodied, interactive intelligence. From the perspective of modality development, then, spatial intelligence marks the turning point where multimodal intelligence evolves from surface-level perception to deep, structured understanding, well beyond a mere byproduct of increasing sensory complexity. The focus of AI research must accordingly shift from modeling and aligning modalities to modeling the structured world that those modalities collectively describe, the path toward a closed-loop paradigm from perception to world modeling, and from world modeling to causal reasoning.

1.1.4 Disciplinary Convergence

Modern AI research is undergoing a profound disciplinary evolution, centered around *modality enrichment* and *spatiophysical coupling*. This paradigm shift reflects a convergence of traditionally siloed domains, spanning Computer Vision (CV), Natural Language Processing (NLP), Audio Processing, Robotics, Geographic Information Systems (GIS), and Urban Simulation, toward a unified framework for modeling intelligent agents that are physically embedded in and interact with the real world.

In the vision domain, the research focus has gradually shifted from static image processing to dynamic scene understanding. These developments provide a perceptual foundation for large-scale world modeling. Language modalities have also evolved beyond static language modeling (*e.g.*, BERT [42], GPT [43]), toward multimodal semantic fusion tasks such as Visual Question Answering (VQA) [44], Vision-and-Language Navigation (VLN) [45], and embodied dialogue, which have become critical interfaces in the expansion of large language models into the multimodal regime. The audio modality has similarly moved beyond traditional speech recognition to include audio event detection, ambient sound understanding, and even 3D auditory modeling, all of which support spatial cognition and situational awareness in real-world interactive environments [46, 47].

Meanwhile, the rapid development of robotics and embodied AI is pushing the evolution of AI systems from “perception-understanding" pipelines to closed-loop “perception-action" systems. AI is outgrowing its existence as software on a computer, becoming embodied, real-time, and spatially grounded. In domains such as autonomous driving, robotic manipulation, and wearable AI systems, tasks increasingly require deep spatial reasoning capabilities for layout perception, path planning, and object manipulation [48, 49]. These tasks demand integration not only of vision and language capabilities but also of knowledge from control theory, motion planning, and state estimation, marking a shift toward truly multidisciplinary AI systems.

The expansion of modalities is also reshaping traditional disciplines through interdisciplinary integration. In GIS and urban computing, AI models are increasingly combined with spatial coordinates, map representations, and temporal population data, enabling applications in smart city planning, trajectory prediction, and disaster simulation. In Earth sciences and remote sensing, AI is employed for multimodal surface analysis, climate modeling, and environmental monitoring, pushing the frontier toward 3D Earth modeling and dynamic ecological understanding [50]. In architectural design and digital twin systems, AI supports full lifecycle 3D spatial modeling, simulation, and predictive analytics for complex structures and spaces.

Cognitive science and neuro-inspired AI (NeuroAI) are playing an increasingly pivotal role in this trend. Foundational research on spatial intelligence in human cognition, such as hippocampus-mediated cognitive maps, spatial memory, and embodied cognition theory, is gradually informing the design of AI spatial perception systems [51, 52]. As Fei-Fei Li puts it, spatial intelligence is *“the scaffolding upon which our cognition is built”*, the substrate that anchors intelligence to the physical world [5]. Thus, the path toward AGI must transcend abstract symbolic computation and advance toward grounded-in-reality agent intelligence. This trend reshapes the research trajectory of AI and, at the same time, redefines the underlying infrastructure of general intelligence itself, with spatial intelligence standing as a central pillar of the emerging world-centered era.

1.2 Historical Evolution and Milestones



The history of spatial intelligence is best understood as a sequence of shifts in *how space is represented* and *what that representation is expected to support*. Over decades, the field has repeatedly shifted among explicit structure and learned abstraction, offline reconstruction and online interaction, and local geometry and world-level state. To make this evolution concrete, we organize the landscape into five eras, each characterized by a dominant representation paradigm and its associated methodological commitments: from early symbolic and cognitive notions of spatial structure, to probabilistic geometric

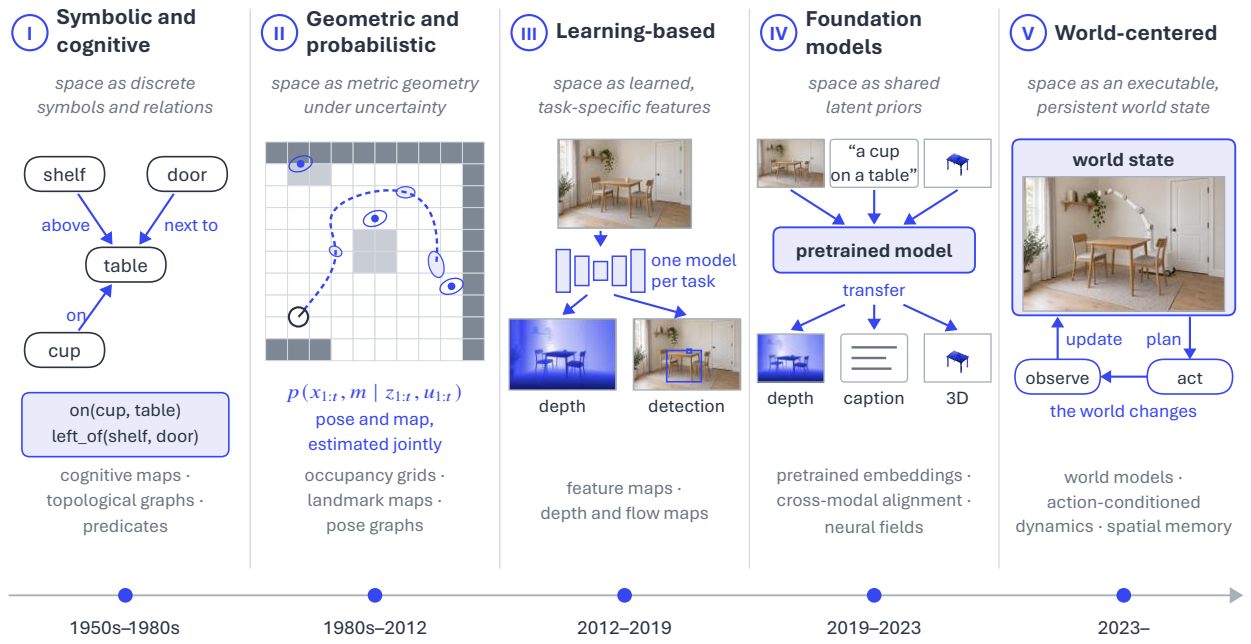


Figure 1.2: The historical evolution of spatial intelligence across five representation-centric eras, from explicit symbolic structure, through estimated geometry and learned features, to executable world states. The visual examples highlight the dominant representation and its role in each era. Dates indicate approximate shifts in emphasis, not exclusive boundaries; earlier paradigms remain in use.

estimation, to learning-based spatial perception, to large foundation representations that are transferable and cross-modal, and finally to world-centered spatial intelligence where representations must be executable, updatable, and tightly coupled with action in dynamic environments [53, 54, 55, 56, 5]. Figure 1.2 shows how each era would represent the same scene, Table 1.1 condenses what each era took space to be and what it left unsolved, and Figure 1.3 places about a hundred representative works from six fields on a common time axis.

1.2.1 Era 1: Symbolic and Cognitive Spatial Computing (1950s–1980s)

Symbolic Space and Cognitive Structure as the Starting Point. This era marks the cognitive and symbolic origin of spatial intelligence. Under severe limits in computation and data, space was primarily treated as a structured object that could be explicitly described, reasoned about, and manipulated. Accordingly, researchers favored symbolic rules, relational descriptions, and graph-like structures to encode environments, viewing spatial representation as a verifiable "spatial description language" that enables planning and logical inference [53, 57]. The lasting contribution of this stage is not empirical performance, but two foundational commitments. First, goal-directed behavior requires an internal world representation. Second, spatial representations are inherently compositional, consisting of objects, relations, and connectivity, which naturally supports rule-based spatial reasoning [58].

From a cognitive science perspective, this period also established the idea that spatial ability is not mere perception but an internal, operable mental model. Concepts such as cognitive maps and spatial memory motivated the view that an intelligent system should maintain structured internal states that support navigation, mental imagery, and transformation of spatial relations [59, 60]. In parallel, early robotics provided an engineering prototype of this view: the classic sense-plan-act pipeline connected symbolic spatial models with planners and execution, demonstrating how an agent can translate limited sensing into purposeful behavior within an environment [61, 62].

Table 1.1: The five eras of spatial intelligence compared along six dimensions: how each treated space, its core representations, what the representation was for, its characteristic methods, the fields that drove it, and what it left unsolved for the next era.

	I. Symbolic and cognitive 1950s–1980s	**II. Geometric and probabilistic** 1980s–2012	**III. Learning-based** 2012–2019	**IV. Foundation models** 2019–2023	**V. World-centered** 2023–
Space treated as	discrete symbols and relations	metric geometry under uncertainty	learned, task-specific features	shared latent priors	an executable, persistent world state
Core representations	cognitive maps, topological graphs, predicates	occupancy grids, landmark maps, pose graphs	feature maps, depth and flow maps, point and voxel features	pretrained embeddings, cross-modal alignment, neural fields	world models, action-conditioned dynamics, spatial memory
What it is for	planning and inference by symbol manipulation	a consistent pose and map from noisy sensing	perception objectives: detection, depth, motion	transfer across tasks, modalities, and environments	long-horizon prediction, planning, and interaction
Characteristic methods	heuristic search, rule-based planning, logic	Bayesian filtering, nonlinear least squares, multi-view geometry	supervised and self-supervised deep networks	large-scale pretraining, contrastive alignment, instruction tuning	action-conditioned prediction, memory, closed-loop learning
Main fields	cognitive science, classical AI, early robotics	robotics, photogrammetry and vision, GIS	computer vision, autonomous driving	multimodal AI, neural rendering, embodied AI	embodied AI, simulation and digital twins, AR/VR
Left unsolved	meaning asserted by rules, not inferred from noisy sensing	features and pipelines engineered by hand	one model per task; no persistent world model	no persistent state that action updates	open: long-horizon consistency under motion and partial observability

Representative Milestones in Techniques and Representations. A widely cited conceptual root is Tolman's proposal of "cognitive maps" (1948), which framed navigation as relying on internal map-like structures rather than stimulus-response associations [63]. In classical AI, the Dartmouth Workshop (1956) crystallized the symbolic paradigm and encouraged formal, rule-based representations of intelligence, providing intellectual soil for early spatial reasoning systems [64, 65]. Planning and search then became the main computational language for spatial decision making: A* heuristic search (1968) [66] offered a principled graph search framework for shortest-path style navigation problems, and STRIPS (1971) [67] formalized actions as operators over states, making spatial tasks expressible as sequences of symbolic state transitions [68, 69].

In robotics, Shakey (late 1960s to early 1970s) became a milestone system that integrated perception, symbolic spatial modeling, planning, and acting in a closed pipeline [61]. Although the sensory and actuation capabilities were primitive by modern standards, the system concretely instantiated the era's thesis that spatial intelligence can be built by maintaining symbolic representations and using them to plan actions. In neuroscience and cognitive modeling, O'Keefe and Nadel's *The Hippocampus as a Cognitive Map* (1978) further systematized the cognitive map account, reinforcing the interpretation that internal spatial representations are functional substrates for memory and navigation [51].

Across these milestones, several representation families repeatedly appeared and stabilized: symbolic spatial models and rule-based spatial descriptions; topological graphs capturing connectivity and place relations; qualitative spatial relations such as containment, adjacency, and support; landmark-based

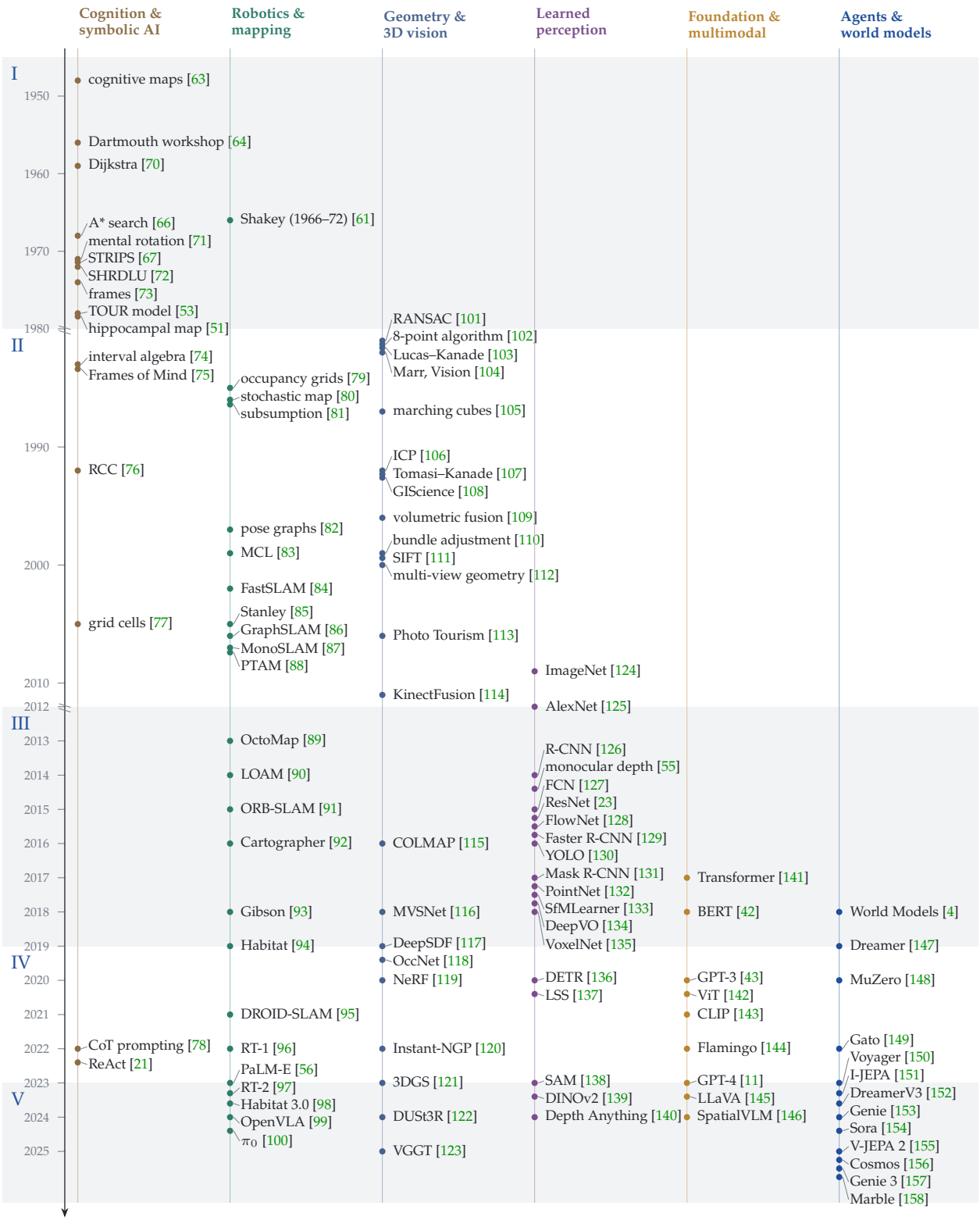


Figure 1.3: Representative works in spatial intelligence by field and year, set against the five eras of Table 1.1 (shaded bands, I–V). Each marker sits at the year of the original work, and the bracketed number cites it; where several works share a year, labels are spread and joined to their markers by short leaders. The time axis is compressed before 1980 and expanded after 2012.

representations for navigation; and grid-world abstractions that later became a convenient substrate for sequential decision making and, eventually, reinforcement learning (RL) formulations [159]. Together, they define the representation-centric signature of Era 1: spatial intelligence begins with explicit structure, grounded in cognition-inspired internal maps and implemented through symbolic reasoning and sense-plan-act-style control.

1.2.2 Era 2: Geometric and Probabilistic Spatial Modeling (1980s–2012)

From Symbolic Descriptions to Metric Geometry under Uncertainty. With the maturation of sensors (laser range finders, wheel odometry, early vision), computing infrastructure, and probabilistic estimation theory, spatial representation shifted away from purely symbolic description toward mathematical modeling of continuous geometry with uncertainty. Space was increasingly treated as a measurable physical entity whose state must be inferred, rather than a rule-based structure whose meaning is asserted. Consequently, the central question of this era shifted from describing the world in symbols to maintaining spatial consistency under noise, drift, and partial observability. This transition also re-centered the representation itself: a map is not a static record, but a set of uncertain state variables, typically the joint configuration of pose and environment, which must be updated and refined through Bayesian estimation and optimization [54, 160].

In this view, representation is inseparable from the inference machinery that maintains it. Coordinate frames and metric spaces became the common language that connects sensing, estimation, and control, while probabilistic modeling provided a principled way to fuse heterogeneous measurements and quantify uncertainty. As a result, a representation-centric principle was consolidated: spatial intelligence requires not only a geometric model of the environment, but also an explicit account of how that model is updated over time as new evidence arrives [161, 162, 163]. This era also laid much of the groundwork for long-term autonomy by formalizing how an agent can localize itself, build maps, and navigate using a consistent geometric state.

In parallel, macro-scale spatial disciplines pushed representation standardization beyond local robotics maps. Surveying, geodesy, cartography, remote sensing, and GIScience strengthened coordinate reference systems, map projections, topological relations, and spatial databases, making large-scale spatial representations structured, queryable, and interoperable at city-to-Earth scale [108]. At the same time, computer vision developed a rigorous multi-view geometry toolkit that defined how 3D structure and camera motion can be recovered from images, reinforcing the idea that geometry is an estimable latent state rather than an explicit symbolic description [112].

Representative Techniques and Milestones. A representative milestone in this era is the emergence of probabilistic state estimation as the dominant language for maintaining spatial consistency under noise and partial observability. Kalman filtering and its nonlinear variants (EKF and related formulations) became widely adopted in robotics and autonomous systems during the 1980s and 1990s, enabling spatial quantities such as pose, velocity, and sensor biases to be represented as random variables with explicit uncertainty propagation [164]. This shift is representation-centric: it turns "space" into a belief state that must be continuously updated, rather than a deterministic symbolic description. Building on this estimation view, occupancy grid mapping in the mid-1980s provided a canonical probabilistic representation that directly fuses range sensing into a discretized spatial field of free versus occupied space [79, 54]. Although conceptually simple, occupancy grids became a durable interface between sensing and downstream planning, because they encode geometry together with uncertainty in a form amenable to incremental updates. Related metric map abstractions, coordinate frames, and sensor models matured in parallel, standardizing the idea that maps are measurable geometric objects anchored to physical reference systems [165, 89].

The SLAM problem then consolidated these ideas into a unified formulation where the agent's pose and map are jointly inferred [166]. EKF-SLAM, whose stochastic-map foundations were laid in the late 1980s

[161] and refined through the 1990s into the early 2000s [167], established the coupled pose-landmark state as a central representation, while particle-filter-based SLAM, including FastSLAM (2002) [84], demonstrated that sampling and factorization could scale inference by exploiting conditional independencies. Beyond their algorithmic contributions, these lines of work stand as milestones in the operationalization of spatial representation: the map becomes a latent variable tied to motion and observation processes, with consistency defined probabilistically.

As systems expanded in scale and duration, graph-based spatial representations rose to prominence in the early 2000s. Pose graphs and factor graphs [82] recast spatial estimation as a global constraint-satisfaction problem, enabling loop closure and large-scale consistency through nonlinear optimization. GraphSLAM [86] and pose-graph optimization frameworks provided a modular representation where constraints from heterogeneous sensors and revisits could be integrated as edges in a unified objective [168]. This was a conceptual milestone because it decoupled spatial consistency from strict temporal recursion and made map structure explicit as an optimization-ready graph.

In computer vision and photogrammetry, multi-view geometry matured into a systematic toolkit for recovering 3D structure from images. Bundle adjustment, a core optimization primitive with roots in analog photogrammetry and consolidated for computer vision in the late 1990s, jointly refines camera poses and 3D points to enforce global geometric consistency across views [110]. This family of methods made camera-centric coordinate frames, point clouds, and calibrated reconstructions standard spatial representations, and it established geometry and optimization, rather than semantics, as the primary organizing principles of spatial understanding in vision during this era. Complementary to these estimation and reconstruction pipelines, geometric registration and fusion techniques advanced practical 3D mapping. Iterative Closest Point (ICP) and its robust refinements enabled consistent aggregation of partial point measurements across viewpoints [106, 169]. In parallel, early volumetric representations, including voxel grids and truncated signed distance function style formulations, began to encode surfaces and free space within unified volumetric fields [105, 109, 114, 170]. While computationally constrained at the time, these representations introduced the field-based view of space that would later become influential for differentiable and implicit modeling.

Finally, at macro scales, GIS and surveying communities standardized spatial representations through coordinate reference systems, projections, geodesy, and spatial databases. Digital elevation models, map products derived from remote sensing, and queryable geospatial layers enabled spatial reasoning at city, regional, and planetary scales with an emphasis on consistency, interoperability, and long-horizon stability [108, 171]. Although historically less coupled to embodied agents, these infrastructures are milestones for spatial intelligence because they demonstrate how spatial representation can function as a persistent, structured substrate supporting navigation, planning, and analysis beyond local robotic maps.

1.2.3 Era 3: Learning-based Spatial Modeling (2012–2019)

From Engineered Geometry to Task-driven Learnable Spatial Representations. This era marks a fundamental shift in how spatial representations are constructed. We date its onset to 2012, when AlexNet's ImageNet result made deep learning the dominant paradigm for visual perception [125]; the ImageNet dataset itself (2009) and early feature-learning work seeded the transition, but 2012 is when learned representations displaced hand-crafted ones as the default across spatial perception tasks. With the rise of machine learning, and especially deep learning, spatial modeling moves away from pipelines dominated by hand-crafted geometry, explicit probabilistic inference, and manually designed features. Instead, spatial structure begins to be encoded implicitly within neural networks through learned feature maps, latent embeddings, and task-optimized intermediate representations. Space is no longer represented solely as a map, graph, or state vector defined by engineers, but as something that can be learned directly from data through end-to-end optimization [124, 55, 119].

A defining characteristic of this period is that spatial representations are learned in a strongly task-driven manner. Networks learn spatial cues to solve specific objectives such as object detection, semantic segmentation, depth estimation, visual odometry, or 3D recognition. As a result, spatial knowledge is often embedded in discriminative feature spaces tailored to particular datasets and benchmarks. While these representations dramatically improve performance on individual tasks, they rarely constitute a unified or reusable world model. Spatial intelligence in this era remains fragmented, with representations optimized locally for perception subtasks rather than globally for cross-task reasoning or long-term world understanding [127, 172, 133].

Another important aspect of this era is the parallel but largely uncoordinated progress across multiple domains. In computer vision, learning-based approaches revolutionize image, video, and 3D perception. In robotics, learning begins to augment classical pipelines for perception and control, but geometry-heavy systems remain dominant for reliability. Remote sensing adopts deep models to learn spatial patterns from multispectral, hyperspectral, SAR, and LiDAR data at large scales. Medical imaging leverages deep networks to learn volumetric spatial representations from CT and MRI scans. Autonomous driving stacks integrate learned perception modules with geometric localization and mapping. Despite shared reliance on learned spatial features, these communities operate with different benchmarks, representations, and evaluation criteria, reinforcing the task-specific nature of spatial modeling in this era [173, 174, 35].

Representative Techniques and Milestones. A widely recognized milestone is the introduction of deep convolutional neural networks (CNNs) for visual perception, catalyzed by AlexNet in 2012, which establishes CNNs as the dominant backbone for spatially grounded perception tasks [125, 175]. This shift is quickly reinforced by region-based detection frameworks such as R-CNN and its successors from 2014 onward, which embed object-level spatial reasoning directly into learned feature hierarchies [172, 131]. ResNet in 2015 further deepens representational capacity and becomes a de facto foundation for most spatial perception pipelines [23]. In three-dimensional perception, PointNet in 2017 marks a milestone by introducing point clouds as a first-class input modality for deep networks [132]. This work opens the door to learned spatial representations beyond images, influencing robotics, autonomous driving, and indoor scene understanding. Concurrently, early voxel-based and sparse convolution methods begin to encode space volumetrically, albeit at limited resolution due to computational constraints [176].

Learning-based depth estimation emerges as another key milestone. Between 2014 and 2016, monocular and stereo depth prediction networks demonstrated that geometric cues such as disparity and scale can be inferred from data rather than explicit triangulation [55, 177]. Around the same period, learning-based visual odometry [134] and ego-motion estimation appeared [178], operating alongside classical SLAM systems. This coexistence highlights a transitional moment where geometry-driven consistency and learning-driven representation evolve in parallel rather than replacing one another [134, 179]. Self-supervised and unsupervised learning of spatial structure became influential from 2016 to 2018. Methods that exploit photometric consistency, view synthesis, and temporal coherence enable networks to learn depth and motion without dense ground truth supervision [133, 180]. These approaches strengthen the idea that spatial representations can be learned from raw sensory streams, reinforcing the data-driven paradigm while remaining largely task-bound.

Autonomous driving perception stacks introduce another representation shift toward bird's-eye-view and multi-view feature fusion. By the late 2010s, learned BEV representations and cost volumes became central to multi-camera and LiDAR-based perception, enabling spatial reasoning aligned with downstream planning requirements [181, 116, 182]. This reflects an increasing emphasis on representations that are geometrically meaningful yet learned. Toward the end of this era, early attempts at neural scene representations and coordinate-based networks appear. Works exploring implicit coordinate mappings and continuous scene embeddings between 2018 and 2019 lay conceptual groundwork for later developments in implicit fields and

neural rendering [119, 183]. Although these models are not yet general or scalable, they signal a growing interest in treating scenes, rather than tasks, as the primary object of representation.

Overall, Era 3 establishes learned spatial representations as the dominant paradigm. However, these representations remain largely discriminative, task-specific, and fragmented across domains. They provide powerful spatial priors for perception but stop short of forming unified, reusable world models, setting the stage for the emergence of large foundation spatial modeling in the subsequent era.

1.2.4 Era 4: Large Foundation Spatial Modeling (2019–2023)

From Task-Specific Learning to Foundation Spatial Representations. The defining shift of this era is the rise of spatial representations trained at scale as foundation-level constructs, a change of kind rather than an incremental improvement on individual spatial tasks. For the first time, spatial modeling is explicitly shaped through large-scale pretraining over diverse data sources, tasks, and modalities, rather than being optimized only within narrow task boundaries. Spatial structure is no longer learned as an auxiliary signal for detection, depth, or reconstruction, but is instead distilled into shared latent representations through massive data exposure and objective diversity. This marks a decisive representation-centric transition, where the primary goal becomes learning general-purpose spatial priors that can be transferred, adapted, and reused across downstream tasks with minimal supervision.

A key enabler of this transition is the combination of large-scale datasets, scalable architectures, and unified training objectives. Vision Transformers establish architectures capable of modeling global spatial structure over high-resolution inputs, while contrastive and generative pretraining objectives align spatial patterns across vision, language, and generation. Through training on internet-scale image-text pairs, video corpora, and synthetic data, spatial representations acquire statistical regularities that go beyond any single environment or task. As a result, space increasingly functions as a shared coordinate system and embedding substrate that supports grounding, retrieval, reasoning, and conditional generation across modalities. Spatial intelligence in this era is therefore inseparable from scale, as both data diversity and model capacity are essential for forming transferable spatial abstractions.

Another hallmark of this era is the elevation of the scene as a first-class unit of representation under large-scale learning. Neural scene representations, most notably neural radiance fields (NeRFs) [119], treat a scene as a learnable and differentiable function defined over continuous space, enabling optimization from sparse observations. Their significance lies not in implicitness itself, but in demonstrating that scene-level spatial structure can be recovered, manipulated, and reused through learning. In parallel, diffusion models introduce powerful generative priors that encode spatial coherence in latent spaces trained on massive datasets, enabling controllable synthesis of images, videos, and three-dimensional content. Together, these developments elevate space from something to be perceived into a scalable, generative, and composable substrate shaped through foundation-level pretraining.

Representative Techniques and Milestones. Before the emergence of vision-specific foundation models, a critical architectural shift had already taken place with the introduction of the Transformer [141]. Originally developed for sequence modeling in natural language processing, attention-based architectures demonstrated that large-scale pretraining could yield highly transferable and task-agnostic representations, as exemplified by models such as BERT [42]. The key insight was not tied to language itself, but to the ability of self-attention to capture global structure and support scalable representation learning. This realization set the stage for extending the same architectural paradigm to visual data, where spatial structure could likewise be learned through large-scale pretraining rather than handcrafted inductive biases. Following this, one central architectural milestone is the introduction of Vision Transformers [142] in 2020, which demonstrates that large-scale pretraining on visual data can produce transferable and globally structured spatial representations. This architectural shift lays the groundwork for foundation visual models that can be adapted across tasks

without redesigning spatial pipelines. Then, cross-modal alignment becomes a defining capability with the release of CLIP in 2021, which aligns visual representations with natural language through large-scale contrastive learning [143]. This alignment provides a universal interface for spatial semantics, enabling zero-shot transfer, retrieval, and language-conditioned spatial reasoning. Spatial representations learned in this manner escape the confines of a fixed label space and can be flexibly queried and recombined through language.

Concurrently, the emergence of LLMs marks a decisive inflection point in foundation modeling. Unlike earlier neural models trained at modest scale, LLMs demonstrate that sufficiently large-scale pretraining on diverse data can induce qualitatively new capabilities, including abstraction, composition, and long-horizon reasoning [11, 12]. Crucially, these abilities are not hard-coded but emerge from scale, data diversity, and architectural simplicity. Language becomes the first modality in which foundation models exhibit general-purpose intelligence, rapidly positioning LLMs as a universal computational substrate that extends beyond NLP into vision, robotics, simulation, and scientific domains. For spatial intelligence, this shift is pivotal: it introduces a representation space in which spatial structure, relations, and plans can be explicitly described, manipulated, and interfaced with external perceptual or physical systems, thereby elevating language from an auxiliary annotation channel to a central medium for spatial abstraction and control. Also, Chain-of-Thought (CoT) style reasoning and tool-augmented prompting begin to systematically populate spatial intelligence pipelines [78]. The key milestone is not CoT per se, but the emergence of *structured intermediate representations* induced by LLMs, such as stepwise spatial decomposition, implicit scene graphs, explicit coordinate or layout hypotheses, and program-like plans that interface with mapping, perception, or simulation modules. In practice, CoT becomes a mechanism to externalize and manipulate spatial states as compositional symbols that can be checked, revised, and grounded via perception or environment feedback, thereby tightening the representation loop from language to spatial structure and back.

In parallel, LLMs begin to serve as coordinators of scene-level reasoning rather than passive text processors. PaLM-E [56] in 2023 represents a critical milestone by integrating language models with embodied sensory inputs and robotic actions, demonstrating that a single foundation backbone can operate over language, vision, and control signals within a shared representational space. This integration highlights that scene representations must bridge symbolic reasoning and physical execution, rather than treating them as separate layers. Also, 2023 marks a clear milestone where Vision-Language-Action (VLA) [97] models crystallize as a representation-centric paradigm. The defining contribution of these systems is not embodiment itself, but the treatment of action as a first-class modality within a shared spatial latent interface. VLA models learn to map perceptual states, such as images, video, and proprioception, together with language instructions, into temporally grounded action tokens or control variables. This reframes spatial representation from a purely geometric or perceptual construct into an *action-aligned internal state*, where affordances, kinematic constraints, and task-relevant spatial variables are explicitly encoded in executable form. Enabled by large foundation backbones, this shift establishes a concrete representational bridge from scene-level spatial models to agent-level world states.

Building upon these representational advances, the foundation paradigm begins to visibly reshape embodied and interactive systems. Early generalist agent models, such as Gato [149] in 2022, signal an attempt to unify perception, action, and spatial representation within a single pretrained architecture. From 2022 to 2023, LLM- and VLM-backed embodied systems increasingly adopt foundation spatial representations as priors for online decision making in navigation and manipulation tasks [56]. High-profile humanoid and mobile robot demonstrations in 2023 further highlight this trend, integrating LLM or VLM planners with perception stacks and low-level controllers. Although many such systems remain engineering-driven rather than standardized academic benchmarks, they mark a qualitative shift: spatial representations turn from offline reconstructions into online task states that must remain coherent under motion, manipulation, and partial observability. These developments collectively signal the maturation of foundation spatial

representations into embodied world models, setting the stage for the transition toward world-centered spatial intelligence.

NeRFs [119], introduced in 2020, represent another milestone by modeling scenes as continuous functions defined over spatial coordinates and viewing directions. This formulation enables high-fidelity view synthesis and establishes a functional perspective on scene representation. Follow-up work extends this idea to signed distance functions and other continuous fields, reinforcing the notion of scene-level spatial modeling as a reusable representation primitive. Generative modeling advances rapidly during this period as well. Diffusion probabilistic models [184], introduced around 2020, provide a stable and scalable framework for learning generative priors over high-dimensional spatial signals. By 2022, large-scale text-to-image diffusion systems demonstrate that spatial layout, object relations, and scene structure can be controlled through language-conditioned latent spaces [185]. These models turn generative spatial representations into widely accessible foundation priors.

In three-dimensional representation, the introduction of 3D Gaussian Splatting [121] in 2023 significantly improves the practicality and efficiency of differentiable scene representations. While conceptually aligned with neural rendering, its impact lies in making scene-level spatial representations more scalable and deployable. This development fits naturally within the broader surge of foundation scene representations rather than defining a separate era. Closely following this trend, four-dimensional (4D) spatial modeling and generation, which explicitly incorporate temporal dynamics into scene representations, begin to rapidly prevail in 2024 and attract unprecedented attention from the community [186]. Building upon continuous scene representations and generative priors established in three-dimensional settings, 4D models aim to represent scenes as time-conditioned functions or dynamic fields that capture motion, deformation, and long-term temporal consistency. This includes emerging paradigms such as dynamic NeRF variants [187, 188], spatiotemporal Gaussian representations [189], video-based neural fields [190], and diffusion-based 4D generation models [191, 192]. The significance of this milestone lies in extending foundation spatial representations from static geometry to evolving worlds, thereby strengthening the role of space as a reusable and temporally grounded latent substrate. Although still closely tied to the foundation modeling paradigm of this era, the rise of 4D spatial representations foreshadows the subsequent shift toward world-centered spatial intelligence, where temporal evolution and agent interaction become first-class constraints.

Across domains such as remote sensing and medical imaging, large-scale pretraining and multimodal alignment also begin to emerge. Efforts to pretrain spatial representations on massive satellite imagery corpora or align medical images with textual reports reflect the spread of foundation spatial modeling beyond conventional vision benchmarks [193, 194]. Simulation and synthetic data generation further support this trend by enabling data scaling for spatial pretraining when real-world annotations are scarce [195, 196, 197].

1.2.5 Era 5: World-Centered Spatial Intelligence (2023–Present)

From Spatial Representations to World States. This era marks a decisive transition from spatial modeling toward world-centered intelligence. The defining shift lies in a fundamental redefinition of what spatial representations are required to support, well beyond the introduction of another three-dimensional encoding scheme. For the first time, spatial representations are systematically constrained to be executable, interactive, continuously updatable, and tightly coupled with agent behavior. They are no longer designed to serve isolated perception modules or one-shot generation tasks, but to function as persistent internal states that sustain long-running systems embedded in dynamic environments. As a result, representation quality is no longer judged primarily by reconstruction accuracy or visual fidelity, but by whether it can support ongoing interaction, decision making, and adaptation over extended time horizons [4, 5].

A central conceptual shift underlying this transition is the expansion of context. The dominant paradigm moves from context from self to context as world. Rather than reasoning solely over prompts, local observations, or short-horizon inputs, systems are now required to organize cognition around an evolving

world state that integrates spatial structure, temporal dynamics, physical constraints, and increasingly social context. In this view, spatial intelligence operates as an internal world state machine. It is written by perception, modified by action, maintained by memory, and queried by planning and prediction. This reframing represents a departure from representations as static input encodings toward representations as operational substrates that mediate perception, reasoning, and action in a unified loop [157, 56].

Crucially, this shift also redefines how learning itself is expected to occur. World-centered spatial intelligence cannot be acquired solely through one-off offline pretraining. Instead, representations must be shaped and continually refined through embodied interaction with the environment. Agents are expected to actively explore, test hypotheses, observe consequences, and update their internal world states accordingly. Spatial intelligence thus becomes inseparable from agency and experience. The internal representation must encode not only where things are, but how the world evolves under action, which affordances are available, and how constraints shape future possibilities. Although current systems remain at an early stage of this transition, the trajectory is clear. By treating spatial intelligence as a persistent, executable, and evolving world substrate, this era points toward a path where artificial agents can maintain coherent world models over long time horizons, learn autonomously through interaction, and ultimately approach human-level and potentially superhuman-level intelligence grounded in the structure and dynamics of the real world [150, 198, 199].

Representative Techniques and Milestones. One prominent signal of this era is the emergence of explicit world modeling frameworks that treat perception, prediction, and memory as components of a unified state transition process. Yann LeCun has proposed a series of Joint Embedding Predictive Architecture (JEPA) world models from 2022 to 2025, which reframes representation learning around predicting future world states rather than reconstructing sensory inputs, emphasizing abstraction, temporal coherence, and action relevance [151, 155, 200]. By shifting the learning objective from pixel-level fidelity to state-level consistency, JEPA positions world dynamics as the primary object of representation. Following this direction, a growing body of work between 2023 and 2024 explores world models that integrate prediction, memory, and action, progressively moving from short-horizon forecasting toward persistent and interaction-aware world state representations [152, 22, 201, 202, 203]. These efforts collectively signal a transition from predictive representation learning to explicit world-centered modeling. Within this trajectory, World Labs' Marble, launched by Fei-Fei Li's team in late 2025 as the first commercial generative world model, turns text, image, video, or coarse 3D prompts into persistent, editable, and walkable 3D environments [158]. Its significance for this era is twofold: it demonstrates that world-level spatial representations can be generated rather than only reconstructed, and that the resulting worlds persist as stable, downloadable state rather than transient observations. Marble thereby moves generative spatial modeling from research demonstrations toward deployable world-building infrastructure, reinforcing the view that spatial representations should be maintained, edited, and acted upon over long horizons.

VLA scaling further accelerates this trend. RT-2 in 2023 demonstrates that models pretrained on large-scale web data can transfer semantic and spatial knowledge directly into robotic control policies [97]. Here, spatial representation becomes action-aligned rather than geometry-first. Internal states must encode affordances and task-relevant spatial variables in forms that are immediately executable, reinforcing the notion that representation quality is measured by behavioral competence. Agent-centric frameworks also mature during this period. ReAct-style agents, introduced in 2022, formalize a tight loop between reasoning and acting, where intermediate representations explicitly externalize spatial hypotheses, plans, and state updates [21]. Voyager, released in 2023 for the Minecraft environment, further exemplifies this shift by integrating long-term memory, skill libraries, planning, and exploration into a persistent world model that evolves through interaction [150]. These systems highlight the importance of maintaining long-horizon spatial consistency rather than optimizing isolated decisions.

Simulation platforms and benchmarks increasingly align with this world-centered view. The Habitat family of embodied AI benchmarks continues to push evaluation from static perception tasks toward closed-loop interaction, navigation, and manipulation in simulated environments [94, 204, 98, 205]. This shift in evaluation methodology reinforces the requirement that spatial representations must remain coherent under motion, partial observability, and delayed feedback. From 2024 onward, interactive world models and action-conditioned video models gain prominence. These models treat the world state as a predictive object whose evolution depends explicitly on agent actions, unifying spatial prediction and control [206, 97]. Such approaches position world dynamics as first-class representational targets rather than side effects of perception. More broadly, multimodal agent frameworks emerging between 2024 and 2025 increasingly combine tool use, memory systems, and long-horizon planning. In these systems, spatial representations act as persistent world states that coordinate perception, reasoning, and action over extended time scales [93, 207, 208, 150].

From a representational standpoint, this era is characterized by the rise of world-level state representations. These representations often combine metric maps, semantic abstractions, scene graphs, and memory-augmented structures into unified internal states. Crucially, they encode not only static layout but also action-conditioned dynamics, affordances, and long-term consistency. The representation target shifts from isolated scenes to cross-scale world states that span rooms, cities, and even planetary systems. Such representations are designed to be executable, editable, and safe, enabling planning, intervention, and constraint enforcement within a single coherent framework.

Chapter 2

A Definition in Motion: Revisiting Spatial Intelligence

While spatial intelligence is rapidly emerging as a foundational paradigm in artificial intelligence; its constituent ideas are by no means new. Concepts related to space, geometry, structure, and physical interaction have long existed across multiple specialized research communities, each developing its own assumptions, abstractions, and technical formalisms. What is new is the growing realization that these previously fragmented components must now be understood as parts of a unified and evolving whole, rather than the appearance of spatial reasoning per se. This chapter revisits the notion of spatial intelligence as a moving definition, one that cannot be fixed within a single discipline, representation, or task setting, but must instead be understood through its cross-cutting roles across modern AI systems.

2.1 The Landscape of Spatial Intelligence

At its core, spatial intelligence is not a single technique, model, or task category. Rather, it constitutes a complex interdisciplinary ecosystem, spanning multiple technical paradigms, sensory modalities, scientific disciplines, and application domains. Different communities approach spatial intelligence with distinct motivations and meanings: for some, it concerns geometric consistency and localization; for others, it is about perception–action coupling, embodied reasoning, or spatial grounding of language. As a result, any attempt to define spatial intelligence in isolation risks oversimplification. Understanding spatial intelligence therefore requires acknowledging its breadth, heterogeneity, and layered structure as an evolving research ecosystem rather than a narrowly scoped subfield.

2.1.1 Beyond A Single Dimension

Spatial intelligence is fundamentally *not simple*. Unlike many AI topics that can be reasonably framed around a dominant abstraction or learning paradigm, spatial intelligence spans a wide spectrum of representational, computational, and conceptual elements. It touches almost every stage of the perception–cognition–action loop and manifests differently depending on the task, environment, and embodiment of the agent. This intrinsic complexity is not accidental; it reflects the fact that space itself is a unifying substrate of the physical world, cutting across sensory input, internal representation, and external interaction.

Historically, research on spatial problems in AI has progressed along parallel and often disconnected trajectories. Simultaneous localization and mapping (SLAM), 3D reconstruction, robotic manipulation, geographic information systems, spatial language understanding, and neural rendering all evolved largely

within their own communities, with specialized benchmarks, metrics, and assumptions. While each of these areas addressed a slice of spatial reasoning, they rarely converged on a shared conceptual framework [207]. As intelligent agents are increasingly expected to operate in open, dynamic, and embodied environments, the limitations of this fragmentation have become increasingly evident.

One major source of complexity arises from the fact that spatial intelligence cuts across *multiple technical paradigms*. It cannot be reduced to either symbolic rules or neural representations alone. Instead, it involves the coexistence and interaction of diverse computational approaches:

Cross-Technical Paradigms. Spatial intelligence spans heterogeneous technical formalisms, including:

- **Rule-based and symbolic systems**, such as topological maps, qualitative spatial calculi (*e.g.*, RCC-8 region connection, interval algebra), and logical planners, which offer interpretability and structured priors [57, 209, 76, 210].
- **Probabilistic and geometric methods**, including SLAM, state estimation, and pose graph optimization, which emphasize metric consistency and uncertainty modeling [54, 211].
- **Neural representations**, such as implicit fields, neural radiance fields (NeRF), and learned spatial embeddings, which enable continuous and data-driven modeling of complex environments [119, 121].
- **Large language/foundation models**, which introduce spatial reasoning through language, symbolic abstraction, and multimodal alignment, but often lack direct physical grounding [56, 212].

Equally important is the fact that spatial intelligence is inherently *cross-modal*. Space is not perceived through a single sensory channel, but emerges from the integration of multiple modalities, each offering partial and complementary information:

Cross-Modal Integration. Spatial intelligence operates across a diverse set of sensory and representational modalities:

- **Language**, which encodes spatial relations, instructions, and abstract descriptions (*e.g.*, "behind", "inside", "near") [213].
- **2D visual perception**, including images and video, which provide appearance-level cues and spatial layout hints.
- **3D representations**, such as point clouds, meshes, and volumetric fields, which capture geometric structure and physical extent.
- **4D spatiotemporal signals**, incorporating motion, dynamics, and temporal evolution of scenes.
- **Specialized sensing modalities**, including remote sensing, thermal imagery, radar, and spectral signals, which extend spatial perception beyond human-visible cues [193, 194].

A further, frequently overlooked axis is *spatial scale*. Cognitive research has long established that space at different scales engages different psychological processes: Montello's classic analysis distinguishes figural (table-top), vista (single view), environmental (learned through locomotion), and geographical (map-mediated) spaces [214]. The same stratification recurs in AI systems, where an agent manipulating objects on a desk, a robot navigating a building, and a model reasoning over a city or the Earth's surface demand different representations, memory structures, and update mechanisms [207]. Competence at one scale does not transfer automatically to another, which is one reason benchmark results on room-scale tasks say little about urban-scale spatial reasoning.

Cross-Scale Operation. Spatial intelligence must therefore operate across nested spatial scales:

- **Object and figural scale**, covering manipulation workspaces, shape understanding, and mental transformation of objects.
- **Scene and vista scale**, covering room-level layout, occlusion, and viewpoint reasoning.
- **Environmental scale**, covering buildings, streets, and spaces learned through movement and exploration.
- **Geographic and planetary scale**, covering cities, terrain, and Earth systems mediated by maps and remote sensing [108, 207].

Beyond modality, technique, and scale, spatial intelligence is deeply *cross-disciplinary*. Its theoretical foundations and practical realizations draw simultaneously from several scientific traditions:

Cross-Disciplinary Foundations. Spatial intelligence intersects with:

- **Computer Vision**, through scene understanding, reconstruction, and spatial perception.
- **Robotics and Embodied AI**, where spatial reasoning underlies navigation, manipulation, and interaction.
- **Geographic Information Science (GIS)**, which studies large-scale spatial representation, mapping, and spatial analytics [108, 171].
- **Natural Language Processing and Linguistics**, particularly in grounding linguistic expressions in physical space and in the cross-linguistic study of spatial reference [213, 215].
- **Psychology and Psychometrics**, which pioneered the measurement of human spatial abilities such as mental rotation and spatial visualization [71, 216].
- **Cognitive Science and Neuroscience**, which investigates human spatial cognition, cognitive maps, and embodied perception [63, 217].

Finally, spatial intelligence manifests across a wide spectrum of *application domains*, further complicating any attempt at a narrow definition:

Cross-Application Presence. Spatial intelligence is central to applications such as:

- **Embodied agents and robotics**, including navigation, manipulation, and human–robot interaction [93, 207, 97].
- **Autonomous driving and aerial systems**, where spatial prediction and planning are safety-critical [218, 219].
- **AR/VR and virtual humans**, which demand consistent spatial grounding across perception and interaction [220, 221, 222].
- **Urban computing, digital twins, and Earth systems**, operating at city and planetary scales [223, 224, 225].

Together, these dimensions highlight that spatial intelligence is not a standalone capability but a unifying layer across perception, reasoning, and action. This complexity precludes a minimal definition and instead calls for viewing spatial intelligence as a structured, evolving system that emerges across methods, modalities, and domains.

2.1.2 Disciplinary Perspectives

Although spatial intelligence is increasingly discussed as one concept, its meaning varies substantially across academic disciplines. Each field approaches spatial intelligence through its own problem formulations, representational assumptions, and evaluation criteria. As a result, what counts as spatial intelligence in one domain may only capture a partial aspect of what another domain considers essential. This part reviews

Table 2.1: How different disciplines characterize spatial intelligence: primary focus, representations, canonical tasks, and evaluation emphasis.

Discipline	Primary focus	Representations	Canonical tasks	Evaluation emphasis
Computer Vision & Graphics	Geometry recovery and view-consistent scene appearance	Depth maps, multi-view geometry, meshes, neural fields (NeRF), 3D Gaussian representations	Depth/pose estimation, 3D reconstruction, neural rendering, scene synthesis	Geometric consistency, rendering fidelity, view synthesis quality
Robotics & Embodied AI	Perception–action coupling under physical constraints	Occupancy grids, semantic maps, pose graphs, affordance maps, world models for planning	SLAM, navigation, manipulation, motion planning, closed-loop control	Task success, robustness, real-time performance, generalization in dynamic environments
GIS & Large-scale Spatial Modeling	Global-scale spatial abstraction, querying, and planning	Geodetic coordinate systems, map layers, spatial graphs, spatial databases	Route optimization, spatial analytics, trajectory analysis, urban-scale simulation	Global consistency, query accuracy, scalability, long-horizon planning utility
Psychology & Psychometrics	Measurement of individual spatial abilities and their training	Factor models of spatial ability (rotation, visualization, orientation), standardized test batteries	Mental rotation tests, paper folding, perspective taking, spatial visualization assessment	Test reliability and validity, individual differences, malleability and transfer
Cognitive Science & Neuroscience	Mechanisms of human spatial cognition and internal representations	Cognitive maps, mental imagery, spatial memory representations, neural coding hypotheses	Mental rotation, navigation studies, spatial memory and attention experiments	Behavioral validity, explanatory power, neural plausibility
Linguistics & Spatial Language	How language encodes, schematizes, and constrains spatial relations	Spatial prepositions and closed-class relations, frames of reference (intrinsic, relative, absolute)	Spatial reference resolution, cross-linguistic comparison, grounded instruction understanding	Descriptive adequacy, cross-linguistic coverage, grounding fidelity
Social & Human-centered Perspectives	Spatial behavior in social contexts and shared environments	Proxemics, social space schemas, spatial cues for interaction, situational context models	Human–robot interaction, social navigation, spatial language pragmatics, collaborative tasks	Interaction naturalness, safety, alignment with human norms and intentions

how spatial intelligence is understood from several major disciplinary perspectives, highlighting both their unique emphases and their limitations.

Computer Vision and Graphics. From the perspective of computer vision, spatial intelligence is primarily associated with the ability to infer geometric structure from visual input. Research in this area focuses on problems such as depth estimation, multi-view geometry, camera pose estimation, and 3D scene reconstruction. The core objective is to recover spatially consistent representations from 2D images or videos, often under strong geometric constraints. These representations provide powerful spatial priors, enabling reasoning about layout, scale, and relative positioning of objects.

Closely related, computer graphics emphasizes the synthesis and rendering of spatial scenes. Modern neural rendering approaches, including NeRF and 3D Gaussian Splatting, prioritize appearance fidelity, view consistency, and continuous scene representation. While these methods excel at producing visually coherent 3D environments, they typically do not model agent-centric interaction, decision-making, or task-driven

behavior. As a result, spatial intelligence in vision and graphics is often appearance-centered and offline, rather than action-oriented.

Robotics and Embodied AI. In robotics and embodied AI, spatial intelligence is traditionally defined in terms of an agent's ability to localize itself, navigate environments, and manipulate objects. This includes spatial perception, motion planning in three-dimensional space, and interaction with representations such as occupancy grids, semantic maps, and volumetric models. Here, spatial intelligence is inseparable from embodiment and control, as spatial reasoning directly informs action execution. The central concern in this discipline is not only representing space, but using spatial representations to support real-time decision-making under physical constraints. Consequently, spatial intelligence is evaluated through task success, robustness, and adaptability in dynamic environments. This action-centric perspective highlights aspects of spatial intelligence that are often absent in purely perceptual or representational frameworks.

Geographic Information Systems and Large-Scale Spatial Modeling. At a larger spatial scale, geographic information systems view spatial intelligence through the lens of structured spatial data, coordinate systems, and topological relations. GIS focuses on modeling, querying, and reasoning over large-scale geographic environments using maps, graphs, and spatial databases. These systems excel at spatial querying, route optimization, and long-term planning across city or regional scales. However, traditional GIS frameworks are rarely coupled with embodied agents or interactive learning. Spatial intelligence in this context is largely static, symbolic, and detached from perception or action. This contrast highlights a scale-dependent interpretation of spatial intelligence, where global consistency and abstraction take precedence over local interaction.

Psychology and Psychometrics. Long before AI adopted the term, differential psychology treated spatial ability as a measurable dimension of human intelligence. Shepard and Metzler's mental rotation experiments showed that the time needed to compare rotated 3D shapes grows linearly with their angular disparity, providing the first chronometric evidence of analog spatial representation [71], and the factor-analytic tradition from Thurstone's space factor to Carroll's broad visuospatial factor established spatial ability as a distinct, testable construct [226]. The term *spatial intelligence* itself was popularized by Gardner's *Frames of Mind* (1983), which listed it among seven human intelligences and defined it as the capacity to perceive the visual world accurately, to perform transformations upon one's initial perceptions, and to re-create aspects of visual experience even in the absence of the relevant stimuli; the construct, however, predates that label by half a century. Two findings from this literature matter directly for AI. First, spatial skills are malleable: a meta-analysis across hundreds of training studies shows they improve with practice and transfer across tasks [216], suggesting spatial competence is acquired rather than fixed. Second, the psychometric task inventory, covering mental rotation, perspective taking, and spatial visualization, has become a direct template for probing spatial understanding in multimodal foundation models, so this measurement tradition now feeds AI benchmarking rather than sitting apart from it [227].

Cognitive Science and Neuroscience. In cognitive science and neuroscience, spatial intelligence is studied as an emergent property of interacting neural systems rather than as an engineered representation. Research has identified key brain regions associated with spatial cognition, including the hippocampus for spatial memory and navigation (place cells, and grid cells in entorhinal cortex), the parietal cortex for spatial attention and transformation, and occipito-parietal networks for visuospatial processing [51, 217]. From this perspective, spatial intelligence arises from neural mechanisms supporting cognitive maps, mental imagery, and spatial reasoning; it is primarily internal and cognitive, centered on perception and imagination rather than external interaction.

Linguistics and Spatial Language. A further disciplinary lens comes from linguistics, which asks how language encodes and constrains spatial thought. Talmy's analysis of how language schematizes space [228] and Landau and Jackendoff's account of the asymmetry between rich object naming and the small

closed class of spatial prepositions [213] established that linguistic descriptions compress spatial structure into coarse, topology-like relations. Levinson's cross-linguistic work further showed that languages differ systematically in their spatial frames of reference, distinguishing intrinsic, relative, and absolute systems [215]. This vocabulary has migrated directly into AI evaluation, where egocentric versus allocentric reference handling is now a standard axis of spatial benchmarks for multimodal models, making the linguistics of space an operative rather than merely historical influence.

Social and Human-Centered Perspectives. Beyond physical geometry, spatial intelligence also carries meaning at the social and mental levels. Human beliefs, emotions, and behaviors are situated within social environments that are themselves spatially structured. From this viewpoint, spatial intelligence encompasses the ability to reason about others in shared spaces, interpret spatial cues in social contexts, and align internal mental states with external environments. This perspective extends spatial intelligence beyond physical navigation toward a unified model of the physical and mental world. It points to broader scientific and societal opportunities, including AI systems that operate in unstructured real-world settings and interact naturally with humans. In this sense, spatial intelligence supports a joint modeling of physical space and human cognition, serving as a bridge between external environments and internal mental states.

2.1.3 Toward A Unified View

For decades, research on spatial reasoning and spatial representation has progressed from markedly different perspectives across multiple subfields. These efforts often evolved in isolation, lacking a shared vocabulary, common research objectives, or directly comparable evaluation benchmarks. As a result, techniques developed in areas such as SLAM, 3D vision, scene modeling, world modeling, and embodied planning frequently addressed overlapping problems without converging toward a unified conceptual framework. Spatial intelligence emerges against this backdrop as a response to fragmentation, offering a higher-level abstraction that can organize and align these historically disjoint research trajectories.

Under the emerging spatial intelligence paradigm, a set of shared demands and core concepts has begun to surface. A central driving force behind this convergence is the growing need for agents that exhibit spatial consistency and physical groundedness, and that can operate continuously within dynamic environments. Such agents must integrate perception, representation, and action in a coherent manner. This requirement has catalyzed the fusion of visual perception with geometry-aware modeling, as seen in navigation-oriented neural rendering approaches and interactive NeRF-based systems. Research at the intersection of neural scene representation and active planning has given rise to a new generation of 3D world models that are not only reconstructive, but also executable and actionable. Similarly, vision-language models are increasingly extended to operate in three-dimensional environments, enabling multimodal agents to localize, describe, and interact with objects within voxelized or volumetric scenes.

At the conceptual level, a set of shared primitives is gradually taking shape. Spatial representations, whether instantiated as voxel grids, signed distance fields, neural feature volumes, or topological graph structures, are becoming central components across domains ranging from autonomous driving and AR/VR to assistive robotics. Fundamental spatial relations, such as containment, adjacency, support, and traversability, are being formalized and embedded into reasoning modules, scene graph systems, and relational neural networks. These abstractions provide a common representational substrate that enables knowledge transfer and system interoperability across tasks and modalities.

We note that the research community has yet to settle on a single taxonomy for this emerging field. Recent surveys and position papers organize spatial intelligence along markedly different axes: by task family and benchmark [229], by the cognitive functions that video-based spatial tasks exercise [227], by spatial scale from embodied agents through cities to Earth systems [207], or, in Li's position essay, by the capabilities a world model must possess, namely being generative, multimodal, and interactive [5]. These framings overlap without coinciding, and none has achieved consensus status. This handbook's representation-centric

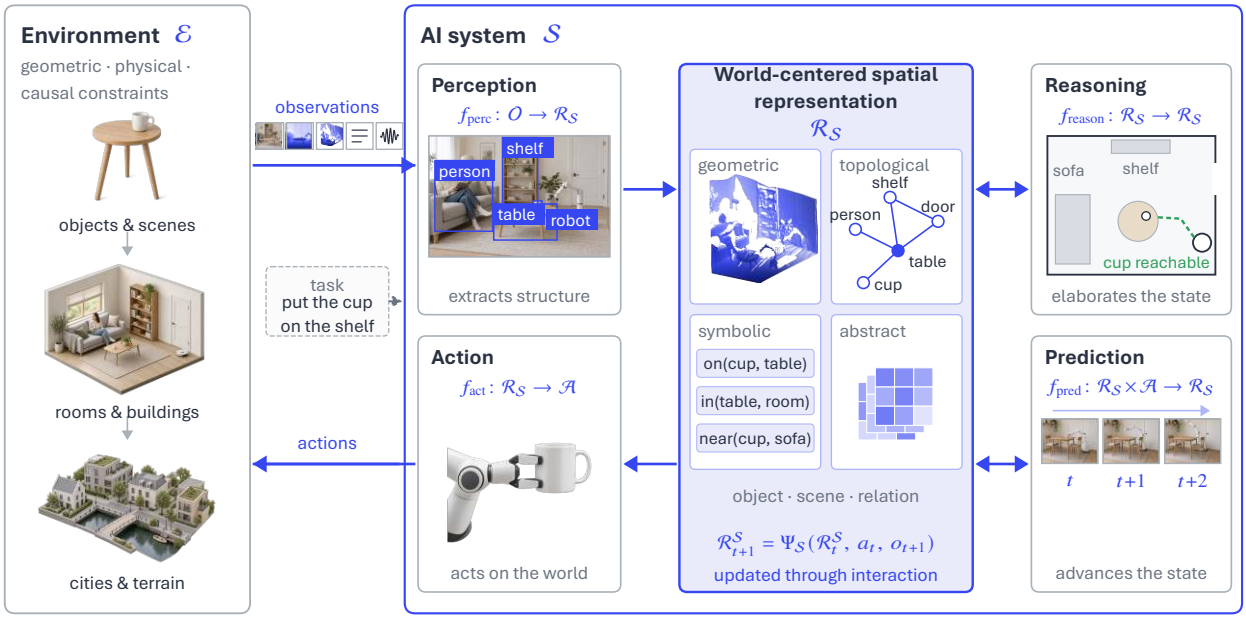


Figure 2.1: Spatial intelligence as a closed loop between an AI system $\mathcal{S}$ and its environment $\mathcal{E}$. Perception writes observations into a world-centered spatial representation $\mathcal{R}_S$ maintained inside the system; reasoning elaborates the current state, prediction advances it under a candidate action, and embodied action reads it to act on the environment, whose response is observed in turn. Each module is labeled with its operator from Section 2.2.2.

organization, which follows how spatial representations are built, used, and acted upon, is our deliberate choice among these alternatives: it is the axis along which the historical evolution of Section 1.2 unfolds most naturally, and the one that lets fragmented communities be compared on common ground.

Taken together, these developments indicate that spatial intelligence is no longer confined to any single domain. It is evolving into a unifying functional substrate that spans perception, reasoning, and interaction; bridges vision, language, and action; and connects symbolic representations with neural models. The unification runs deeper than shared technique. Spatial intelligence aligns and integrates multiple disciplines, including computer vision, robotics, psychology, cognitive science, linguistics, geospatial systems, human-computer interaction, and neuro-inspired AI, and through this alignment it functions as the connective tissue that enables the next generation of generalist AI systems to perceive the world, reason about it, and act within it.

2.2 Defining A Spatial Intelligence Framework

Given the inherently hybrid and interdisciplinary nature of spatial intelligence, it becomes necessary to revisit and refine what we mean by "spatial intelligence" in a precise and unified manner. Rather than treating spatial intelligence as a loosely defined collection of capabilities, we aim to articulate a coherent conceptual framework that can accommodate its diverse manifestations across systems, tasks, and environments. In this section, we therefore propose an explicit definition of spatial intelligence and analyze its core properties in a structured way.

Definition (Spatial Intelligence). *Spatial intelligence is the ability of an AI system to construct, maintain, and manipulate spatial representations of the physical world or environment that support perception, reasoning, prediction, and embodied action, to accomplish tasks within a given environmental context.*

This definition emphasizes several essential aspects. First, spatial intelligence is grounded in representations of the world rather than isolated sensory inputs. Second, it is functional rather than static, supporting perception, reasoning, prediction, and action. Third, it is inherently situated, meaning that spatial intelligence only manifests meaningfully within an environment where tasks are defined and executed. In the following, we unpack this definition by examining its key attributes.

2.2.1 Core Elements

2.2.1.1 Parties

At a minimum, spatial intelligence involves two fundamental parties: an AI system and the physical world or environment in which it operates. Neither can be meaningfully defined without the other.

AI System. The AI system refers to any computational entity capable of perception, reasoning, and decision-making. This includes, but is not limited to, large language models, multimodal foundation models, world models, embodied agents, and robotic systems. Crucially, spatial intelligence is not tied to a specific architecture. Instead, it characterizes a class of capabilities that may be realized through different model families, training paradigms, and system designs [230, 231].

Physical World or Environment. The environment denotes the external context in which the AI system is situated. This typically refers to the physical world governed by geometric, physical, and causal constraints. In some cases, the environment may also include social or human-centric settings, where spatial structure interacts with social norms, human intent, and collective behavior [232]. Spatial intelligence arises precisely from the interaction between the AI system and such structured environments.

2.2.1.2 Spatial Representation

A core component of spatial intelligence lies in the ability to form, maintain, and manipulate world-centered spatial representations. These representations serve as structured world models that are compositional, generalizable, and reusable across tasks.

Spatial representations may vary along several dimensions:

- **Granularity.** Representations can operate at different spatial scales, including object-centric representations, scene-centric representations, and relational representations capturing relative geometry or topology.
- **Modality.** Spatial information may be derived from heterogeneous sensory channels, including pixels, grid maps, point clouds, voxel grids, language descriptions, audio signals, spectral measurements, radar inputs, as well as tactile and force feedback.
- **Level of Abstraction.** Spatial representations may be geometric, such as metric coordinates and surfaces, topological, such as connectivity graphs, or semantic and symbolic, such as object categories and relational predicates.
- **Concrete versus Abstract.** Representations may directly encode observable properties of the physical world or serve as abstract internal models that summarize, compress, or reinterpret spatial structure for reasoning and planning.

Importantly, spatial intelligence does not require committing to a single representational form. Instead, it demands the ability to align and translate across multiple representational levels in a world-centered manner.

2.2.1.3 Dynamic Interaction

Spatial intelligence is not a static property acquired through passive observation. Rather, it emerges through continuous and dynamic interaction with the environment. This interaction includes navigation, manipulation, exploration, and embodied perception. An intelligent system must ground its spatial understanding in embodied experience, for example by learning the structure of a room through movement, or improving depth perception through active viewpoint selection. Interaction also extends to socially situated behaviors. Maintaining appropriate interpersonal distance, coordinating actions with humans, or navigating shared spaces all require spatial understanding informed by social context. As a result, spatial representations must be executable and continuously updated through perception-action feedback loops. We further decompose dynamic interaction into several functional components.

Perception. Perception refers to the extraction of spatial structure from multimodal sensory inputs. This includes recognizing objects and scene elements, estimating distances, identifying spatial layouts, and localizing entities within an environment.

Reasoning and Planning. Spatial reasoning involves inference over spatial relations and dynamics. This includes trajectory prediction, path planning, occlusion reasoning, spatial causality, temporal change modeling, relational inference, and logical or commonsense reasoning about space. Such reasoning spans both low-level problems, such as shortest-path computation, and high-level abstractions, such as understanding how rearranging objects alters the affordances of a space.

Predicting. Prediction in spatial intelligence operates along two axes. One concerns forecasting future spatial states or events in a manner consistent with physical laws and causal structure. The other concerns generative capabilities, such as producing spatially aware images, videos, 3D or 4D scenes, and motion trajectories.

Embodied Action. Embodied action denotes the capacity to interact with physical or simulated environments while continuously pursuing goals. This includes pose adjustment, navigation, grasping, and object manipulation. Typical scenarios include domestic robots and autonomous driving systems. Such action requires tight integration of perception, reasoning, planning, and prediction, while aligning internal spatial models with task objectives, human intent, and social context.

What Constitutes a Strong Spatially Intelligent System

A well-designed spatially intelligent AI system should satisfy the following criteria:

- **High-quality world representation.** It must acquire a rich, consistent, and generalizable spatial representation of the world that captures geometry, relations, and structure across scales.
- **Dynamic interaction capability.** It should support accurate perception, reasoning, prediction, and embodied action through continuous interaction with the environment.
- **Alignment between model, task, and environment.** The internal spatial model must remain aligned with external task objectives, environmental constraints, and feedback signals during execution.

2.2.2 Formal Characterization

To make the definition more precise, we provide a lightweight formalization that captures the core components of spatial intelligence without committing to a specific implementation or learning paradigm.

Let $\mathcal{E}$ denote the environment in which an AI system is situated, representing the physical or simulated world governed by spatial, physical, and causal constraints. Let $\mathcal{S}$ denote the AI system or agent. We assume that tasks are defined with respect to the environment and are drawn from a task space $\mathcal{T}$.

World-Centered Spatial Representation. We define a world-centered spatial representation as a system-induced mapping from the environment to an internal state space:

$$\Phi_{\mathcal{S}} : \mathcal{E} \to \mathcal{R}_{\mathcal{S}}, \tag{2.1}$$

where $\mathcal{R}_{\mathcal{S}}$ denotes the internal spatial representation space maintained by the AI system $\mathcal{S}$. The representation $\Phi_{\mathcal{S}}(e) \in \mathcal{R}_{\mathcal{S}}$ may take various forms, including geometric maps, scene graphs, implicit fields, symbolic layouts, or learned latent embeddings. Crucially, $\mathcal{R}_{\mathcal{S}}$ is not a raw sensory space, but a structured and system-dependent representation space that supports compositionality, generalization, and reuse across tasks.

Functional Use of Spatial Representations. Spatial intelligence is not defined by representation alone, but by how representations are functionally manipulated by the AI system. We characterize this through a family of system-conditioned operators acting on $\mathcal{R}_{\mathcal{S}}$:

$$\mathcal{F}_{\mathcal{S}} = \{f^{\mathcal{S}}_{\text{perc}}, f^{\mathcal{S}}_{\text{reason}}, f^{\mathcal{S}}_{\text{pred}}, f^{\mathcal{S}}_{\text{act}}\}, \tag{2.2}$$

corresponding to perception, reasoning, prediction, and embodied action. Each operator is defined relative to the system $\mathcal{S}$:

$$f^{\mathcal{S}}_{\text{perc}} : \mathcal{O} \to \mathcal{R}_{\mathcal{S}}, \tag{2.3}$$

$$f^{\mathcal{S}}_{\text{reason}} : \mathcal{R}_{\mathcal{S}} \to \mathcal{R}_{\mathcal{S}}, \tag{2.4}$$

$$f^{\mathcal{S}}_{\text{pred}} : \mathcal{R}_{\mathcal{S}} \times \mathcal{A} \to \mathcal{R}_{\mathcal{S}}, \tag{2.5}$$

$$f^{\mathcal{S}}_{\text{act}} : \mathcal{R}_{\mathcal{S}} \to \mathcal{A}, \tag{2.6}$$

where $\mathcal{O}$ denotes observations and $\mathcal{A}$ denotes the action space. The two representation-to-representation operators play distinct roles: $f^{\mathcal{S}}_{\text{reason}}$ elaborates the *current* world state by inferring relations and constraints already implicit in it, whereas $f^{\mathcal{S}}_{\text{pred}}$ advances the state *forward in time*, rolling out how the world would evolve under a hypothesized action (with a null action recovering passive forecasting).

Task-Oriented Intelligence. Given a task $t \in \mathcal{T}$, spatial intelligence manifests as the ability of the AI system $\mathcal{S}$ to select and execute actions based on its internal spatial representation:

$$\pi_{\mathcal{S}} : (\mathcal{R}_{\mathcal{S}}, t) \to \mathcal{A}, \tag{2.7}$$

such that actions are grounded in the system's current world representation and lead to task completion within the environment. This formulation highlights that spatial intelligence is inherently situated and agent-dependent: the same environment may induce different behaviors depending on the internal representation and decision mechanisms of $\mathcal{S}$.

Dynamic Interaction and World State Evolution. Finally, spatial intelligence requires representations to be dynamically updated through interaction. We model the evolution of the internal world representation of system $\mathcal{S}$ as:

$$\mathcal{R}^{\mathcal{S}}_{t+1} = \Psi_{\mathcal{S}}(\mathcal{R}^{\mathcal{S}}_{t}, a^{\mathcal{S}}_{t}, o_{t+1}), \tag{2.8}$$

where $a^{\mathcal{S}}_{t} \in \mathcal{A}$ is the action selected by system $\mathcal{S}$ and $o_{t+1} \in \mathcal{O}$ is the subsequent observation. It emphasizes that spatial representations are executable, interactive, and continuously revised by the system, rather than static encodings of the environment.

This formalization clarifies that spatial intelligence is not tied to a specific modality, algorithm, or architecture. Instead, it is defined by the existence of a system-induced, world-centered spatial representation space $\mathcal{R}_{\mathcal{S}}$ and the ability of the AI system $\mathcal{S}$ to manipulate, update, and exploit this representation to support perception, reasoning, prediction, and embodied action within an environment.

2.2.3 Compatibility with Existing Research and Paradigms

The proposed framework is intentionally designed to be inclusive rather than prescriptive. Rather than introducing a new task taxonomy or replacing existing formulations, it provides a unifying lens through which a wide range of spatial intelligence-related tasks, domains, and research traditions can be consistently described. From this perspective, most existing work can be understood as instantiating or emphasizing only a subset of the core elements defined in our framework, often driven by domain-specific constraints, data availability, or evaluation protocols.

A large body of research in computer vision, graphics, and extended reality focuses primarily on the *representation layer* of spatial intelligence. Tasks such as 3D reconstruction, neural rendering, scene synthesis, video generation, and emerging 4D spatiotemporal modeling concentrate on learning expressive spatial representations $\mathcal{R}_{\mathcal{S}}$, often with little or no emphasis on action, task execution, or long-term interaction. Within our framework, these approaches can be viewed as developing increasingly powerful instantiations of world-centered spatial representations, while largely abstracting away the agent $\mathcal{S}$ and the functional operators that connect representation to behavior. Their contributions lie in advancing the expressiveness, continuity, and generative capacity of spatial representations, even if they remain largely offline or perception-centric.

Robotics, embodied AI, and embedded systems emphasize a different subset of the framework, placing primary focus on action grounding and closed-loop interaction. Vision-Language-Action models, navigation agents, and manipulation systems operationalize the mapping from internal representations to actions, often under real-time and physical constraints. However, in many cases, the underlying spatial representations remain task-specific, shallow, or tightly coupled to particular embodiments. From the perspective of our framework, these systems instantiate the functional operators and dynamic interaction components, while relying on relatively limited or implicit world representations. This explains why many embodied systems exhibit strong short-horizon competence but struggle with long-term consistency, world editing, or cross-task transfer.

Classical and modern SLAM systems provide a complementary example. SLAM explicitly targets the construction and maintenance of a world-centered spatial representation under uncertainty, typically in the form of metric maps, pose graphs, or hybrid geometric-topological structures. Within our framework, SLAM can be understood as a specialized instantiation of the representation update process, with strong guarantees on geometric consistency and state estimation. At the same time, SLAM systems often remain agnostic to high-level reasoning, semantic abstraction, or task-conditioned decision making, highlighting again that they occupy a specific slice of the broader spatial intelligence spectrum rather than the full stack.

Viewed through this unified lens, existing spatial intelligence research does not fragment into incompatible paradigms, but instead forms a landscape of partial realizations. Different communities emphasize representation learning, perception, reasoning, prediction, or action to varying degrees. Our framework provides a common conceptual structure that makes these emphases explicit, clarifies their limitations, and reveals how advances in one area can be systematically integrated with others. In this sense, the framework is not a replacement for existing approaches, but a scaffold for understanding how they collectively contribute to, and fall short of, world-centered spatial intelligence.

Chapter 3

Scope and Structure of This Handbook

This chapter provides a structural overview of the handbook. We clarify the scope, goals, and organization of the volume, and explain how the different parts jointly form a coherent research landscape for spatial intelligence. Specifically, we introduce the core research pillars covered in this handbook, outline their relationships, and offer guidance on how readers from different backgrounds may navigate the content.

3.1 The Research Pillars of Spatial Intelligence

We outline the core research pillars that constitute the structural backbone of this handbook. They traverse the full pipeline from spatial representation and perception to reasoning and generative capabilities, and ultimately extend into concrete real-world applications. Each pillar aligns closely with a corresponding Part of the handbook, serving as a thematic axis for both foundational methods and emerging challenges in spatial intelligence.

- **Representation (Part II: Spatial-aware Representation)**
  This pillar investigates how spatial information is encoded and represented across modalities and dimensions. We begin with single-modality representations, including 2D (*e.g.*, visual frames), 3D (*e.g.*, point clouds, meshes), and 4D (*e.g.*, dynamic scenes, spatio-temporal fields), as well as non-visual modalities such as language and audio. Particular attention is given to cross-modal alignment and fusion mechanisms that unify heterogeneous data sources. These representational strategies provide the necessary structural foundation for downstream reasoning, manipulation, and generation.
- **Perception (Part III, Section 6: Spatial Perception)**
  This pillar focuses on how intelligent systems extract structured knowledge from spatial environments. Topics include object detection, semantic and instance segmentation, depth and surface normal estimation, visual odometry (VO), and ego-motion prediction. These perception modules translate raw sensory inputs into geometric or semantic abstractions, enabling agents to comprehend and navigate the physical world.
- **Reasoning (Part III, Section 7: Spatial Reasoning)**
  Reasoning in spatial intelligence spans multiple paradigms: symbolic reasoning over geometric primitives or logical propositions; geometric reasoning over shapes, topologies, and configurations; neural reasoning using deep architectures; and LLM-driven spatial inference based on large-scale pretrained priors. These methods enable agents to interpret relations, deduce spatial constraints, and plan in dynamic settings.

- **Generation (Part III, Section 8: Spatial Generation)**
  This pillar examines the generative side of spatial intelligence. It encompasses tasks such as 2D/3D scene synthesis, controllable image and video generation, trajectory and motion prediction, and dynamic world creation. These generative processes are often guided by spatial prompts, semantic layouts, or agent intentions, forming a bridge between structured knowledge and creative capability.

- **Applications (Part IV: Applications)**
  Finally, the application pillar illustrates how the above methods power real-world systems. Domains include embodied agents and robotics (*e.g.*, navigation, grasping), autonomous driving and aerial agents, warehouse and wearable AI systems, virtual and augmented reality (XR), geospatial AI (*e.g.*, urban planning, satellite analysis), biomedical and scientific computing (*e.g.*, 3D medical reconstruction), and intelligent assistants. These chapters emphasize system-level integration and highlight how spatial AI capabilities translate into societal impact.

3.2 Goal of the Handbook: Charting a Research Ecosystem



The primary goal of this handbook is not to present a catalog of isolated techniques, but to chart an emerging research ecosystem centered on spatial intelligence. As established in earlier chapters, spatial intelligence is neither a single method nor a narrowly scoped task. Rather, it is a unifying paradigm that connects representation, perception, reasoning, generation, and action through structured models of the world.

From Isolated Methods to an Integrated Landscape. Much of the existing literature on spatial intelligence is fragmented. Some surveys focus deeply on individual subfields, such as SLAM, neural rendering, or 3D generation, while others adopt a high-level perspective that lacks technical grounding. This handbook aims to bridge this gap by organizing spatial intelligence as an interconnected landscape, where advances in one component are explicitly related to others. Our objective is to reveal how seemingly distinct research threads jointly contribute to world-centered intelligence.

Spatial Intelligence as a Connective Substrate. A central thesis of this handbook is that spatial intelligence functions as a connective substrate for next-generation AI systems. It provides the structural glue that links perception with reasoning, language with geometry, and planning with embodied action. By framing spatial intelligence in this way, the handbook draws connections across computer vision, robotics, simulation, geospatial systems, cognitive science, and neuro-inspired AI. This perspective enables a shared conceptual vocabulary through which contributions from different communities can be meaningfully aligned.

Identifying Gaps and Open Tensions. Beyond synthesizing existing work, this handbook deliberately highlights unresolved gaps in the current research landscape. These include the disconnect between symbolic and neural spatial representations, the limited grounding of perception in actionable world models, and the difficulty of integrating spatial reasoning into long-horizon embodied agents. By making these tensions explicit, we aim to clarify where current paradigms fall short and where future breakthroughs are most likely to emerge.

A Blueprint for an Open Research Program. Ultimately, this handbook is intended as a blueprint for an open and evolving research program. Rather than proposing a closed taxonomy or a definitive solution, it offers a structured lens through which spatial intelligence can be studied, extended, and reimagined. We invite readers not only to learn from the systems and methods surveyed here, but also to rethink spatial intelligence as a first-class organizing principle in the design of generalist intelligent agents.

3.3 How to Read This Handbook



IVEN the interdisciplinary nature of spatial intelligence, which spans AI, computer vision, robotics, graphics, GIS, and cognitive science, it is crucial to provide tailored reading paths that help each type of reader gain the most value from this handbook.

▶ **For Beginners and Students** (e.g., *undergraduates, junior graduate students, newcomers to spatial AI)*

- Begin with **Part I: Introduction and Background** (Chapters 1–3) to grasp the motivation, historical evolution, and research scope.
- Then proceed to **Part II: Spatial-aware Representation** to build a solid understanding of core representations in 2D, 3D, and 4D, including geometry and topology.
- Optionally skim **Part IV: Applications** to get a glimpse of real-world use cases, such as robotics, autonomous driving, and AR/XR.

▶ **For Researchers and Academics** (e.g., *PhD students, postdocs, faculty, or interdisciplinary scholars in AI-related fields)*

- Focus on **Part III: Spatial Intelligence, from Perception to Generation**, which includes chapters on:
  - Spatial perception, from detection and segmentation to depth and odometry
  - Symbolic, geometric, neural, and LLM-driven spatial reasoning
  - Generative spatial models for scenes, videos, and worlds
- Also study **Part IV: Applications** for cross-domain synergies, particularly in GeoAI, healthcare, urban simulation, and embodied cognition.
- Consider **Part V: Discussions and Futures** for open problems, benchmarking gaps, and research directions.

▶ **For System Developers and Engineers** (e.g., *practitioners building simulation environments, digital twin platforms, or perception pipelines)*

- Prioritize chapters in **Part III (Generation)** and **Part IV (Applications)** that introduce:
  - Modular spatial generation frameworks
  - Multi-agent simulation engines and control loops
  - Deployed system domains (*e.g.*, robotic manipulation, warehouse automation, wearable AI, aerial agents)
- Use **Part V: Discussions and Futures** to anticipate where deployment bottlenecks and benchmark gaps lie.

▶ **For Cross-disciplinary Explorers** (e.g., *scholars in architecture, biology, geoscience, or cognitive science)*

- Start with **Part I (Introduction and Background)** to acquire shared conceptual grounding.
- Dive into **Part IV (Applications)** for domain-aligned chapters and illustrative case studies.
- Refer to **Part II (Representation)** to gain a technical foothold in spatial structures and encodings.

Overall, this handbook is modular in structure and can be read either linearly or selectively depending on the reader's focus. The chapter cross-references are designed to aid flexible navigation and long-term use.

Part II

Spatial-aware Representation: Building Blocks of Space

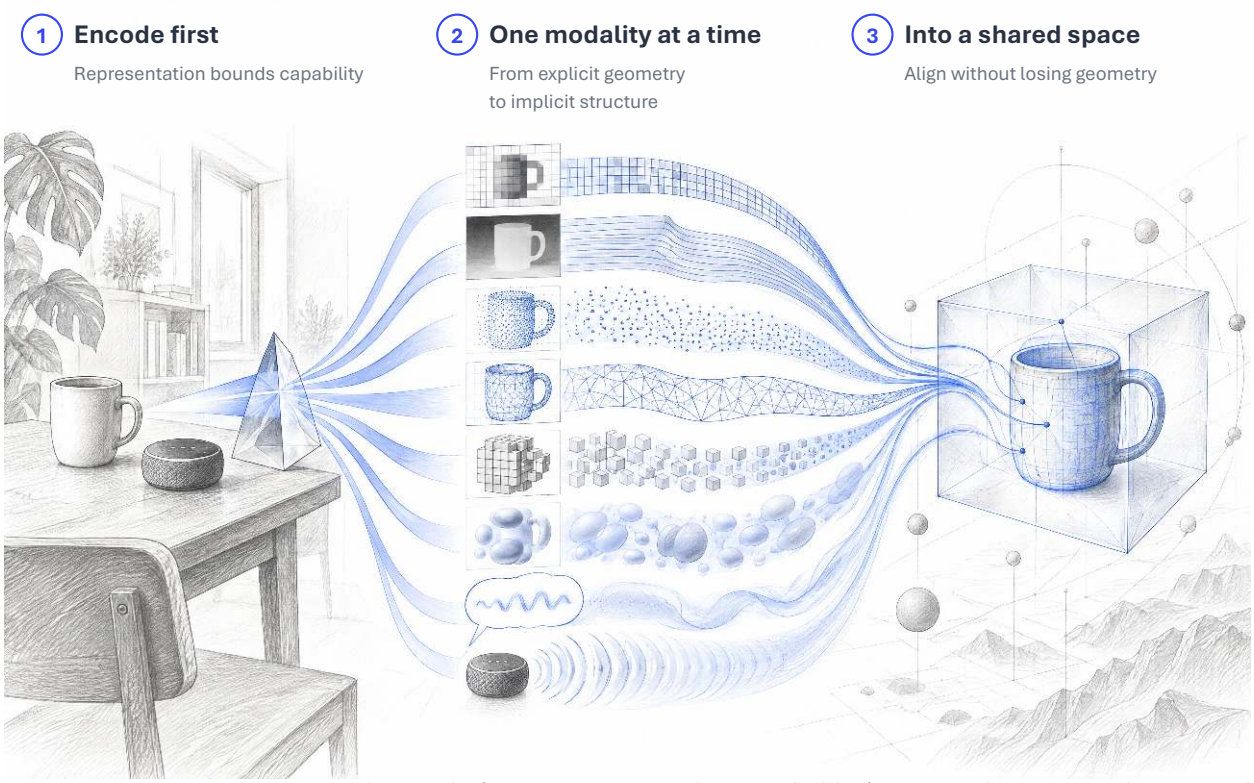


Figure II: Part II in one view. (1) Encode first: a scene must be encoded before it can be perceived, reasoned about, or acted on. (2) One modality at a time: the same object as pixels, depth, points, meshes, voxels, and Gaussians, and then as language and sound, from explicit geometry to implicit structure (Chapter 4). (3) Into a shared space: the separate encodings are aligned in one space without losing the geometry that makes them spatially meaningful (Chapter 5).

Everything a spatially intelligent system does, perceiving a scene, reasoning over it, generating or acting within it, passes through some representation of space. That representation is upstream of the capability: what a system can perceive is bounded by what its encoding can express, and what it can reason about is bounded by what that encoding makes accessible. Before the later parts turn to perception, reasoning, and generation, this part asks the prior question of how spatial structure is captured at all, how the geometry, topology, and semantics of an environment come to be held in a form a machine can compute with.

The question is approached in two movements. The first takes each modality on its own terms, asking how images, video, 3D scenes, language, and sound each encode spatial structure and what each renders explicit or leaves implicit. The second asks how these separate encodings are brought into a shared space, so that signals as different as a pixel grid and a sentence can be aligned without losing the geometry that makes them spatially meaningful. The order is deliberate, since a modality must be understood by itself before its representation can be fused with others, and it is in that fusion that the partial, sensor-bound view offered by any one signal begins to reinforce the rest. Together they form the representational substrate on which the remainder of the handbook is built.

Chapter 4

Single-Modal Spatial Representation

Single-modal spatial representations ask how much of a scene's spatial structure can be recovered from one modality alone, without recourse to cross-modal correspondence. The question matters because every multimodal system is ultimately built from unimodal encoders, and the spatial content a fused representation can carry is bounded by what each modality already captures on its own.

What separates the modalities is not whether they carry spatial information but how overtly they carry it. Vision presents geometry almost directly, arranging appearance and shape over a spatial grid or the volume a scene occupies. Language and sound hold the same structure only obliquely, the one as symbolic relations among named entities, the other as the physical trace that propagation leaves on a signal. Others still are tied to a particular sensor or task, trading generality for fidelity to the data they were built around. This chapter proceeds along that gradient, from the most explicit geometric encodings of space toward the most implicit, examining each modality on its own terms before the question of how they combine is taken up.

4.1 2D

The 2D representations are the most direct and information-dense entry point into spatial intelligence: images and videos provide a regular, discretely sampled grid over the visual field, on top of which a family of structured maps can be defined to expose geometric and semantic properties of the underlying scene. We organize this section around seven such representations (Figure 4.1): the raw *image/video tensor* itself, and six derived maps: *segmentation*, *depth*, *surface normal*, *edge*, *optical flow*, and *2D keypoint* representations. All seven share the image grid as their domain, which makes them straightforward to fuse with one another and to consume with convolutional or transformer backbones; what differs is what each representation *exposes*: appearance, semantics, geometry, boundaries, motion, or sparse structure.

4.1.1 Image and Video Representations

An image can be modeled as a discrete sampling of a continuous visual signal over a 2D spatial domain. Formally, a digital image is represented as a function

$$I : \Omega \to \mathbb{R}^C, \quad \Omega \subset \mathbb{R}^2, \tag{4.1}$$

where Ω denotes the image grid, C is the number of channels (*e.g.*, $C = 1$ for grayscale, $C = 3$ for RGB), and each pixel value encodes intensity or color information at a discrete spatial location [233, 234]. In practice, images are stored as finite tensors $I \in \mathbb{R}^{H \times W \times C}$, where H and W denote image height and width.

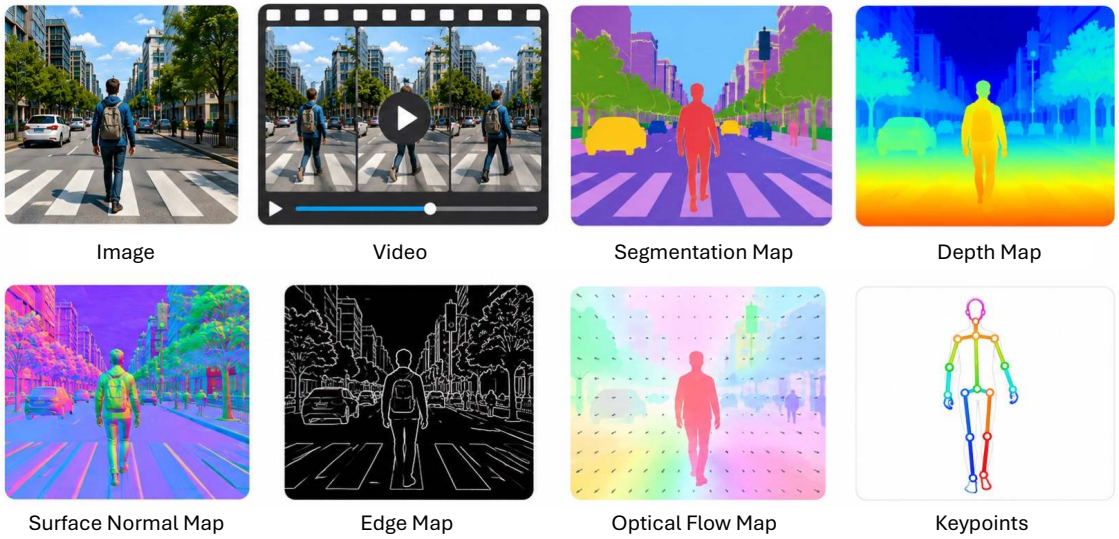


Figure 4.1: The seven 2D representations of this section, all defined over the image grid $\Omega \subset \mathbb{R}^2$: image/video, segmentation, depth, surface normals, edges, optical flow, and keypoints.

A video extends this representation by introducing a temporal dimension. Given a sequence of frames sampled over time, a video can be written as

$$V : \Omega \times \mathcal{T} \to \mathbb{R}^C, \tag{4.2}$$

or equivalently as a 4D tensor $V \in \mathbb{R}^{T \times H \times W \times C}$, where T denotes the number of frames [234]. This formulation preserves spatial structure within each frame while enabling the modeling of temporal dynamics across frames. From a computational perspective, images and videos are treated as structured arrays of numerical values that can be efficiently processed using linear algebra operations and convolutional kernels. This regular grid structure enables weight sharing, local receptive fields, and efficient parallel computation, forming the basis of CNNs and their spatiotemporal extensions [235, 236]. In modern deep learning systems, image and video tensors may undergo normalization, resizing, patch embedding, or tokenization before being fed into neural architectures, including CNNs and vision transformers [142, 237]. For video inputs, spatiotemporal extensions either inflate 2D kernels into 3D convolutions [238, 239] or extend self-attention across space and time, as in the family of video transformers [240, 241].

4.1.2 Segmentation Maps

A segmentation map is a semantic representation that assigns a label or a mask to each pixel, thereby partitioning visual data into meaningful regions such as objects, object parts, or background categories [234, 242]. For a single image defined on $\Omega \subset \mathbb{R}^2$, segmentation can be written as a mapping

$$S : \Omega \to \mathcal{L}, \tag{4.3}$$

where $\mathcal{L}$ is a set of labels. In *semantic segmentation* (SS), $S(u, v)$ denotes the class label of pixel (u, v). In *instance segmentation* (IS), the output is a set of masks $\{M_k\}_{k=1}^K$ with associated class labels $\{\ell_k\}$, where each $M_k : \Omega \to \{0, 1\}$ corresponds to an object instance [127, 131]. *Panoptic segmentation* (PS) unifies SS and IS by predicting a single, non-overlapping labeling of pixels that distinguishes both "stuff" and countable object instances [243].

These tasks extend naturally to videos. Given a sequence of frames $\{I_t\}_{t=1}^T$, *video semantic segmentation* (VSS) predicts per-frame semantic labels $\{S_t\}$, while *video instance segmentation* (VIS) predicts instance masks

with temporal consistency across frames, often requiring instance association (tracking) over time [242]. *Video panoptic segmentation* (VPS) further combines VSS and VIS, producing temporally consistent panoptic outputs and is commonly evaluated with metrics that account for both segmentation quality and tracking consistency [242]. Closely related settings include *video object segmentation* (VOS) [244], which segments foreground objects across time under semi-supervised or unsupervised protocols, and *referring* segmentation, which predicts masks grounded by natural language expressions in images (RIS) [245] or videos (RVOS) [246]. Segmentation maps are typically formed by dense prediction models that produce either hard labels or per-pixel class probabilities $P(u, v, \ell)$ (and, for instance-level tasks, a set of object masks and scores). A key advantage of segmentation is its explicit and interpretable structure: it converts low-level pixels into object- and region-level representations that support downstream reasoning and decision-making, and it can be fused naturally with geometry (*e.g.*, depth, normals) and language in multimodal systems [242]. Consequently, segmentation maps are central to applications such as autonomous driving, medical image analysis, robotic manipulation, visual editing, and human-computer interaction.

4.1.3 Depth Maps

A depth map is a geometric representation that augments a 2D image with per-pixel distance information, describing the spatial layout of a scene from a given viewpoint [234]. Formally, given an image domain $\Omega \subset \mathbb{R}^2$, a depth map is a scalar function

$$D : \Omega \to \mathbb{R}, \tag{4.4}$$

where $D(u, v)$ denotes the distance from the camera center to the first visible surface along the viewing ray corresponding to pixel (u, v). Depending on the sensing modality and formulation, this value may represent metric depth, inverse depth, or disparity [112]. Depth maps can be obtained through multiple mechanisms, including stereo vision via triangulation [112], active sensors such as structured-light or time-of-flight cameras [247], and learning-based monocular depth estimation [55]. When combined with known camera intrinsics, a depth map enables the back-projection of image pixels into 3D space, thereby serving as a bridge between 2D image representations and 3D geometry. A key advantage of depth maps lies in their compact, image-aligned structure, which allows them to be processed using standard convolutional architectures and easily fused with the RGB appearance information. Compared to volumetric representations, depth maps are memory-efficient while still encoding meaningful geometric cues. As a result, they are widely used in applications such as 3D reconstruction, novel view synthesis [119], robotic navigation, and autonomous driving [173].

4.1.4 Surface Normal Maps

A surface normal map is a geometric representation that encodes local surface orientation at each pixel of an image. Given an image domain $\Omega \subset \mathbb{R}^2$, a surface normal map assigns to each pixel a unit vector

$$\mathbf{N} : \Omega \to \mathbb{S}^2, \tag{4.5}$$

where $\mathbf{N}(u, v) = (n_x, n_y, n_z)$ represents the direction of the surface normal in camera or world coordinates, and $\mathbb{S}^2$ denotes the unit sphere. Surface normal maps can be derived from depth maps or 3D geometry via local differentiation, or directly estimated from RGB images using learning-based methods [248]. Classical approaches to normal estimation rely on photometric stereo or geometric reconstruction [249], while modern methods employ CNNs to regress surface normals from monocular images [250]. Regardless of the estimation procedure, surface normal maps remain image-aligned and encode rich local geometric information that complements depth and segmentation cues. Surface normal maps can capture fine-grained shape and orientation details that may be ambiguous or noisy in raw depth estimates. Because surface normals are invariant to absolute depth scale, they provide robust geometric constraints for scene understanding.

Consequently, surface normal maps are widely used in applications such as 3D reconstruction, object pose estimation, scene layout analysis, realistic rendering, and as auxiliary supervision in multi-task learning [250].

4.1.5 Edge Maps

An edge map is a structural representation that highlights salient intensity or color discontinuities in an image, which often correspond to object boundaries and significant geometric transitions [251]. Given an image $I : \Omega \to \mathbb{R}^C$, an edge map is typically constructed by measuring local image gradients, for example,

$$E(u, v) = \|\nabla I(u, v)\|, \tag{4.6}$$

where ∇I denotes the spatial gradient operator. Classical edge detection methods rely on hand-crafted operators such as Sobel, Laplacian of Gaussian, and the Canny detector [251], while modern learning-based approaches directly predict the edge probabilities using convolutional neural networks [252]. Edge maps may be binary, indicating the presence or absence of an edge, or real-valued, encoding edge strength, orientation, or confidence. The primary advantage of edge maps is their ability to significantly reduce visual complexity while preserving essential geometric structure. This abstraction makes them relatively robust to changes in illumination and appearance. Consequently, edge maps are widely used in image segmentation, contour detection, shape analysis, line drawing, and as intermediate cues for higher-level tasks such as object recognition and depth estimation [234].

4.1.6 Optical Flow Maps

An optical flow map is a motion representation that encodes the apparent 2D displacement of pixels between consecutive frames in a video sequence [253]. Given two image frames I_t and I_{t+1}, the optical flow is defined as a vector field

$$\mathbf{F}(u, v) = (f_x(u, v), f_y(u, v)), \tag{4.7}$$

where (f_x, f_y) describes the horizontal and vertical displacement of pixel (u, v) from time t to $t + 1$. Classical optical flow estimation relies on assumptions such as brightness constancy and spatial smoothness, leading to variational formulations [253, 254]. More recently, deep neural networks have been proposed to directly regress dense flow fields from image pairs or short video clips [128, 255]. Regardless of the estimation method, the resulting optical flow map remains image-aligned, preserving spatial correspondence with the input frames. The main advantage of optical flow maps is their ability to capture rich dynamic information about scene motion, including object motion, camera motion, and motion parallax, while remaining computationally efficient. As a result, optical flow is a core representation in video analysis and robotic perception tasks such as motion segmentation, action recognition, video interpolation, and visual odometry.

4.1.7 2D Keypoints

A 2D keypoint set is a sparse image representation that encodes the image-plane locations of a finite set of semantically or geometrically meaningful points, such as corners, joints, or landmarks [234]. Unlike dense maps that assign a value to every pixel, keypoint representations summarize visual content with a small number of salient points, trading pixel-level coverage for compactness and geometric explicitness. Formally, given an image defined on a domain $\Omega \subset \mathbb{R}^2$, a keypoint representation is a set of K points

$$\mathcal{K} = \{(x_k, y_k)\}_{k=1}^{K}, \quad (x_k, y_k) \in \Omega, \tag{4.8}$$

where each keypoint corresponds to a specific semantic or geometric entity. In learning-based formulations, keypoints are often represented implicitly as confidence or heat maps:

$$H_k : \Omega \to [0, 1], \tag{4.9}$$

with the keypoint location obtained by finding the maximum or expected value of H_k [256]. 2D keypoints can be obtained using classical feature detectors, such as Harris corners, SIFT, and SURF [257, 111], or through modern deep neural networks that directly regress keypoint locations or predict dense heat maps [258, 256]. Depending on the task, keypoints may represent generic interest points, object-specific landmarks, or articulated structures such as the human body joints. The detection *task* that produces such a set, including its extension to landmark detection and human pose estimation, is discussed in detail in Section 6.1.3 of Chapter 6. A key advantage of 2D keypoint representations is their sparsity and compactness: they significantly reduce the dimensionality of visual data while preserving critical structural information. This makes keypoints computationally efficient, robust to background clutter, and well-suited for geometric reasoning and correspondence matching. Human pose estimation is a specialized and highly structured instance of 2D keypoint detection that focuses on localizing a predefined set of anatomical landmarks of the human body, such as joints and facial keypoints, in images or videos [259]. In the 2D setting, the task aims to estimate the image coordinates of body joints, typically represented as a set of keypoints $\mathcal{P} = \{(x_j, y_j)\}_{j=1}^{J}$, where J denotes the number of joints in a human skeletal model. These keypoints are not independent: they are constrained by the underlying human body structure, which is often modeled using kinematic trees, graphical models, or learned structural priors.

4.2 3D



The 3D representations lift the 2D pixel grid into the ambient space the scene actually occupies, replacing image-plane appearance with explicit geometry and topology. They are the substrate on which downstream tasks that require metric reasoning, occlusion handling, viewpoint manipulation, or physical simulation are built, from autonomous driving and robotic manipulation to medical visualization and immersive graphics. Compared with the depth and surface normal maps of Section 4.1, which still inhabit the image grid, 3D representations expose geometry directly and can therefore describe parts of the scene that are not visible in any single view.

We organize this section around the eight 3D representations that dominate current practice (Figure 4.2, Table 4.1): RGB-D images, voxels, point clouds, meshes, implicit fields, neural radiance fields (NeRF), 3D Gaussian Splatting (3DGS), and hybrids of the above. They differ in what they store (surfaces vs. volumes, explicit primitives vs. neural functions), how they trade off fidelity against memory and rendering speed, and how amenable they are to editing and to gradient-based optimization.

- **RGB-D Image**. Pairs a standard RGB image with a per-pixel depth map; view-dependent and cannot capture occluded surfaces from a single frame.
- **Voxel**. A uniform 3D grid of volumetric elements; conceptually simple and well suited to inherently volumetric data, but memory and resolution scale as $O(N^3)$.
- **Point Cloud**. An unstructured set of 3D points, typically the native output of LiDAR or photogrammetry; faithful to the raw measurement but lacks surface connectivity.
- **Mesh**. Surfaces defined by vertices and faces; the dominant representation in graphics for its GPU-friendliness, at the cost of a fixed connectivity that complicates topology change.
- **Implicit Representation**. Defines surfaces as the level set of a continuous function (*e.g.*, Signed Distance Functions); resolution-agnostic and topology-flexible, but slow to render.
- **NeRF (Neural Radiance Field)**. A neural function over a 5D input domain (3D location + 2D viewing direction) that supports state-of-the-art novel-view synthesis.
- **3DGS (3D Gaussian Splatting)**. Represents a scene as anisotropic 3D Gaussian primitives rendered through a differentiable splatting rasteriser, reaching NeRF-level quality at real-time speeds.

Table 4.1: Qualitative comparison of 3D representations (relative ratings, not measured values).

Representation	Fidelity	Memory Cost	Render Speed	Topology Flexibility	Editability
RGB-D Image	Low (2.5D)	Low	High	View-dependent	Low
Voxel	Low–Medium	High ($O(N^3)$)	Medium	High	Medium
Point Cloud	Medium	Medium (scalable)	High	High	Medium
Mesh	High	Low (GPU-friendly)	Very High	Low (fixed connectivity)	High
Implicit (SDF)	High	Low	Low (volume query)	High	Low
NeRF	Very High	Low	Low (ray marching)	High	Very Low
3DGS	Very High	Medium	Very High (real-time)	High	Medium
Hybrid	Very High	Medium	High	Adaptive	Medium–High

- **Hybrid Methods**. Combinations of the above (*e.g.*, DMTet, Tri-plane) that pair explicit primitives with implicit decoders to balance fidelity, efficiency, and gradient-based optimizability.



RGB-D Image Voxel Point Cloud Mesh SDF

Figure 4.2: Examples of 3D representations: RGB-D image, voxel grid, point cloud, triangle mesh, and a signed distance field (SDF) slice.

4.2.1 RGB-D Images

RGB-D images are a popular 2.5D representation that captures both photometric and geometric information from a single viewpoint. This data format consists of two co-registered images: a standard RGB color image providing appearance and texture details, and a corresponding depth map where each pixel value encodes the distance from the sensor to the corresponding point in the scene. Formally, it can be defined as a function mapping a 2D image grid Ω to a 4D space combining color and depth:

$$I_{rgbd} : \Omega \to \mathbb{R}^3 \times \mathbb{R}_{\geq 0}, \tag{4.10}$$

where the first three channels represent RGB color and the fourth channel encodes the distance from the sensor to a point in the scene. In practice, RGB-D data is stored as a tensor $I \in \mathbb{R}^{H \times W \times 4}$.

The primary advantage of RGB-D data lies in its efficiency and accessibility. It is captured by low-cost sensors such as Microsoft Kinect or Intel RealSense [260], and its image-like grid structure enables the direct application of powerful 2D convolutional neural networks (CNNs) to tasks such as semantic segmentation and object detection [261, 262, 263]. This fusion of color and depth provides a rich, multi-modal input for scene understanding. However, a significant limitation is its view-dependent nature; an RGB-D image represents only a partial view of the 3D world, suffering from occlusions and providing no information about surfaces hidden from the camera. Despite this, its real-time acquisition and processing capabilities make it indispensable in robotics for navigation [264] and grasping [265, 266, 267], in augmented reality (AR) for scene interaction, and as a foundational input for 3D reconstruction pipelines that fuse multiple frames into a complete model [114].

4.2.2 Voxel Grids

The voxel grid is a foundational 3D representation that discretizes space into a uniform grid of volumetric elements, or *voxels* (a portmanteau of "volume" and "pixel"). Analogous to a pixel in 2D, a voxel grid can be modeled as a function mapping discrete 3D coordinates to a set of attributes:

$$G : \{1, \dots, D\} \times \{1, \dots, H\} \times \{1, \dots, W\} \to \mathcal{A}, \tag{4.11}$$

where D, H, W are the grid dimensions and $\mathcal{A}$ can represent occupancy, color, density, or material properties, thereby enabling a detailed volumetric description of an object.

A key advantage of this representation is its ability to model complex geometries and internal structures, which are often challenging for surface-based methods like polygon meshes to capture. The conceptual simplicity of voxels, which can be seen as the three-dimensional extension of 2D pixels, facilitates straightforward implementation and manipulation. Consequently, voxel grids are natively suited for applications involving inherently volumetric data, such as medical imaging (*e.g.*, MRI and CT scans) [268], where biological tissues are naturally captured as a 3D array of values. They are also widely adopted in computer graphics and gaming for tasks like procedural generation and creating destructible environments, as popularized by games like *Minecraft*.

4.2.3 Point Clouds

Point clouds offer a direct and unstructured 3D representation, defined as a set of points in 3D space:

$$P = \{p_i\}_{i=1}^{N}, \quad p_i = (x_i, y_i, z_i, \mathbf{a}_i) \in \mathbb{R}^3 \times \mathbb{R}^C. \tag{4.12}$$

Each point p_i consists of its geometric coordinates (x_i, y_i, z_i) and an optional C-dimensional attribute vector $\mathbf{a}_i$ encoding properties such as color, intensity, or surface normals. They are the native output of active 3D scanning technologies like LiDAR (Light Detection and Ranging) and structured-light scanners [269], as well as passive methods like photogrammetry [115].

The primary strength of point clouds lies in their raw and unstructured nature; they represent geometry without imposing any explicit topological connectivity (unlike meshes) or spatial discretization (unlike voxels). This makes them a faithful and often highly precise representation of the captured data. However, this very lack of inherent structure poses significant challenges for downstream processing. The unordered and potentially sparse distribution of points makes tasks like feature extraction, segmentation, and reconstruction non-trivial, necessitating specialized algorithms and deep learning architectures (*e.g.*, PointNet [132] and its successors [270]) that are designed to operate directly on irregular point sets.

4.2.4 Meshes

A mesh is a canonical surface representation in computer graphics, defining an object's boundary with a set of geometric and topological primitives. Formally, a triangle mesh is a tuple $M = (V, F)$, where

$$V = \{v_i\}_{i=1}^{|V|} \subset \mathbb{R}^3, \; F \subset V \times V \times V. \tag{4.13}$$

Here, V is a set of vertices representing points in 3D space, and F is a set of faces (typically triangles) that define connectivity between vertices.

The key characteristic of a mesh is its explicit topological structure, which encodes the connectivity between these elements. This structure strikes an effective balance between descriptive power and computational efficiency. Its prevalence is due to several key advantages: meshes can represent complex, smooth surfaces with arbitrary fidelity through adaptive tessellation (*i.e.*, varying the polygon density); moreover, this representation is natively supported by modern Graphics Processing Units (GPUs), enabling highly efficient

real-time rendering. The explicit connectivity also facilitates the straightforward computation of essential surface properties like normals and curvature, which are critical for realistic lighting and physical analysis. However, as a purely surface-based representation, standard meshes are ill-suited for modeling volumetric or internal structures, and representing objects with complex or changing topologies can be algorithmically challenging.

4.2.5 Implicit Representations

Implicit representations define 3D geometry not as a discrete set of primitives (*e.g.*, points or polygons), but as the level set of a continuous function:

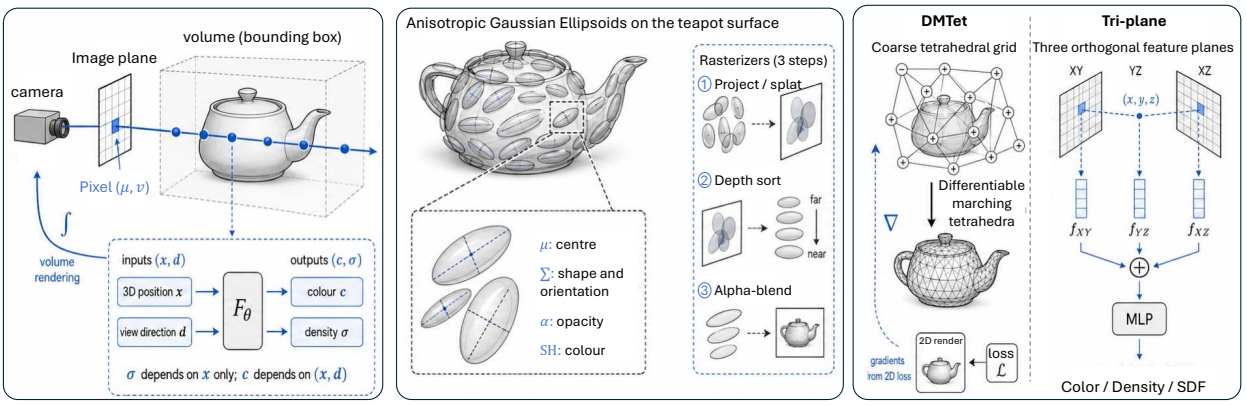
$$f(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}. \tag{4.14}$$

The surface $\mathcal{S}$ is implicitly defined as the set of points where the function evaluates to a specific value, typically zero:

$$\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 | f(\mathbf{x}) = 0\}. \tag{4.15}$$

A particularly powerful and popular form of this is the **Signed Distance Function (SDF)**. An SDF maps any spatial coordinate $\mathbf{x}$ to the closest distance to the surface, where the sign indicates whether the point is inside ($f(\mathbf{x}) < 0$) or outside ($f(\mathbf{x}) > 0$) the object. This formulation inherently guarantees watertight, closed surfaces and provides a smooth, continuous description of the geometry, making it resolution-agnostic. A key advantage of implicit fields is their ability to elegantly handle complex and dynamic topologies; operations like merging, splitting, or creating holes do not require complex and error-prone re-meshing procedures. This makes them ideal for physical simulations, such as fluid dynamics and shape manipulation.

This paradigm has experienced a major resurgence with the rise of **neural fields** or **neural implicit representations**. Instead of storing the function in a grid, a neural network is trained to approximate it [118] (*e.g.*, DeepSDF [117]). This approach is memory-efficient and can learn continuous, high-fidelity shape priors from data. Extensions such as Neural Radiance Fields (NeRF) further generalize implicit fields to jointly model geometry and appearance, and are discussed in detail in Section 4.2.6. To visualize or use implicit surfaces in traditional rendering pipelines, an explicit mesh representation is often extracted via algorithms like Marching Cubes [105].



(a) NeRF (b) 3DGS (c) Hybrid (DMTet + Tri-plane)

Figure 4.3: Neural and hybrid 3D representations: (a) NeRF, (b) 3DGS, and (c) hybrid methods (DMTet and Tri-plane).

4.2.6 Neural Radiance Fields

Neural Radiance Fields (NeRF) represent a paradigm shift in novel view synthesis and implicit 3D scene representation [119]. The core innovation is to represent a static scene as a continuous function with a 5D

input domain (3D location + 2D viewing direction), parameterized by a simple Multi-Layer Perceptron (MLP). This neural network, denoted as F_Θ, maps a 3D spatial coordinate $\mathbf{x} = (x, y, z)$ and a 2D viewing direction $\mathbf{d} = (\theta, \phi)$ to a volume density σ and an emitted color $\mathbf{c} = (r, g, b)$. A crucial design choice is that density σ depends only on the location $\mathbf{x}$, while the color $\mathbf{c}$ depends on both location and viewing direction $(\mathbf{x}, \mathbf{d})$. This allows NeRF to capture complex, non-Lambertian surface effects such as specular reflections and translucency.

To render an image from a novel viewpoint, rays are cast through the scene, and their colors are computed by integrating the predicted color and density values along each ray using principles of classical volume rendering and numerical quadrature. By optimizing the network's weights to minimize the photometric reconstruction error between rendered and ground-truth images, NeRF can learn a highly detailed and coherent representation of the scene from only a sparse set of posed input images. The resulting state-of-the-art photorealism has established NeRF as a foundational technique, inspiring a vast body of follow-up research aimed at improving its speed, scalability, and applicability to dynamic scenes [120, 271, 272, 273].

4.2.7 3D Gaussian Splatting

3D Gaussian Splatting (3DGS) has recently emerged as a groundbreaking alternative to neural radiance fields, achieving state-of-the-art quality for novel view synthesis while enabling real-time rendering [121]. Unlike the implicit representation of NeRF, 3DGS represents a scene as a set of N anisotropic 3D Gaussian primitives, each parameterized by:

- **Position** $\boldsymbol{\mu} \in \mathbb{R}^3$: the 3D center of the Gaussian;
- **Covariance** $\boldsymbol{\Sigma} \in \mathbb{R}^{3\times3}$: shape and orientation, allowing thin planes through to volumetric wisps;
- **Color**: Spherical Harmonics (SH) [274] coefficients capturing view-dependent appearance;
- **Opacity** $\alpha \in [0, 1]$: transparency.

The density contributed by a single primitive at a query point $\mathbf{x} \in \mathbb{R}^3$ is

$$G(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \exp\left(-\tfrac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x} - \boldsymbol{\mu})\right). \tag{4.16}$$

$\boldsymbol{\Sigma}$ is decomposed into a scaling matrix $\mathcal{S}$ and rotation matrix $\mathcal{R}$, such that

$$\boldsymbol{\Sigma} = \mathcal{R}\mathcal{S}\mathcal{S}^T\mathcal{R}^T, \tag{4.17}$$

with per-primitive scaling and rotation stored as $s \in \mathbb{R}^{N\times3}$ and quaternions $r \in \mathbb{R}^{N\times4}$. The key to 3DGS's real-time performance lies in the differentiable splatting rasteriser introduced together with the representation [121], which extends earlier differentiable rasterization work on meshes [275]. Instead of costly per-pixel ray marching, 3DGS projects (or "splats") these 3D Gaussians onto the 2D image plane, creating 2D Gaussian splats. These splats are then sorted and alpha-blended in a highly-parallelized, feed-forward process that is exceptionally well-suited for modern GPU architectures. Differential splatting applies a viewing transform W and Jacobian J to compute the transformed covariance $\boldsymbol{\Sigma}' = JW\boldsymbol{\Sigma}W^TJ^T$, enabling novel view rendering. Each pixel color $\mathcal{C}$ is obtained by blending the N overlapping primitives along the ray:

$$\mathcal{C} = \sum_{i\in N} c_i\alpha_i \prod_{j=1}^{i-1}(1 - \alpha_j), \tag{4.18}$$

where c_i and α_i denote the color and opacity of the i-th primitive. The entire model, initialized from a sparse point cloud (typically from Structure-from-Motion), is optimized end-to-end against training images. The optimization process also includes an adaptive density control mechanism to dynamically add, split, or prune Gaussians to best represent the scene's geometry. This combination of an explicit, flexible primitive with a fast, differentiable rasterizer has set a new standard for performance and quality in 3D scene reconstruction.

4.2.8 Hybrid Methods

To harness the complementary strengths of different 3D representations, hybrid methods have emerged as a powerful paradigm. These approaches combine two or more representations, integrating the high-fidelity and continuous nature of implicit functions with the computational efficiency and explicit structure of primitives like meshes or voxels. This section reviews two influential hybrid techniques: Deep Marching Tetrahedra (DMTet) and Tri-plane.

Deep Marching Tetrahedra (DMTet). Deep Marching Tetrahedra (DMTet) [276] is an innovative hybrid technique that bridges learned implicit fields with explicit polygonal meshes. Its core concept involves using a neural network to predict an implicit representation of a shape, such as a Signed Distance Function (SDF) [277] or an occupancy field. A differentiable version of the marching tetrahedra algorithm subsequently extracts a high-quality, explicit triangle mesh surface from this implicit representation.

The primary advantage of DMTet lies in its end-to-end differentiability. The entire pipeline, from the implicit field prediction to the final mesh generation, allows gradients to flow from a 2D-image-based loss function back to the 3D shape parameters. This enables direct optimization of the mesh geometry using 2D supervision, leading to highly detailed and accurate models. Furthermore, DMTet can learn to produce an adaptive mesh, allocating higher tetrahedral density to regions of fine detail, thus optimizing computational resources while maintaining high fidelity.

Tri-plane. The Tri-plane representation [278] offers an efficient, factorized approach to modeling 3D scenes and objects. Instead of encoding information in a dense 3D volume, it decomposes the 3D space into three orthogonal 2D feature planes (typically aligned with the XY, YZ, and XZ axes). This design significantly reduces the memory and computational complexity associated with 3D representations.

The working principle is to derive the feature for any 3D point by projecting it onto the three 2D planes, querying the features at the corresponding projected coordinates, and aggregating them (*e.g.*, through summation or concatenation). This aggregated feature vector is then fed into a small MLP to decode the final property of the point, such as its color, density, or SDF value. This "explicit-grid-plus-implicit-decoder" structure is the essence of its hybrid nature. It benefits from the efficiency of 2D grid-based feature storage, which can be processed by powerful 2D CNNs, while avoiding the prohibitive memory cost of a full 3D grid.

4.3 4D



The 4D representations extend the 3D representations of Section 4.2 into the temporal dimension, modeling not only the geometry and appearance of a scene at a single instant but also how both evolve over time. They are the substrate for tasks that involve dynamic content (moving objects, deformable bodies, fluid simulation, novel-view synthesis of casually captured videos), where a sequence of independent 3D reconstructions would either fail to maintain temporal coherence or be prohibitively expensive to recompute frame by frame.

We organize this section around five 4D representations (Figure 4.4), each obtained by adding time to one of the 3D representations of Section 4.2: RGBD video, point clouds, meshes, neural radiance fields, and deformable 3D Gaussian Splatting. The voxel and implicit-field families of Section 4.2 admit analogous time extensions but are less commonly used in the 4D literature, and we therefore omit them as standalone subsections here.

4.3.1 RGBD Video

RGBD videos $\mathcal{V} = \{(I_t, D_t)\}_{t=1}^{T}$, with $I_t \in \mathbb{R}^{H\times W\times 3}$ and $D_t \in \mathbb{R}^{H\times W}$, are a straightforward extension of RGBD images (see Section 4.2.1) to the temporal domain. Equivalently, they can be stored as a 4D tensor

Table 4.2: Representative methods for each 3D representation family, chosen to show its historical progression and dominant design choices.

Representation	Representative Methods	Key Idea / Contribution
RGB-D Image	KinectFusion [114], DADA [263], SemanticFusion [279]	Fuses color and per-pixel depth for real-time reconstruction and semantic understanding.
Voxel Grid	VoxNet [280], 3D U-Net [268], OGN [281]	Direct 3D CNNs on regular grids; octree variants reduce memory for high resolution.
Point Cloud	PointNet [132], PointNet++ [270], Point Transformer [282], PointNeXt [283]	Permutation-invariant and attention-based architectures for unordered point sets.
Mesh	MeshCNN [284], Pixel2Mesh [285], AtlasNet [286]	Learning on mesh edges and single-image mesh generation from data.
Implicit (SDF/occupancy)	Occupancy Networks [118], DeepSDF [117], IM-Net [287], NeuS [288]	Neural networks encode continuous surfaces via occupancy or signed-distance fields.
NeRF	NeRF [119], Plenoxels [289], Instant-NGP [120], Mip-NeRF 360 [272]	5D radiance fields; later works accelerate training and reduce aliasing.
3DGS	3D Gaussian Splatting [121], Mip-Splatting [290], Scaffold-GS [291], 4DGS [191]	Anisotropic Gaussian primitives with differentiable rasterization for real-time rendering.
Hybrid	DMTet [276], Tri-plane [278], NeRFactor [292]	Combines explicit grids or meshes with compact implicit decoders.

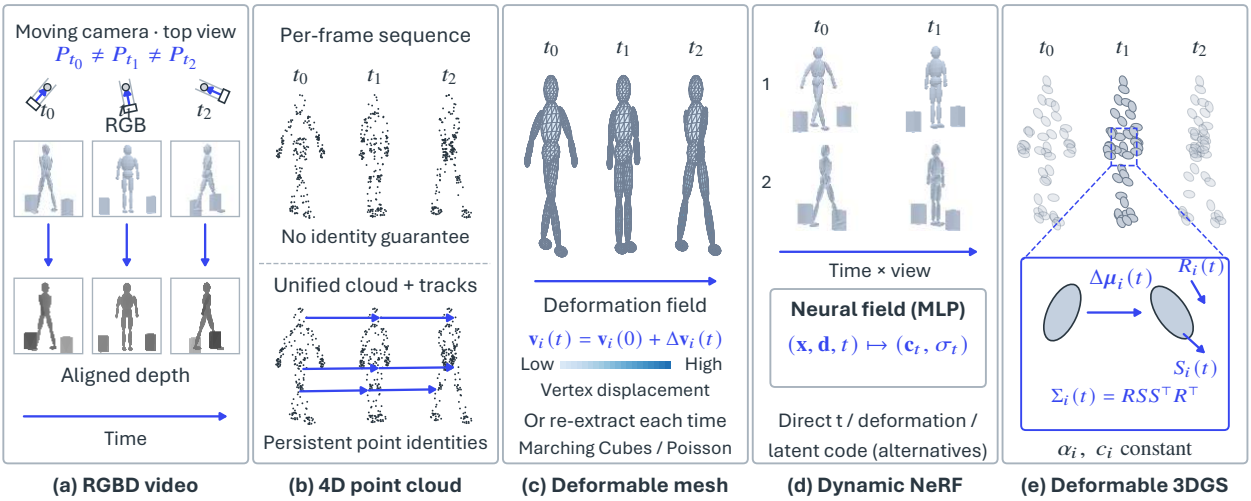


Figure 4.4: Five 4D representations, each a 3D primitive evolving over time: (a) RGBD video, (b) 4D point cloud, (c) deformable mesh, (d) dynamic NeRF, and (e) deformable 3DGS. The RGBD example uses changing camera poses across time to observe the scene from different viewpoints; the top-view camera icons use the same azimuths as the paired RGB and depth renderings. The point-cloud panel distinguishes independent frame-wise samples from persistent point trajectories. Mesh colors indicate vertex displacement; re-extraction is an alternative to preserving vertex and face identities. In (d), rows are views and columns are times. In (e), R and S abbreviate the time-dependent $R_i(t)$ and $S_i(t)$; constant opacity and color describe the illustrated parameterization. All scenes are controlled synthetic illustrations, not experimental reconstructions.

$\mathcal{V} \in \mathbb{R}^{T\times H\times W\times 4}$, with the last channel holding depth. RGBD videos can be captured using depth sensors, such as Microsoft Kinect or Intel RealSense, which provide synchronized RGB and depth data. These videos are commonly used in applications such as motion capture, gesture recognition, and augmented reality. There are also many learning-based methods [293, 294, 295] that directly predict consistent RGBD videos from monocular video sequences $\mathcal{I} = \{I_t\}_{t=1}^{T}$. However, such a representation may struggle to handle complex dynamic scenes with significant occlusions or topology changes. If we want to further reconstruct a unified 4D scene in a canonical space, we may also need to provide or estimate camera poses $\mathcal{P} = \{P_t\}_{t=1}^{T}$

for each frame, as well as point trajectories over time to establish temporal correspondences, *e.g.* via joint pose-trajectory optimization in recent video-based SfM systems [296].

4.3.2 Point Cloud

Point clouds are another popular representation for 3D scenes (see Section 4.2.3), and they can be extended to 4D by incorporating temporal information. A 4D point cloud can be represented as $\mathcal{P} = \{(\mathbf{p}_i(t), \mathbf{c}_i(t))\}_{i=1}^{N}$, where $\mathbf{p}_i(t) \in \mathbb{R}^3$ is the 3D position of point i at time step t, and $\mathbf{c}_i(t) \in \mathbb{R}^3$ is the corresponding color. To map an uncalibrated, continuous video sequence $\mathcal{I} = \{I_t\}_{t=1}^{T}, I_t \in \mathbb{R}^{H\times W\times 3}$, to a 4D point cloud representation, it generally has two types of representations: (1) a sequence of point clouds $\mathcal{P} = \{\mathcal{P}_t\}_{t=1}^{T}$ in a fixed coordinate system, where each $\mathcal{P}_t = \{(\mathbf{p}_i(t), \mathbf{c}_i(t))\}_{i=1}^{H\times W}$ is an independent point cloud at time step t; (2) a unified point cloud $\mathcal{P} = \{(\mathbf{p}_i(t), \mathbf{c}_i(t))\}_{i=1}^{N}$, where N is the total number of points across all time steps, and each point i has a trajectory over time. The first representation is easier to obtain, as it can be directly derived from the depth maps estimated from each frame using monocular depth estimation methods [296]. However, it may suffer from temporal inconsistencies and a lack of point correspondences across frames. The second representation, in contrast, requires explicit temporal correspondences (either ground-truth tracks, optical-flow lifting [255], or jointly optimized point trajectories [296]), and is therefore harder to acquire, but it offers persistent point identities and consistent trajectories that are useful for downstream 4D reconstruction, motion analysis, and dynamic NeRF/3DGS initialization.

4.3.3 Mesh

We can extend the mesh representation (see Section 4.2.4) to 4D by representing a sequence of 3D meshes over time. A 4D mesh is a sequence $\{\mathcal{M}_t\}_{t=1}^{T}$ with $\mathcal{M}_t = (\mathbf{V}_t, \mathbf{F}_t)$, where $\mathbf{V}_t \in \mathbb{R}^{N\times 3}$ and $\mathbf{F}_t \in \mathbb{N}^{M\times 3}$ are the vertices and faces of the mesh at time t, respectively. To reconstruct a 4D mesh using deep learning, researchers typically sample a set of points on the mesh surface at the original time step $t = 0$, and then learn a deformation field to model the motion of these points over time [297]. Then, the deformed points at each time step can be used to reconstruct watertight manifold meshes using Marching Cubes [105] or Poisson Surface Reconstruction [298].

4.3.4 NeRF

As discussed in Section 4.2.6, NeRF [119] is a continuous, volume-based 3D representation that maps a 3D position $\mathbf{x} \in \mathbb{R}^3$ and a 2D viewing direction $\mathbf{d} \in \mathbb{S}^2$ to a color $\mathbf{c} \in \mathbb{R}^3$ and a volume density $\sigma \in \mathbb{R}_{\geq 0}$ via a learned function $F_\Theta : (\mathbf{x}, \mathbf{d}) \to (\mathbf{c}, \sigma)$. This function is typically implemented using a neural network, which is trained on a set of input images and their corresponding camera poses. To extend NeRF to dynamic scenes, we can introduce a temporal dimension t to the input of the function, resulting in a 4D representation $F_\Theta : (\mathbf{x}, \mathbf{d}, t) \to (\mathbf{c}_t, \sigma_t)$. This allows the model to capture changes in the scene over time, enabling the rendering of novel views at different time steps. In practice, the temporal dimension can be represented in various ways. For example, it can be trained to directly take a continuous time variable as input, and output displacement $\Delta\mathbf{x}(t)$ to deform the 3D position $\mathbf{x}(t) = \mathbf{x}(0) + \Delta\mathbf{x}(t)$ at time step t [187]. Alternatively, the deformation can be parameterized using a dense $\mathrm{SE}(3)$ field [299], which models the rigid transformations (rotation and translation) of points in the scene over time. Other approaches [300] may also incorporate additional latent codes to represent different time steps, enabling the model to capture more complex temporal variations.

4.3.5 Deformable 3D Gaussian Splatting

Given a static scene representation $\mathcal{G} = \{(\alpha_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, c_i)\}_{i=1}^{G}$, which contains G 3D Gaussians [121] (see Section 4.2.7), where $\alpha_i \in [0, 1]$ is the opacity, $\boldsymbol{\mu}_i \in \mathbb{R}^3$ is the mean (3D position), $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3\times 3}$ is the covariance

matrix, and $c_i : \mathbb{S}^2 \to \mathbb{R}^3$ is the color function that maps a viewing direction to an RGB color, we extend it to a *deformable* 3D Gaussian Splatting representation for dynamic scenes by letting both the mean and the covariance vary in time: $\mathcal{G}(t) = \{(\alpha_i, \boldsymbol{\mu}_i(t), \boldsymbol{\Sigma}_i(t), c_i)\}_{i=1}^{G}$ [191]. The mean is parameterized by a deformation field, $\boldsymbol{\mu}_i(t) = \boldsymbol{\mu}_i(0) + \Delta\boldsymbol{\mu}_i(t)$, where $\boldsymbol{\mu}_i(0)$ is the initial position at $t = 0$ and $\Delta\boldsymbol{\mu}_i(t)$ is the time-dependent offset. The covariance is decomposed in the same way as in the static case, $\boldsymbol{\Sigma}_i(t) = \mathcal{R}_i(t)\mathcal{S}_i(t)\mathcal{S}_i(t)^T\mathcal{R}_i(t)^T$, with the rotation $\mathcal{R}_i(t)$ and scaling $\mathcal{S}_i(t)$ themselves outputs of the deformation field at time t. The opacity α_i and color function c_i are typically kept constant over time. In practice, this means that the appearance of the scene does not change significantly over time, while the latest work [301] also explores time-varying colors and opacities. This representation allows for efficient rendering of dynamic scenes by accounting for the temporal changes in the positions and shapes of the 3D Gaussians.

4.4 Language

Unlike the visual and geometric representations of the previous sections, which encode space explicitly through pixels, points, or surfaces, language represents space *symbolically*, naming entities and stating the relations among them (*e.g.*, "the chair is next to the table" or "the store is across the street from the park") and thereby capturing relational, topological, and functional structure that metric representations leave implicit. The question this section pursues is not how language *describes* space, which is self-evident, but whether models trained only on language come to *encode* it. Answering that requires first understanding how language representations are learned at all, so the section moves from the evolution of language representation learning, from masked-language pretraining to large autoregressive models, to the more surprising finding that such sequence-only training induces latent, spatially structured world models rather than mere text statistics. The scope here is language in isolation; how language is *aligned* with visual, 3D, and audio modalities is deferred to Chapter 5.

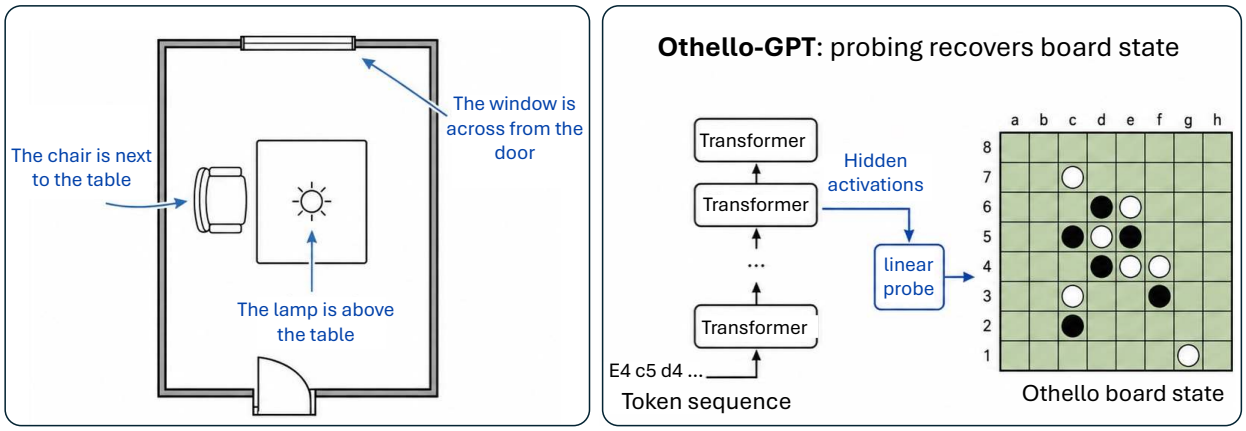


Figure 4.5: Spatial information implicit in language-only representations. *Left*: natural-language descriptions of spatial relations. *Right*: a linear probe on Othello-GPT hidden activations recovers the board state from move sequences alone.

4.4.1 The Evolution of Language Representation Learning

Language representation learning has evolved rapidly over the past decade, from masked-language pretraining (starting with BERT [42] and continued by RoBERTa [302], XLNet [303], and T5 [304]) to autoregressive large language models trained purely on next-token prediction, exemplified by GPT-2 and GPT-3 [305, 43]. Across this progression, a recurring observation is that representation quality scales with model size, data

size, and the sophistication of the pretraining objective [306], and that pretrained representations encode richer structure than the surface text statistics they were trained on [307, 308]. This last property, that language-only training produces representations carrying structure beyond text, is what makes language representations load-bearing for spatial intelligence, as the next subsection makes explicit.

4.4.2 Language Representation Learning Induces Spatial Intelligence

Although LLMs are optimized for sequence prediction, recent work suggests that they can develop structured internal representations that resemble world models [309, 310, 311]. One particularly pertinent demonstration of this on spatial intelligence is Othello-GPT, a transformer trained solely on sequences of legal moves from the board game Othello [312].

Despite receiving no prior knowledge of the board layout or game rules, Othello-GPT achieves near-perfect accuracy in predicting legal moves (actions in a physical space). Probing experiments reveal that nonlinear classifiers can reliably reconstruct the entire board state from hidden activations [313, 307, 314]. Crucially, interventional studies show that manipulating internal representations of specific tiles alters the model's predictions in a way consistent with counterfactual board states, providing causal evidence of spatial encoding [315, 316].

These findings align with broader research indicating that LMs acquire implicit representations of abstract or grounded concepts. For example, models have been shown to encode syntactic structure [308], color spaces [311], and even logical state-tracking [309]. Work on chess transcripts similarly shows that LMs can track board positions without explicit supervision [317]. Compared to systems like AlphaZero, which are built with strong inductive biases and explicit access to board states [318, 319], Othello-GPT highlights the surprising extent to which sequence-only training can lead to emergent spatial reasoning.

In sum, it has been demonstrated that spatially structured representations can emerge in LMs trained only on language-only sequences. This suggests that autoregressive sequence prediction is sufficient, in certain domains, to induce latent world models that encode spatial relations and can causally drive action predictions.

4.5 Audio



UDIO is fundamentally tied to spatial information. Unlike vision, which directly captures the geometric layout of a scene, acoustic signals encode spatial structure only *implicitly*, through the physical transformations imposed on sound during propagation. Recovering space from sound is therefore a matter of reading these transformations backward: from the perceptual cues that human hearing exploits, to the computational representations that formalize them, to the learned models and cross-modal couplings that ground sound in the surrounding geometry. This section follows that arc. Human auditory perception interprets the resulting acoustic cues, shaped by these transformations, to infer the spatial layout of the environment. The primary cues are binaural: the *Interaural Time Difference (ITD)* and the *Interaural Level Difference (ILD)*, both arising from the geometry of the head and the spatial separation of the ears [320, 321]. ITD reflects the difference in arrival time between the two ears and is highly reliable for localizing low-frequency sounds whose long wavelengths exceed head size. Conversely, ILD captures sound pressure level differences caused by head-shadowing effects and serves as a dominant cue for high-frequency localization where such attenuation is more pronounced. However, binaural cues alone cannot fully resolve depth, elevation, or front-back ambiguities.

To address these limitations, the auditory system further exploits *monaural spectral cues* and *environmental reverberation*. Direction-dependent spectral filtering produced by the torso, head, and pinnae introduces characteristic notches and peaks, enabling inference of elevation and disambiguation of positions on the cone of confusion [322]. In addition, the room impulse response (RIR) shaped by early reflections, late reverberation,

and surface materials encodes information about source distance, enclosure geometry, and environmental structure [323, 324]. These monaural and environmental cues complement binaural information, jointly supporting robust spatial inference under naturalistic conditions.

Computational models formalize these perceptual cues through structured acoustic representations. Head-related transfer functions (HRTFs) [325] capture listener-specific spectral filtering and are foundational to perceptual research and spatial audio rendering. In parallel, spherical-harmonics–based ambisonics representations [326] provide a scene-centric encoding of the sound field around a point in space, with controllable angular resolution. Together, these representations describe how sound interacts with the body and the environment, enabling principled approaches to spatial reasoning, source localization, and scene reconstruction. Taken together, the ability of the auditory system to integrate binaural, monaural, and environmental cues demonstrates that sound carries rich implicit structure about the spatial configuration of sources and the surrounding environment.

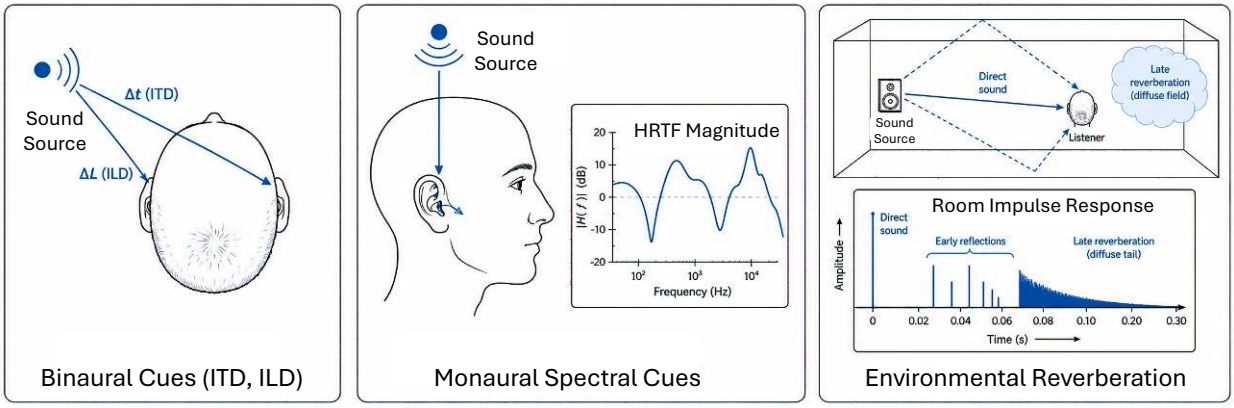


Figure 4.6: Three families of acoustic spatial cues: binaural cues (ITD, ILD), monaural spectral cues (direction-dependent filtering by torso, head, and pinnae, summarized by the HRTF), and environmental reverberation (room impulse response).

Recent learning-based approaches aim to capture spatial structure directly from raw audio, going beyond analytic models. Neural encoders trained on spatially annotated or simulated data learn to infer direction, distance, and environmental geometry from multi-channel waveforms or spectrograms [47, 327]. Self-supervised frameworks further exploit the natural co-occurrence between sound and space: SoundSpaces [46] and VisualEchoes [328] train models to predict sound propagation or reconstruct echoes from static images. These learned representations encode not only spatial acoustics but also grounding in surrounding geometry, forming a key component of spatial-aware representation.

In real-world environments, spatial audio interacts closely with other sensory modalities. Audio–visual models align acoustic cues with visual observations to localize sounding objects and understand where events occur [329, 330, 331]. Embodied agents further leverage auditory signals to perceive layout and navigate toward acoustic targets, as demonstrated in audio–visual navigation [332, 333] and egocentric audio reasoning [46]. Through such cross-modal grounding, audio provides complementary spatial cues that enhance robustness under visual occlusion and enable reasoning about events beyond the agent's field of view. This makes audio a powerful complementary modality for spatial representation, especially in scenarios where visual information is partial, noisy, or unavailable.

4.6 Domain-Specific Representations

The representations surveyed so far are organized by modality, each general enough to span many applications. A large body of practice, however, develops representations tailored to a specific

sensor or downstream task rather than to a modality as such, and these are collected here. What unites them is not a shared signal but a shared design logic: each encodes the inductive biases of its data and trades spatial fidelity against semantic compressibility in the way its sensors and target tasks demand. This section walks through five such domains: embodied robotics, virtual reality, geospatial and remote sensing, clinical and scientific imaging, and indoor intelligent assistants, examining in each the representations that have become canonical and the constraints that shaped them. Beginning with robot perception and planning, systems commonly take as input RGB-D sequences and point clouds (often fused over time into semantic maps that ground observations into a common 3D frame [114, 334]), and enable tracking, map building, and manipulation. Many such systems also encode visual input into compressed spatial representations, such as keypoints, dense correspondences [335], and point tracks, used to parameterize 6-DoF grasps [336] and object-centric actions. When available, free space is typically represented with occupancy grids [79, 89] and signed-distance representations [109], and is often additionally augmented with semantic labels such as traversability and cost for planning. To further enable action from perception, planners frequently operate on affordance representations [337] over parts and candidate contact points, encoding feasible interactions such as graspability or predicted contact quality, thereby enabling policies to execute tasks successfully.

Real-time virtual reality systems often represent scenes with triangle meshes and physically based rendering (PBR) materials, where geometry, normals, and texture define appearance, while 6-DoF camera poses keep virtual content registered to the environment. To enable flexible interaction, such systems include captured geometry (*e.g.*, spatial meshes, planes, or TSDF and SDF volumes [109, 114]) and depth-based occlusion layers, which are often paired with collision proxies and semantic labels. Toward photoreal view synthesis, methods increasingly use neural scene representations (*e.g.*, NeRF-style radiance fields and 3D Gaussian splats [119, 121]), to support novel-view rendering and relighting. Additionally, for the purposes of enabling characters in virtual reality systems, such characters are commonly represented with skeletal pose and expression parameters (*e.g.*, SMPL-style body models [338] or Blendshapes), and their navigation relies on navmeshes combined with acceleration structures (*e.g.*, octrees), to handle moving agents in simulation at high frame rates.

In remote sensing applications, geographic information systems typically represent the world (the Earth), as georeferenced raster fields, for example, optical multispectral and hyperspectral imagery as well as SAR backscatter. These fields are aligned to explicit coordinate reference systems and sensor geometry metadata. Such systems encode terrain through multiple elevation products (*e.g.*, DEM, DTM, DSM), and compress human infrastructure into vector networks (*e.g.*, roads, power grids, pipelines), often across scales. And increasingly, they use LiDAR to not only output raw point clouds, but also to output rasterized height maps that support flexible 3D reasoning. These frameworks also commonly operate on multi-temporal stacks and spatiotemporal data with validity masks, which can generate change maps for land-use dynamics and disaster response, as well as feed digital-twin pipelines that generate orthomosaics and photogrammetric point clouds [115], for simulation and virtual modeling.

Clinical and scientific imaging operate on volumetric scans (*e.g.*, CT and MRI), in which 3D voxel grids capture geometry and intensities [268], and explicit spatial transforms (*e.g.*, rigid, affine, and nonlinear deformation fields) align subjects and modalities into a shared atlas space. In addition, functional measurements (*e.g.*, fMRI BOLD and calcium imaging) are often stored as 4D spatiotemporal sensors, but are frequently compressed into time series and activation maps after parcellation. When available, such systems represent structural delineations (*e.g.*, tissue, organ, lesion) as segmentation masks and labeled maps [174], which are often paired with surface meshes (*e.g.*, cortical surfaces) that enable surface-based topological comparison across subjects. For diffusion imaging applications, tensor and orientation fields encode local diffusion directions and magnitudes, while tractography streamlines uses these fields to reconstruct fiber pathways and uncover possible structural connectivity. These diverse representations across granularities and temporal scales are often mapped onto one another through atlas spaces and parcellations, enabling multimodal fusion and cross-subject analysis.

Intelligent assistants that operate in indoor environments typically ground language in 3D space using multi-scale maps, for example, 3D reconstructions and floorplans anchored by an estimated 6-DoF agent pose [94, 339, 340], which are often combined with semantic maps that encode navigable routes. Context for interactions is captured in structured world models (*e.g.*, scene graphs with typed entities and relations [341, 342, 343]), which can be augmented with additional semantics like anchors, keypoints, and regions of interest. For the purposes of intelligent and persistent assistance, representations that encode some form of memory are necessary, for example, episodic records and long-horizon object trajectories, which are often represented as probabilistic beliefs over object locations in a room. In addition, assistant systems can optionally contain geofences and occupancy masks, to help define actionable regions for planning.

In summary, many domains have developed representations specifically designed for specialized sensors and encoding the inductive biases of the underlying data structure. We see that these representations model a tradeoff between (spatial) fidelity and (semantic) compressibility, reflecting how the aforementioned sensors sample space, what priors are built in, and what downstream tasks are to be supported.

Chapter 5

Cross-Modal Spatial Representation

The aim of cross-modal spatial representation is a latent space in which heterogeneous perceptual signals, images, 3D scenes, video, audio, and language are aligned so that the spatial structure each carries is preserved rather than averaged away. Where the previous chapter asked how much geometry a single modality holds on its own, the concern here is the opposite and harder one: how signals that encode space in incompatible formats- a pixel grid, a world-frame point set, a waveform, a sentence- can be brought into correspondence without the alignment itself distorting what it aligns.

No single recipe achieves this. Methods differ in what they treat as the unit of alignment, in the objective that binds one modality to another, in how tightly they anchor the space to physical geometry, and in the linguistic interface through which it is queried. These choices are not independent, and the chapter first makes them explicit as a small set of design axes before turning to the individual cross-modal pairings in which they play out.

5.1 Overview

We define four axes that structure this cross-modal representation learning problem, which we use to organize the remainder of the chapter. The first axis is the *alignment strategy*, which we categorize into: contrastive alignment, generative alignment, and programmatic alignment. Contrastive alignment learns a shared embedding space by pushing together paired samples and pulling away negative samples. Such alignment scales to weakly paired data and enables retrieval and coarse grounding, but often needs additional spatial supervision to understand fine-grained relations grounded to specific visual regions. In contrast, generative alignment predicts across modalities (*e.g.*, captioning from vision or language-conditioned decoding into images, 3D, or video), which gives rich supervision and coverage, but also tends to drift from true underlying geometry unless it is further constrained, for example, by multi-view consistency. Finally, programmatic alignment aligns visual entities to explicit symbols (*e.g.*, attributes, relation graphs, or concepts in domain-specific languages), which define spatial semantics and naturally interface with program-based, verifiable reasoning, but often requires stronger supervision and trades off expressiveness for structure.

The second axis is *spatial modality*, which ranges from view-aligned 2D to world-aligned 4D. Here, two-dimensional tokens (*e.g.*, pixels, patches) are expressive and broadly applicable, but rely on some form of geometric structure to capture faithful spatial relationships. Three-dimensional tokens (*e.g.*, points, meshes, implicit fields, Gaussian primitives) better encode metric scale and rigid-body invariance (*i.e.* SE(3) consistency: rotations and translations of the scene do not change the underlying semantics), so that linguistic queries are grounded in world coordinates rather than in image projections. However, they are

less ubiquitous and often rely on partial and noisy reconstructions of geometry. Extending 3D through time yields 4D traces (*e.g.*, tracks), that bind actions to expressive motion and change, but significantly increase computational complexity and require reliable alignment data, while being sensitive to complex semantics such as identity switches. These different spatial modalities correspond to and build upon the single-modality representations of Chapter 4.

The third axis is the *language interface*, from free-form text to structured interfaces. Free-form text (*e.g.*, natural language) enables open-vocabulary understanding for referring expressions, retrieval, and instruction following, but comes with corresponding ambiguity in terms of frames of reference, scale, and units. In contrast, structured interfaces (*e.g.*, programs) reduce such ambiguity by explicitly describing spatial semantics, enabling integration with symbolic reasoning, but are less flexible in capturing the full range of natural language descriptions.

The fourth axis is the *training regime*, which defines both the level of supervision and coverage. Paired supervision, for example, from captioned videos or referring expressions in 3D scenes, gives strong alignment, but is expensive and can be biased toward frequent compositions and co-occurrences. Weakly paired internet data, for example, from alt-text or transcripts, enable diversity, but, due to low signal-to-noise ratios, often yield noisier spatial grounding.

Below, we organize the chapter by cross-modal pair, using *spatial modality* (which pair) and *alignment strategy* (which objective) as the primary organisational axes; the *language interface* and *training regime* axes recur as cross-cutting attributes referenced throughout. We give an overview of 2D-text alignment (Section 5.2), 3D-text alignment (Section 5.3), audio-text alignment (Section 5.4), and vision-vision alignment (Section 5.5); each follows the scope and axes defined here.

5.2 2D-text



The most developed cross-modal setting pairs images with language. Its defining tension is that a text-aligned 2D representation lives in the image projection, not the world: appearance and layout sit on the pixel grid, yet the relations a caption asserts (left of, above, behind) are only implicit there. How faithfully those relations survive alignment turns on two choices: the granularity at which the image is represented, and the objective that binds it to text, and this section is organized around them. Models often fall into three broad families of visual representations, across different levels of granularity. First are global embeddings that compress an entire image into a single vector paired with a sentence embedding. Here, dual-encoder approaches such as CLIP [143] and ALIGN [344] show that large-scale contrastive pretraining over noisy web data yields strong zero-shot classification and retrieval performance. Second, many works have proposed local tokens tied to image subregions, in which vision-language models, often transformers (*e.g.*, ViLBERT [24], LXMERT [25], UNITER [345], OSCAR [346]) input region features extracted by bottom-up detectors (*e.g.*, Faster R-CNN [172]) into cross-modal attention blocks, enabling word-region alignment, and hence fine-grained reasoning over object proposals. A third class of methods makes relations explicit; here, models extract intermediate representations that are directly aligned with spatial predicates such as "left-of" or "on." NSCL [347] and LEFT [348] learn hybrid representations that explicitly bind named visual concepts with latent features. This representational spectrum, from global to region to relation-based, forms the basis for the objectives below.

Contrastive retrieval serves as a common objective for large-scale pretraining, as it learns a shared metric space, without requiring dense and expensive supervision. As representative examples, CLIP [143] and ALIGN [344] maximize agreement between paired image and text embeddings, enabling robust zero-shot transfer; LiT [349] shows that contrastively tuning only the text encoder on top of a locked image encoder can further improve such transfer; and CoCa [350] fuses contrastive learning with captioning in a single encoder-decoder architecture to combine discriminative and generative strengths. Another line of

work adapts contrastive alignment to more fine-grained regions. RegionCLIP [351] lifts global contrast to region-text pairs, FILIP [352] proposes late token-token interaction to align words with patches efficiently, X-VLM [353] learns multi-grained alignment such that entities and attributes can be localized and described consistently across granularities, and ALBEF [354] conducts image-text contrast with cross-modal fusion and momentum-distilled pseudo-targets to stabilize learning on noisy web data.

In addition, masked modeling with textual conditioning has been shown to provide a complementary self-supervised signal that encourages cross-modal consistency. LXMERT [25] and UNITER [345] pretrain with masked language modeling, masked region modeling, and image-text matching to build word-region alignment. VLMo [355] generalizes this idea with a mixture-of-modality experts, and BEiT-3 [356] performs masked data modeling over images, text, and image-text pairs in a single multiway transformer.

With more labels, generative objectives can be applied for this representation learning problem, and also complement self-supervised ones. Here, methods teach models to produce language conditioned on images, intuitively, to generate tokens that act as spatially grounded explanations. As representative examples, SimVLM [357] shows that a prefix language-modeling objective trained on weak supervision can improve performance on captioning and visual question answering, BLIP [358] adds caption bootstrapping to denoise web supervision, and GIT [359] simplifies a scalable image-encoder and text-decoder architecture.

In a more fine-grained manner, grounding objectives can directly optimize spatial correspondence between language and 2D pixels, points, or boxes. Here, text-conditioned detection and phrase grounding unify detection and language. MDETR [360] learns a DETR-style [136] detector modulated by text, GLIP [361] pretrains on both detection and grounding corpora and self-generates boxes from image-text pairs to scale supervision, and Grounding DINO [362] augments strong detection backbones with grounded pretraining for open-set detection. At the pixel level, DenseCLIP [363] adapts image-text matching to pixel-text matching for dense prediction, and GroupViT [364] shows that segmentation structure can appear from text supervision alone.

Across these categories, we see that 2D-text representations can emerge from diverse approaches, for example, differing in whether spatial structure is implicit (*e.g.*, in attention over patches) or explicit (*e.g.*, in region tokens), and whether supervision is weakly paired and contrastive, generative, or explicitly grounded. Models that learn 2D vision-language representations now increasingly combine all contrastive, captioning, grounding, and masked objectives to solve a wide range of downstream tasks, from (more semantic) referring expression comprehension to (more visually grounded) localization.

5.3 3D-text



Moving from the image plane to the scene itself removes the projection gap that limits 2D-text alignment: language is now bound to geometry expressed in world coordinates, so that metric scale and the invariance of meaning under rotation and translation become properties the representation can actually enforce. The difficulty shifts accordingly, from recovering relations that pixels hide to obtaining and aligning 3D structure that is often partial, noisy, and expensive to annotate. As before, the representation families and the alignment objectives acting on them jointly organize the section. Two complementary families of representations have emerged in this domain. First, object-centric and scene-centric encoders learn world-aligned tokens, then align them to language via contrastive or generative objectives. ULIP [365] instantiates this paradigm by aligning point-cloud features to an image-text space, scaling supervision through synthetic triplets. OpenShape [366] extends this direction further with curated 3D corpora and hard negative mining to support open-world 3D retrieval, and OpenScene [39] co-embeds dense features for 3D scenes with text and image pixels in CLIP feature space. Additional projection strategies have also been explored, for example, adapting 2D vision-language models to 3D by rendering depth or multi-view images

and transferring CLIP's alignment to point clouds; PointCLIP [367] and CLIP2Point [368] do so and enable adapters for cross-modal transfer.

A second family binds language to implicit scene fields. Here, LERF [369] optimizes a dense 3D language field inside NeRF [119], by rendering CLIP embeddings across views and enables open-vocabulary queries that return relevancy volumes. CLIP-Fields [370] similarly distills 2D vision-language cues into 3D implicit maps to support semantic search and view localization. In addition, replacing NeRF with 3D Gaussian Splatting [121] yields efficient language fields; for example, LangSplat [371] combines CLIP features with Gaussians for fast, precise open-vocabulary querying. These methods build on advances in explicit 3D primitives and real-time rendering to maintain high-fidelity representations while training efficiently.

In this 3D-text domain, objectives also follow the representation choices. Contrastive alignment remains common, for example, ULIP trains with InfoNCE [372] over point clouds-images-text, while OpenShape scales both data and hard-negative mining. Projection-based approaches transfer contrastive objectives to depth, and multi-view renders to align point clouds with text labels. Grounded objectives enforce explicit 3D correspondence between language and geometry. 3D referring expression datasets and models (*e.g.*, ScanRefer [373], ReferIt3D [374], 3D-VisTA [375], BUTD-DETR [376], NS3D [377], InstanceRefer [378]) localize targets from language descriptions and learn to understand complex relations dependent on viewpoints.

Here, geometry-aware representations are critical for aligning language with the physical world. Ideal 3D-text representations should respect SE(3) structure (*e.g.*, rotations and translations should not spuriously change semantics), yet cross-modal alignment should still encode view-dependent language (*e.g.*, "the chair to the left of the table"). To this end, priors from equivariant architectures, such as the SE(3), Transformer [379] and Vector Neurons [380], can be integrated into 3D-text models. Implicit fields enable different options; while NeRF-based language fields are not strictly equivariant, multi-view consistency and cross-view regularization can improve their spatial coherence. Alternatively, Gaussian splatting uses a canonical world frame and enables efficient supervision during language-field optimization. Overall, these representation choices and objectives give rise to a wide range of approaches for 3D-text alignment, from tri-modal contrastive pretraining to implicit or splat-based language fields.

5.4 Audio-text

Sound aligns to language along a different primary dimension than vision does. What an acoustic signal encodes first is time, the sequence of events as they unfold, and only secondarily, and coarsely, does it carry spatial attributes such as direction of arrival or distance. Audio-text representation learning therefore aims for embeddings that are open-vocabulary and temporally grounded, with spatial awareness added where the labels permit. The section again moves from the representation granularity to the objectives that align it with text. One such direction is to embed audio-text representations at a coarse level, for example, with global encoders mapping an entire audio clip and a full sentence into a joint space, enabling zero-shot retrieval. Here, CLAP [381] and LAION-Audio [382] scale contrastive language-audio pretraining with curated datasets, and show strong transfer across retrieval benchmarks. In addition, AudioCLIP [383] extends CLIP to a tri-modal image-text-audio setting, to leverage visual-text semantics, and Wav2CLIP [384] distills CLIP's alignment into an audio encoder when audio-text pairs are scarce.

Beyond global representations, a line of work that aims to extract representations at a more fine-grained scale learns temporally local structure, for example, by operating on spectrogram patch tokens with transformer architectures. The Audio Spectrogram Transformer (AST) [385] and PaSST [386] show that attention over time-frequency patches captures long-range temporal dependencies, and HTS-AT [387] further returns frame-level maps that are useful for event localization. These tokenizations make it straightforward to bind subclip semantics to text (*e.g.*, "a door slams, then footsteps echo"). Self-supervised masked objectives, *e.g.*, SSAST [388], MAE-AST [389], and Audio-MAE [390], additionally pretrain such backbones by reconstructing

masked patches, while BEATs [391] adds an iterative tokenizer-encoder scheme that encourages high-level semantic abstraction.

Across advances in representations and objectives, an emergent and increasingly common paradigm is to first pretrain an audio backbone with masked modeling, then align embeddings to text via contrastive losses, and finally, optionally add a captioning head to increase supervision. In tasks where spatial semantics is important, incorporating spatial labels, such as direction-of-arrival, helps refine the audio-text representation space. By leveraging these approaches, models can learn to align predicates such as "left-of" or "far" with meaningful acoustic features. These methods enable models to learn embeddings that are not only open-vocabulary and temporally grounded, but also (coarsely) spatially aware.

5.5 Vision-vision



common approach to improving vision-language alignment is to first align vision with vision, intuitively, enforcing geometric consistency such that visual features are anchored to a persistent world model before combining with language. Concretely, the goal of vision-vision representation learning is to fuse visual inputs (*e.g.*, multiview RGB) into a set of world-aligned scene tokens, such that they are invariant to viewpoint changes and can be later queried by language. Here, we discuss three broad families of approaches: lifting 2D features into 3D scene tokens, projecting these 3D tokens back into 2D for supervision, and using audiovisual correspondence to add constraints that further anchor the world model.

Lifting 2D to 3D follows two complementary directions. The most classical and canonical approach first estimates camera poses and geometry with structure-from-motion (SfM) and simultaneous localization and mapping (SLAM), then adds semantics to the generated map. Here, commonly used frameworks include COLMAP for SfM [115], KinectFusion and ElasticFusion for dense RGB-D mapping [114, 392], and ORB-SLAM2 for SLAM [393]. Learning-based reconstructions build upon this direction by back-projecting per-view features into a 3D volume and predicting TSDF or occupancy directly from posed images, for example, as in NeuralRecon [394]. Though components may differ, all of these approaches output view-agnostic scene tokens that can later be guided by linguistic supervision and queries.

Projecting 3D back to 2D provides self-supervision. Differentiable rendering in neural fields anchors world-aligned tokens to multiview observations, and enforces cross-view consistency [119]. Distilling 2D foundation features into 3D fields adds semantic knowledge without manual, expensive 3D labels. Here, works typically transfer 2D features into a co-registered 3D feature volume, which can be rendered to any view [395, 396]. In addition, distillation is not limited to 2D language targets alone; for example, ImageBind [397] proposes a multisensor feature target (image-text-audio-depth-thermal-IMU), which enables transfer of cross-sensor semantics even when there are few explicit labels. Although these distillations are vision-vision processes, they are explicitly designed for language alignment later. The key benefit is that now vision-language alignment is conducted on a geometry-consistent representation, rather than directly on view-specific features.

Audio-visual alignment can further constrain this vision-vision world model; self-supervised objectives have been designed to enforce consistency between audio and visual motion, which allows models to localize sound sources and disambiguate co-occurring events [398, 399, 400]. Such cues provide directional or timing semantics that help anchor linguistic queries to specific world-aligned entities, especially in cases with multiple candidate entities. Across this variety of approaches, vision-vision representation learning methods all aim to learn a persistent world model that is geometrically and temporally consistent before adding any linguistic supervision, therefore making downstream cross-modal alignment more data-efficient and spatially faithful to the physical world.

Part III

Spatial Intelligence: From Perception to Generation

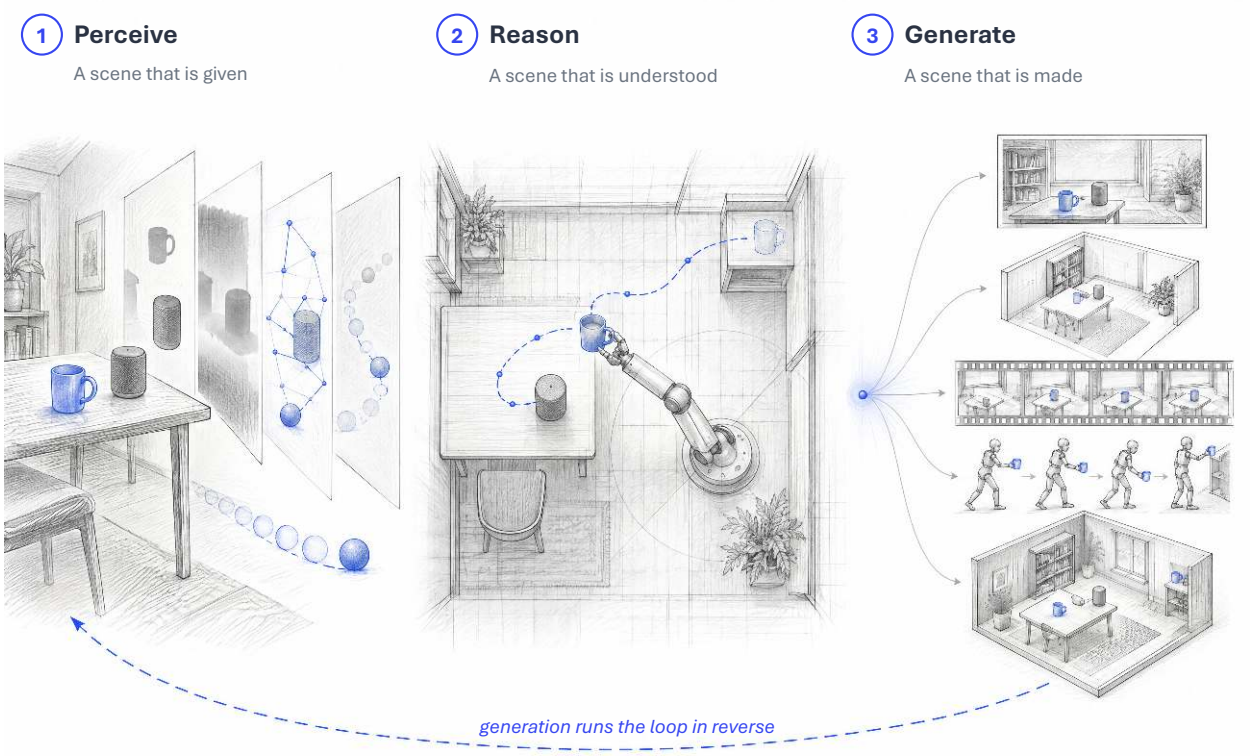


Figure III: Part III in one view. (1) Perceive: a given scene is separated into objects, depth, relational structure, and motion. (2) Reason: the understood scene supports inference and planning, here a path that carries the cup around an obstacle to the shelf. (3) Generate: an intent unfolds into images, 3D scenes, video, motion, and whole worlds, so that generation runs the loop in reverse.

If the previous part asked how space is represented, this one asks what a system does with that representation. Spatial intelligence, in the working sense of this handbook, is the ability to turn spatial structure into understanding and understanding into action, and it resolves into three capabilities that build on one another: perceiving space, reasoning over it, and generating it. These are less separate subfields than three phases of a single loop running from sensory input to spatial artifact.

Perception comes first because everything downstream depends on it: it converts raw observation into structured spatial evidence, answering what is present, where it lies, and how it moves. Reasoning takes that evidence as given and draws consequences from it, inferring relations, planning routes and actions, and predicting how a configuration will change under intervention. Generation closes the loop by running the process in reverse, mapping an intent (a prompt, a constraint, a partial observation) back into concrete spatial content: an image, a shape, a scene, a motion, an entire simulated world. Read in sequence, the three chapters trace the path from a scene that is given, to one that is understood, to one that is made.

Each of the three rests on the representations built in the previous part and supplies capabilities that the applications of the next will draw on; together they are the core of what makes a system spatially intelligent.

Chapter 6

Spatial Perception

Spatial perception is the foundational layer of spatial intelligence: it turns raw sensory input into the structured spatial evidence that everything downstream depends on, answering what is present, where it lies, and how it moves. It is what stands between an agent and a scene, and the reliability of every later inference is bounded by the reliability of what perception delivers.

The chapter follows a single progression, from the elemental to the structured. It begins with recognizing the discrete entities that populate a scene, grounds them in metric geometry, then binds them into relational and scene-level structure, extends that structure across time, and finally broadens it beyond vision by fusing complementary modalities. Each step presupposes the ones before it, so that perception builds from isolated percepts toward a dynamic, multimodal model of space that reasoning and generation can act on.

6.1 Spatial Entity Recognition

Before space can be measured or organized; its contents must be named. Spatial entity recognition is that entry point: it answers *what is present*, and every later geometric and relational step builds on the entities it names. Recognition operates at several granularities, and the section is ordered accordingly: detecting discrete object instances, delineating dense regions of pixels or points by segmentation, pinning down sparse keypoints and landmarks, and finally assigning a holistic scene label. What unites them is that they recover identity and category, not yet geometry; where each entity sits and how it is oriented is taken up next.

6.1.1 Object Detection

6.1.1.1 Task Definition

Given a 2D image I, *2D object detection* identifies instances of predefined object classes and localizes each one in the image plane. The output is a set of detections

$$\mathcal{D} = \{(c_i, b_i, s_i)\}_{i=1}^{N}, \tag{6.1}$$

where c_i is the predicted class, $b_i = (x_i, y_i, w_i, h_i)$ a 2D bounding box (center and size, or corner coordinates), and $s_i \in [0, 1]$ a confidence score. In short, the task answers *what objects are where*. Figure 6.1 summarizes the 2D task and its detector families.

3D object detection keeps the same set-of-detections output but places each box in metric 3D space. The input is a LiDAR point cloud $\mathcal{P} = \{\mathbf{p}_j\}_{j=1}^{M}$ with $\mathbf{p}_j \in \mathbb{R}^3$ (often with an intensity channel), one or more

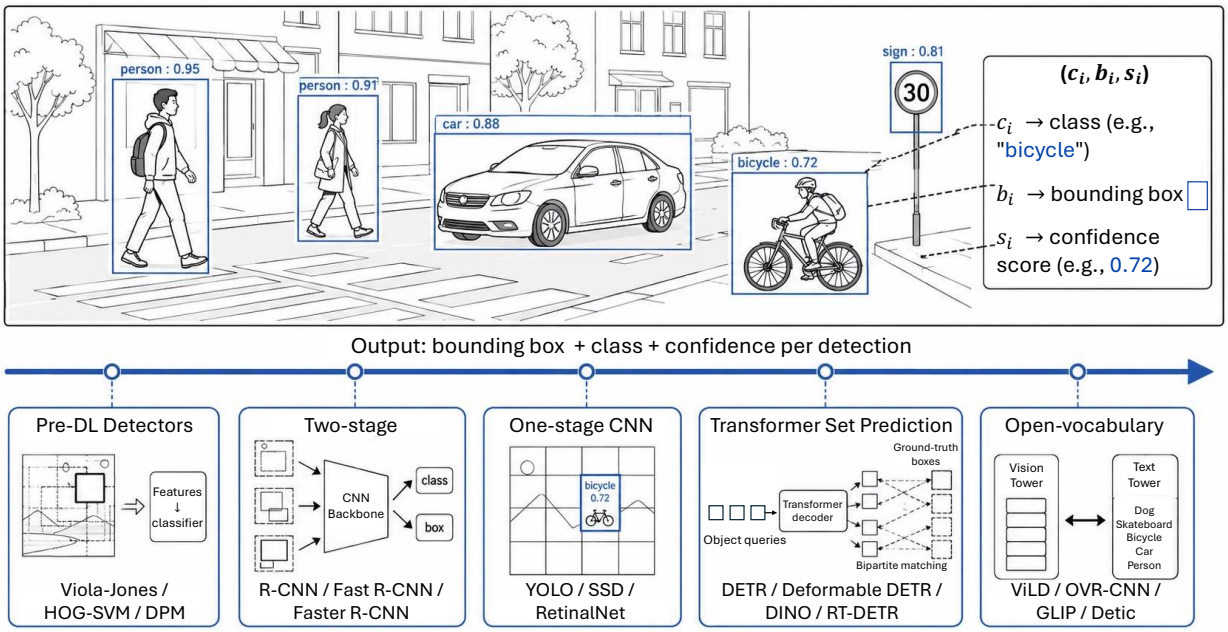


Figure 6.1: 2D object detection: each detected instance is a tuple (c_i, b_i, s_i) of class, bounding box, and confidence score (top), and the progression of detector families with representative methods under each (bottom).

calibrated images, or both, and each box becomes

$$b_i = (x_i, y_i, z_i,\ l_i, w_i, h_i,\ \theta_i), \tag{6.2}$$

a 3D center, the physical length, width, and height, and a heading angle θ_i about the vertical axis; indoor benchmarks often use axis-aligned boxes and drop θ_i. Because the box is expressed in the sensor or ego frame rather than the image plane, a 3D detection already carries metric position and heading, which is why driving and robotics stacks consume it directly.

6.1.1.2 Methodology

Before Deep Learning. Early detectors [401] ran a sliding window over multi-scale image pyramids and classified each candidate region with hand-crafted features and shallow models. Viola–Jones [402] combined Haar-like features with an AdaBoost cascade for real-time face detection; HOG features with linear SVMs [403] became the standard pipeline for pedestrians and generic objects; and deformable part-based models (DPMs) [404] represented an object as parts connected by deformable constraints. Manual feature design, dense-window evaluation, and limited robustness to pose, scale, and occlusion capped this line and motivated the move to learned features.

Two-stage Detectors. R-CNN [126] first generates category-independent region proposals and then classifies each with a CNN, a large accuracy gain over hand-crafted pipelines bought with high computational cost. Fast R-CNN [405] shares convolutional computation across proposals through ROI pooling and trains classification and box regression jointly; Faster R-CNN [172] adds a Region Proposal Network so that proposals are nearly free and the whole detector trains end-to-end. Sparse R-CNN [406] later replaced dense anchors with a small set of learnable proposals, reaching end-to-end detection without a transformer.

One-stage Detectors. One-stage detectors drop the proposal step and predict classes and boxes directly from the feature map. YOLO [130] casts detection as a single regression over the whole image, trading

localization accuracy for speed; YOLOv2 and YOLOv3 [407, 408] add anchors, multi-scale prediction, and stronger backbones, and later versions [409, 410] refine training and deployment. SSD [411] predicts from multiple feature maps at different scales, and RetinaNet [412] introduces the focal loss to counter the extreme foreground–background imbalance, bringing one-stage accuracy level with two-stage methods. Recent YOLO models move to anchor-free, decoupled heads (YOLOv8 [413]) and NMS-free end-to-end training (YOLOv10 [414]), converging with the set-prediction detectors below.

Transformer-based Detectors. DETR [136] reformulates detection as direct set prediction with a transformer encoder–decoder, removing anchors and non-maximum suppression at the price of slow convergence and weak small-object performance. Deformable DETR [415] attends to a sparse set of sampling points, which fixes both; Conditional DETR [416] conditions decoder attention on spatial queries for faster training; DINO [417] adds denoising training and better query initialization; and RT-DETR [418] brings the family to real-time speed.

Open-Vocabulary Detection. Open-vocabulary detection (OVD) recognizes objects outside a fixed label set by aligning region features with text embeddings. The line evolved from embedding distillation to large-scale vision–language pretraining to data-driven vocabulary expansion: ViLD [419] and OVR-CNN [420] distill or learn a shared region–text embedding space from image–text pairs; GLIP [361] unifies detection with phrase grounding under grounded language–image pretraining; and Detic [421] shows that image-level labels over a large vocabulary suffice to train the classifier of an open-vocabulary detector. OWL-ViT [422], Grounding DINO [362], and YOLO-World [423] push the same idea toward simpler transfer, tighter grounding, and real-time speed; see the surveys [424, 425] for a fuller account.

3D Object Detection. 3D detectors differ mainly in how they turn sensor data into something a network can process; the detection heads on top largely reuse the 2D families above. On LiDAR, VoxelNet [135] learns features inside voxels and runs 3D convolutions over the grid, and PointPillars [426] collapses the vertical axis into pillars so that an ordinary 2D CNN on the bird's-eye-view (BEV) plane does the rest at real-time speed. Point-based detectors keep the raw points: PointRCNN [427] generates proposals from PointNet++ features [270] of segmented foreground points, PV-RCNN [428] combines voxel and point features, and CenterPoint [429] replaces anchors with BEV center heatmaps, the 3D counterpart of anchor-free 2D detection. Indoor scenes are cluttered and object centers often fall in empty space, which favors voting: in VoteNet [430] surface points vote for object centers before grouping, and 3DETR [431] carries DETR-style set prediction over to point clouds.

Camera-only 3D detection has to infer the depth that LiDAR measures. Monocular detectors regress depth, size, and orientation on top of a 2D detection (Section 6.2.3 treats them as metric estimation). Pseudo-LiDAR [432] instead converts a predicted depth map into a point cloud and runs a LiDAR detector on it, showing that much of the camera–LiDAR gap came from the representation rather than from depth quality alone. Multi-camera systems lift image features into a shared BEV space, either bottom-up, as Lift-Splat-Shoot [137] does by predicting a depth distribution per pixel and splatting features onto the ground plane, or top-down, as DETR3D [433] and BEVFormer [434] do by projecting 3D queries or BEV cells into the images to sample features. Fusion combines both sensors, from Frustum PointNets [435], which crops the point cloud with 2D image detections, to BEVFusion [181], which merges camera and LiDAR features on a common BEV map. Cube R-CNN [436] trains a single image-based detector across indoor and outdoor domains on Omni3D. Deployment constraints for driving are discussed in Section 9.2, and video detection, which adds cross-frame association to the same output set, under tracking in Section 6.4.2.

6.1.1.3 Datasets and Metrics

Table 6.1 summarizes the benchmarks that shaped the field. **PASCAL VOC** [444] fixed the evaluation protocol and remained the reference through the early deep-learning era despite its modest scale. **MS COCO** [440],

Table 6.1: Summary of object detection datasets across modalities and domains, with image and bounding-box counts per official split.

Dataset	Modality, Domain	Class	train		validation		test	
			images	boxes	images	boxes	images	boxes
VOC-2007 [437]	image, general	20	2,501	6,301	2,510	6,307	4,952	14,976
VOC-2012 [438]	image, general	20	5,717	13,609	5,823	13,841	10,991	-
ILSVRC-2014 [439]	image, general	200	456,567	478,807	20,121	55,502	40,152	-
MS-COCO-2015 [440]	image, general	80	82,783	604,907	40,504	291,875	81,434	-
MS-COCO-2017 [440]	image, general	80	118,287	860,001	5,000	36,781	40,670	-
Objects365-2019 [441]	image, general	365	600,000	9,623,000	39,000	479,000	100,000	170,000
OpenImages [442]	image, general	600	1,743,042	14,610,229	41,620	303,980	125,436	937,327
ImageNet VID [439]	video, general	30	3,862	-	555	-	937	-
LVIS [443]	image, general	1,203	100,000	-	-	-	-	-
Omni3D [436]	3D (image), general	98	175,573	-	19,127	-	39,452	-

Table 6.2: Reported performance of representative object detection methods on COCO val2017 under the standard COCO protocol; AP_S / AP_M / AP_L break down by object size (small / medium / large).

Method	Backbone	#P (M)	FLOPs (G)	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L	Year
Two-stage detectors										
Faster R-CNN [172]	R50-FPN	41.5	207	37.4	58.1	40.4	21.2	41.0	48.1	2015
Sparse R-CNN [406]	R50-FPN	106	174	45.0	64.1	49.4	26.9	47.2	59.5	2021
One-stage CNN detectors										
SSD512 [411]	VGG-16	27	88	28.8	48.5	30.3	10.9	31.8	43.5	2016
RetinaNet [412]	R50-FPN	38	198	36.5	55.0	39.1	19.9	40.3	48.7	2017
YOLOv3 [408]	DarkNet-53	62	156	33.0	57.9	34.4	18.3	35.4	41.9	2018
Modern YOLO										
YOLOv5-s [410]	CSPDarkNet	7.2	16.5	37.4	–	–	–	–	–	2020
YOLOv8-x [413]	CSP-ELAN	68.2	257.8	53.9	–	–	–	–	–	2023
YOLOv10-l [414]	CSP-ELAN	24.4	120.3	53.2	–	–	–	–	–	2024
Transformer set-prediction detectors										
DETR [136]	R50, 500 ep	41	86	42.0	62.4	44.2	20.5	45.8	61.1	2020
Deformable DETR [415]	R50, 50 ep	40	173	43.8	62.6	47.7	26.4	47.1	58.0	2021
DINO [417]	R50, 4-sc., 12 ep	47	279	49.0	66.6	53.5	32.0	52.3	63.0	2023
RT-DETR [418]	R50	42	136	53.1	71.3	57.7	34.8	58.0	70.0	2024

with complex everyday scenes and segmentation masks alongside boxes, then became the standard for both detection and segmentation. **Open Images** [442], **Objects365** [441], and **LVIS** [443] scale the vocabulary to hundreds and then over a thousand categories, with LVIS deliberately emphasizing rare classes, which turned it into the canonical long-tail and open-world benchmark.

Evaluation follows the *precision–recall* framework. A detection $(\hat{c}, \hat{b}, \hat{s})$ is correct when $\hat{c}$ matches the ground-truth class and $\hat{b}$ overlaps the ground-truth box b above an *Intersection over Union* (IoU) threshold

$$\mathrm{IoU}(\hat{b}, b) = \frac{|\hat{b} \cap b|}{|\hat{b} \cup b|}. \tag{6.3}$$

Sorting the detections of a class by confidence traces a precision–recall curve whose area is the *Average Precision* (AP); the *mean Average Precision* (mAP) averages AP over classes. Benchmarks differ chiefly in the IoU threshold: PASCAL VOC uses a single $\mathrm{IoU} = 0.5$ (AP_{50}), whereas COCO averages over ten thresholds from 0.5 to 0.95 to obtain its primary AP and reports AP_{50} and AP_{75} as diagnostic slices. COCO further stratifies by object size into AP_S (area $< 32^2$ pixels), AP_M (32^2–96^2), and AP_L ($> 96^2$), which exposes the persistent difficulty of small objects.

3D detection is benchmarked mainly on driving data, KITTI [173], nuScenes [445], and the Waymo Open Dataset [446], and indoors on SUN RGB-D [447] and ScanNet [448]; Omni3D (Table 6.1) pools image-based 3D annotations from both. The IoU above is computed between 3D boxes or their BEV footprints: KITTI reports 3D and BEV AP at IoU 0.7 for cars and 0.5 for pedestrians and cyclists, and indoor benchmarks report mAP at 3D IoU 0.25 and 0.5. nuScenes matches detections by center distance on the ground plane (0.5, 1, 2, and 4 m) instead of IoU and folds translation, scale, orientation, velocity, and attribute errors into its nuScenes Detection Score (NDS); Waymo weights AP by heading accuracy (APH).

Table 6.2 reports COCO val2017 results across the three families. On the canonical ResNet-50 backbone, AP has climbed from 37.4 for Faster R-CNN to 53.1 for RT-DETR over roughly a decade, with much of the recent gain attributable to transformer-based set prediction (DETR $\rightarrow$ Deformable DETR $\rightarrow$ DINO) and improved denoising-style query initialization. The AP_S column highlights the well-documented small-object weakness of vanilla DETR (20.5); deformable attention and denoising progressively close this gap (Deformable DETR 26.4, DINO 32.0, RT-DETR 34.8). On the efficiency frontier, the modern YOLO line (YOLOv8-x at 53.9, YOLOv10-l at 53.2) matches RT-DETR in accuracy at substantially higher throughput, so the traditional accuracy–speed divide between CNN one-stage detectors and transformer set prediction has largely closed.

6.1.1.4 Challenges

The accuracy–efficiency trade-off remains unresolved. The transformer-based detectors that top COCO and open-vocabulary leaderboards are still expensive at inference: even with deployment stacks such as TensorRT, their latency and parameter counts rule out most mobile and embedded targets, while the lightweight real-time detectors that fit those budgets trail them in accuracy. The same tension recurs for every recognition task in this section. **Open-world detection degrades on the hardest instances.** Open-vocabulary detectors extend recognition beyond a fixed label set, yet still miss small objects and mislabel rare categories, precisely the long-tail regime that motivated LVIS. **Detection inside multimodal large language models is still immature.** Recent systems fold detection into MLLMs as a grounding primitive that ties image regions to language, but even the strongest closed-source models post markedly lower numbers than specialist detectors on standard benchmarks [424, 425]. **3D detection is bounded by depth.** For camera-only detectors, localization error is dominated by depth error, which grows with distance [449]; LiDAR measures depth directly but its returns thin out with range, so distant and small objects are seen through a handful of points.

6.1.2 Segmentation

Segmentation partitions an image, a video, or a 3D scan into meaningful regions. Before deep learning it was a mid-level grouping problem, clustering pixels by similarity somewhere between low-level edge detection and high-level recognition; learned models turned it into dense prediction, with class labels, instance identities, and temporal consistency attached to every region. Figure 6.2 shows the principal segmentation map types produced on a common scene.

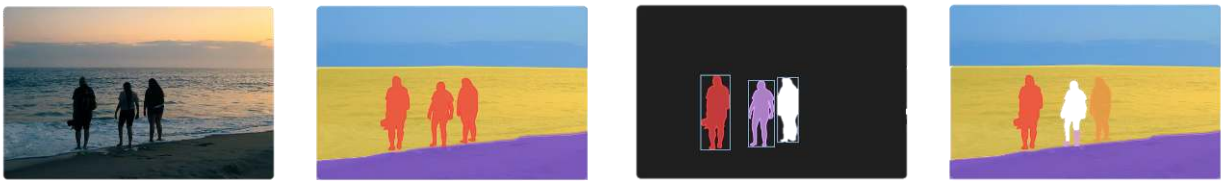


Figure 6.2: Examples of different segmentation maps: semantic, instance, and panoptic on a shared scene.

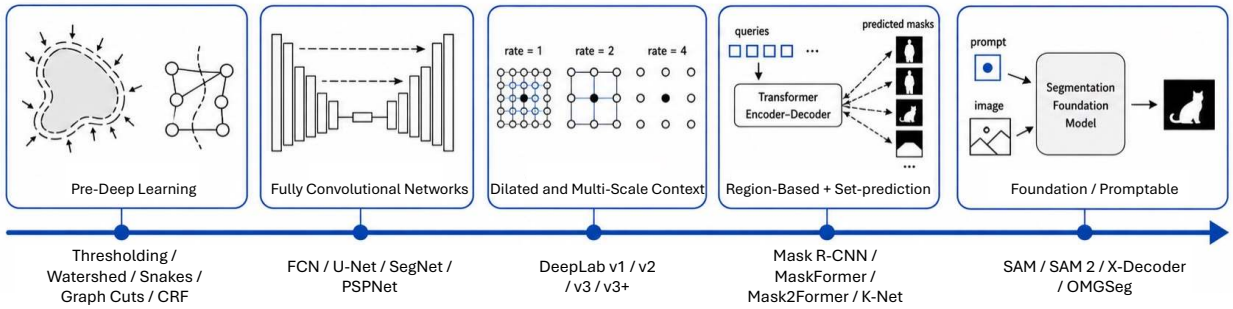


Figure 6.3: Methodological progression of image and video segmentation, from pre-deep-learning methods to foundation and promptable segmenters, with representative methods under each stage.

6.1.2.1 Task Definition

Semantic Segmentation: Given an input image $I \in \mathbb{R}^{H\times W\times 3}$, *semantic segmentation* assigns a class label to every pixel, producing a dense label map

$$S \in \{1,\dots,C\}^{H\times W}, \tag{6.4}$$

where C is the number of classes; instances of the same class are not distinguished.

Instance and Panoptic Segmentation: *Instance segmentation* detects and segments each object instance separately. The output is a set of labeled masks

$$\mathcal{Y} = \{(m_i, c_i)\}_{i=1}^{N}, \tag{6.5}$$

where $m_i \in \{0,1\}^{H\times W}$ is the binary mask of the i-th instance and c_i its class; masks may overlap, and only countable *thing* classes are covered. *Panoptic segmentation* keeps the same output form but requires the K masks to form a non-overlapping partition of the image that covers *thing* instances and *stuff* regions alike, each stuff class as a single mask.

Video Segmentation: Given a video clip $V \in \mathbb{R}^{T\times H\times W\times 3}$, *video segmentation* predicts temporally consistent masks across frames, output as a set of mask tubes

$$\mathcal{Y} = \{(m_i, c_i)\}_{i=1}^{N}, \quad m_i \in \{0,1\}^{T\times H\times W}, \tag{6.6}$$

each tracking an object or region over time, with semantic, instance, and panoptic variants as above.

3D Segmentation: Given a point cloud $\mathcal{P} = \{\mathbf{p}_j\}_{j=1}^{M}$ with $\mathbf{p}_j \in \mathbb{R}^3$ and optional color or intensity, *3D semantic segmentation* labels every point,

$$S \in \{1,\dots,C\}^{M}, \tag{6.7}$$

and 3D instance and panoptic segmentation predict point masks $m_i \in \{0,1\}^{M}$ under the same set form and overlap constraints as their image counterparts; voxels or mesh vertices take the place of points for other 3D representations. The definitions carry over unchanged. What changes is the input: the points are unordered, unevenly spaced, and sparse, and that is where 3D methods depart from image methods.

6.1.2.2 Methodology

Figure 6.3 summarizes the method families surveyed below.

Before Deep Learning. Classical segmentation grouped pixels either by appearance or by minimizing an energy. The first family includes *thresholding* of pixel intensities [450], *region growing* that merges similar neighbors [451], *k-means clustering* in feature space [452], and the *watershed* transform, which treats the image as a topographic surface and segments by catchment basins [453]. The second family includes *active contours (snakes)* that evolve a curve toward object boundaries [454], *graph cuts* that solve certain segmentation

energies to global optimality [455], and *Markov* and *conditional random fields* that impose spatial coherence probabilistically [456, 457]; the last of these carried over directly into early learning-based segmenters.

Fully Convolutional Networks. FCNs [127] replace fully connected layers with convolutions, enabling end-to-end dense prediction at arbitrary input size. U-Net [174] adds a symmetric encoder–decoder with skip connections that fuse low- and high-level features, SegNet [458] reuses max-pooling indices for cheap upsampling and boundary recovery, and PSPNet [459] aggregates global and local context through pyramid pooling.

Dilated and Multi-Scale Context Modeling (DeepLab Series). The DeepLab series introduces atrous (dilated) convolution to enlarge the receptive field without losing resolution (v1 [460]), Atrous Spatial Pyramid Pooling for multi-scale context (v2 [461], refined with batch normalization and stronger backbones in v3 [462]), and an encoder–decoder that recovers boundary detail from low-level features (v3+ [463]).

Region-Based Mask Prediction (Mask R-CNN Family). Mask R-CNN [131] extends Faster R-CNN with a parallel mask branch and ROIAlign, which removes the quantization error of ROI pooling and makes pixel-accurate masks possible. PANet [464], Hybrid Task Cascade [465], and PointRend [466] refine it through better feature flow, cascaded box–mask prediction, and point-wise boundary rendering respectively; this two-stage, region-based family dominated instance segmentation until transformers arrived.

Transformer Backbones and Representation Learning. Transformer-based segmentation, following DETR, favors unified query-based formulations that bridge detection and segmentation, predict masks end-to-end without hand-crafted post-processing, and allow joint learning across tasks and modalities. The first step was the backbone: SETR [467] applies a pure ViT to semantic segmentation with a simple decoder, SegFormer [468] pairs a lightweight hierarchical transformer with an MLP decoder for a favorable accuracy–efficiency trade-off, and Swin Transformer [469] uses shifted-window attention to produce CNN-like multi-scale features and has become a default segmentation backbone. Self-supervised pretraining such as MAE [470] supplies ViT backbones that transfer well to dense prediction.

Query-based Transformer Segmentation. Query-based methods cast segmentation as set prediction with object or mask queries. MaX-DeepLab [471] is the first purely mask-based panoptic framework without box supervision; MaskFormer [472] unifies semantic, instance, and panoptic segmentation by predicting mask embeddings with class labels; K-Net [473] treats segmentation as kernel prediction with iterative refinement; and Mask2Former [474] adds masked cross-attention and multi-scale decoding, setting a strong baseline across all three tasks. QueryInst [475] and Panoptic SegFormer [476] are variants that respectively generate instance-specific kernels from queries and decouple semantic from instance queries.

Video and Multimodal Transformer-based Segmentation. VisTR [477] predicts spatio-temporal mask tubes end-to-end from object queries; Video K-Net [478] unifies video semantic, instance, and panoptic segmentation by associating queries across time; and TubeFormer [479] adds tube-level attention for long-term consistency. X-Decoder [480] and OMGSeg [481] use multimodal pretraining to cover open-vocabulary, class-agnostic, image and video segmentation within a single model.

SAM series. SAM [138] made segmentation promptable: trained on the SA-1B dataset of over one billion masks, it decouples a heavy image encoder from a light prompt-conditioned mask decoder and produces masks from points, boxes, or text with strong zero-shot generalization. SAM 2 [482] extends this to video, propagating masks across frames with a memory mechanism while keeping interactive control.

3D Point Cloud Segmentation. Point-cloud segmenters split by how they impose structure on the points (Section 4.2.3). Point-based networks consume raw points: PointNet [132] applies a shared MLP to every point and pools symmetrically, PointNet++ [270] adds hierarchical neighborhoods, DGCNN [483] builds k-nearest-neighbor graphs that are recomputed in feature space at every layer, KPConv [484] defines

Table 6.3: Summary of representative segmentation datasets across 2D and 3D modalities; split sizes count images (or LiDAR scans).

Dataset	Purpose	Classes	Resolution	Samples		
				Train	Valid	Test
• *2D*						
Pascal VOC [444]	Generic	21	Variable	1464	1449	Private
MS COCO [440]	Generic	80+	Variable	82,783	40,504	81,434
ADE20K [492]	Generic	150	Variable	20,210	2,000	3,000
COCO-stuff [493]	Generic	172	Variable	118,000	5,000	20,000
IDD [494]	Urban/Driving	34	Variable	6,993	981	2,029
Dark Zurich [495]	Urban/Driving	19	1920×1080	8,377	50	151
ACDC [496]	Driving	30 (19)	2048× 1024	1,600	406	2,000
PartImageNet [497]	Generic-Part	158	Variable	16,540	2,957	4,598
SiftFlow [498]	Outdoor	33	256×256	2,688	-	-
BSD [499]	Generic	30	Variable	12,000	-	-
Youtube-Objects [500]	Generic/Video	10	480×360	126	-	-
Cityscapes [501]	Urban	30 (8)	2048 ×1024	2975	500	1525
CamVid [502]	Urban/Driving	32	960×720	701	-	-
Mapillary Vistas [503]	Street	65	-	18,000	-	-
• *3D*						
NYU-D V2 [504]	Indoor	40 (13)	640×480	795	-	654
SUN-3D [505]	SfM/Video	-	640×480	415 seqs	-	-
SUN RGB-D [447]	Indoor	37	Variable	5,285	-	5,050
ScanNet [448]	Indoor	20	Variable	1,201	312	100
SemanticKITTI [506]	LiDAR/Driving	28	Point cloud	19,130	4,071	20,351

convolution kernels by points placed in space, and RandLA-Net [485] relies on random sampling to scale to scenes of millions of points. Sparse voxel networks such as MinkowskiNet [486] and SPVNAS [176] quantize the points and convolve only occupied voxels, which keeps 3D convolution affordable on room- and street-scale scans. Projection methods take the opposite route: RangeNet++ [487] unrolls a LiDAR sweep into a range image, runs a 2D segmenter in real time, and projects the labels back onto the points. Transformers entered with Point Transformer [282], and Point Transformer V3 [488] replaces exact neighbor search with serialized point orderings so that attention scales to large scenes.

Instance segmentation in 3D retraces the 2D history. PointGroup [489] predicts per-point offsets toward instance centers and clusters the shifted points, a bottom-up grouping; Mask3D [490] applies Mask2Former-style queries to sparse voxel features and predicts instance masks directly. Because 3D labels remain scarce, open-vocabulary 3D segmentation mostly borrows from 2D foundation models: OpenScene [39] distills CLIP-aligned pixel features into points, and OpenMask3D [491] classifies class-agnostic 3D instance masks with CLIP features aggregated over multi-view images.

6.1.2.3 Datasets and Metrics

Table 6.3 summarizes the main benchmarks. **PASCAL VOC** [444] anchored the classical semantic-segmentation era with 20 foreground classes and no instance labels. **MS COCO** [440], with semantic, instance, box, and panoptic annotations over complex object-centric scenes, is the shared testbed for all three modern task types. **Cityscapes** [501] and **Mapillary Vistas** [503] specialize to street scenes, the latter stressing long-tail classes and diverse geographies, and **ADE20K** [492] targets holistic scene parsing over 150 object and stuff classes. The 3D rows of the table cover indoor RGB-D scans (NYU-D V2, SUN RGB-D, ScanNet) and outdoor LiDAR (SemanticKITTI), and S3DIS [507] is a further indoor standard.

Table 6.4: Reported performance of representative segmentation methods on standard benchmarks: ADE20K mIoU and Cityscapes mIoU (semantic), COCO Mask AP (instance), and COCO PQ (panoptic). Numbers follow each paper's own protocol (split, single-/multi-scale) and are indicative only; "–" marks a metric not reported.

Method	Backbone	ADE20K mIoU	Cityscapes mIoU	COCO Mask AP	COCO PQ	Year
•*FCN family and dilated convolution*						
FCN-8s [127]	VGG-16	29.4	62.7	–	–	2015
PSPNet [459]	R101	43.3	80.2	–	–	2017
DeepLab v3+ [463]	Xception-65	45.6	82.1	–	–	2018
• *Region-based instance segmentation*						
Mask R-CNN [131]	R50-FPN	–	–	35.4	–	2017
HTC [465]	R50-FPN	–	–	38.4	–	2019
• *Transformer backbones*						
SETR-MLA [467]	ViT-L	50.3	82.2	–	–	2021
SegFormer-B5 [468]	MiT-B5	51.0	84.0	–	–	2021
Swin-L + UperNet [469]	Swin-L	53.5	–	–	–	2021
• *Universal query-based segmentation*						
MaskFormer [472]	Swin-L	55.6	–	–	52.7	2021
Mask2Former [474]	Swin-L	57.7	–	50.1	57.8	2022

Segmentation evaluation reuses IoU (Section 6.1.1) with masks in place of boxes. For semantic segmentation the headline metric is the *mean Intersection over Union* (mIoU),

$$\mathrm{mIoU} = \frac{1}{C}\sum_{k=1}^{C}\frac{\mathrm{TP}_k}{\mathrm{TP}_k + \mathrm{FP}_k + \mathrm{FN}_k}, \tag{6.8}$$

where C is the number of classes and TP_k, FP_k, FN_k count the true-positive, false-positive, and false-negative pixels for class k. Because the average is unweighted, mIoU is sensitive to rare classes, which is one motivation behind ADE20K and Mapillary Vistas; pixel accuracy is dominated by frequent classes (*e.g.*, sky, road) and is less informative.

For *instance segmentation*, the COCO mask-AP protocol matches predictions to ground truth by mask IoU and averages AP over IoU thresholds 0.5–0.95. For *panoptic segmentation*, *Panoptic Quality* [243] is $\mathrm{PQ} = \mathrm{SQ} \times \mathrm{RQ}$, the average IoU of matched segments times the F1 score over matched, false-positive, and false-negative segments, treating things and stuff uniformly. For *video object segmentation*, the DAVIS $\mathcal{J}\&\mathcal{F}$ metric averages region similarity ($\mathcal{J}$, mask IoU) and contour accuracy ($\mathcal{F}$, computed on mask boundaries). 3D semantic segmentation reports the same mIoU computed over points or voxels, and 3D instance segmentation on ScanNet reports AP at mask IoU thresholds averaged over 0.5–0.95, with AP_{50} and AP_{25} as looser slices.

Table 6.4 reports results across these four protocols. The clearest trend is on semantic segmentation, where ADE20K mIoU has roughly doubled in under a decade, from FCN-8s's 29.4 to Mask2Former's 57.7; the largest single jumps track two shifts, first to transformer backbones (SETR, SegFormer, Swin-L) and then to universal query-based formulations. That second shift is what unifies the table: MaskFormer is the first entry to report semantic mIoU and panoptic PQ within one framework, and Mask2Former extends the same architecture to all three tasks with competitive results on each. Instance segmentation followed a partly separate path, climbing from Mask R-CNN (35.4) and HTC (38.4) on R50-FPN to 50.1 once Mask2Former is paired with a Swin-L backbone, which confirms that transformer backbones lift mask quality at the instance level as well, not only semantic accuracy.

6.1.2.4 Challenges

The cost of self-attention at high resolution, over long video, and on dense 3D data sharpens the accuracy–speed tension noted for detection, and two problems are specific to segmentation. **Long-video, multi-object consistency is hard.** Holding spatial precision and temporal coherence together remains difficult in cluttered scenes, particularly for video panoptic segmentation and segmentation-based multi-object tracking, where objects enter, leave, and occlude one another over long horizons. **Segmentation is moving toward the open world.** The field is shifting from fixed label sets to class-agnostic, open-vocabulary, and multimodal settings, in which language and other auxiliary signals are folded in to improve spatial reasoning and adaptability, a direction that SAM, X-Decoder, and OMGSeg already anticipate.

6.1.3 Keypoint and Landmark Detection

Keypoint and landmark detection localizes a set of semantically consistent, precisely positioned points that characterize object geometry in images or videos: eye corners, skeletal joints, or structurally distinctive parts of man-made objects. Compared with region-based representations, keypoints give a sparser, explicitly geometric encoding that often survives deformation, rotation, and partial occlusion. Figure 6.4 summarizes the task families and the method families surveyed below.

6.1.3.1 Task Definition

Given an input image I, a parametric model f_θ estimates a set of point locations

$$f_\theta(I) = \{\mathbf{p}_i\}_{i=1}^{N}, \quad \mathbf{p}_i \in \mathbb{R}^2, \tag{6.9}$$

where each $\mathbf{p}_i$ is the image coordinate of a salient visual or structural element. Landmark detection fixes a predefined semantic identity for each point (a particular eye corner, a particular joint); the two terms are otherwise used interchangeably.

3D keypoint detection predicts $\mathbf{p}_i \in \mathbb{R}^3$ instead. From images, the depth of each point must be inferred; from depth maps or point clouds $\mathcal{P}$, it is observed, and the detector chooses or regresses points on the measured surface. For human pose, 3D joints are usually expressed in camera coordinates relative to a root joint (the pelvis), which leaves global depth unconstrained, as a single image cannot fix it.

The task varies along three axes. The input modality sets what geometry is directly available: most detectors work on 2D images (RGB, grayscale, or infrared), where keypoints must be inferred from appearance alone, whereas point clouds expose spatial structure directly; RGB-D or RGB-infrared fusion adds robustness in clutter and poor lighting, and video supplies temporal continuity through motion blur and transient occlusion. The prediction target ranges from planar image coordinates, which underpin most benchmarks, to 3D keypoints required in robotics and autonomous driving, to 4D trajectories tracked over time. The application scenario fixes what the points mean. Research has concentrated on humans: body joints for pose estimation, facial landmarks for expression recognition and identity verification, and finger joints for gesture-based interaction and sign language recognition [508]. Keypoints are also defined on everyday items, industrial parts, and vehicles for manipulation, inspection, and driving (*e.g.*, monocular 3D detection from vehicle keypoints [509]), on animals for behavior analysis (multi-animal DeepLabCut [510]), and on anatomical structures in medical imaging, where universal landmark detectors cover several modalities and anatomies [511].

6.1.3.2 Methodology

Classical pipelines combined local feature detectors with statistical shape models that handled occlusion and deformation through explicit geometric constraints. Moravec [512], Harris [257], and Shi–Tomasi [513] corners were followed by SIFT [111], SURF [514], FAST [515], BRIEF [516], and ORB [517], which improved

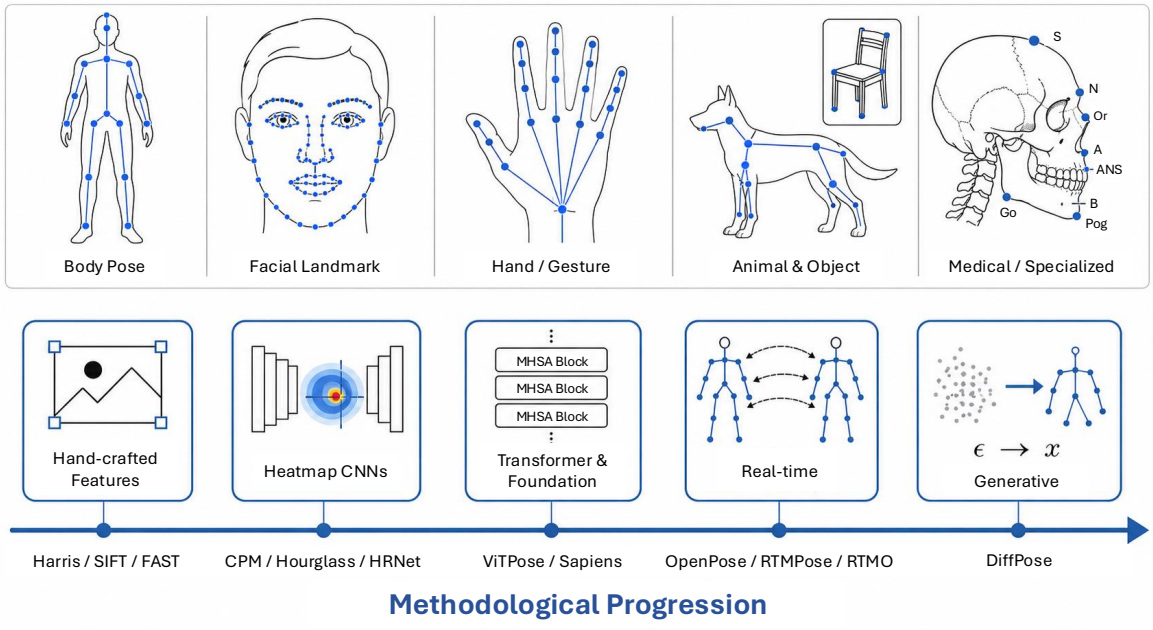


Figure 6.4: Keypoint and landmark detection: representative task families (top) and the methodological progression, mainly of 2D detectors, from hand-crafted feature detectors to generative models, with representative methods under each stage (bottom).

scale invariance, speed, and robustness. For faces and hands, active shape models [518], active appearance models [519], and constrained local models with regularized mean-shift fitting [520] provided the parametric frameworks.

Learning-based detection began with supervised descent for face alignment [521] and, in 2014, with DeepPose [258], which regressed human joints directly with a deep network. Heatmap regression then dominated through 2020: Tompson *et al*. [522, 523] coupled convolutional networks with graphical models, convolutional pose machines [524] and stacked hourglass networks [256] refined heatmaps over repeated stages, and HRNet [525] kept a high-resolution branch throughout the network for precise localization.

Multi-person estimation splits into two strategies. Top-down methods detect each person first and then estimate keypoints inside the box, which is accurate but scales with the number of people; bottom-up methods predict all keypoints at once and then group them, which is faster but makes grouping the hard part. Part Affinity Fields [526], consolidated as OpenPose [527], are the canonical bottom-up design; YOLO-Pose [528], RTMPose [529], and RTMO [530] push real-time multi-person estimation across both paradigms.

Vision transformers and foundation models made keypoint detection general-purpose: ViTPose [531] set simple transformer baselines for human pose in 2022, and Sapiens [532] in 2024 provides a single foundation model for human-centric vision tasks.

3D human pose estimation grew out of these 2D detectors. Lifting methods take detected 2D keypoints as input and regress the 3D joints: a fully connected network on 2D joints alone was a strong baseline [533], temporal convolutions over 2D pose sequences (VideoPose3D [534]) and spatio-temporal transformers (MotionBERT [535]) use motion to reduce the depth ambiguity, and DiffPose [536] casts lifting as diffusion. Direct methods predict 3D from pixels, as in the volumetric heatmaps of Pavlakos *et al*. [537]. With a calibrated multi-view rig most of the ambiguity disappears, and learnable triangulation [538] fuses per-view

Table 6.5: Reported performance of representative keypoint detection methods on COCO Keypoints (val2017). AP is computed under the OKS-based protocol.

Method	Backbone	Input	#Params (M)	GFLOPs	AP	AP_{50}	AP_{75}	AR	Year
HRNet-W32 [525]	HRNet-W32	256×192	28.5	7.1	74.4	90.5	81.9	79.8	2020
HRNet-W48 [525]	HRNet-W48	384×288	63.6	32.9	76.3	90.8	82.9	81.2	2020
YOLOv5s6-Pose [528]	YOLOv5-s	960×960	15.1	–	63.8	87.6	69.6	70.4	2022
YOLOv5m6-Pose [528]	YOLOv5-m	960×960	41.4	–	67.4	89.1	73.7	73.9	2022
YOLOv5l6-Pose [528]	YOLOv5-l	960×960	87.0	–	69.4	90.2	76.1	75.9	2022
ViTPose-B [531]	ViT-B	256×192	86	–	75.8	–	–	81.1	2022
ViTPose-L [531]	ViT-L	256×192	307	–	78.3	–	–	83.5	2022
ViTPose-H [531]	ViT-H	256×192	632	–	79.1	–	–	84.1	2022
RTMPose-s [529]	CSPNeXt-s	256×192	5.5	0.68	69.6	–	–	–	2023
RTMPose-m [529]	CSPNeXt-m	256×192	13.6	1.93	73.6	–	–	–	2023
RTMPose-l [529]	CSPNeXt-l	256×192	27.7	4.16	74.8	–	–	–	2023
RTMO-s [530]	CSPDarkNet	640×640	9.9	–	67.7	87.8	73.7	71.5	2024
RTMO-m [530]	CSPDarkNet	640×640	22.6	–	70.9	89.0	77.8	74.7	2024
RTMO-l [530]	CSPDarkNet	640×640	44.8	–	72.4	89.9	78.8	76.2	2024

heatmaps into a 3D estimate. Recovering a full body mesh rather than a joint set is taken up with human motion in Section 8.4.

Classical corner detectors were in turn replaced by learned ones: SuperPoint [539] learns detection and description jointly under self-supervised homographic adaptation, DISK [540] selects keypoints with a reinforcement-style objective, and SuperGlue [541] and the detector-free LoFTR [542] recover dense correspondences directly. These serve as correspondence rather than semantic points and are the workhorses of the structure-from-motion, visual-odometry, and SLAM pipelines in the spatial-localization section that follows.

On point clouds, keypoints mostly serve registration and pose estimation. Intrinsic Shape Signatures [543] select points whose local neighborhood has a distinctive eigenvalue structure; USIP [544] learns, without labels, detectors that stay stable under rigid transformation, and D3Feat [545] predicts a detection score and a descriptor for every point, the 3D analogue of SuperPoint. Category-level 3D keypoints are a different kind: Suwajanakorn *et al*. [546] discover keypoints that are consistent across instances of a category through a relative-pose objective, and kPAM [547] uses such keypoints as the interface for robotic manipulation. Keypoint-based 6D object pose, which predicts the 2D projections of 3D model keypoints and solves PnP, belongs to pose estimation (Section 6.2.5).

Table 6.5 reports COCO Keypoints (val2017) results for the CNN, transformer, and real-time families.

6.1.3.3 Datasets and Metrics

Keypoint datasets are mostly embedded in broader pose and action benchmarks rather than built in isolation; Table 6.6 lists the most used. Human body datasets are the most mature and trace the field's expansion: MPII [548] and COCO Keypoints [440] fixed in-the-wild 2D evaluation, PoseTrack [549] added temporal association in video, Human3.6M [550] supplies controlled multi-view 3D ground truth for 2D-to-3D lifting, DensePose [551] replaces sparse joints with dense surface correspondences, and CrowdPose [552] stresses crowded, occluded scenes. Face datasets moved from large pose variation (AFLW [553]) and high-density annotation (HELEN [554]) to occlusion (COFW [555]), the standardized 68-point protocol of 300-W [556] and its large-pose synthetic extension 300-W-LP [557], and the 98-point, hard-condition WFLW [558]. Hand datasets contend with fine articulation and self-occlusion through multi-view capture (CMU Panoptic HandDB [559]), synthetic 3D labels (RHD [560]), real 3D pose and shape (FreiHAND [561]), and interacting hands (InterHand2.6M [562]). Animal-Pose [563] and AP-10K [564] carry the human recipe to animals across

Table 6.6: Representative datasets for keypoint and landmark detection across human body, face, hand, animal, and object domains. "KP" denotes the number of annotated keypoints per instance; "Mod." denotes modality (2D image, 3D pose, or multi-view). Sample counts follow the official splits where available; "–" marks values not provided by the source paper.

Dataset	Year	Domain	Mod.	KP	#Samples (train / val / test)
• *Human body*					
MPII [548]	2014	In-the-wild body	2D	16	25K images (train+test)
COCO Keypoints [440]	2014	In-the-wild multi-person	2D	17	57K / 5K / 20K images
PoseTrack [549]	2017	Video body tracking	2D	15	593 / 170 / 375 videos
Human3.6M [550]	2014	Indoor multi-view body	3D	32	3.6M frames, 11 subjects
DensePose [551]	2018	Dense body surface	2D	SMPL surface	∼50K images (COCO subset)
CrowdPose [552]	2019	Crowded multi-person	2D	14	10K / 2K / 8K images
• *Face*					
AFLW [553]	2011	In-the-wild face	2D	21	25K faces (22K images)
HELEN [554]	2012	High-resolution face	2D	194	2K / – / 330 images
COFW [555]	2013	Occluded face	2D	29	1.3K / – / 507 images
300-W [556]	2013	Aggregated face	2D	68	3.1K / – / 689 images
300-W-LP [557]	2016	Large-pose synthetic face	2D	68	61K images
WFLW [558]	2018	In-the-wild face (challenging)	2D	98	7.5K / – / 2.5K images
• *Hand*					
CMU Panoptic HandDB [559]	2017	Multi-view hand	Multi-view	21	14K manual + bootstrapped
RHD [560]	2017	Synthetic hand	2D + 3D	21	41K / – / 2.7K frames
FreiHAND [561]	2019	Real hand pose + shape	3D	21	33K / – / 4K samples
InterHand2.6M [562]	2020	Two-hand interaction	3D	42	2.6M frames
• *Animal & Object*					
Animal-Pose [563]	2019	5-class mammal pose	2D	20	∼4K images
AP-10K [564]	2021	54-species animal pose	2D	17	10K images
ApolloCar3D [565]	2019	Vehicle keypoints (driving)	2D	66	5.3K images
KeypointNet [566]	2020	Multi-category 3D objects	3D	5–23	8.3K models

species, while ApolloCar3D [565] and KeypointNet [566] define keypoints on rigid vehicles and 3D object categories.

Metrics measure the distance between predicted and ground-truth points, normalized for scale. The End-Point Error (EPE) is the raw Euclidean distance. The Normalized Mean Error (NME) divides it by a reference length:

$$\mathrm{NME} = \frac{1}{N}\sum_{i=1}^{N}\frac{\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2}{d}, \tag{6.10}$$

where $\hat{\mathbf{x}}_i$ and $\mathbf{x}_i$ denote the predicted and ground-truth locations of the i-th keypoint, and d is a task-dependent normalization factor (*e.g.*, inter-ocular distance for faces). The Percentage of Correct Keypoints (PCK) instead thresholds the error,

$$\mathrm{PCK@}\alpha = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2 \le \alpha \cdot s)\,, \tag{6.11}$$

where α controls the tolerance and s is the reference scale, giving an accuracy rate that is easy to compare across methods; the Cumulative Error Distribution (CED) curve and its Area Under the Curve (AUC) summarize behavior across thresholds.

For large-scale benchmarks, Object Keypoint Similarity (OKS) incorporates object scale and keypoint-specific uncertainty,

$$\mathrm{OKS} = \frac{\sum_i \exp\left(-\frac{\|\hat{\mathbf{x}}_i - \mathbf{x}_i\|_2^2}{2s^2\kappa_i^2}\right)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)}. \tag{6.12}$$

Here s denotes the object scale (the square root of the segmented area), κ_i is a per-keypoint normalization constant calibrating the fall-off, and $v_i \in \{0, 1, 2\}$ is the ground-truth visibility flag for keypoint i, where $v_i = 0$ indicates unlabeled, $v_i = 1$ indicates labeled but occluded, and $v_i = 2$ indicates labeled and visible. OKS plays the role of IoU in computing average precision and recall on MS COCO [440]. For 3D human pose, the Mean Per-Joint Position Error (MPJPE) averages the Euclidean joint error in millimeters after aligning the root joints; PA-MPJPE applies a Procrustes alignment of scale, rotation, and translation first, and 3DPCK thresholds the error at 150 mm, as on MPI-INF-3DHP [567]. Point-cloud keypoint detectors are scored by repeatability under rigid transformation and, downstream, by registration recall on benchmarks such as 3DMatch [568].

6.1.3.4 Challenges

Robustness in unconstrained scenes is still the first problem [569, 570]: occlusion, extreme pose, and non-rigid deformation leave partial or ambiguous configurations, and symmetric parts (left versus right limbs, the fingers of a self-occluded hand) are routinely swapped. Generalization is the second [571, 572, 573]: keypoint definitions differ across datasets even for the same object class, so models trained on one benchmark transfer poorly to another, and the gap widens for animals, objects, and anatomical structures, where annotation is scarce and demographic or species bias enters the training data. The accuracy–speed tension discussed for detection applies here as well [574, 575]: high-precision models are too heavy for embedded targets, and lightweight ones lose accuracy precisely under occlusion and fast motion.

6.1.4 Scene Classification

Going beyond objects, regions, and keypoints, scene classification (SC) assigns a single semantic label to a whole image describing the environment it portrays [576]. Where object recognition names the things in a picture ("car", "tree", "person"), scene classification names the *where*: "a busy street", "a serene beach", "a modern kitchen". It asks "*What kind of place is this?*", a judgment that rests on the global layout and gist of the image rather than on any single object [577, 578], and it supplies the holistic prior on which an embodied or virtual agent conditions its later behavior [579, 580, 219, 581].

6.1.4.1 Task Definition

Formally, scene classification learns a mapping $f : I \mapsto y \in \mathcal{Y}$ from an image I to a label y in a predefined category set $\mathcal{Y}$, or, in the multi-label case, to a subset $Y \subseteq \mathcal{Y}$ covering several facets of the scene [582] ("urban" for area, "parking lot" for place, "night" for condition). Beyond the single-/multi-label split, the task is instantiated by domain, with indoor [583] and outdoor [584] scenes, remote-sensing imagery [585, 586], and driving contexts such as highways and tunnels [580], and by modality, extending from still images to video [587] and acoustic scenes [588].

6.1.4.2 Methodology

Figure 6.5 summarizes the progression from hand-crafted global descriptors through deep features to vision–language models.

Early methods summarized the whole image into one discriminative vector of low-level texture, color, and gradient statistics. The Spatial Envelope model [589] describes a scene by a few perceptual dimensions

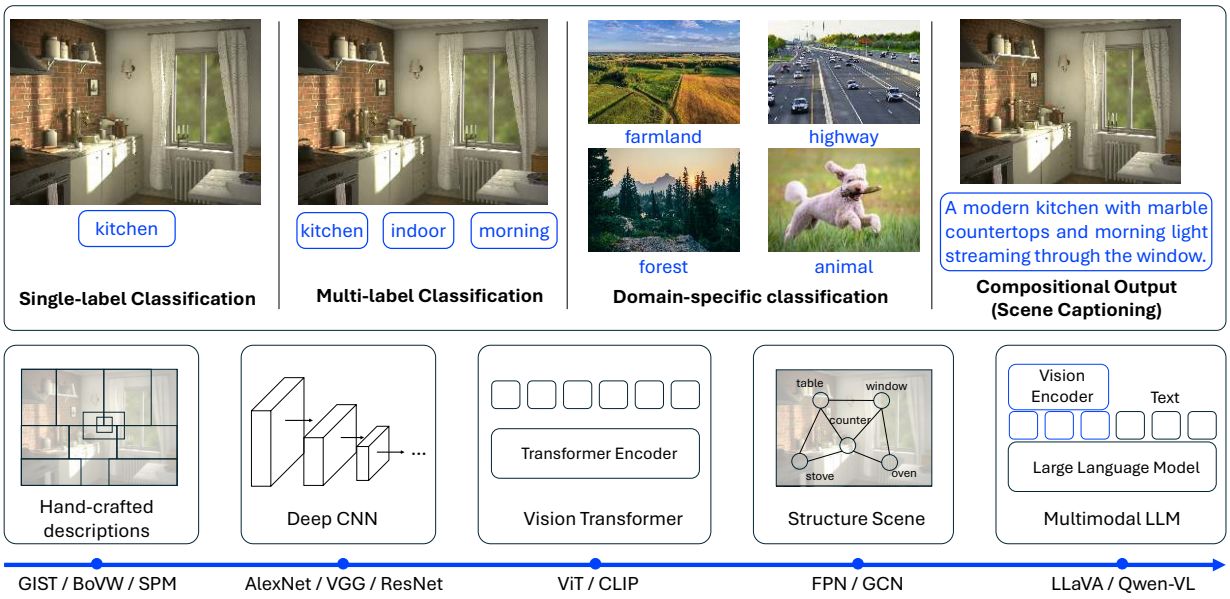


Figure 6.5: Scene classification: four task formulations, from single-label classification to compositional captioning output (top), and the methodological progression from hand-crafted descriptors to multimodal LLMs, with representative methods under each stage (bottom).

(naturalness, openness, roughness, expansion, ruggedness), a descriptor commonly called GIST. Bag-of-Visual-Words (BoVW) [590] treats the image as a histogram of quantized local features such as SIFT [111], discarding most spatial information [591]; Spatial Pyramid Matching (SPM) [592] restores coarse layout by computing BoVW histograms over a pyramid of sub-regions and concatenating them.

Deep networks learn scene features end to end. CNNs [125, 593] such as AlexNet [125], VGG [594], ResNet [23], and DenseNet [595], pre-trained on ImageNet [439, 125], proved to be strong global feature extractors for scenes, and Places-CNN [596] re-trained them on the large-scale Places benchmark to establish the standard transfer recipe. Vision Transformers [142] match or exceed CNNs with a class-token embedding that attends over all patches, in the manner of BERT [42].

A separate line models layout explicitly, since a stadium is recognized by its global oval while a classroom differs from an office by the mid-scale arrangement of desks and blackboard. Multi-scale pooling over the CNN feature hierarchy (FPN-style [597]) captures structure at several scales, and scene graphs [598] make objects and their relations explicit, so that propagating over them with graph convolutional networks [599] yields layout-aware scene features [600].

Vision–language models (VLMs) turn scene classification into open-vocabulary, zero-shot matching. Contrastive models such as CLIP [143] embed the image and the textual names of all candidate classes in one space and pick the most similar class [601], needing no fine-tuning beyond prompt design [602]; the recipe extends to 3D point clouds [368]. Autoregressive multimodal large language models (MLLMs; LLaVA [145], Qwen-VL [603], Gemma 3 [604]) instead generate the label as text in response to an instruction [605]. With rich semantics available for free, the research focus has shifted from one global label toward compositional understanding of what is inside the scene [606].

6.1.4.3 Datasets and Metrics

Table 6.7 lists the standard benchmarks. The small early sets, Scene 15 [592] and the indoor MIT 67 [583], were superseded by SUN 397 [607, 608], which fixed a near-exhaustive category list, and by Places [596, 609], which scaled the same categories to millions of images across its Places205, Places88, Places365-Standard,

Table 6.7: Representative scene classification datasets. "Mod." denotes modality; "–" marks a value not fixed by the source (scene-category subsets vary across protocols).

Dataset	Year	Domain	Mod.	#Classes	#Images (protocol)
Scene 15 [592]	2006	Indoor + outdoor, grayscale	2D	15	4,448
MIT Indoor 67 [583]	2009	Indoor	2D	67	15,620 (80 / 20 per class)
SUN 397 [607, 608]	2010	Indoor + outdoor	2D	397	108,754 (50 test per class)
Places205 [596]	2014	Indoor + outdoor	2D	205	2.5M
Places365-Standard [609]	2017	Indoor + outdoor	2D	365	∼1.8M
Places365-Challenge [609]	2017	Indoor + outdoor	2D	365	∼8M
NYU Depth v2 [504]	2012	Indoor	RGB-D	–	1,449 labeled pairs
SUN RGB-D [447]	2015	Indoor	RGB-D	–	10,335

Table 6.8: Reported performance of representative scene classification methods on Places-365 val and SUN 397. "–" marks a metric not reported by the corresponding source.

Method	Backbone	Places-365 Top-1	SUN 397 Top-1	Year
• *Supervised*				
Places365-AlexNet [609]	AlexNet	53.2	–	2017
Places365-VGG [609]	VGG-16	55.2	–	2017
Places365-ResNet [609]	R152	54.7	–	2017
• *Vision-language zero-shot*				
CLIP [143]	ViT-L/14	–	63.2	2021

and Places365-Challenge subsets. NYU Depth v2 [504] and SUN RGB-D [447] add aligned depth and a scene category per image for indoor RGB-D classification.

Scene classification is evaluated primarily with top-1 accuracy (the fraction of images for which the highest-scoring predicted class matches the ground truth) and top-5 accuracy (the fraction of images where the ground-truth class is among the top five predictions), with the latter especially useful on long-tail benchmarks such as Places-365 and SUN 397. For multi-label settings, the mean Average Precision (mAP) and class-wise F1 are reported instead, mirroring the protocol used in multi-label image recognition.

Table 6.8 reports results on the two most-cited scene benchmarks. Supervised CNNs on Places-365 cluster in a narrow band of roughly 53–55% top-1, with diminishing returns from AlexNet to ResNet, while CLIP reaches 63.2% top-1 on SUN 397 zero-shot [143], on par with supervised scene classifiers without any target-set training.

6.1.4.4 Challenges

Closed-set accuracy on Places-365, SUN 397, and MIT 67 is now recovered zero-shot by CLIP [143] and by MLLMs (LLaVA [145], Qwen-VL [603], Gemma 3 [604]), so "what kind of place is this" is no longer the frontier. The task has climbed a granularity ladder, from one label to attribute sets to grounded, compositional description across image, video, and 3D, and the open problems sit at that richer end, where MLLMs are weaker than their coarse-label accuracy suggests.

Fine-grained and long-tail discrimination. MLLMs still trail dedicated classifiers and CLIP-style models on fine-grained and long-tail categories, where the distinction between, *e.g.*, a "bistro" and a "brasserie" turns on subtle cues [610].

Compositional and spatial grounding. The natural output is no longer a label but a description whose claims can be grounded in specific objects and layouts [606]. MLLMs describe fluently but ground weakly: objects are referenced vaguely rather than by region or box, and the persistent failure is structural rather than semantic, since models name the constituents of a scene correctly yet err on relative layout, depth ordering, orientation, and cross-view or 3D consistency. The cause is representational: 2D patch tokens are optimized

for semantic alignment with language, not geometry, so fine spatial cues are lost once flattened into the language stream, and spatial-cognition benchmarks show even strong MLLMs degrading sharply as tasks move from naming toward layout and 3D reasoning [611, 612].

Hallucination and faithfulness. Free-form description reintroduces a failure mode that closed-set classification never had: MLLMs assert objects, attributes, or relations absent from the image, and their reasoning traces rarely point to verifiable spatial evidence [612].

6.2 Spatial Localization and Position Estimation

Recognition says what is present; it does not say where. Spatial localization and position estimation supply the missing geometry, situating recognized entities in a metric frame and recovering how they are oriented within it. The section climbs a ladder of geometric abstraction: from the per-pixel quantities of depth and surface normals, through metric attributes such as height and distance, to full SE(3) pose, first of the moving camera and then of rigid objects in the scene. Throughout, the concern is the static geometry of a single instant; how these quantities evolve is deferred to spatio-temporal perception.

6.2.1 Depth Estimation

Depth estimation predicts, for every pixel, the distance from the camera to the surface it observes, turning a 2D image into the 3D structure that reconstruction, navigation, and manipulation build on. Figure 6.6 summarizes the input configurations and the method families surveyed below.

6.2.1.1 Task Definition

Given an image I and optional auxiliary observations $\mathcal{V}$ (a second view, a multi-view set, a sparse sensor measurement, or neighboring video frames), a depth estimator predicts a depth map

$$f_\theta(I, \mathcal{V}) = D \in \mathbb{R}_+^{H \times W}, \tag{6.13}$$

where $D(u)$ is the distance along the optical axis from the camera center to the surface seen at pixel u. The task splits along three axes.

Sensing. Active methods emit light (structured light, time-of-flight, LiDAR) and measure its return, providing reliable metric depth even on textureless surfaces but tying accuracy to the sensor and degrading under strong ambient light or at long range. Passive methods infer depth from ordinary images alone; they need no special hardware and have become the dominant research line, at the cost of sensitivity to texture and illumination.

Scale. Metric depth gives distances in physical units, which 3D reconstruction and navigation require. Relative depth recovers only pixel ordering, or depth up to an affine transform; it cannot support metric measurement but is easier to supervise and generalizes better because heterogeneous datasets can be mixed without a consistent scale.

Density and Input Configuration. Classical pipelines produced sparse depth at feature points or semi-dense depth along high-gradient edges; dense per-pixel prediction is now the norm. The input configuration fixes how much geometry is available: active sensing measures depth directly; stereo recovers it from the disparity between two calibrated views; multi-view estimation enforces consistency across many views, which improves robustness to occlusion and matching ambiguity at the price of computation and known camera poses; monocular estimation has no geometric constraint at all and relies on learned statistical, structural, and semantic cues, which is why it is ill-posed in scale yet the most widely applicable setting.

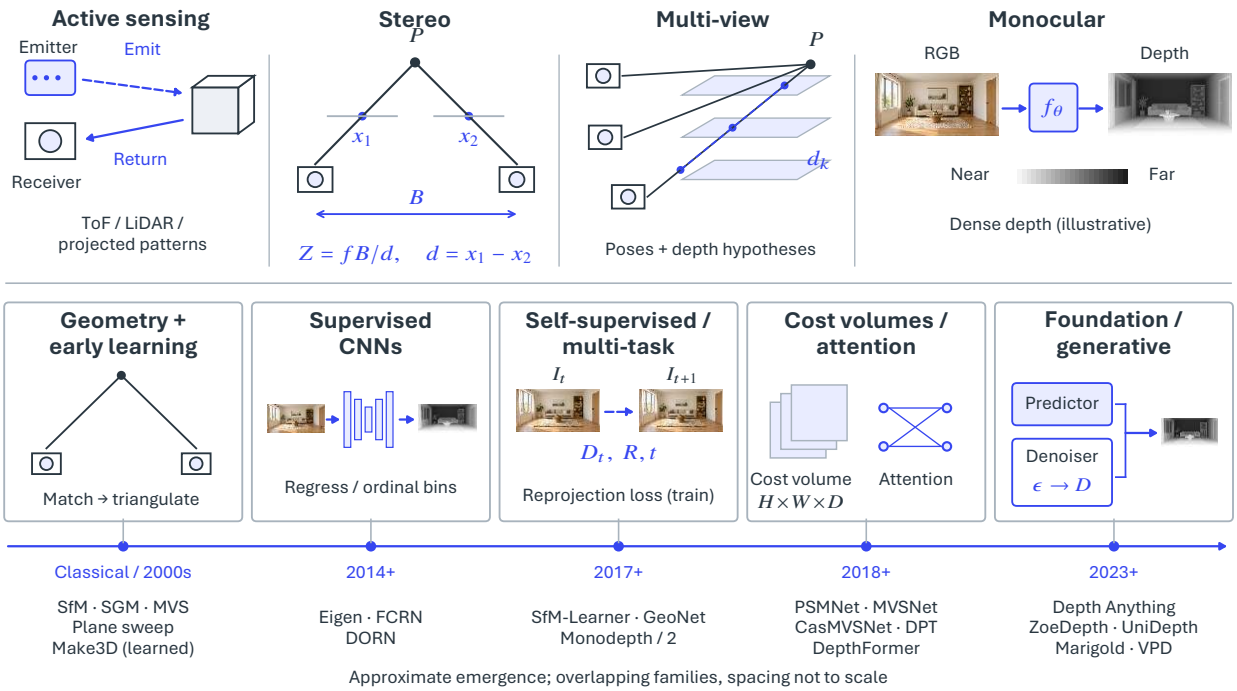


Figure 6.6: Depth estimation: four input configurations (top) and five overlapping method families with representative methods on an approximate emergence timeline (bottom). Cost volumes and attention are distinct mechanisms; denoising applies to the generative branch. The depth maps are schematic illustrations, with brighter values indicating nearer surfaces.

6.2.1.2 Methodology

Geometric Methods. Classical depth estimation solves geometry explicitly. In binocular stereo, two rectified cameras with focal length f and baseline B observe a point whose projections differ by the disparity $d = x_1 - x_2$; similar triangles give

$$Z = \frac{fB}{d}, \tag{6.14}$$

so depth follows once correspondences are found. Stereo matching is therefore a correspondence problem, usually posed as an energy that balances data fidelity against smoothness; Semi-Global Matching approximates the global optimum by aggregating costs along several 1D paths and remains a standard dense baseline. Structure from Motion (SfM) extends triangulation to many uncalibrated views, recovering camera poses and a sparse point cloud up to global scale, and supplies the poses that Multi-View Stereo (MVS) densifies. MVS typically proceeds by plane sweep: for a reference pixel u_r and a hypothesized depth d_k, back-project with the intrinsics K_r, transform into a source view with the relative pose (R, t), reproject, and keep the depth whose reprojections are most photometrically consistent across source views,

$$u_s \sim K_s \left(R\, d_k K_r^{-1} u_r + t \right), \qquad D(u_r) = \arg\min_{d_k} \sum_s \left\| I_s(u_s) - I_r(u_r) \right\|. \tag{6.15}$$

Plane sweep is thus triangulation turned into a search over depth planes, and it is the template that MVSNet [116] and CasMVSNet [613] later made differentiable as learned cost volumes.

Two single-image cues complete the classical picture. Depth from defocus exploits the thin-lens relation $1/f = 1/z + 1/z'$: a point off the focal plane blurs into a circle of confusion whose diameter grows with its distance from that plane, so measured blur can be inverted for depth. Shape from shading inverts a reflectance model, typically Lambertian, to recover surface orientation from intensity and integrates it

into depth; it is treated under surface normals in §6.2.2. Both are ill-posed under unknown illumination, reflectance, or optics and survive today mainly as auxiliary cues.

Active Depth Sensing. Active sensors measure depth rather than infer it. Time-of-flight systems time the round trip of emitted light: LiDAR scans the scene with laser beams and returns a sparse but long-range point cloud, whereas ToF cameras flood the scene and recover a dense depth map at every pixel, best suited to short-range indoor use. Structured light projects a coded pattern (the Kinect v1 used a pseudo-random infrared speckle) and triangulates from its deformation; active stereo projects texture onto the scene only to make ordinary stereo matching stable on featureless surfaces.

Early Learning-Based Methods. Data-driven depth began in the mid-2000s with monocular estimation. Saxena *et al*. [614] regressed depth from hand-crafted multi-scale features with a Markov random field enforcing spatial coherence, and Make3D [38] extended the model to the 3D position and orientation of scene patches while releasing a laser-scanned outdoor dataset that served as an early benchmark.

Supervised CNN Regression. Eigen *et al*. [55] first applied CNNs to monocular depth with a coarse-to-fine two-scale network, and their evaluation protocol became the standard; a follow-up [250] showed that the same architecture predicts depth, normals, and semantic labels with minor changes. Deep Convolutional Neural Fields [615] paired CNN unaries with a continuous CRF, FCRN [616] introduced a fully convolutional encoder–decoder with efficient upsampling of ResNet features, and DORN [617] recast depth as ordinal regression over a spacing-increasing discretization.

Learned Stereo and Multi-View Stereo. Learning entered the geometric pipelines first as a replacement for individual components: MC-CNN [618] computed stereo matching costs with a Siamese network in place of hand-crafted similarity. Mayer *et al*. [619] then showed that networks can regress disparity and scene flow end to end, and GC-Net [182] and PSMNet [620] embedded the cost volume and its regularization inside the network. The same move reached multi-view geometry with SurfaceNet [621] and MVSNet [116], which infer depth from several posed images, and R-MVSNet [622] and CasMVSNet [613] improved their memory footprint and resolution.

Self-Supervised and Multi-Task Learning. Ground-truth depth is costly, so a second line replaced it with geometric consistency. SfM-Net [623] and SfM-Learner [133] learn depth and camera motion jointly from unlabeled video by minimizing photometric reprojection error between adjacent frames; Monodepth [624] obtains the same signal from stereo pairs, and Monodepth2 [180] unifies both sources in one framework. GeoNet [625] adds optical flow to the joint estimate. Because all of these train a monocular network with stereo or multi-view supervision, the old separation between input configurations began to dissolve. Alongside, multi-task sharing (depth with semantic segmentation [626]), ImageNet-pretrained encoders with structural losses (DenseDepth [627, 124, 595]), local planar guidance (BTS [628]), and mixed-dataset training for zero-shot transfer (MiDaS [629]) each pushed accuracy and generalization.

Transformers and Foundation Models. Self-attention [141] gives every pixel access to global context, which helps precisely where local convolutions fail: textureless regions, occlusions, and large scenes. DPT [630] established the pattern of a Vision Transformer [142] encoder with a light dense decoder, and DepthFormer [631] combined global attention with local convolutional refinement. Scaling data and model size then produced depth foundation models that are trained once and transferred zero-shot: ZoeDepth [632] and Depth Anything [140] learn from large mixed corpora with flexible supervision and generalize across scenes and capture conditions without per-dataset adaptation.

Video and Generative Depth. Video adds temporal constraints. ManyDepth [633] aggregates neighboring frames into a cost volume at test time, whereas MonoViT [634] keeps the self-supervised video objective but strengthens single-frame reasoning with a Transformer. A separate line treats depth as conditional generation: VPD [635] showed that text-to-image diffusion backbones carry usable geometric features,

Marigold [636] reframed monocular depth as conditional denoising and obtained globally coherent maps under hard conditions, and GeoWizard [637] extended the recipe to joint depth and normal prediction. The same diffusion-as-perception idea recurs for surface normals and object pose later in this section.

Event-Based Depth. Event cameras record asynchronous per-pixel brightness changes [638, 639], offering microsecond latency and high dynamic range where frame cameras fail. Early event-based depth relied on geometric constraints and temporal accumulation [640, 641, 642, 643]; learned event stereo [644, 645], event–image fusion [646, 647], and self-supervised or hardware-aware frameworks [648, 649, 650] followed. The line remains a niche complement for extreme dynamics; see the survey [651].

6.2.1.3 Datasets and Metrics

Table 6.9 groups the main datasets into Type I, which ship explicit depth maps for direct supervision, and Type II, which require depth to be derived from LiDAR, stereo, or SfM. Three waves are visible. Indoor RGB-D collections (NYU Depth v2 [504], TUM RGB-D [652], SUN RGB-D [447], later ScanNet [448] and Matterport3D [340]) and the LiDAR-based KITTI [653, 654] fixed the indoor and outdoor protocols that still anchor the field, with Make3D [38] as the laser-scanned precursor. Relative-depth supervision from internet photos (DIW [655], ReDWeb [656], MegaDepth [657]) and synthetic data (Scene Flow [619]) then relaxed the need for metric sensors, while ETH3D [658], iBims-1 [659], ApolloScape [660], and Cityscapes stereo pairs [501] served precision evaluation and self-supervised training. The current wave is driven by autonomous driving and rendering: multi-sensor fleets (nuScenes [445], Waymo Open [446], Argoverse 2 [661], KITTI-360 [662], DDAD [663]) supply millions of LiDAR-calibrated frames, photorealistic synthetic sets (Hypersim [664], Virtual KITTI 2 [195], UrbanSyn [665], TartanAir [666]) supply exact ground truth even for transparent surfaces and hard lighting, and DIODE [667], BlendedMVS [668], ARKitScenes [669], Mannequin [670], and WildRGB-D [671] cover high-precision scans, handheld LiDAR, and dynamic or cluttered capture.

Depth is evaluated with pixel-wise regression metrics. The root mean squared error and its logarithmic variant,

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\hat{d}_i - d_i\right)^2}, \qquad \mathrm{RMSE}_{\log} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\log \hat{d}_i - \log d_i\right)^2}, \tag{6.16}$$

measure absolute deviation, the log form damping the dominance of far-range errors. Because an error of fixed size matters more at close range, relative errors normalize by the true depth,

$$\text{Abs Rel} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left|\hat{d}_i - d_i\right|}{d_i}, \qquad \text{Sq Rel} = \frac{1}{N}\sum_{i=1}^{N}\frac{\left(\hat{d}_i - d_i\right)^2}{d_i}, \tag{6.17}$$

and Abs Rel is the primary metric on most benchmarks. For monocular prediction, whose scale is ambiguous, the scale-invariant log error discounts a global scale shift,

$$\text{SI-Log} = \frac{1}{N}\sum_{i=1}^{N}\left(\log \hat{d}_i - \log d_i\right)^2 - \frac{1}{N^2}\left(\sum_{i=1}^{N}\left(\log \hat{d}_i - \log d_i\right)\right)^2. \tag{6.18}$$

Threshold accuracy reports the fraction of pixels within a multiplicative tolerance,

$$\delta = \frac{1}{N}\left|\left\{ i \;\middle|\; \max\left(\frac{\hat{d}_i}{d_i}, \frac{d_i}{\hat{d}_i}\right) < \tau \right\}\right|, \qquad \tau \in \{1.25,\, 1.25^2,\, 1.25^3\}. \tag{6.19}$$

Ordinal metrics, which score only the pairwise depth ordering, are used where supervision is relative; they track perceived layout but say nothing about metric precision.

Table 6.9: Depth estimation datasets grouped by supervision type: Type I datasets provide explicit depth maps for direct supervision; Type II datasets require depth to be derived from raw LiDAR, stereo pairs, or SfM, often through projection or self-supervision.

Name	Year	Scene	Type	Metric	Setup	Volume	Data Source
• *Type I: Datasets with Explicit Depth Maps (Direct Supervision)*							
Make3D [38]	2009	Outdoor	Real	Metric	Mono	534	Laser Scanner
NYU Depth v2 [504]	2012	Indoor	Real	Metric	Mono	1.4K	Kinect (SL)
TUM RGB-D [652]	2012	Indoor	Real	Metric	Mono	39 Seqs	Kinect (SL)
SUN RGB-D [447]	2015	Indoor	Real	Metric	Mono	10K	RGB-D Cameras
DIW [655]	2016	Varied	Real	Relative	Mono	495K	Internet Photos
KITTI (Depth) [654]	2017	Outdoor	Real	Metric	Stereo	93K	LiDAR (Velodyne)
ScanNet [448]	2017	Indoor	Real	Metric	Mono	2.5M	Structure Sensor
Matterport3D [340]	2017	Indoor	Real	Metric	Mono	194K	Matterport (RGB-D)
ETH3D [658]	2017	Both	Real	Metric	Multi-V	13 Scans	Laser Scanner
ApolloScape [660]	2018	Outdoor	Real	Metric	Stereo	140K	LiDAR + Stereo
iBims-1 [659]	2018	Indoor	Real	Metric	Mono	100	Camera + Laser
MegaDepth [657]	2018	Outdoor	Real	Relative	Mono	130K	Internet + SfM
ReDWeb [656]	2018	Varied	Real	Relative	Mono	3.6K	Internet + Stereo
DIODE [667]	2019	Both	Real	Metric	Mono	26K	Camera + LiDAR
BlendedMVS [668]	2020	Both	Syn+Real	Metric	Multi-V	17K	MVS Reconstruction
Hypersim [664]	2020	Indoor	Syn	Metric	Mono	77K	Simulation
DDAD [663]	2020	Outdoor	Real	Metric	Multi-V	76K	HD LiDAR
Virtual KITTI 2 [195]	2020	Outdoor	Syn	Metric	Stereo	21K	Simulation (Unity)
TartanAir [666]	2020	Varied	Syn	Metric	Stereo	∼1M	Simulation (UE4)
ARKitScenes [669]	2021	Indoor	Real	Metric	Mono	5K Clips	iPad LiDAR
UrbanSyn [665]	2023	Outdoor	Syn	Metric	Mono	7.5K	Simulation
WildRGB-D [671]	2024	Varied	Real	Metric	Mono	20K	RGB-D Cameras
• *Type II: Datasets Requiring Depth to be Derived from Raw Geometry (LiDAR, Stereo or SfM)*							
KITTI (Raw) [653]	2012	Outdoor	Real	Metric	Stereo	42K	LiDAR (Velodyne)
Cityscapes [501]	2016	Outdoor	Real	Metric	Stereo	25K	Stereo Camera
Scene Flow [619]	2016	Varied	Syn	Metric	Stereo	39K	Simulation (Blender)
NuScenes [445]	2019	Outdoor	Real	Metric	Multi-V	1.4M	LiDAR
Waymo Open [446]	2019	Outdoor	Real	Metric	Multi-V	200K	LiDAR
Mannequin [670]	2019	Varied	Real	Relative	Mono	170K	Internet + SfM
KITTI-360 [662]	2020	Outdoor	Real	Metric	Stereo	100K+	LiDAR + Stereo
Argoverse 2 [661]	2022	Outdoor	Real	Metric	Multi-V	6M	LiDAR

On KITTI, supervised regressors such as BTS [628] and DPT [630] reach Abs Rel around 0.06 with $\delta_{1.25}$ near 0.96, while the self-supervised Monodepth2 [180] reaches Abs Rel near 0.10 without any ground-truth depth. On NYU Depth v2, ZoeDepth [632] and Depth Anything [140] reach $\delta_{1.25}$ of 0.96 and 0.98 after fine-tuning, and UniDepth [672] matches them zero-shot, which makes transfer a credible alternative to per-benchmark training. Generative models such as Marigold [636] match or exceed regression on scale-invariant metrics while producing sharper boundaries, so recent gains come more from representation and data scale than from architectural depth. Boundary sharpness, temporal stability, and latency are still not evaluated in a standard way.

6.2.1.4 Challenges

Monocular depth is ill-posed: distinct 3D scenes project to the same image, so a single view fixes depth only up to scale and predictions drift across scenes. Scene content adds its own failures. Moving objects break the static-world assumption behind self-supervised and multi-view methods, and in video the result is flicker between frames and scale drift over time. Transparent and reflective materials and textureless regions give weak or misleading appearance cues; thin structures and sharp boundaries lose detail; low

light, adverse weather, and underwater imagery degrade models trained on clean data. These scene effects compound into a domain gap: models are tied to the geometry and depth distribution of their training data, the indoor–outdoor divide is especially wide, real depth labels are scarce and noisy, and synthetic data still transfers imperfectly.

Current work attacks these problems from several sides. Depth foundation models [140, 672, 632] supply strong zero-shot priors, and Video Depth Anything [295] adds temporal modules to one of them for consistent video depth, while video depth diffusion (DepthCrafter [294], ChronoDepth [673], RollingDepth [674]) pursues the same goal generatively. Coupling depth with 3D scene representations (DepthSplat [675]) improves spatial consistency, language cues help anchor metric scale [676], and lightweight architectures such as MonoLENS [677] and LMDepth [678] target the real-time budgets of driving, robotics, and AR/VR.

6.2.2 Surface Normal Estimation

Surface normal estimation predicts the local orientation of visible surfaces from images or videos. Where depth answers "how far is this point from the camera?", a normal answers "which way does the surface face?". That makes normals a compact, scale-invariant description of local 3D geometry: a normal field exposes planes, folds, object boundaries, and contact surfaces even when metric depth is ambiguous, which is why it serves reconstruction [288], relighting and intrinsic decomposition [679, 248], physical interaction modeling [680], and egocentric perception [681].

The learning problem looks simple: from an RGB image, RGB-D frame, multi-view set, or video, output one unit vector per valid pixel. The inverse problem underneath is badly constrained, since the same intensity can be explained by different combinations of shape, reflectance, and illumination, and camera intrinsics change how image gradients map to 3D orientation. Progress has therefore come from three directions at once: more and better supervision, stronger geometric inductive biases, and visual priors inherited from large dense-prediction and diffusion models; since 2024 the last of these has turned normal estimation into one output head of broader geometric foundation models.

6.2.2.1 Task Definition

Let $I \in \mathbb{R}^{H\times W\times 3}$ be an image, $\mathcal{V}$ optional auxiliary observations (neighboring frames, stereo or multi-view images, depth, intrinsics, or poses), and Ω the set of valid pixels. The goal is to estimate

$$f_\theta(I, \mathcal{V}) = \hat{\mathbf{N}}, \qquad \hat{\mathbf{N}}(u) = \hat{\mathbf{n}}_u \in \mathbb{S}^2, \quad u \in \Omega, \tag{6.20}$$

where each $\hat{\mathbf{n}}_u$ is a unit vector in the camera frame or in a gravity- or world-aligned frame. Supervision comes from depth sensors, fitted meshes, rendered assets, multi-view reconstruction, or indirectly from photometric and intrinsic-image losses, and camera metadata matters because a pixel's viewing ray depends on focal length and field of view. Monocular prediction from one RGB image is the canonical benchmark setting; RGB-D variants use measured or predicted depth as input or supervision through the differential relation between depth gradients and normals, multi-view and video variants enforce viewpoint or temporal consistency, object-centric variants target CAD-like objects, hands, faces, and bodies, and panoramic variants predict under equirectangular or fisheye geometry, where pinhole assumptions break.

6.2.2.2 Methodology

Surface normal estimation has moved from physics-based inverse rendering to supervised dense prediction and, most recently, to foundation-model and diffusion-based geometry estimation. Figure 6.7 summarizes the method families; the last two below are grouped as one family in the figure.

Photometric and intrinsic-decomposition methods. The classical origin is shape from shading and photometric stereo. Photometric stereo recovers orientation from several images taken under different

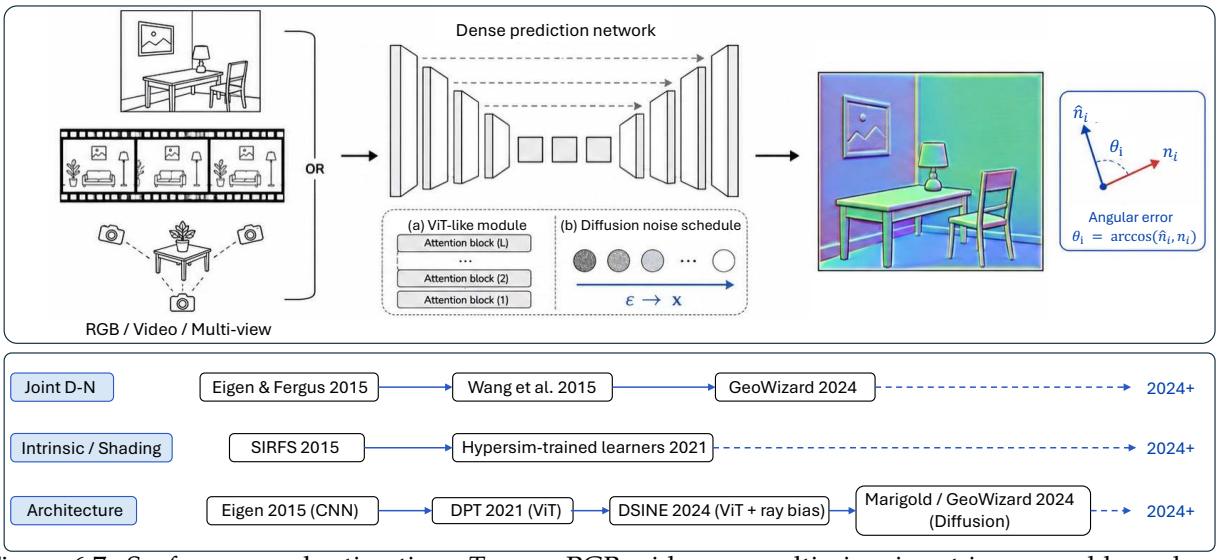


Figure 6.7: Surface normal estimation. *Top*: an RGB, video, or multi-view input is mapped by a dense prediction network ((a) ViT-like or (b) diffusion variant) to a per-pixel unit normal field $\hat{\mathbf{n}} \in \mathbb{S}^2$, scored by angular error. *Bottom*: method timelines along three threads: joint depth–normal prediction, intrinsic/shading decomposition, and backbone architecture.

lighting; Woodham's formulation [249] shows that normals follow once illumination and reflectance are sufficiently constrained. Single-image shape from shading is harder because a change in intensity may signal a change in orientation, albedo, illumination, or material; SIRFS [248] is the canonical optimization-era method, jointly recovering shape, illumination, and reflectance under natural-image and geometric priors. These methods identify the physical ambiguities that every later learned model inherits, and physically based rendering datasets such as Hypersim [664], where shading, materials, depth, and normals are jointly available, keep their logic in play.

Supervised CNN and multi-task dense prediction. Eigen and Fergus [250] trained one multi-scale convolutional network to predict depth, normals, and semantic labels, establishing that normals can be regressed directly from RGB and that geometric tasks benefit from shared representations. Later CNN systems improved encoder capacity, decoder resolution, and boundary preservation, usually with normals as one branch of a multi-task perception model, and Taskonomy [682] and Omnidata [683] systematized the paradigm by training dense predictors across many 3D scans and tasks. A CNN normal estimator runs in one forward pass and drops into real-time reconstruction, AR, or robotics pipelines; its local features, however, blur thin structures and lack the global context needed for consistent orientation across large planes.

Transformer and camera-aware discriminative predictors. Vision Transformers let every output pixel attend to the whole image. DPT [630] showed that Transformer features support strong dense prediction across depth and semantics, and the same global context helps normals: co-planar walls, floors, and object faces may be far apart in the image yet geometrically coupled. Pure attention discards one geometric fact, that pixels correspond to camera rays; DSINE [684] injects ray-direction information and shows that camera geometry remains valuable even for large learned backbones.

Diffusion and foundation-model priors. A more recent line casts normal estimation as conditional generation or as one head of a general geometric foundation model. VPD [635] showed that text-to-image diffusion models carry useful features for dense perception, and GeoWizard [637] extended the Marigold recipe of §6.2.1 to joint depth and normal prediction under a shared generative prior. Metric3Dv2 [685] took the discriminative route, training a camera-aware monocular model for metric depth and normals across

Table 6.10: Representative datasets and benchmarks for surface normal estimation. The "Normal supervision/use" column indicates whether normals are directly rendered, derived from depth or meshes, or indirectly related through intrinsic-image supervision. Counts are indicative of commonly used versions and may vary by split or preprocessing protocol.

Dataset	Year	Domain	Source	Normal supervision / use
NYU Depth v2 [504]	2012	Indoor	RGB-D	Normals derived from Kinect depth; canonical indoor monocular normal benchmark.
SUN RGB-D [447]	2015	Indoor	RGB-D	Depth-derived normals across diverse indoor scenes and sensors.
ScanNet [448]	2017	Indoor	RGB-D video + mesh	Mesh- or depth-derived normals with camera trajectories; useful for video and reconstruction-linked supervision.
Matterport3D [340]	2017	Indoor panorama	RGB-D scans + mesh	High-resolution reconstructed geometry; supports room-scale and panoramic normal evaluation.
Taskonomy [682]	2018	Indoor	3D scans	Large multi-task supervision including surface normals, depth, reshading, and semantics.
Replica [339]	2019	Indoor synthetic / scanned	Rendered geometry	Clean normals from high-quality reconstructed scenes; useful for controlled evaluation.
DIODE [667]	2019	Indoor / outdoor	LiDAR depth	Depth-derived normals; stresses cross-domain generalization beyond NYU-style indoor scenes.
Hypersim [664]	2021	Indoor synthetic	PBR rendering	Ground-truth depth, normals, materials, and lighting for geometry + intrinsic learning.
Omnidata [683]	2021	Indoor multi-source	3D scans	Large-scale mid-level vision training data with normals and many related tasks.

heterogeneous datasets. StableNormal [686] addressed a practical weakness of diffusion-based normals: stochastic denoising yields high-quality but high-variance maps, so reducing variance matters as much as improving the mean. The generative formulation samples uncertain or textureless regions from a learned distribution instead of forcing a single local regression, at the cost of iterative-denoising latency, which motivates distillation and single-step variants.

Temporal, multi-view, and reconstruction-coupled methods. Normals matter most when predictions must persist across time or support 3D reconstruction. Video normal estimation enforces temporal consistency by warping predictions with optical flow or camera poses and penalizing flicker; NormalCrafter [687] targets temporally consistent sequences for downstream video reconstruction and editing rather than per-frame accuracy alone. Multi-view settings derive normals from reconstructed meshes or regularize them through cross-view photometric consistency, and neural surface reconstruction such as NeuS [288] consumes normal-like differential geometry implicitly through signed-distance gradients, so normal estimation and reconstruction now supervise each other.

Material- and domain-specific normal estimation. A smaller branch revisits what generic models still get wrong. Transparent, reflective, and laboratory objects violate Lambertian assumptions because their gradients come from refraction, reflection, or the background rather than the surface; TransNormal [688] targets transparent objects with a dedicated benchmark and model, making explicit that open-world normal prediction cannot be solved by indoor RGB-D supervision alone.

6.2.2.3 Datasets and Metrics

Table 6.10 shows that normal supervision is usually inherited from another representation rather than annotated directly. RGB-D datasets (NYU Depth v2 [504], SUN RGB-D [447], ScanNet [448]) supply measured depth from which local planes or mesh normals are computed, which fixes the indoor evaluation regime but also means the labels are only as good as the depth sensor and are dominated by man-made indoor structure. Scanned and synthetic collections address coverage and label quality from opposite sides: Matterport3D [340], Taskonomy [682], and Omnidata [683] scale supervision across many environments and tasks, whereas Replica [339] and Hypersim [664] give clean rendered normals with physically meaningful lighting at the price of a synthetic-to-real gap. DIODE [667] pushes toward indoor–outdoor generalization, yet outdoor normals remain far less standardized than outdoor depth because sparse LiDAR is a poor match for dense orientation supervision; intrinsic-image sets such as IIW [689] and MPI Sintel Intrinsics [690] constrain shading and reflectance for inverse-rendering formulations without being normal benchmarks themselves. The most visible gap is temporal: although ScanNet and other RGB-D videos contain sequences, there is no widely adopted video normal benchmark, so temporal stability is reported qualitatively or through method-specific warping losses, even though flicker often hurts reconstruction and AR consumers more than a small rise in mean angular error.

Evaluation is angular because normals are unit vectors. With $\hat{\mathbf{n}}_i, \mathbf{n}_i \in \mathbb{S}^2$ the predicted and ground-truth normals at valid pixel i and N the number of valid pixels, the per-pixel error in degrees is

$$e_i = \arccos\big(\mathrm{clip}(\hat{\mathbf{n}}_i^\top \mathbf{n}_i, -1, 1)\big)\,, \tag{6.21}$$

summarized by its mean (sensitive to boundary noise and invalid-depth artifacts), its median (robust to a few severe failures), occasionally its RMSE, and by threshold accuracies

$$\mathrm{Acc}_\tau = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(e_i < \tau), \qquad \tau \in \{11.25^\circ, 22.5^\circ, 30^\circ\}, \tag{6.22}$$

which expose changes in the fraction of practically usable pixels that a small shift in mean error can hide. Training objectives usually minimize the cosine loss $1 - \hat{\mathbf{n}}_i^\top \mathbf{n}_i$, monotonic in angle for unit vectors, but benchmarks report angles for interpretability. Beyond per-pixel angle, joint depth–normal models are checked for agreement between predicted normals and normals differentiated from depth, reconstruction-oriented work asks whether normal discontinuities align with object boundaries and whether large planes stay coherent, and video methods report temporal consistency by warping predictions between adjacent frames with flow or pose,

$$\mathrm{TempCons} = \frac{1}{M}\sum_{(i,t)} \left\| \hat{\mathbf{n}}_i^{\,t} - \mathcal{W}_{t+1\to t}(\hat{\mathbf{n}}^{\,t+1})_i \right\|_2, \tag{6.23}$$

after rotating camera-frame normals into a common frame. Protocol details move rankings: depth-derived normals are unstable at discontinuities, papers mask invalid pixels differently, and some evaluate in the camera frame while others rotate into a gravity-aligned frame, so the table below records which paper produced each number.

Table 6.11 makes two things visible. Within one protocol, camera-aware discriminative models lead: DSINE roughly halves the median error of the 2015 baseline, and Metric3Dv2 reaches 13.3° mean error zero-shot, so projective inductive bias and large camera-diverse pretraining still pay off even with strong backbones. Diffusion-based estimators trail these numbers on NYU Depth v2 under the StableNormal protocol, but they trade accuracy on clean indoor scenes for coherence in textureless or ambiguous regions and for a usable uncertainty, which the single-frame benchmark does not reward; the post-GeoWizard line has accordingly shifted the target from single-frame accuracy toward stable, usable geometry, StableNormal controlling diffusion variance and NormalCrafter [687] enforcing temporal coherence.

Table 6.11: Representative surface-normal estimators on NYU Depth v2 (lower mean / median angular error and higher $\mathrm{Acc}_{11.25^\circ}$ are better). Rows marked † are taken from the re-evaluation in StableNormal [686], under whose protocol DSINE scores 18.6° / 9.9° / 56.1%; the other rows are as reported by the cited paper (Omnidata as re-evaluated by DSINE [684]). Numbers across protocols are indicative, not strictly comparable.

Method	**Family**	**Year**	**Mean↓**	**Median↓**	**$\mathbf{Acc}_{11.25^\circ}$ ↑**
Eigen & Fergus [250]	Multi-scale CNN, multi-task	2015	20.9	13.2	44.4
Omnidata [683]	Large-scale mid-level supervision	2021	23.1	12.9	45.8
DSINE [684]	Ray-aware discriminative	2024	16.4	8.4	59.6
Metric3Dv2 (ViT-g, zero-shot) [685]	Geometric foundation model	2024	13.3	7.0	66.4
Marigold (normal variant)† [636]	Diffusion prior	2024	20.9	11.1	50.5
GeoWizard† [637]	Joint depth + normal diffusion	2024	20.4	11.9	47.0
StableNormal† [686]	Low-variance diffusion	2024	19.7	10.5	53.0

6.2.2.4 Challenges

An RGB image alone does not determine shape, lighting, and reflectance: a dark pixel may be a tilted surface, a dark material, a cast shadow, or a specular highlight, and learned models resolve the ambiguity with priors rather than removing it, which is why estimators look confident on common indoor layouts and fail systematically on rare materials, unusual lighting, and out-of-distribution scenes. The persistent failure modes follow from this. *Reflective and transparent surfaces* produce gradients caused by the environment rather than the surface. *Textureless and weakly lit regions* give little evidence, and networks hallucinate planar sheets. *Thin structures and depth discontinuities* mix receptive fields and inherit depth-derived label noise, blurring normals around chair legs, plants, railings, and silhouettes. *Wide-field and panoramic cameras* break models trained under narrow pinhole assumptions, which is what makes ray-aware designs such as DSINE [684] necessary. *Dynamic and deformable objects* flicker because most supervision comes from static scans or rigid synthetic scenes.

Dataset bias compounds the problem: canonical benchmarks are indoor and sensor-derived, so outdoor surfaces, hands, faces, deformable and transparent objects, and egocentric manipulation stay underrepresented, and no accepted benchmark jointly measures cross-domain generalization, temporal stability, and downstream reconstruction quality. TransNormal [688] is a corrective for one slice of the non-Lambertian regime (glass, liquid, mirror, metal, wet roads), which remains far from solved. The directions now taking shape answer these gaps directly: geometric foundation models that merge depth, normals, segmentation, material, and calibration into one dense-prediction backbone, following Omnidata [683] through Metric3Dv2 [685] and GeoWizard [637]; diffusion distillation and variance control that keep the coherence and uncertainty of generative models while recovering real-time, deterministic inference [686]; temporally consistent, reconstruction-coupled normals [687]; and physically grounded models for non-Lambertian materials, where data scale is no substitute for modeling refraction and reflection.

6.2.3 Height and Distance Estimation

Height and distance estimation recovers absolute, real-world lengths: the dimensions of an object (the height of a person), the separation between scene entities (the gap between two vehicles), or the displacement of a moving object over time, from which speed follows. Whereas depth estimation (§6.2.1) recovers a dense per-pixel distance from the sensor to every visible surface, height and distance estimation targets a few metric quantities defined between scene entities rather than relative to the sensor, and is therefore tied to specific object instances and to a downstream measurement requirement. Radar, ultrasonic, and radio ranging measure such distances directly and are mature; this subsection covers the vision-based problem.

6.2.3.1 Task Definition

The input is one or more images or a video together with a specification of what to measure: the height of an object, the distance between two objects, or the distance an object travels. The output is a length in physical units, or a displacement over a time interval, from which average speed is read off (**speed estimation**).

Every vision-based route to a metric length passes through the camera model, shown in Figure 6.8. Under the pinhole model a world point P with homogeneous coordinates $\mathbf{P}_w$ projects to the pixel

$$\mathbf{p} \sim K\,[\,R \mid \mathbf{t}\,]\,\mathbf{P}_w, \tag{6.24}$$

where the extrinsics $(R, \mathbf{t})$ move the point into the camera frame and the intrinsic matrix K (focal length, principal point, pixel aspect) maps metric image-plane coordinates to pixels. Real lenses deviate from this model; Brown's model [691] corrects the dominant radial term as

$$\begin{pmatrix} x_{\text{cor}} \\ y_{\text{cor}} \end{pmatrix} = \begin{pmatrix} x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{pmatrix}, \tag{6.25}$$

with a tangential term for sensor–lens misalignment, and camera calibration [692] recovers K together with the distortion coefficients. Without calibration, every pixel-to-length conversion inherits an unknown scale.

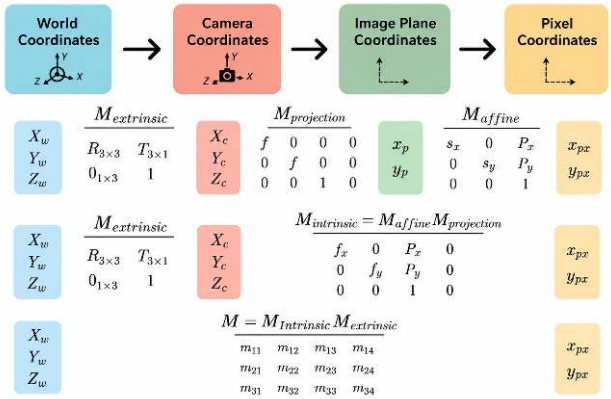


Figure 6.8: Pinhole camera model: world coordinates map to camera, image-plane, and pixel coordinates through the extrinsic, projection, and affine matrices; the last two compose into the intrinsic matrix $M_{\text{intrinsic}}$ and all three into the full projection M.

Beyond calibration, geometric pipelines rely on a small set of recurring intermediate quantities, illustrated in Figure 6.9: vanishing points, where parallel world lines converge and from which camera orientation follows; the horizon line, which fixes the camera's tilt relative to the ground; and a ground-plane homography, which lets lengths on a dominant plane such as a road or floor be measured once a single reference length is known. Stereo pipelines add rectification, which aligns the two image planes so that correspondences lie on the same scanline, and disparity matching. Learning-based pipelines keep object detection (Section 6.1.1) as the front end and replace the explicit geometry with regression of depth, 3D center, and dimensions from region features, optionally conditioned on a dense depth map and refined over time when video is available.

6.2.3.2 Methodology

Vision-based height and distance estimation is organized along two axes: the sensor configuration (stereo versus monocular), which decides whether metric depth is available by triangulation, and the inference paradigm (geometry-based versus learning-based), which decides whether scene geometry is solved from projective constraints or inferred from data. Figure 6.10 lays out the resulting families; it draws the uncalibrated case as a third branch for readability, but it is best read as the weakly constrained regime of

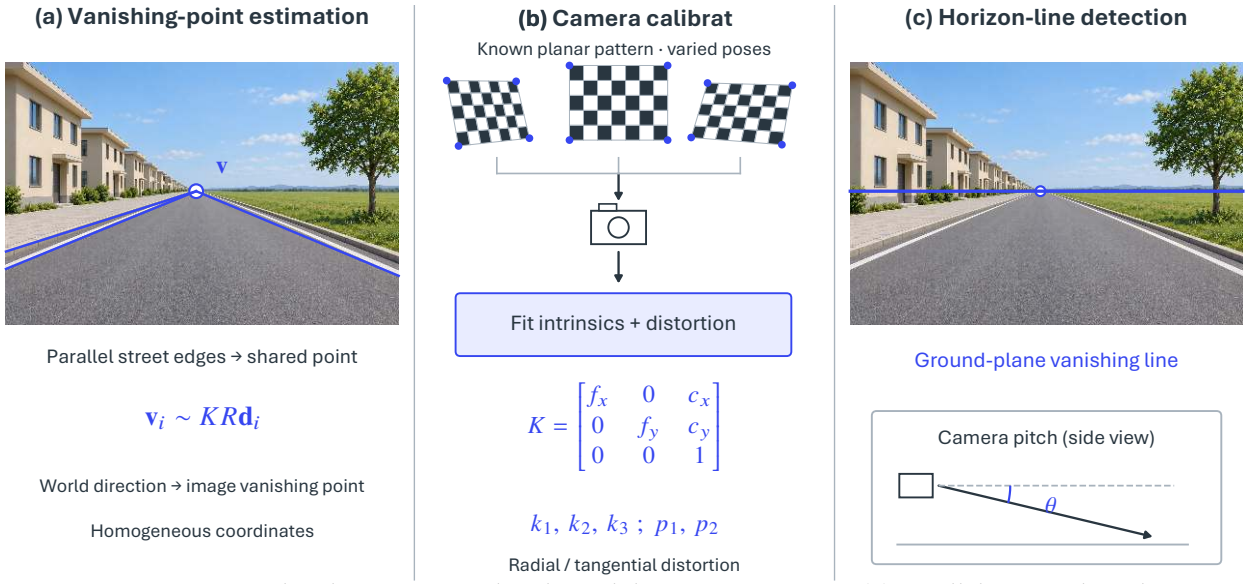


Figure 6.9: Geometric subtasks supporting height and distance estimation: (a) parallel street edges determine a vanishing point; (b) multiple views of a known planar pattern estimate camera intrinsics and lens distortion; (c) the ground-plane horizon passes through horizontal-direction vanishing points. Street geometry and the separate camera-pitch inset are illustrative.

monocular or stereo inference in which some geometric information is missing and must be supplied by priors.

Stereo Vision. A calibrated stereo rig recovers metric depth directly from disparity, $Z = fB/d$ with focal length f, baseline B, and disparity d. Classical pipelines run calibration, rectification, matching, disparity estimation, and triangulation [692, 693, 112]; their accuracy rests on calibration quality and on matching reliability in textureless, reflective, or occluded regions, and object distance is read off by aggregating depth over the detected region. Multi-view stereo adds views for consistency without changing the triangulation principle [694, 116]. Learned stereo replaces hand-crafted matching costs with learned features and cost volumes [182, 620, 695] and, in recent systems, folds distance estimation into stereo 3D object detection, predicting category, 2D extent, depth, and 3D box jointly [696, 697, 698]. Learning improves robustness, but the metric scale still comes from the calibrated baseline and intrinsics.

Monocular Vision. A single image does not fix absolute scale, so monocular methods recover it from explicit geometric constraints, learned priors, or external reference information [112, 699]. Classical monocular metrology uses projective geometry: vanishing points of parallel world lines give camera orientation [700, 112], and a homography between a dominant ground plane and the image enables metric measurement on that plane once one reference length is known [699, 701]. The approach is interpretable and accurate in controlled scenes but needs clean geometric cues, reliable calibration, and visible parallel lines or planar supports. Learning-based monocular methods detect objects in 2D and regress their depth, 3D center, and physical dimensions directly from image features [702, 703, 704, 705, 706, 707, 708, 449], implicitly learning the inverse relation between image extent and distance together with perspective, context, and occlusion cues, without an explicit vanishing-point or homography step.

Foundation-Model-Assisted Metric Estimation. A newer route uses pretrained metric or metric-aligned depth models as an intermediate representation. DPT, ZoeDepth, Metric3D, and Depth Anything predict dense depth from a single image with strong cross-domain generalization [630, 632, 709, 140]; pooling the predicted depth inside a detected region or instance mask then gives object distance, and combining masks, depth, and simple geometric constraints gives object height. This removes the need for explicit vanishing-

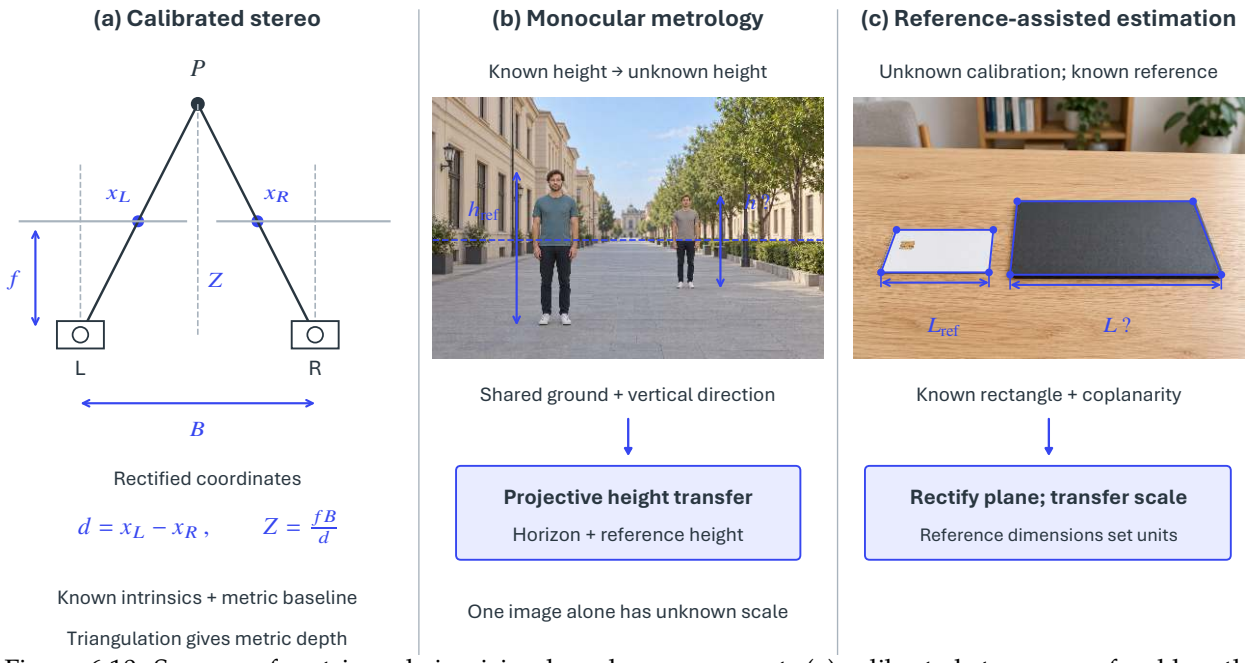


Figure 6.10: Sources of metric scale in vision-based measurement: (a) calibrated stereo uses focal length, disparity, and a known baseline; (b) monocular metrology transfers a reference height using ground-plane and vertical-direction geometry; (c) a reference-assisted example of weakly calibrated monocular inference uses a known planar rectangle to rectify a shared plane and set scale. Scenes are illustrative; the panels do not imply a universal accuracy ranking.

point detection and manual scene calibration and offers a practical path to metric estimation on in-the-wild images. In the fully uncalibrated case, exact metric recovery from one image remains ill-posed [112]; practical systems inject scale through a known reference object, a known camera height, a ground-plane assumption, category-level size priors, or one of the pretrained metric-depth models above [699, 703, 632, 709].

6.2.3.3 Datasets and Metrics

Table 6.12 summarizes the benchmarks, which fall into three groups by sub-task. Object-size estimation is evaluated on the indoor RGB-D sets CAMERA25 and REAL275 [710], where category-level dimensions are scored jointly with pose on controlled tabletop scenes. Sensor-to-object distance is rarely benchmarked on its own; it is evaluated through the 3D-detection annotations (box centers, sizes, velocities) of the driving datasets KITTI [653], nuScenes [445], Waymo Open [446], and Argoverse 2 [661], which test metric localization at scale across multi-camera rigs and scenes, with BDD100K [711] as a monocular video source. Speed and displacement estimation is dominated by fixed-camera traffic video: AI City Challenge 2018 [712], BrnoCompSpeed [713], UTFPR [223], and QMUL [714], plus the UAV-mounted Beihang videos [715]; these are the only benchmarks that score a physically meaningful motion quantity rather than static geometry.

Each sub-task is scored with the same triplet: an absolute error in physical units, a relative error for cross-range comparison, and a threshold accuracy for downstream binary decisions. For object size, the metric is the mean absolute error in physical units (centimeters for human height, millimeters for category-level pose),

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |\hat{s}_i - s_i|, \tag{6.26}$$

with the mean absolute percentage error $\mathrm{MAPE} = \frac{1}{N}\sum_i |\hat{s}_i - s_i|/s_i \times 100\%$ alongside for cross-class comparison; when size is scored jointly with pose, the 3D IoU of predicted and ground-truth boxes is

Table 6.12: Height and distance estimation benchmarks, grouped by the sub-task they score (object size, sensor-to-object distance via 3D boxes, or vehicle speed).

Dataset	Task	Modality	View	Domain	Scale	Resolution
CAMERA25 [710]	Size	RGB-D image	Monocular	Mixed-reality indoor	300K images	-
REAL275 [710]	Size	RGB-D image	Monocular	Indoor	8,000 images	-
KITTI 3D [653]	Distance	RGB + LiDAR	Monocular	Vehicle	15K frames	-
nuScenes [445]	Size / distance	RGB + LiDAR + radar	Multi-view	Vehicle	1,000 scenes	-
Waymo Open Dataset [446]	Size / distance	RGB + LiDAR	Multi-view	Vehicle	1,150 segments	-
Argoverse 2 Sensor [661]	Size / distance	RGB + LiDAR	Multi-view	Vehicle	1,000 scenes	-
AI City Challenge 2018 [712]	Speed	Video	Binocular	Vehicle	45 hours	1920×1080, 800×410 @30 Hz
BrnoCompSpeed [713]	Speed	Video	Monocular	Vehicle	18 videos × 1 hour	1920×1080 @50 Hz
UTFPR [223]	Speed	Video	Monocular	Vehicle	5 hours	1920×1080 @30 Hz
QMUL [714]	Speed	Video	Monocular	Vehicle	1 hour	360×288 @25 Hz
Beihang UAV Videos [715]	Speed	Image	Monocular	Vehicle	70 images	960×540
BDD100K [711]	Size / distance	Video	Monocular	Vehicle	100K videos	720p @30 Hz

thresholded, the NOCS protocol [710] reporting $\mathrm{Acc}@\tau$ at $\tau \in \{5\%, 10\%, 20\%\}$ IoU and at $\{5, 10\}$ cm translation. Inter-object distance reuses MAE, RMSE, and MAPE on distances $\hat{d}_i$ and the depth-style ratio accuracy δ_τ of §6.2.1 with $\tau \in \{1.10, 1.25\}$; because stereo error grows as $\Delta Z \propto Z^2/(fB)$, KITTI-style protocols [653] stratify by range (near < 10 m, mid 10–50 m, far > 50 m) so that far-field degradation is not averaged away. Speed estimation reports MAE, RMSE, and MAPE in km/h, but the dominant device is a percent-within-tolerance band: AI City Challenge 2018 [712] fixes ± 3 km/h and combines it with detection recall into its S1 score, and BrnoCompSpeed [713] adds the median absolute error because single-frame outliers from motion blur or detection jitter distort the mean. Two protocol details change rankings: per-vehicle averaging (BrnoCompSpeed) weighs trajectories differently from per-frame averaging matched to track IDs (AI City), and monocular and uncalibrated methods are sensitive to the assumed focal length, principal point, and reference length, so their reported accuracy is only meaningful together with the calibration assumptions.

Performance is sub-task specific. Category-level size recovery on REAL275 [710] remains far from millimeter accuracy in cluttered scenes. On BrnoCompSpeed [713], calibration-aware pipelines reach low single-digit km/h errors on the original sessions but degrade when transferred to unseen camera placements, exposing the brittleness of fixed-camera homography assumptions, and on the AI City 2018 speed track [712] the leading entries pair strong detection-and-tracking front ends with multi-frame speed regression, so the S1 score reflects detection recall as much as raw speed accuracy. Across all three sub-tasks the bottleneck is calibration, of the camera and of the scene-plane reference, rather than the regression backbone.

6.2.3.4 Challenges

Calibration sensitivity is the dominant open problem: small errors in intrinsics or in the scene-plane reference propagate non-linearly into every recovered length, and methods that treat focal length and reference size as latent variables, or refine calibration online from multi-view consistency, generalize across camera setups where fixed-calibration pipelines fail. The second gap is coverage: benchmarks are confined to indoor tabletop size estimation and vehicle-traffic speed, leaving pedestrian speed in crowds, package dimensions

in logistics, and indoor inter-object distance essentially unmeasured. Fusing visual estimates with LiDAR, radar, or inertial sensing would lift robustness in low light and under occlusion, but the vision-only setting treated here still lacks a benchmark that scores metric accuracy jointly with calibration uncertainty.

6.2.4 Visual Odometry and Camera Pose Estimation

Visual odometry (VO) and camera pose estimation recover the position and orientation of a camera from visual observations. They provide the coordinate scaffold that lets a system stitch individual images into a persistent spatial representation: depth, surface normals, scene flow, maps, and 3D reconstructions all become more useful once they can be expressed in a shared camera or world frame. VO usually means estimating *incremental* motion between consecutive frames, often as a trajectory $\{\mathbf{T}_t\}_{t=1}^{T}$ in $SE(3)$; camera pose estimation is broader and may recover a query image's absolute 6-DoF pose in a known scene, the relative pose between two views, or camera motion and scene geometry jointly from an unordered image collection. The field was built on geometric optimization and SLAM, and its center of gravity is now shifting toward learned correspondence, differentiable optimization, dense geometric priors, and feed-forward geometric foundation models that predict cameras and geometry together.

6.2.4.1 Task Definition

Let $\mathbf{T}_t = [\mathbf{R}_t|\mathbf{t}_t] \in SE(3)$ denote the pose of camera t with respect to a world frame. Given images $\{I_t\}$ (monocular, stereo, RGB-D, multi-view, or video), optional intrinsics $\{\mathbf{K}_t\}$, optional inertial readings, and sometimes a prior map or database, the goal is to estimate relative transformations $\Delta\mathbf{T}_{i\to j} = \mathbf{T}_j\mathbf{T}_i^{-1}$, an absolute pose $\mathbf{T}_q$ for a query image, or a full trajectory together with sparse or dense scene geometry, and occasionally the intrinsics themselves when calibration is unknown. The variants differ in what is assumed known. **Monocular VO** is scale ambiguous without external priors; **stereo and RGB-D VO** recover metric scale from a known baseline or measured depth; **visual-inertial odometry** uses IMU measurements to stabilize scale and fast rotation; **SLAM** adds mapping and loop closure to online odometry; **SfM** recovers poses and structure from unordered collections; **map-based relocalization** assumes an existing scene representation and places a query in it; and **map-free relocalization and feed-forward reconstruction** skip explicit map construction by predicting relative pose or globally aligned point maps directly from images.

6.2.4.2 Methodology

Figure 6.11 summarizes five method families, ordered from explicit geometry to learned geometric priors to feed-forward scene-level geometry.

Geometry-first SfM, VO, and SLAM. Classical pipelines detect repeatable features, match them across views, fit epipolar geometry with robust estimators, and refine poses and 3D points by bundle adjustment [110, 716]. COLMAP [115] made this recipe robust enough for internet photo collections, and GLOMAP [717] revisits the pose-graph and averaging components of global SfM for large-scale reconstruction. For sequential input, visual SLAM adds mapping and loop closure to frame-to-frame tracking: MonoSLAM [87], PTAM [88], and ORB-SLAM [91] define the feature-based branch, LSD-SLAM [718] and DSO [719] the direct branch. Explicit geometric consistency is the family's strength; weak texture, dynamic objects, rolling shutter, illumination shifts, and small-baseline or planar degeneracies are its failure modes.

Learned VO and self-supervised view synthesis. End-to-end VO regresses camera motion from image pairs or short windows, DeepVO [134] being the standard early example with convolutional features and recurrent temporal modeling. Self-supervised VO instead learns depth and ego-motion together by minimizing photometric reconstruction error between adjacent frames, with SfM-Learner [133] as the canonical starting point. This made monocular video supervision scalable, but the geometry remains bound to photometric

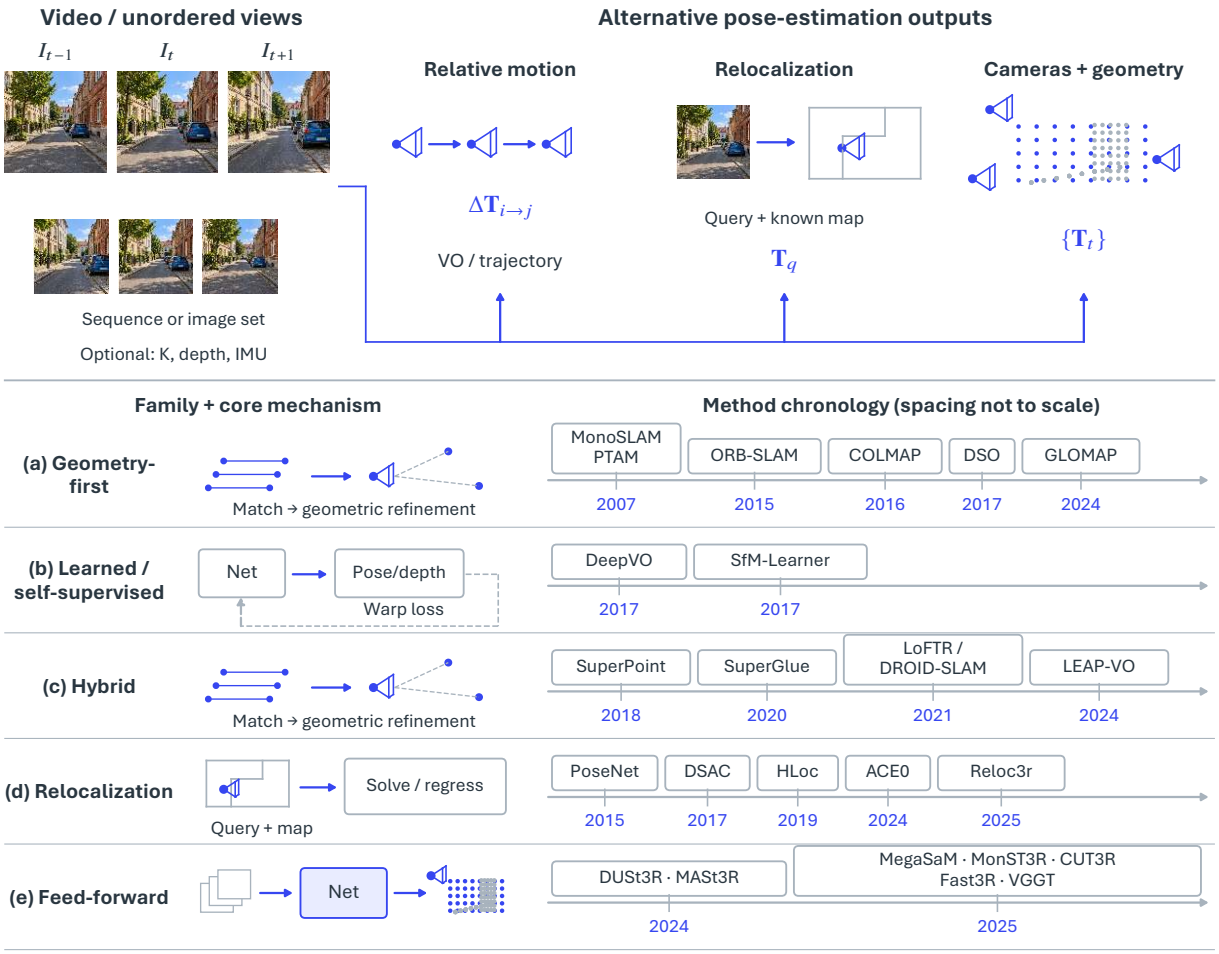


Figure 6.11: Visual odometry and camera pose estimation. *Top*: a video or unordered image set is mapped to relative motion $\Delta\mathbf{T}_{i\to j}$, an absolute pose $\mathbf{T}_q$ in a known scene, or camera poses $\{\mathbf{T}_t\}$ jointly with scene structure. *Bottom*: method lanes (a) geometry-first, (b) learned and self-supervised VO, (c) hybrid learned–geometric, (d) relocalization, and (e) feed-forward foundation models. Mechanism arrows indicate operations; dated timelines summarize representative methods in publication order, with non-proportional spacing and no implied computational dependency. The dashed warp-feedback path denotes self-supervised training.

assumptions (static Lambertian scenes, stable intrinsics, sufficient parallax) and to monocular scale ambiguity unless stereo, IMU, or learned metric priors are added.

Hybrid learned geometry and differentiable optimization. The most successful learned systems keep geometric optimization and learn the parts that are hard to hand-design. Learned features and matchers (SuperPoint [539], SuperGlue [541], LoFTR [542]) improve the correspondence front end, and VGG-SfM [720] and ACE0 [721] push learned features and scene-coordinate learning into SfM and relocalization. DROID-SLAM [95] is the representative differentiable-optimization system, coupling dense correspondence, depth, and camera motion through recurrent updates and a dense bundle-adjustment layer, and LEAP-VO [722] stabilizes motion over longer windows through long-term point tracking.

Relocalization and map-based pose estimation. Given a query image and a known scene or database, relocalization estimates the query's 6-DoF pose. PoseNet [723] popularized absolute pose regression, which is fast but less accurate than geometry. Scene-coordinate regression, from Shotton *et al.* [724] to DSAC [725] and ACE-style relocalizers [721], predicts a 3D scene coordinate per pixel and solves pose with PnP and RANSAC;

Table 6.13: Representative datasets and benchmarks for visual odometry, camera pose estimation, relocalization, and feed-forward scene geometry. Counts and splits vary across protocols.

Dataset / benchmark	Year	Domain	Input	Primary use
KITTI Odometry [173]	2012	Outdoor driving	Stereo + GPS/IMU	Classical and learned VO; segment drift evaluation.
TUM RGB-D [652]	2012	Indoor	RGB-D + mocap	Indoor RGB-D SLAM and trajectory accuracy.
7-Scenes [724]	2013	Indoor	RGB-D video	Camera relocalization and scene-coordinate regression.
Cambridge Landmarks [723]	2015	Outdoor landmarks	RGB	Outdoor absolute pose regression and relocalization.
EuRoC MAV [733]	2016	Aerial / industrial indoor	Stereo + IMU	Visual-inertial odometry under fast motion.
ScanNet [448]	2017	Indoor	RGB-D video + mesh	Learned pose, reconstruction, and dense geometry.
MegaDepth [657]	2018	Internet landmarks	SfM supervision	Relative pose, matching, depth, and geometric pretraining.
RealEstate10K [734]	2018	Indoor / real-estate video	Video + recovered poses	Self-supervised video geometry and view synthesis.
Virtual KITTI 2 [195]	2020	Synthetic driving	Rendered stereo / depth / pose	Controlled sim-to-real VO and depth evaluation.
DL3DV-10K [735]	2024	Large-scale scenes	Multi-view video / images	Generalizable 3D reconstruction and pose-aware training.

hierarchical localization [726] retrieves candidate database images, matches, and verifies geometrically, and remains the high-accuracy reference. Relative-pose regressors trained on much larger collections, such as Reloc3r [727], revisit the speed and generalization trade-off and show that regression recovers a role when paired with scale and data.

Feed-forward geometric foundation models. The main post-2024 shift is models that predict scene geometry and camera relations directly, often without SfM initialization. DUSt3R [122] regresses dense 3D point maps from image pairs, so that matching, triangulation, and pose recovery become one learned feed-forward problem; MASt3R [728] grounds image matching in this 3D representation, and MASt3R-SfM [729] builds an unconstrained SfM pipeline on it. MegaSaM [296] and MonST3R [730] extend the idea to casual dynamic video, where static-scene assumptions fail; CUT3R [731] adds persistent state for continuous 3D perception, Fast3R [732] scales feed-forward reconstruction to very large image sets, and VGGT [123] predicts camera poses, depth, and point maps jointly with a visual-geometry-grounded Transformer. Geometry does not disappear in this family; it moves from a hand-coded optimization loop into a learned prior that supplies initialization, dense correspondence, or complete camera-and-scene predictions.

6.2.4.3 Datasets and Metrics

Table 6.13 shows three kinds of benchmark serving three different questions. Trajectory benchmarks (KITTI Odometry [173], TUM RGB-D [652], EuRoC MAV [733]) measure drift under a fixed sensor setup and cover a narrow slice of deployment: urban driving, small rooms, micro-aerial-vehicle flights. Relocalization benchmarks (7-Scenes [724], Cambridge Landmarks [723]) ask whether a query can be placed in a known scene and favor methods with scene-specific maps, retrieval, or scene-coordinate regression, so they are not comparable to online VO. Large internet and video corpora (MegaDepth [657], RealEstate10K [734], ScanNet [448], DL3DV-10K [735]) serve mostly as training data for generalizable geometry models, which need diverse baselines, unknown intrinsics, unordered images, dynamic content, and dense reconstruction targets.

Table 6.14: Representative methods under two shared protocols. *Top*: monocular SLAM / VO, absolute trajectory error [m] averaged over the 11 EuRoC MAV sequences and the 9 TUM-RGBD freiburg1 sequences, as reported in DROID-SLAM [95] ("fails k" = lost tracking on k sequences, so no average is defined). *Bottom*: multi-view camera pose, AUC@30 on RealEstate10K (unseen) and CO3Dv2, as reported in VGGT [123]; the learned methods were trained on CO3Dv2 but not RealEstate10K. Numbers are from the cited evaluating paper and are comparable only within a block.

Method	**Family**	**Year**	**EuRoC ATE↓**	**TUM fr1 ATE↓**
ORB-SLAM [91]	Feature-based SLAM	2015	fails 2 / 11	fails 6 / 9
DSO [719]	Direct sparse odometry	2017	0.601	–
ORB-SLAM3 [736]	Feature-based multi-map SLAM	2021	fails 1 / 11	fails 5 / 9
DROID-SLAM [95]	Differentiable dense BA	2021	**0.022**	**0.038**
Method	**Family**	**Year**	**Re10K AUC@30↑**	**CO3Dv2 AUC@30↑**
COLMAP + SuperPoint/SuperGlue [115, 539, 541]	Incremental SfM, learned matching	2016 / 2020	45.2	25.3
DUSt3R [122]	Pairwise point-map regression	2024	67.7	76.7
MASt3R [728]	3D-grounded matching	2024	76.4	81.8
VGGSfM v2 [720]	Learned differentiable SfM	2024	78.9	83.4
CUT3R [731]	Continuous 3D perception with state	2025	75.3	82.8
Fast3R [732]	Large image-set feed-forward	2025	72.7	82.5
VGGT [123]	Visual geometry grounded Transformer	2025	**85.3** (93.5 w/ BA)	**88.2** (91.8 w/ BA)

Evaluation is correspondingly layered. With $\mathbf{T}_i$ and $\hat{\mathbf{T}}_i$ the ground-truth and estimated poses, the absolute trajectory error measures global discrepancy after a rigid or similarity alignment $\mathbf{S}$,

$$\mathrm{ATE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left\|\mathrm{trans}\left(\mathbf{T}_i^{-1}\mathbf{S}\hat{\mathbf{T}}_i\right)\right\|^2}, \tag{6.27}$$

and the relative pose error measures local motion over an interval Δ,

$$\mathrm{RPE}(\Delta) = \frac{1}{N-\Delta}\sum_{i=1}^{N-\Delta}\left\|\mathrm{Log}\left[(\mathbf{T}_i^{-1}\mathbf{T}_{i+\Delta})^{-1}(\hat{\mathbf{T}}_i^{-1}\hat{\mathbf{T}}_{i+\Delta})\right]\right\|; \tag{6.28}$$

KITTI reports the same idea as translational and rotational drift averaged over fixed-length path segments. Whether scale is aligned before computing ATE, or scale drift is scored separately, decides whether monocular, stereo, RGB-D, and visual-inertial methods can be compared at all. Pairwise matching and feed-forward pose methods report rotation error and translation-direction error in degrees, often aggregated as the area under the accuracy curve (AUC) below angular thresholds; relocalization reports the fraction of queries within thresholds such as $(5\,\mathrm{cm}, 5^\circ)$ indoors or meter-level outdoors, or median translation and rotation error; and feed-forward geometry models add depth error, point-map error, Chamfer distance, registration rate, or view-synthesis quality, since their cameras are judged by whether they support a usable 3D representation.

Table 6.14 puts the two regimes side by side. On trajectory benchmarks the lesson is robustness as much as accuracy: the classical systems that are highly accurate when they track lose tracking on a fraction of EuRoC flights and on most TUM freiburg1 sequences, whose rolling shutter, motion blur, and fast rotation defeat feature matching, whereas DROID-SLAM's learned dense correspondence inside a bundle-adjustment loop

tracks every sequence at centimeter-level error. On multi-view pose the feed-forward models have overtaken the COLMAP pipeline by a wide margin and keep improving on unseen data (RealEstate10K), with VGGT ahead of DUSt3R-style pairwise regression at a fraction of the runtime, and a short bundle-adjustment refinement still adding several points, which is the hybrid pattern the challenges below return to. The two blocks are not comparable to each other, and that is the point: a method optimized for low drift under a calibrated rig and one optimized for zero-shot sparse-view geometry are judged by different questions.

6.2.4.4 Challenges

Geometric ambiguity sets the floor. Monocular cameras cannot recover metric scale without priors; pure rotation, planar scenes, and very small baselines make epipolar geometry ill-conditioned; rolling shutter and unknown intrinsics violate the assumptions built into many solvers. Learned models can hide these failures behind priors but cannot restore missing information, which is why visual-inertial fusion, calibration, and depth priors remain necessary even with large geometry models. Deployment adds its own: dynamic agents corrupt the static-world assumption behind classical SLAM and bundle adjustment; textureless rooms, repetitive corridors, glass, water, and specular surfaces yield unreliable correspondences; appearance shift across time of day, weather, season, and camera hardware breaks place recognition; and long-horizon operation still needs loop closure, map maintenance, and uncertainty handling, none of which a feed-forward prediction supplies on its own.

The strongest trend is the convergence of SLAM, reconstruction, and foundation geometry. DROID-SLAM [95] showed that optimization can be learned without abandoning geometric structure, and the feed-forward models above predict dense geometry, cameras, and correspondences together; the near-term frontier is hybrid, with feed-forward geometry supplying initialization, dense priors, and generalization and optimization enforcing metric consistency, loop closure, and uncertainty-aware correction. Two further shifts matter for spatial intelligence. Pose estimation must become **dynamic-scene aware**, estimating camera and object motion jointly as MegaSaM [296] and MonST3R [730] begin to do, rather than discarding moving content as outliers; and camera pose should be judged by **downstream usefulness**, whether it supports stable depth and normal prediction, reconstruction, embodied navigation, AR anchoring, or world-model memory. As pose becomes one output of larger geometric perception systems, the boundary between VO, SLAM, scene reconstruction, and spatial memory will keep blurring.

6.2.5 Pose Estimation

6-DoF object pose estimation recovers a rigid transformation

$$\mathbf{T}_o = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in SE(3), \tag{6.29}$$

which maps an object's canonical or model frame into the current camera frame, giving a rotation $\mathbf{R} \in SO(3)$ and a translation $\mathbf{t} \in \mathbb{R}^3$ per detected instance. It is the camera pose problem of §6.2.4 with the reference frame reversed: the camera is held fixed and each rigid object is located within it, using much of the same machinery (2D–3D correspondence, PnP solvers, RANSAC, and render-and-compare refinement) but with object-centric CAD models and pose annotations as supervision. It is also distinct from the keypoint and landmark detection of §6.1.3: the target is the full 6-DoF pose of a rigid body, not semantic points on an articulated or deformable one. Object pose is the geometric primitive consumed by robotic manipulation and grasping (revisited in Chapter 9), AR/VR object overlay, assembly guidance, bin picking, and the metric localization of other agents in driving.

Three axes organize the literature. The *assumption axis* runs from instance-level methods that assume a high-fidelity CAD mesh per object, through category-level methods that assume only a canonical frame per category, to model-free methods that see a previously unseen object through a handful of reference

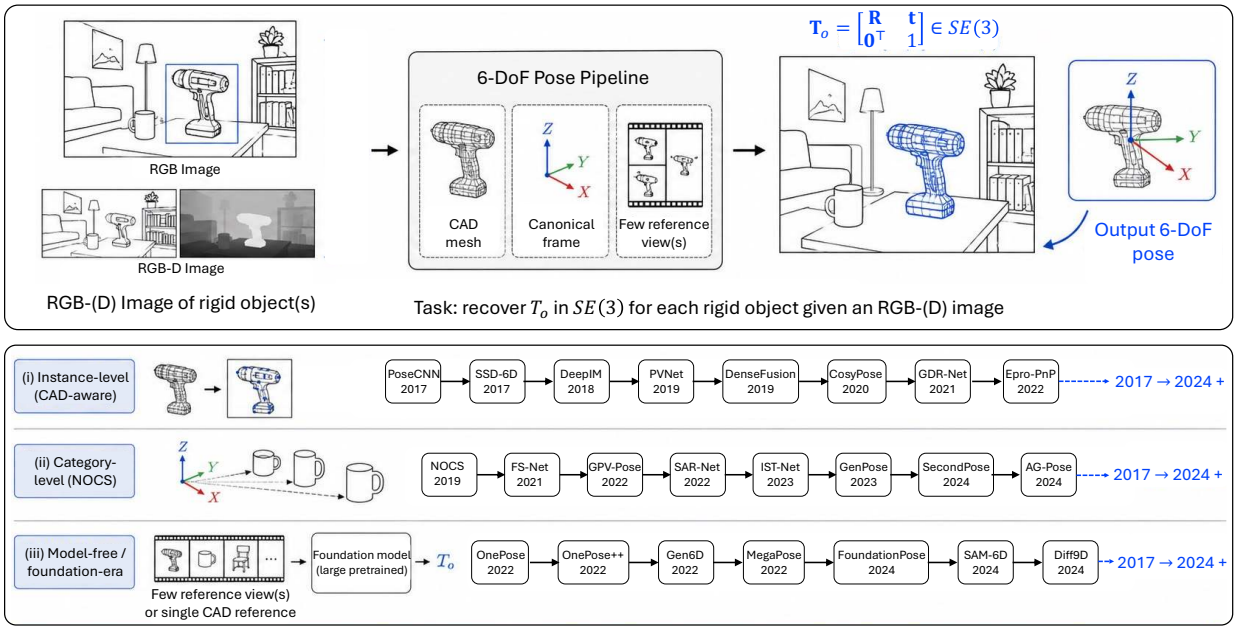


Figure 6.12: 6-DoF object pose estimation. *Top*: recover $\mathbf{T}_o \in SE(3)$ for each rigid object in an RGB(-D) image, given a CAD mesh, a canonical frame, or a few reference views. *Bottom*: method timelines for (i) instance-level CAD-aware, (ii) category-level (NOCS) canonical-frame, and (iii) model-free, foundation-era pose estimation.

images. The *representation axis* separates direct $SE(3)$ regression from keypoint- or dense-correspondence pipelines that hand pose recovery to PnP, and from render-and-compare refinement of an initial pose. The *foundation-model axis* tracks the shift from per-object training to cross-object generalization built on segmentation, vision, and depth foundation models.

6.2.5.1 Task Definition

The input is an RGB image, an RGB-D image, or a short multi-view burst of a scene containing one or more rigid objects, together with object-specific priors that depend on the setting: a CAD mesh or a small reference image set per object (instance-level and model-free), or a category label with a canonical frame (category-level), optionally with camera intrinsics and an upstream 2D detection or mask. The output is a pose $\mathbf{T}_o \in SE(3)$ per detected object in the camera frame, augmented in the category-level setting with a metric scale $s \in \mathbb{R}^+$ (9-DoF pose). For symmetric objects the prediction is properly an equivalence class of poses under the object's symmetry group, which the ADD-S and BOP metrics below account for.

6.2.5.2 Methodology

Figure 6.12 schematizes the task and the three method-family trajectories, which recent methods often straddle.

Instance-Level 6-DoF: CAD-Aware Pipelines. With a mesh of each target object available, pose estimation becomes alignment of the mesh with image evidence, and synthetic renderings of the CAD models are the main training signal. PoseCNN [737] set the modern baseline by regressing an object's image-space center and a quaternion rotation with a CNN, and introduced the YCB-Video benchmark. A second family decouples the network from the geometric solver by predicting projections of sparse 3D keypoints and solving PnP with RANSAC; PVNet [738] votes per pixel for each keypoint, which makes it robust to occlusion, while SSD-6D [739] folds a discretized viewpoint classification into a detector and skips PnP entirely. A third family regresses dense object coordinates instead of sparse keypoints: DenseFusion [740] fuses RGB-D features per

pixel for iterative refinement, GDR-Net [741] couples dense coordinate regression with a geometry-guided PnP layer, ZebraPose [742] encodes the surface with a coarse-to-fine binary code, and SurfEmb [743] learns continuous correspondence distributions that stay robust under symmetry and occlusion; EPro-PnP [744] replaces the non-differentiable PnP solver with a probabilistic surrogate so the whole pipeline trains end to end. A fourth family refines by render-and-compare: DeepIM [745] iteratively minimizes the residual between a rendered CAD view and the image, and CosyPose [746] extends this to consistent multi-view, multi-object refinement and led the BOP 2020 leaderboard. The same machinery reaches far from the tabletop, for example spacecraft pose under extreme scale variation [747].

Category-Level 6-DoF + Size: Canonical Frame Recovery. Without a per-instance mesh but with a category that admits a canonical frame, pose is recovered up to that frame together with absolute scale, a 9-DoF prediction. NOCS [710] introduced the formulation, mapping every instance of a category into a Normalized Object Coordinate Space and regressing dense correspondences there; its REAL275 benchmark is the standard category-level evaluation. Later work re-parameterizes the problem: FS-Net [748] decouples rotation, translation, and size and speeds up inference through a shape-based pathway, GPV-Pose [749] grounds rotation in geometry-guided point-wise voting, SAR-Net [750] recovers a deformed shape prior alongside pose to handle intra-class variation, and IST-Net [751] drops the shape prior and learns the canonical transformation implicitly. SecondPose [752] and AG-Pose [753] import DINOv2 semantics and instance-adaptive keypoints for further gains on REAL275, and GenPose [754] casts category-level pose as score-based diffusion sampling, the precursor of the 9-DoF diffusion model below. The family's standing assumption, that a meaningful canonical frame exists per category, holds for mugs and cameras but fails for heterogeneous semantic groupings, which motivates the model-free line.

Model-Free / Novel-Object Pose: Foundation-Model Era. The largest recent shift is away from per-object and per-category training toward objects first seen at test time, given only a few reference images or a CAD model. From a short reference video, OnePose [755] triangulates a feature point cloud and matches test queries against it, OnePose++ [756] removes the dependence on repeatable keypoints with a dense-matching front end, and Gen6D [757] unifies detection, viewpoint selection, and refinement without per-object fine-tuning. From a CAD model alone, MegaPose [758] trains render-and-compare on millions of synthetic objects and transfers zero-shot, FoundationPose [759] consolidates model-based and model-free operation under large-scale synthetic pretraining and leads the BOP novel-object track, GigaPose [760] cuts the cost of render-and-compare by sampling templates in a two-DoF rotation space and recovering the remaining parameters from a single patch correspondence, and SAM-6D [761] uses the Segment Anything Model [138] as a zero-shot detection front end before learned pose matching. A further sub-line drops pose-specific training altogether and matches frozen foundation features: FoundPose [762] establishes 2D–3D correspondences between DINOv2 patch descriptors of the query and pre-rendered templates, and FreeZe [763] fuses frozen 3D geometric descriptors with 2D foundation features and recovers pose by RANSAC registration; a FreeZe variant took first place for unseen-object localization and detection in the BOP Challenge 2024 [764], overtaking the render-and-compare line. Diff9D [765] reformulates 9-DoF category-level pose as diffusion sampling, the object-pose instance of the diffusion-as-perception pattern seen for depth and normals, which yields calibrated multi-hypothesis predictions (natural for symmetric objects) at the cost of latency. The foundation-model line has thus largely dissolved the instance-versus-category-versus-model-free distinction; the active question is how cheap the per-test-object supervision can become.

Sensing Modalities. Across these axes, the input modality fixes which supervisory signals exist and which symmetries can be resolved. RGB-only pipelines [737, 738, 741, 744, 755] suit mobile and AR deployment but face scale–depth ambiguity on texture-poor objects; RGB-D fusion (DenseFusion [740], CosyPose [746] with depth refinement, FoundationPose [759]) is substantially more accurate on BOP at the cost of a depth sensor; depth-only methods are rare but remain competitive on highly symmetric objects where shading carries little

Table 6.15: Representative datasets and benchmarks for 6-DoF object pose estimation. "–" marks a value not fixed by the source or varying across protocol versions.

Dataset	Year	Setting	Objects	Frames	Role
LINEMOD [766]	2012	Instance, textured household, RGB-D	15	∼1.2K / object	Entry benchmark; introduced ADD
LINEMOD-Occluded [767]	2014	Instance, heavy mutual occlusion	8	re-annotated LM scene	Occlusion stress test
YCB-Video [737]	2017	Instance, RGB-D video, moving camera	21	∼133K	Sequential pose and tracking
T-LESS [768]	2017	Instance, texture-less, symmetric industrial parts	30	–	Symmetry / texture-poor stress test
HomebrewedDB [769]	2019	Instance, household scenes	–	–	Out-of-distribution generalization
NOCS-REAL275 (+CAMERA) [710]	2019	Category-level 9-DoF, RGB-D	6 categories	∼8K real (+ synthetic)	Category-level standard
SPEED [770]	2019	Spacecraft, extreme scale and lighting	1	–	Beyond-tabletop deployment
BOP Challenge [771, 772, 773]	2018–	Unified protocol over LM-O, T-LESS, YCB-V, ITODD, IC-BIN, TUD-L, HB, HOPE	–	–	Headline leaderboard; novel-object track since 2023

rotational evidence. Monocular depth foundation models (§6.2.1) are increasingly used as plug-in depth providers at inference, which narrows the RGB / RGB-D gap without an active sensor.

6.2.5.3 Datasets and Metrics

Table 6.15 lists the recurring benchmarks. Instance-level evaluation grew from LINEMOD [766], the texture-rich entry point that introduced the ADD metric, through its occlusion-heavy re-annotation LINEMOD-Occluded [767], which modern methods report as the more informative of the two, to the video setting of YCB-Video [737] and the texture-less, symmetric industrial parts of T-LESS [768], where symmetry handling decides the ranking; HomebrewedDB [769] serves as an out-of-distribution test for methods trained on the BOP core data, and SPEED [770] stretches the task to spacecraft under scale and illumination ranges no tabletop set covers. NOCS-REAL275 with its synthetic companion CAMERA [710] is the category-level standard. The BOP Challenge [771, 772, 773] consolidates the instance-level datasets under one protocol and has been the field's headline leaderboard since 2017; its 2023 edition separates a classical CAD-known track from a novel-object track, formalizing the model-free shift. Modern papers report at minimum on LINEMOD-Occluded, YCB-Video, and a BOP-core subset, novel-object methods add the BOP unseen-object track, and category-level methods center on REAL275.

Each metric answers to a particular failure mode of pose error, and the symmetry-aware variants exist because many objects admit several valid poses. The Average Distance of Model Points (ADD) is the standard for asymmetric objects: with predicted pose $(\hat{\mathbf{R}}, \hat{\mathbf{t}})$, ground truth $(\mathbf{R}, \mathbf{t})$, and mesh vertices $\mathcal{M}$,

$$\text{ADD} = \frac{1}{|\mathcal{M}|} \sum_{\mathbf{x} \in \mathcal{M}} \left\| (\mathbf{R}\mathbf{x} + \mathbf{t}) - (\hat{\mathbf{R}}\mathbf{x} + \hat{\mathbf{t}}) \right\|, \tag{6.30}$$

and a pose counts as correct when $\text{ADD} < 0.1\, d$ for object diameter d, the ADD-0.1d threshold introduced with LINEMOD [766]. For symmetric objects ADD-S replaces direct vertex correspondence with nearest-neighbor matching,

$$\text{ADD-S} = \frac{1}{|\mathcal{M}|} \sum_{\mathbf{x}_1 \in \mathcal{M}} \min_{\mathbf{x}_2 \in \mathcal{M}} \left\| (\mathbf{R}\mathbf{x}_1 + \mathbf{t}) - (\hat{\mathbf{R}}\mathbf{x}_2 + \hat{\mathbf{t}}) \right\|, \tag{6.31}$$

Table 6.16: Representative 6-DoF object pose results on LINEMOD-Occluded (higher is better). ADD(-S)-0.1d anchors the RGB methods; BOP-AR is reported for the refinement and foundation-era systems. The two columns are different metrics and are not directly comparable.

Method	Family	Modality	Train assumption	Year	ADD-0.1d ↑	BOP-AR ↑
PoseCNN [737]	Instance / direct	RGB	per-object	2017	24.9	–
PVNet [738]	Instance / keypoint+PnP	RGB	per-object	2019	40.8	–
GDR-Net [741]	Instance / coord. reg.	RGB	per-object	2021	62.2	–
CosyPose [746]	Instance / refinement	RGB	per-object	2020	–	63.3
SurfEmb [743]	Instance / dense corr.	RGB-D	per-object	2022	–	75.8
MegaPose [758]	Model-free / render&comp.	RGB-D	cross-object	2022	–	58.3
FoundationPose [759]	Model-free / found.	RGB-D	cross-object	2024	–	78.8

under the same threshold; the two are reported together so that symmetric and asymmetric objects are diagnosed separately. For AR overlay the relevant error is in image space, so the 2D projection error averages the pixel distance between mesh vertices projected under the two poses, with 5 px as the usual threshold. Category-level evaluation on REAL275 counts a prediction correct when

$$\angle(\mathbf{R}^{-1}\hat{\mathbf{R}}) < 5^\circ \quad \text{and} \quad \|\mathbf{t} - \hat{\mathbf{t}}\| < 5\,\text{cm}, \tag{6.32}$$

and adds 3D box IoU at 50% and 75% to score pose and size jointly. The BOP Challenge [771] averages recall over three symmetry-aware errors, each thresholded at several operating points: the Visible Surface Discrepancy (VSD), which scores only the visible object surface and is therefore robust to occlusion, the Maximum Symmetry-aware Surface Distance (MSSD), the worst-case surface deviation over all valid symmetries, and its image-space analogue MSPD,

$$\text{AR} = \tfrac{1}{3}\big(\text{AR}_{\text{VSD}} + \text{AR}_{\text{MSSD}} + \text{AR}_{\text{MSPD}}\big), \tag{6.33}$$

averaged over the BOP core datasets. ADD and ADD-S remain the default for LINEMOD-style instance-level work, BOP-AR for cross-dataset comparison, and projection error for AR; the metrics persist side by side because no single norm captures both strict pose error and the equivalence-class structure that VSD and MSSD make explicit.

Table 6.16 reports LINEMOD-Occluded, the most widely used instance-level benchmark under the BOP protocol. Among RGB methods scored by ADD(-S)-0.1d, explicit geometry steadily closes the gap over direct regression: PoseCNN's 24.9 rises to 40.8 with per-pixel keypoint voting (PVNet) and to 62.2 with geometry-guided dense coordinate regression (GDR-Net). On the symmetry-aware BOP-AR, the render-and-compare CosyPose (63.3) and the dense-correspondence SurfEmb (75.8) are now matched or exceeded by model-free systems with no per-object training, FoundationPose reaching 78.8 against MegaPose's earlier 58.3; this is the 2023–2024 shift that made novel-object pose a deployment target rather than a research curiosity. The gap between the two families is widest on the symmetric objects of T-LESS, where losses that model symmetry (BOP-AR-aware objectives, ADD-S optimization) generalize better than those assuming a single asymmetric pose.

6.2.5.4 Challenges

The onboarding bottleneck has shifted from pose to detection. Moving to novel objects relocated the limiting stage rather than removing it. On the BOP Challenge 2024 [764], 6-DoF *localization* of unseen objects has nearly caught up with the seen-object regime (the strongest training-free entry, a FreeZe variant [763], trails the best seen-object method by a few points of average recall), whereas *detection* of unseen objects still lags its seen-object counterpart by roughly a third and is now the largest error source in the novel-object stack. The open problem is zero-shot detection and segmentation of a never-seen instance in clutter, together with the cost of *onboarding* it: BOP 2024 added a track in which an object is registered from a short reference video,

Any6D [774] registers from a single RGB-D anchor image, and how few views, or whether a single image or a text prompt, suffice is the defining frontier question. The same challenge pivoted toward high-resolution AR/VR headset recordings, a signal that the benchmark-realism gap, including articulated and deformable objects, open-vocabulary targets named rather than modeled (POPE [775]), and egocentric capture, is now a first-order concern.

Absolute metric scale remains ambiguous for model-free targets. The true size of an unseen object is under-determined from RGB alone, since scale and depth trade off along the viewing ray. Category-level 9-DoF methods absorb this by regressing size in a canonical frame [710], and model-free systems lean on a depth sensor or a known CAD extent; when neither is available, scale is the residual unknown separating a plausible pose from a usable one, which Any6D [774] makes explicit by coupling pose recovery with scale alignment, and monocular depth priors inherit the affine ambiguity of §6.2.1.

Accuracy trades sharply against latency. The methods that top the novel-object leaderboard are the least suited to closed-loop use: on BOP 2024 [764] the most accurate unseen-object method runs in tens of seconds per image, close to an order of magnitude slower than both the fastest seen-object pipelines and the practical sub-second entries, which buy speed with accuracy. Heavy segmentation, render-and-compare, and diffusion priors (Diff9D [765]) are the cause, which is why one-correspondence template matching (GigaPose [760]), distillation, and selective invocation of expensive components have become load-bearing for robotic and AR loops at frame rate.

Symmetry and geometric ambiguity are irreducible. Symmetric and texture-less parts of the kind that fill T-LESS [768] admit a continuous or discrete family of photometrically indistinguishable poses, and any loss assuming a unique target diverges on them; transparent and specular surfaces violate the brightness-constancy and depth-sensing assumptions behind correspondence and RGB-D fusion; rotation parameterization adds degeneracies that continuous 6D or axis-angle encodings reduce but do not remove; and heavy occlusion, the regime of LINEMOD-Occluded [767], starves the estimator of the evidence needed to resolve any of them. Generative multi-hypothesis pose (Diff9D [765]) predicts a distribution instead of a point, but turning that distribution into a *calibrated* confidence a planner can trust in safety-critical manipulation and driving is still largely unaddressed.

Across these problems, pose pipelines increasingly consume foundation-model priors as plug-in modules (promptable segmentation for zero-shot detection, monocular depth for inference-time geometry) rather than training pose-specific backbones, and the boundary between pose estimation, 6-DoF tracking, and neural reconstruction is dissolving, as FoundationPose [759] folds estimation and tracking into one loop and the reconstruction discussion of Chapter 8 pursues further. Promptable, real-time, uncertainty-aware pose is the target the next generation of methods converges on.

6.3 Spatial Structure Construction



ITH entities recognized and their positions fixed, what remains is how they hang together. Spatial structure construction turns independently localized entities into an organized configuration: which objects relate to which, how those relations compose into a scene, and how the scene as a whole is laid out. The section builds outward in scope, from pairwise spatial relations, through the scene graphs that assemble them, to room-level layout; it then lets structure take on time and scale, as the event structure of action detection and the whole-environment structure of mapping and SLAM. Structure in real perception is seldom purely static, and what the section yields is the explicit spatial configuration that downstream reasoning and planning act on.

6.3.1 Spatial Relation Extraction

Spatial Relation Extraction (SRE) identifies the relative spatial configuration of objects, such as "table-left of-chair" or "car-behind-bus". It turns detections or segmentations into spatial relation knowledge that feeds reasoning, scene graph generation [776, 777, 778], autonomous driving [779, 780, 781, 782], and embodied AI [783, 784]. It is narrower than Scene Graph Generation (SGG), which runs the full pipeline from detection to relation prediction and includes non-spatial predicates ("person-holding-book"); here SRE covers only geometric, topological, and projective spatial predicates.

6.3.1.1 Task Definition

Given a visual scene (a 2D image, a video, a point cloud, or an RGB-D stream) and two target object instances specified by tags, boxes, or masks, SRE outputs their spatial relationship either as a triplet (s, p, o) with p drawn from a spatial predicate vocabulary, or as a natural-language sentence describing the relation and its context. Three subtasks differ in output granularity. **Visual Spatial Classification (VSC)** [785, 786] predicts the predicate from a closed label set ("on", "in", "next to"). **Visual Spatial Description (VSD)** [787, 788] generates a free-form sentence that can carry attributes and context beyond the bare predicate. **Quantitative spatial understanding** [789, 229] replaces qualitative predicates with metric quantities (distances, absolute sizes, 3D orientations), which navigation and manipulation require. The field has moved through these three in order, from closed-set classification to generation to metric quantification.

6.3.1.2 Methodology

Figure 6.13 summarizes the task and the three method families surveyed below.

Rule-based approaches. Rule-based pipelines [779, 787, 790] first run off-the-shelf perception (a 2D detector with a bird's-eye-view transform [791], or a 3D scene extractor [792]) to obtain each object's 3D position, size, orientation, and appearance in a camera-centered frame whose vertical axis is aligned with gravity, and then apply deterministic geometric rules to the centroids to produce (O_1, p, O_2) triplets [787]. Let $(x_{O_i}, y_{O_i}, z_{O_i})$ denote the world-coordinate centroid of object O_i, with y aligned to the gravity axis. Then

$$\begin{aligned} &\Delta x = |x_{O_1} - x_{O_2}|, \quad \Delta y = |y_{O_1} - y_{O_2}|, \quad \Delta z = |z_{O_1} - z_{O_2}|, \\ &\quad \Delta y > \tau \wedge y_{O_1} > y_{O_2} \;\Rightarrow\; \text{“above”}, \\ &\quad \Delta x > \Delta z \wedge \Delta x > \tau \;\Rightarrow\; \text{“left”/“right”}, \end{aligned} \tag{6.34}$$

where τ is a small threshold (*e.g.*, 0.2 m). Such rules are cheap but brittle: 2D-to-3D back-projection misplaces occluded or oddly posed objects (a rear vehicle whose centroid lands ambiguously on the BEV plane), fixed rules cannot express functional or contextual relations ("person riding bicycle", a book more likely on a shelf than under it), and every new domain needs its rules rewritten.

Learning-based approaches. Zhao *et al*. [787] decoupled spatial relation generation from generic captioning and trained it with a joint visual–spatial objective; 3DVSD [788] moved the same task into 3D scene modeling and added structure-aware sampling that holds up under occlusion.

MLLM-based approaches. Multimodal large language models carried the line from image-level predicates toward finer and more metric grounding. SpatialVLM [229] coupled monocular depth with the visual encoder during pre-training so the model answers metric queries such as "How far is A from B?", outperforming GPT-4V on spatial VQA. SpatialBot [793] exposes pixel-level depth queries through a depth API and trains progressively, translating spatial reasoning into higher robotic grasping success. SpatialRGPT [789] sharpens granularity from the whole image to regions, accepting boxes or masks and auto-generating 8.7M region-level spatial samples through scene-subgraph sampling (S^3). SpatialLLM [794] pairs a hybrid CLIP–DINOv2 encoder with staged data injection to hold qualitative and quantitative capabilities in one model.

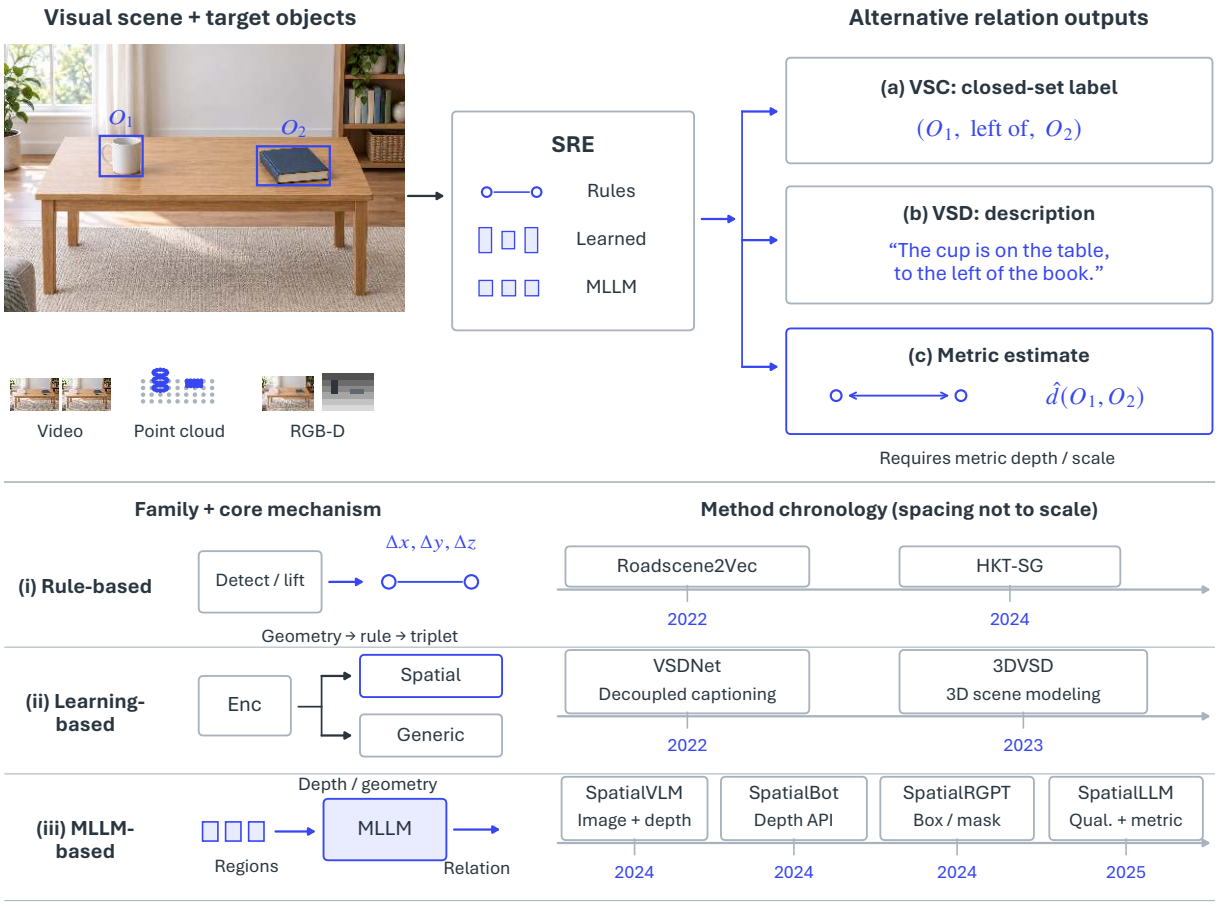


Figure 6.13: Spatial relation extraction. *Top*: given a visual scene (image, video, point cloud, or RGB-D) and two target objects (O_1, O_2), predict (a) a closed-set relation label (VSC), (b) a free-form description (VSD), or (c) a quantitative metric estimate. *Bottom*: (i) rule-based, (ii) learning-based decoupled, and (iii) MLLM-based mechanisms with method timelines (chronological spacing is not proportional). Metric outputs require depth or scale information.

6.3.1.3 Datasets and Metrics

Table 6.17 lists the benchmarks. They differ mainly in how relations were collected: manual labeling of robot-captured lab scenes in SpatialAwareRobot [795]; adversarial crowdsourcing in SpatialSense [785], where annotators propose relations that fool a baseline and only those count, which makes the dataset deliberately hard; the merge of SpatialSense and Visual Genome [796] into VSRC [787], whose 143K free-form descriptions make it the standard VSD benchmark; the 34-preposition, English–French annotations of SpatialVOC2K [797]; and the automatically generated, depth-aware SpaceSGG [776], released as 10K scene descriptions, 20K question–answer pairs, and 20K multi-turn conversations over 20K scenes.

Each subtask has its own evaluation regime, and the metrics are not comparable across them. VSC is scored as classification: predicate accuracy and per-relation precision/recall on SpatialSense [785] and VSRC [787], with the driving-scene systems Roadscene2Vec [779] and HKT-SG [790] reporting per-relation F1 or accuracy on their own data; results on SpatialSense's adversarial split lag well behind those on conventional crowdsourced data, which is the contextual-relation difficulty taken up under Challenges. VSD is scored as generation: VSDNet [787] and 3DVSD [788] report BLEU-4, METEOR, and CIDEr on VSRC, and LLaVA-SpaceSGG [776] reports QA accuracy on SpaceSGG. Quantitative understanding is the newest regime: SpatialVLM [229] and SpatialRGPT [789] report distance- and size-estimation accuracy against depth-aligned

Table 6.17: Benchmark datasets for spatial relation extraction. "–" marks an open vocabulary without a fixed category count.

Dataset	Scene Type	Modality	Annotation	Samples	Relation Categories
SpatialAwareRobot [795]	Robotics	Image	Manual	1,000	7
SpatialSense [785]	Generic	Image	Manual	11,569	9
VSRC [787]	Generic	Image	Manual	29,000	9
SpatialVOC2K [797]	Generic	Image	Manual	2,026	34
SpaceSGG [776]	Generic	Image	Automated	20,000	–

ground truth on auto-generated test sets (SpatialVLM-Bench, SpatialRGPT-Bench), SpatialLLM [794] reports combined qualitative and quantitative VQA accuracy, and SpatialBot [793] adds robotic grasp success as an embodied proxy. MLLM-based models dominate the qualitative leaderboards but still fall short on metric-scale reasoning, which motivates the unified representations discussed next.

6.3.1.4 Challenges

Spatial relations are not purely geometric. "A cup on a table" is plausible and "a cup on a wall" is not, yet the two differ only in object identity, so rule-based and purely geometric methods cannot capture functional or commonsense constraints; a model needs to know what objects are for and where they tend to be, not just their boxes or point coordinates. Metric grounding is the second gap. MLLMs handle contextual and functional predicates well but judge distances, sizes, and angles poorly; injecting geometry and depth during pre- or post-training, or exposing dedicated spatial tools, is the current route, and the longer-term goal is a representation that holds discrete predicates ("near") and continuous quantities ("2 meters away") together, since navigation and manipulation need both. Generalization is the third. SpatialSense [785] and VSRC [787] cover a small slice of real scenes, and models trained on them degrade under new object appearances, scene complexity, lighting, and occlusion; SpatialVLM [229] and its successors generalize better, but unseen and dynamic environments remain largely untested.

6.3.2 Scene Graph Generation

Scene graph generation (SGG) [798, 799, 800] parses visual content into a graph of triplets, each joining a subject and an object through a predicate, for example $(\text{person}, \text{rides}, \text{horse})$. The graph records not only which objects are present but how they interact, which is what retrieval [598, 801], captioning [802, 803], autonomous driving [804, 805, 779], and robotics [806, 807, 808] consume downstream.

6.3.2.1 Task Definition

Given a visual scene, SGG produces a graph

$$\mathcal{G} = \{(s_i, p_i, o_i)\}_{i=1}^{M}, \qquad s_i, o_i \in \mathcal{O},\ p_i \in \mathcal{P}, \tag{6.35}$$

where each entity in $\mathcal{O}$ is an object instance with a localization (a bounding box, or a mask in the panoptic setting) and a class label, and p_i is a predicate from a relation vocabulary $\mathcal{P}$ describing how s_i and o_i interact.

The task is instantiated by input modality. **Image SGG (ISG)** [796, 809] on static 2D images is the original and most studied setting and anchors most benchmarks. **Video SGG (VSG)** [810, 811] adds time, so relations become dynamic and temporally extended, and temporal context disambiguates interactions a single frame leaves open. **Textual SGG (TSG)** [812, 813] parses a sentence ("A person is riding a horse in a field") into a graph, treating SGG as structured language parsing. **3D SGG (3DSG)** [342, 343] operates on point clouds or meshes and models relations in metric 3D space, free of the perspective ambiguity of images. **Universal SGG (USG)** [778] aims at one graph from several simultaneous modalities (image, video, 3D, text). Two

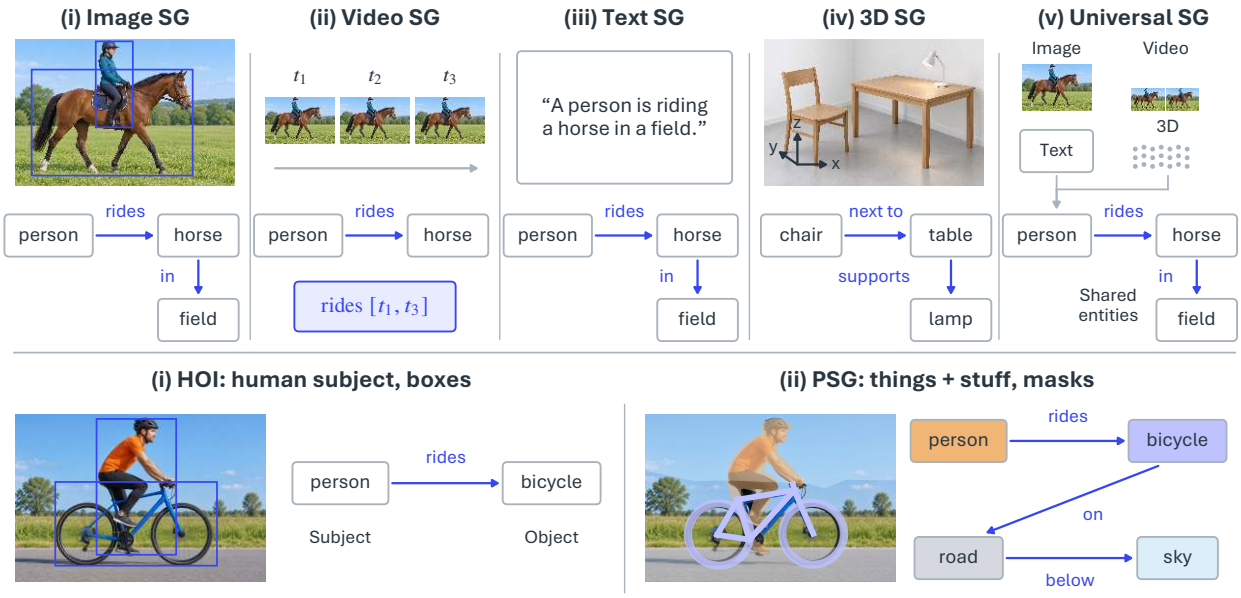


Figure 6.14: Scene graph generation by input modality (top) and sub-task setting (bottom). *Top*: image, video, text, 3D, and universal (multimodal) SGG, each with a sample input and output graph. *Bottom*: (i) human–object interaction (HOI), restricted to human subjects at box level, and (ii) panoptic SGG (PSG), grounding "things" and "stuff" on pixel masks.

further variants change the level of detail rather than the modality. **Human–Object Interaction detection (HOI)** [814, 815] restricts the subject to humans and has become a field of its own because of pose variability and the subtlety of intent. **Panoptic SGG (PSG)** [816, 817] grounds nodes on panoptic segmentation masks instead of boxes, so relations can involve "stuff" (sky, road) and are localized without the imprecision of boxes.

6.3.2.2 Methodology

Figure 6.14 illustrates the five modality settings and the two specializations with a representative input and output each.

Two-stage methods. Two-stage methods [798, 809, 818, 819, 820, 821] were the standard for years: a pre-trained detector such as Faster R-CNN [172] produces boxes and per-object features, and a separate relation model predicts a predicate for each pair. Specializing each stage made the graphs accurate, but detector errors propagate uncorrected into relation prediction, the two models are trained and tuned separately, and object interactions are only modeled after detection has already committed.

One-stage methods. One-stage methods [822, 823, 824, 825] fold detection and relation prediction into one end-to-end network. Pixel-based [824] and fully convolutional [825] formulations came first; after DETR [136], most one-stage SGG adopted set prediction with learned queries, and the methods differ in what the queries represent. RelTR [826] uses paired subject–object queries so the Transformer predicts relations between pairs directly; SGTR [827] decouples subject, object, and predicate queries and composes them, which lets each component be predicted on its own; and PSGTR [816] uses triplet queries that emit scene-graph triplets without a detector at all.

Prior knowledge. Because predicates are long-tailed and many relations are not visually decidable, SGG has long imported priors. Language priors [828, 829, 830] supply semantic embeddings that help rare predicates;

Table 6.18: Benchmark datasets for scene graph generation.

Dataset	Modality	Obj. Class	BBox	Rela. Class	Triplets	Instances
Visual Phrase [838]	Image	8	3,271	9	1,796	2,769
Scene Graph [598]	Image	266	69,009	68	109,535	5,000
VRD [829]	Image	100	-	70	37,993	5,000
Open Images v7 [442]	Image	600	3,290,070	31	374,768	9,178,275
Visual Genome [796]	Image	5,996	3,843,636	1,014	2,347,187	108,077
GQA [839]	Image	200	-	310	-	3,795,907
VrR-VG [840]	Image	1,600	282,460	117	203,375	58,983
UnRel [841]	Image	-	-	18	76	1,071
SpatialSense [785]	Image	3,679	-	9	13,229	11,569
SpatialVOC2K [797]	Image	20	5,775	34	9,804	2,026
OpenSG [816]	Image (panoptic)	133	-	56	-	49K
AUG [842]	Image (overhead)	76	-	61	-	-
STAR [843]	Image (satellite)	48	219,120	58	400,795	31,096
ReCon1M [844]	Image (satellite)	60	859,751	64	1,149,342	21,392
SkySenseGPT [845]	Image (satellite, instruction)	-	-	-	-	-
Traffic Scene Graph [804]	Image (traffic)	2,266	-	4,272	-	451
ImageNet-VidVRD [846]	Video	35	-	132	3,219	100
VidOR [847]	Video	80	-	50	-	10,000
Action Genome [848]	Video	35	0.4M	25	1.7M	10,000
AeroEye [849]	Video (drone)	56	-	384	-	2.2M
PVSG [817]	Video (panoptic)	126	-	57	4,587	400
ASPIRe [850]	Video (interlacements)	-	-	4.5K	-	1.5K
Ego-EASG [851]	Video (egocentric)	407	-	235	-	-
IESG [805]	Video (traffic)	9	-	16	-	-
3DSSG [342]	3D	528	-	39	-	48K
PSG4D [852]	4D	46	-	15	-	-
4D-OR [853]	4D (operating room)	12	-	14	-	-
MM-OR [854]	4D (operating room)	-	-	-	-	-
EgoExOR [855]	4D (operating room)	36	-	22	568,235	-
TSG Bench [856]	Text	-	-	-	11,820	4,289
FACTUAL [812]	Text, Image	4,042	-	1,607	40,149	40,369
DiscoSG-DS [857]	Text, Image	4,018	-	2,033	68,478	8,830

statistical priors [831, 832, 821] model object-pair co-occurrence; knowledge graphs such as ConceptNet [833] add commonsense [834, 835]; and LLMs bring world knowledge [836, 837].

Beyond images. The non-image settings reuse these designs and add modality-specific machinery. Video SGG tracks object tubes and aggregates relations over time: TRACE [810] decomposes relation prediction by temporal segment and STTran [811] applies a spatio-temporal Transformer over frame-level graphs. Textual SGG is a parsing problem, for which the FACTUAL parser [812] fine-tunes a sequence-to-sequence model to emit faithful triples. 3D SGG predicts relations from point-cloud or mesh segments with graph networks [342], and SceneGraphFusion [343] does so incrementally from an RGB-D stream so the graph grows with the reconstruction. Panoptic SGG is served by the triplet-query PSGTR [816] and its video extension [817], and USG [778] shares one graph decoder across image, video, 3D, and text inputs.

6.3.2.3 Datasets and Metrics

Table 6.18[1] lists the benchmarks by modality. The image benchmarks run from the small precursors Visual Phrase [838], Scene Graph [598], and VRD [829], which first exposed the long-tailed predicate distribution, to Visual Genome [796], Open Images [442], and GQA [839], with VrR-VG [840] and UnRel [841] carved out to test visually relevant and zero-shot relations respectively. Visual Genome is the primary benchmark but its raw release (33,877 object and 40,480 relationship synsets) is noisy and heavily long-tailed, so the community works

[1]The table is extended from the GitHub repository: https://github.com/ChocoWu/Awesome-Scene-Graph-Generation

Table 6.19: Representative image SGG methods on Visual Genome (VG150), scene graph detection (SGDet) with graph constraint, as reported in the comparison of EGTR [822]. Two-stage methods use a Faster R-CNN detector with a ResNeXt-101-FPN backbone; one-stage methods are DETR-based with the backbone shown. "–" = not reported.

Method	Family / backbone	Year	R@50	R@100	mR@50	mR@100
Neural Motifs [809]	Two-stage, X-101-FPN	2018	32.1	36.9	5.5	6.8
BGNN [818]	Two-stage, X-101-FPN	2021	31.0	35.8	10.7	12.6
SGTR [827]	One-stage, DETR-R101	2022	24.6	28.4	12.0	15.2
RelTR [826]	One-stage, DETR-R50	2023	27.5	–	10.8	–
EGTR [822]	One-stage, Deformable DETR-R50	2024	30.2	34.3	7.9	10.1

with cleaned subsets: the broad cleaning in Table 6.18 keeps roughly 6K object and 1K predicate classes, and the de-facto SGG benchmark VG150 restricts these to the 150 most frequent object and 50 most frequent predicate classes. Spatial-only relations are covered by SpatialSense and SpatialVOC2K (see the preceding subsection), panoptic grounding by OpenSG [816], and non-standard viewpoints by the overhead AUG [842], the satellite STAR [843], ReCon1M [844], and instruction-based SkySenseGPT [845], and the traffic-scene graphs of [804]. Video benchmarks began with ImageNet-VidVRD [846] and the trajectory-annotated VidOR [847], then specialized to human activity (Action Genome [848]), drone footage (AeroEye [849]), panoptic video (PVSG [817]), fine topological interlacements (ASPIRe [850]), egocentric capture (Ego-EASG [851]), and imminent traffic events (IESG [805]). 3DSSG [342] is the standard 3D benchmark; PSG4D [852] adds time, and the operating-room sets 4D-OR [853], MM-OR [854], and EgoExOR [855] bring multimodal (RGB-D, audio, robot logs) and egocentric surgical scenes. On the text side, TSG Bench [856] evaluates LLMs on graph generation from narratives, while FACTUAL [812] and DiscoSG-DS [857] pair images with text for factually consistent, cross-modal graphs.

Evaluation separates what is given from what is predicted. In **Predicate Classification (PredCls)** boxes and object labels are given and only predicates are predicted; in **Scene Graph Classification (SGCls)** boxes are given and both object labels and predicates are predicted; in **Scene Graph Generation (SGGen, also SGDet)** everything is predicted, and a triplet counts only if both boxes overlap their ground truth at IoU ≥ 0.5, with **Phrase Detection** relaxing this to the union box and SGGen+ also crediting correctly detected singleton nodes. Panoptic SGG replaces box IoU with mask IoU. The standard metric is recall of ground-truth triplets among the top-K predictions,

$$R@K = \frac{|Top_K \cap GT|}{|GT|}, \tag{6.36}$$

under a graph constraint (one predicate per pair) or without it. Because $R@K$ is dominated by frequent predicates, **mean Recall@K** ($mR@K$) averages recall per predicate class and has become the long-tail metric of record; the gap between $R@K$ and $mR@K$ drove much of the debiasing literature [819, 818, 821]. Weighted triplet recall, micro- versus macro-averaging over images, the stricter mAP_{rel} / mAP_{phr} of the Open Images challenge, and retrieval-based R-Precision are reported in specific settings.

Each modality has its own benchmark and protocol, so leaderboards are not comparable across types: video SGG uses ImageNet-VidVRD, VidOR, and Action Genome with the same triad plus temporal-IoU thresholds; textual SGG uses set F1 over triples on FACTUAL and LLM-judged action coverage on TSG Bench; 3D SGG uses object-, predicate-, and triplet-level recall on 3DSSG; panoptic SGG uses mask-IoU $R@K$ on OpenSG and PVSG. Table 6.19 therefore reports only the image setting, where VG150 SGDet is shared.

Table 6.19 shows the trade-off that still shapes image SGG. The two-stage Motifs baseline keeps the highest overall recall, but its mean recall is the lowest, which is the frequent-predicate bias in numbers; BGNN's debiasing doubles mean recall at a small cost in $R@K$. The one-stage DETR-style models start several recall points lower, since they must detect and relate in one pass with a weaker backbone, yet SGTR and RelTR

reach the best mean recall of the group, and EGTR closes most of the overall-recall gap to two-stage methods while running end to end.

6.3.2.4 Challenges

From single-modality to universal SGG. Methods are built per modality, yet real scenes are multimodal, with images carrying visual detail and text carrying abstract concepts and events. A universal scene graph [778] would express all of them in one cross-modal representation; the hard part is identifying the same objects, events, and relations across modalities and suppressing cross-modal noise, and granularity has to follow the application rather than a fixed ontology, since robotic interaction or driving may need relations at a level of detail that image-level graphs do not provide.

Integrating SGG with MLLMs. MLLMs are strong at language and weaker at structured visual understanding. Scene graphs offer a way in: pretraining objectives that mask and predict graph nodes (objects, attributes, relations) across image, video, and text inputs would teach a language model to read structure rather than only the next token, and the scene-graph-guided MLLMs of the preceding subsection [776] are an early instance.

From images to embodied use. Most SGG analyzes static images or videos, but the graph is most useful when it is kept live: robots update it as they act [806, 807, 808], and driving systems read relations among lanes, signs, and pedestrians from it [804, 805, 779]. As these uses grow, the scene graph shifts from a product of perception to the state on which decision and interaction operate.

6.3.3 Layout Estimation and Room Parsing

Layout estimation recovers the structural geometry of an indoor space, the corners, walls, floor, and ceiling that bound it, as a simplified 3D "wireframe" called the room layout. Room parsing extends this by also detecting, localizing, and labeling the objects inside (furniture and fixtures), giving a complete description of the scene. For outdoor scenes, the analogous problem is recovering a road layout in bird's-eye view from a street image [858]. Figure 6.15 shows the inputs and output representations for the structural part of the task.

6.3.3.1 Task Definition

Given an image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ (a perspective view, a 360° panorama, or a multi-view or video capture), layout estimation recovers the dominant surfaces that bound the navigable volume of a room as a *room layout* $\mathcal{L} = \{P_k\}_{k=1}^{K}$, where each P_k is a planar polygon in 3D; equivalently the output may be plane parameters $(\mathbf{n}_k, d_k)$ or a per-pixel structural-surface label map. Under a vertical-wall model, the layout compresses to a floor-plan polygon with floor and ceiling heights y_f, y_c. Outdoor road-layout methods instead predict road regions and vehicle occupancy in bird's-eye view. Room parsing adds per-object instance masks $\{m_i\}$ or boxes with class labels, drawn from a fixed ontology or, in the open-vocabulary setting, from free-form language.

6.3.3.2 Methodology

Pre-deep-learning approaches. Classical room layout estimation rests on the **Manhattan assumption** [8]: the scene is built from surfaces aligned with three mutually orthogonal directions, as walls, floors, and ceilings in most built environments are. Hedau *et al*. [859] introduced the task, modeling a cluttered indoor image jointly as a 3D box layout (the room as if empty) and per-pixel surface labels (visible walls, floor, ceiling, and objects). Three mutually orthogonal vanishing points are first estimated with a robustified variant of

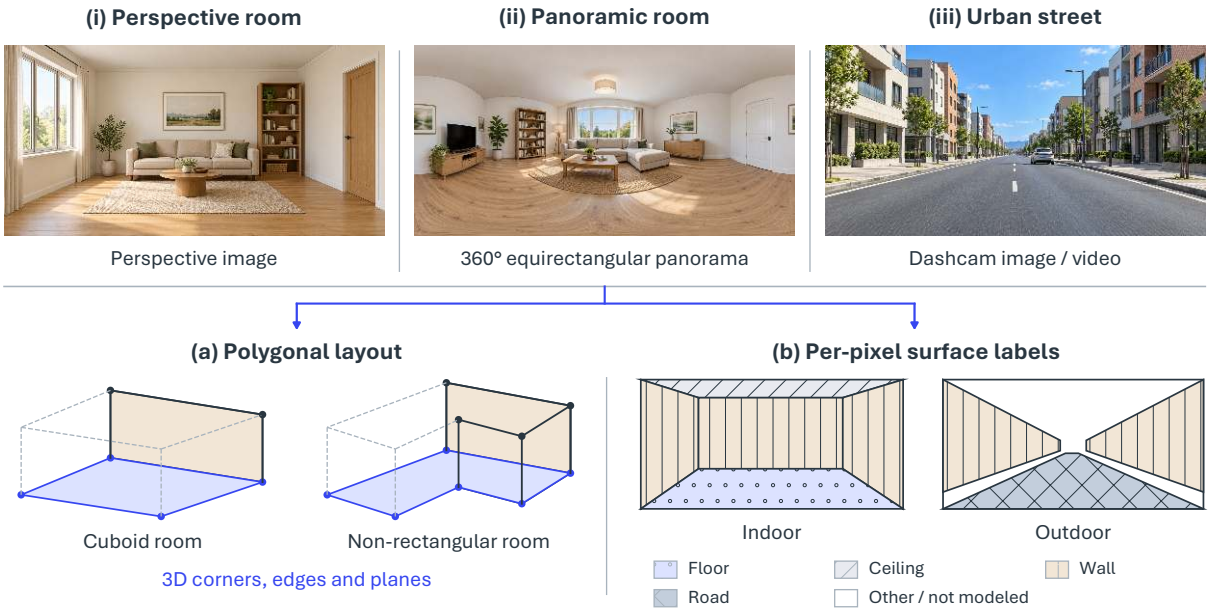


Figure 6.15: Inputs and output representations of structural layout estimation. Top: (i) indoor perspective images, (ii) 360° panoramas, and (iii) outdoor street views. Bottom: (a) polygonal room layouts and (b) schematic indoor/outdoor surface-label maps, illustrating representations that may be predicted separately or jointly. Object-level parsing and outdoor bird's-eye-view outputs are not depicted.

Rother's method [860], in which a line segment l votes for a candidate vanishing point p with

$$v(l,p) = |l| \cdot \exp\left(-\frac{\alpha(l,p)^2}{2\sigma^2}\right), \tag{6.37}$$

where $\alpha(l,p)$ is the angle between l and the line from the midpoint of l to p, $|l|$ is the segment length, and σ is a fixed bandwidth; candidates are selected greedily, removing the lines that supported each selected point. The vanishing points fix the orientation of the box; sampling pairs of rays through two of them generates box-layout hypotheses, whose remaining edges follow from the third vanishing point, and a learned structured regressor scores the hypotheses and selects the layout.

Deep learning approaches. Mallya and Lazebnik [861] first used a fully convolutional network to predict informative edge maps, then kept the classical hypothesis-generation-and-ranking step, scoring hypotheses with features from the learned edges together with line membership and geometric context. Later deep estimators are best organized by input modality and geometric representation, which vary independently: panorama input does not imply a cuboid output, and non-cuboid geometry need not break the Manhattan assumption. Figure 6.16 groups the representative methods of 2015–2021 along these lines.

(i) Perspective-image corner and edge predictors. RoomNet [862] regresses an ordered set of room corners with a recurrent encoder–decoder conditioned on one of 11 predefined cuboid topologies, and LayoutNet [863] predicts corner and boundary probability maps and fits a cuboid to them. The topology constraint simplifies reconstruction, but a single perspective view often leaves part of the room outside the frame.

(ii) Panorama-based methods. A 360° equirectangular panorama removes the field-of-view truncation of a perspective image, though not occlusion. LayoutNet [863] predicts 2D corner and boundary maps from a gravity-aligned panorama; HorizonNet [864] compresses the layout into three 1D per-column signals (floor-wall boundary, ceiling-wall boundary, and wall-wall corner indicator), adds Pano Stretch augmentation, and reconstructs general Manhattan layouts as well as cuboids; HoHoNet [865] learns latent horizontal

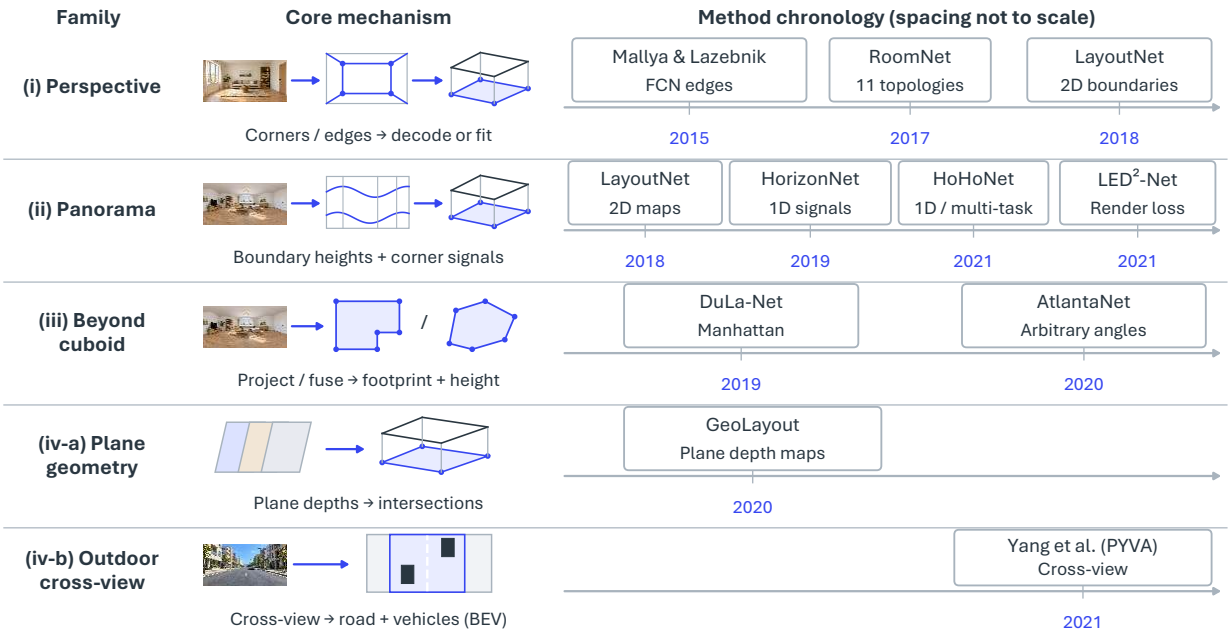


Figure 6.16: Representative layout-estimation methods, 2015–2021. Each lane pairs a core-mechanism illustration with a publication timeline: (i) perspective corner and edge prediction, (ii) panorama representations, (iii) beyond-cuboid floor plans, and (iv-a, iv-b) indoor plane geometry and outdoor cross-view prediction. Timeline spacing is not proportional; arrows indicate chronology, not method dependencies.

features decodable into layout, depth, and semantics; and LED2-Net [866] supervises the layout through differentiable depth rendering, comparing depth rendered from the predicted layout with the target geometry.

(iii) Beyond-cuboid and Atlanta-world methods. Floor-plan prediction separates the number of walls from the constraints on their orientation. DuLa-Net [867] combines equirectangular and ceiling–floor projections to estimate a floor plan under a Manhattan prior, which admits non-rectangular rooms with orthogonal walls. AtlantaNet [868] projects a gravity-aligned panorama onto horizontal planes above and below the camera and predicts a room footprint and height under the Atlanta-world model, which keeps walls vertical but allows arbitrary horizontal orientation, including curved footprints. Both still assume vertical walls, so slanted walls and ceilings remain out of reach.

(iv) Plane-based indoor geometry and outdoor cross-view prediction. GeoLayout [869] predicts pixel-level plane parameters, infers the depth map of each dominant plane, and reads layout boundaries off the intersections of those depth maps, a geometric alternative to regressing polygon vertices; the accompanying Matterport3D-Layout dataset covers cuboid and non-cuboid rooms. For outdoor scenes, Yang *et al.* [858] use a cross-view transformation with cycle-consistency constraints to predict road layout and vehicle occupancy in bird's-eye view from a monocular front-facing image.

6.3.3.3 Datasets and Metrics

Table 6.20 summarizes the benchmarks. Perspective-image evaluation grew from Hedau [859] (308 web and LabelMe images with box layouts and surface labels) through its augmented successor Hedau+ and the more cluttered SUNbox drawn from SUN2012 [607], both built by Mallya and Lazebnik [861] to train FCN edge predictors, to LSUN Room Layout [870], the standard perspective benchmark with corner positions and cuboid topologies on over 5K images. Corridor and video data serve navigation: Yang [871] labels ground versus wall in corridor images, FreeYourCamera [872] records six-degree-of-freedom smartphone walks that frequently violate box and Manhattan assumptions, and MichiganIndoorCorridor [873] provides non-Manhattan corridors with glass and reflections. Panoramic and 3D evaluation rests on PanoContext [874]

Table 6.20: Layout estimation and room parsing benchmarks. Scale is in images unless noted.

Dataset	Year	Scene	Modality	Scale	Annotation
Hedau [859]	2009	Room	Image	308	Box layout + surface labels
Hedau+ [861]	2015	Room	Image	337	Box layout (Hedau + 75 bedroom images)
SUNbox [861]	2015	Room	Image	543	Box layout, cluttered SUN2012 scenes
LSUN Room Layout [870]	2015	Room	Image	5,394	Corners + 11 cuboid topologies
Yang [871]	2016	Corridor	RGB-D image	967	Ground / non-ground masks
FreeYourCamera [872]	2013	Mixed	Video (6-DoF)	10 seq. (84 frames)	Layout under non-box, non-Manhattan motion
MichiganIndoorCorridor [873]	2011	Corridor	Video	11 seq. (∼3,300 frames)	Ground + three walls on first frame
PanoContext [874]	2014	Room	Panorama	∼500	Cuboid layout + furniture
Matterport3D-Layout [869]	2020	Room	RGB-D image	7,360	Plane parameters, cuboid + non-cuboid
Stanford 2D-3D-S [875]	2017	Building	RGB-D panorama	∼70K	Layout, depth, normals, semantics, mesh

Table 6.21: Panorama-based layout estimation, 3D IoU (%) on PanoContext, Stanford 2D-3D-S, and Matterport3D-Layout (overall), as reported in the comparison of LED2-Net [866]. Higher is better.

Method	Representation	Year	PanoContext	Stanford 2D-3D-S	Matterport3D-Layout
LayoutNet [863]	2D corner + boundary maps	2018	74.48	76.33	75.82
DuLa-Net [867]	Dual projection, Manhattan plan	2019	77.42	79.36	75.05
HorizonNet [864]	1D column representation	2019	82.17	79.79	78.73
AtlantaNet [868]	Top-down footprint, Atlanta world	2020	78.76	82.43	80.02
LED2-Net [866]	Differentiable depth rendering	2021	**82.75**	**83.77**	**81.52**

(cuboid layouts and furniture for about 500 SUN360 panoramas, the standard testbed for LayoutNet, HorizonNet, and LED2-Net), Stanford 2D-3D-S [875] (about 70K RGB-D panoramas with depth, normals, semantics, and meshes over 270 rooms, enabling joint layout-depth-semantics evaluation), and Matterport3D-Layout [869], derived from Matterport3D [340] with RANSAC-fitted plane ground truth for cuboid and non-cuboid rooms.

Metrics follow the output representation. Perspective benchmarks (Hedau, LSUN) report *pixel error*, the fraction of pixels whose predicted surface label differs from the ground truth, and *corner error*, the mean distance between predicted and true corners normalized by image size; the edge-map methods additionally report the contour-detection F-scores at the optimal dataset scale (ODS, one threshold for the whole set) and optimal image scale (OIS, the best threshold per image). Panoramic and 3D benchmarks (PanoContext, Stanford 2D-3D-S, Matterport3D-Layout) report 3D and 2D *room IoU* between the predicted and ground-truth layout volumes or floor plans, with frequency-weighted IoU and pixel accuracy as segmentation-style complements. A method trained for one input type rarely transfers to the other without re-training, so perspective and panoramic results are not compared directly, and the outdoor road-layout protocol [858] is separate again.

Table 6.21 compares the panorama methods under one protocol. The 1D column representation of HorizonNet gained about eight IoU points over LayoutNet's 2D maps on PanoContext, and the geometric supervision of LED2-Net adds a further point or two on every benchmark; the beyond-cuboid methods pay

for their generality on cuboid-heavy PanoContext (DuLa-Net, AtlantaNet below HorizonNet) and recoup it on Matterport3D-Layout, whose non-cuboid rooms are exactly what they were designed for.

6.3.3.4 Challenges

The geometric priors that make layout estimation tractable also bound it. Cuboid models cannot express alcoves or L-shaped plans; Manhattan models keep orthogonal walls even when the footprint is free-form; the Atlanta-world model of AtlantaNet [868] frees wall orientation but not wall tilt; and plane-based methods such as GeoLayout [869] approximate curved geometry only with many planes. Slanted enclosures, split levels, and the outdoor bird's-eye-view target of [858], which is a different output space altogether, all sit outside the standard formulations. Clutter and occlusion are the second obstacle: furniture hides the corners, boundaries, and vanishing lines that both classical and learned methods read, so estimators must hallucinate hidden structure from layout priors, and confidently wrong boundaries without uncertainty are the typical failure. Generalization follows from the data: benchmarks are indoor, mostly rooms and corridors, and models trained on one set degrade on another's appearance, architecture, and lighting, while sparse, unposed multi-view capture breaks the pose assumptions of classical pipelines; the feed-forward geometry models of §6.2.4 are the natural starting point for layout from such input.

6.3.4 Action Detection

Action detection (AD) finds human actions in untrimmed video: for each action instance it returns the time interval $[t_s, t_e]$ in which it occurs and its category c. Where action recognition classifies a clip that has already been trimmed to one action, detection must also decide when actions start and end, in streams that contain many of them. It belongs to spatial structure construction because it organizes the entities and relations of a scene along the time axis into events, the temporal counterpart of the scene graphs above and the basis of video SGG and human–object interaction. Figure 6.17 illustrates the task.

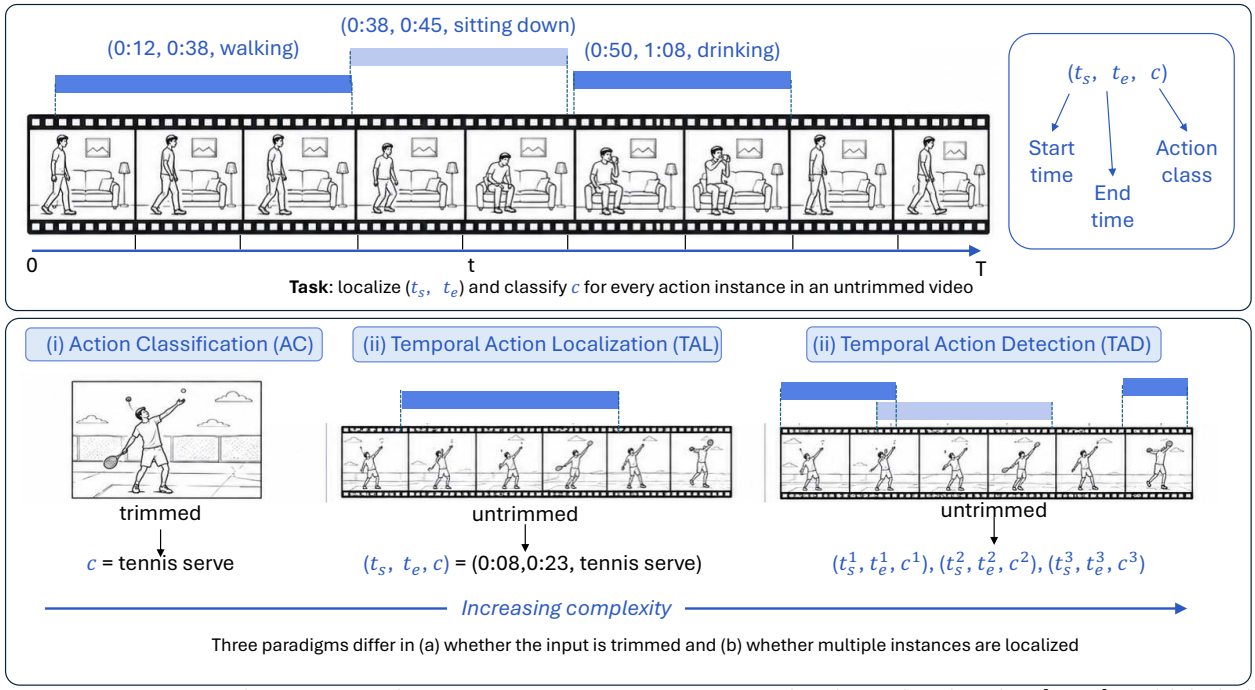


Figure 6.17: Action detection: each action instance in an untrimmed video is localized to $[t_s, t_e]$ and labeled with class c (top). *Bottom*: three settings of increasing complexity: (i) action classification of a trimmed clip, (ii) temporal action localization of one interval, and (iii) temporal action detection of multiple instances.

6.3.4.1 Task Definition

Given an untrimmed video (or a continuous multi-sensor stream), AD outputs a set of tuples (t_s, t_e, c) with start time, end time, and category for every action instance. Three settings differ in what is assumed about the input. **Action classification (AC)** takes a trimmed clip and predicts one category, with no localization. **Temporal action localization (TAL)** predicts start and end times in untrimmed video, usually assuming one dominant action per video. **Temporal action detection (TAD)** is the general case: multiple instances per video, possibly overlapping or concurrent, across temporal scales, each to be localized and classified.



Figure 6.18: Representative methods and mechanisms for (i) action classification, (ii) temporal localization with boundary estimation and alternative supervision signals, and (iii) temporal detection with anchor-based, anchor-free, boundary-based, and query-based formulations. Method timelines show publication order, not computational dependencies; spacing is not proportional. Same-year methods are grouped. The localization and detection panels emphasize different aspects of closely related tasks.

6.3.4.2 Methodology

Figure 6.18 organizes the methods by the three settings; each builds on the previous one's representations.

Action Classification. Two-stream CNNs [876] model appearance from RGB frames in one stream and motion from dense optical flow in another, fusing the two at the output. 3D CNNs [877] convolve over space and time jointly, C3D [238] and I3D [239] being the standard examples, at the cost of a large parameter count that R(2+1)D [878] and P3D [879] reduce by factoring 3D convolution into 2D spatial and 1D temporal steps. TSN [880] samples a few snippets sparsely across a long video and supervises at video level, which scales long-range temporal modeling cheaply, and Transformers [881, 882] model global spatio-temporal dependencies among sampled tokens beyond the local reach of convolution.

Temporal Action Localization. Early TAL ran a classifier over dense, fixed-size sliding windows, which was both slow and blind to the wide range of action durations. Structured segment networks [883] instead

model a proposal's starting, ongoing, and ending stages and score its completeness, which refines boundary regression. Boundary-sensitive methods [884] go further and predict start and end probabilities at every snippet, then pair high-confidence boundary points into proposals; this flexible boundary generation is the component on which later detection systems rest.

Because precise boundary annotation is expensive, weakly supervised TAL (WSTAL) [885, 886, 887] learns from video-level labels alone. The task becomes multiple-instance learning: a video is positive if at least one segment contains the action, class activation sequences are derived from video features, and foreground segments are thresholded into proposals. Trained only with a classification loss, such models fire on the most discriminative moments rather than the whole action, so proposals are incomplete and high confidence does not imply accurate boundaries. Point-level supervision (P-TAL) [888, 889, 890] adds one labeled frame per instance, which tends to fall near the temporally most salient, roughly central moment, and uses it to realign confidence and infer complete boundaries from that single anchor.

Temporal Action Detection. TAD methods differ in how they generate and score temporal proposals. Anchor-based detection [891] transplants Faster R-CNN and SSD to the time axis: a dense set of multi-scale temporal anchors is classified and regressed, first in two stages (S-CNN [892]) and then in one (SSAD [893]); fixed anchors produce many redundant proposals and limit boundary precision, which anchor-free variants such as AFSD [894] address by regressing boundaries from a temporal center representation. Boundary-based detection [895, 884] is bottom-up: start and end probabilities are predicted over the whole timeline and boundary matching scores every candidate pairing, which gives flexible, precise boundaries; TSA-Net [896] adds multi-dilation temporal convolution for multi-scale context and BGCN [897] graph convolution over co-occurring proposals. Query-based detection [898, 899, 900, 901] follows DETR: a small set of learned queries attends to global video features in a Transformer encoder–decoder and decodes the action set in parallel, removing anchors, NMS, and hand-designed matching at the cost of heavier computation. Across the three, anchors give dense parallel scoring but coarse boundaries, boundary methods give precise boundaries but complex matching, and query methods give the simplest end-to-end pipeline; all three keep moving toward longer-range context, continuous temporal representations, and less frame-level annotation.

6.3.4.3 Datasets and Metrics

Table 6.22 lists the benchmarks by role. UCF101 [902] and HMDB51 [903] are trimmed-clip classification sets; AVA [904] adds spatio-temporal boxes on movie footage. THUMOS'14 [905], whose 20 classes carry dense temporal boundaries, and ActivityNet [906], with 200 classes over 20K untrimmed videos, are the standard TAL and TAD benchmarks, extended in scale and granularity by HACS Segment [907] and FineAction [908]. MultiTHUMOS [909], Charades [910], and TVSeries [911] supply dense, multi-label, overlapping actions in sports, homes, and television; Breakfast [912], 50Salads [913], MPII Cooking 2 [914], and COIN [915] cover procedural cooking and instructional activity for segmentation and step detection; and MEVA [916] (thousands of hours of surveillance) and MoVi [917] (motion-captured actors) sit at the large-scale and controlled extremes.

Each setting has its own metric. Action classification reports Top-1 accuracy, $\mathrm{Acc}_{\text{Top-1}} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\hat{y}_i = y_i)$, with Top-$k$ on large label sets; temporal boundaries are given, so localization is not assessed. Localization quality is measured by the temporal intersection over union of predicted and ground-truth intervals,

$$\mathrm{tIoU}(I_p, I_g) = \frac{|I_p \cap I_g|}{|I_p \cup I_g|}, \tag{6.38}$$

a prediction counting as correct above a threshold τ; TAL reports average precision at one or several thresholds. TAD scores detections $(I_p, \hat{y}, c)$ with confidence c by mean average precision over classes and thresholds,

$$\mathrm{mAP} = \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}}\frac{1}{|\{\tau_m\}|}\sum_{\tau_m}\mathrm{AP}_c(\tau_m), \tag{6.39}$$

Table 6.22: Benchmark datasets for action detection.

Dataset	Action Categories	Videos	Duration (hours)	Multi-label (Per-Frame)	Domain
UCF101 [902]	101	13,320	27	–	Sports, Daily Activities
HMDB51 [903]	51	6,849	–	–	Movies
AVA [904]	80	430	108	Yes	Movies
THUMOS'14 [905]	101 (20 w/ boundaries)	18,394	–	No	Sports
MultiTHUMOS [909]	65	413	–	Yes	Sports
Breakfast [912]	48	1,712	77	No	Cooking
50Salads [913]	17	50	5.3	No	Cooking
MPII cooking 2 [914]	59	273	27	No	Cooking
TVSeries [911]	30	27	–	Yes	TV Series
HACS Segment [907]	200	50,000	–	No	Daily Activities
ActivityNet-1.3 [906]	200	19,994	849	No	Daily Activities
Charades [910]	157	9,848	–	Yes	Daily Activities
MoVi [917]	20	–	9	–	Daily Activities
MEVA [916]	37	–	9,300	–	Surveillance
COIN [915]	180	11,827	142	No	Daily Activities
FineAction [908]	106	16,732	–	Yes	Daily Activities

Table 6.23: Representative methods under the two shared protocols. *Top*: action classification, Top-1 accuracy (%) on UCF101 (split 1, two-stream RGB+flow where applicable, with ImageNet or Kinetics pretraining as in each paper). *Bottom*: temporal action detection on THUMOS'14, mAP (%) at tIoU $= 0.5$ and averaged over tIoU $\in \{0.3, \dots, 0.7\}$, as reported by the cited papers and the comparisons in AFSD [894], CoLA [886], and DualDETR [901]; "weak" = video-level labels only. "–" = not reported.

Method	Family	Year	UCF101 Top-1↑
Two-Stream CNN [876]	RGB + optical-flow streams	2014	88.0
TSN [880]	Sparse segment sampling	2016	94.2
R(2+1)D [878]	Factorized 3D convolution	2018	97.3
I3D [239]	Inflated 3D convolution	2017	**98.0**

Method	Family	Year	mAP@0.5↑	Avg. mAP@[0.3:0.7]↑
S-CNN [892]	Anchor-based, two-stage	2016	19.0	–
SSN [883]	Segment-structure proposals	2017	29.8	–
BSN [884]	Boundary-sensitive proposals	2018	36.9	36.8
AutoLoc [885]	Weak, outer–inner contrast	2018	21.2	–
CoLA [886]	Weak, snippet contrastive	2021	32.2	–
AFSD [894]	Anchor-free, salient boundary	2021	55.5	52.0
DualDETR [901]	Query-based (dual DETR)	2024	**70.4**	**66.8**

which penalizes misclassification, loose boundaries, and duplicates together; THUMOS'14 uses $\tau \in \{0.3, \dots, 0.7\}$ and ActivityNet $\{0.5, \dots, 0.95\}$, and proposal methods also report average recall at a fixed proposal budget. Weakly supervised methods are often evaluated on ActivityNet-1.2 rather than 1.3, so their numbers are not interchangeable with fully supervised ones.

Table 6.23 shows how far the two settings have diverged. Trimmed-clip classification on UCF101 was effectively solved once Kinetics-pretrained 3D networks arrived, with I3D and R(2+1)D within a point of each other near 98%. Detection on THUMOS'14 tells a different story: mAP@0.5 moved from 19.0 (S-CNN) to 36.9 (BSN) over the anchor-to-boundary transition, jumped to 55.5 with anchor-free salient-boundary features (AFSD), and reached 70.4 with query-based decoding (DualDETR), while the best weakly supervised method (CoLA, 32.2) matches fully supervised accuracy of only a few years earlier, which quantifies what video-level labels still cost.

6.3.4.4 Challenges

Fully supervised TAD depends on densely annotated boundaries, which scale poorly to long, loosely structured real-world video; weakly and semi-supervised methods relax this with video-level labels or sparse temporal cues [918, 919], at the accuracy cost the table above makes visible. Actions are also still detected in isolation. Coupling TAD with structured temporal representations such as scene graphs or event schemas [778, 803] would let a system infer boundaries, temporal relations, and higher-level activity structure jointly, which matters most in instructional video, driving, and human–robot interaction. Finally, benchmarks and detectors remain visual and closed-set, whereas real action understanding draws on audio, language, gaze, and context and must handle actions outside a fixed vocabulary [920, 921, 922]; multimodal temporal detection, where audio or language sharpens boundaries and disambiguates semantics, and open-vocabulary formulations that loosen the classification-centric setup are the directions large multimodal video corpora now make feasible.

6.3.5 Mapping and SLAM

Mapping builds a spatial representation of an unknown environment from streaming sensor observations, together with an estimate of the agent's trajectory through it. Doing so from on-board sensors requires the agent to localize itself inside the partial map it is still building: the chicken-and-egg problem known as *Simultaneous Localization and Mapping (SLAM)*. SLAM resolves it by estimating trajectory and map jointly from the same sensor stream, and observations of previously visited places supply the external references that bound the drift of purely proprioceptive odometry (IMU, wheel encoders). The result is a globally consistent spatial model that supports localization, path planning, obstacle avoidance, and exploration. Figure 6.20 depicts the pipeline from sensors through front-end odometry and back-end optimization to the final map.

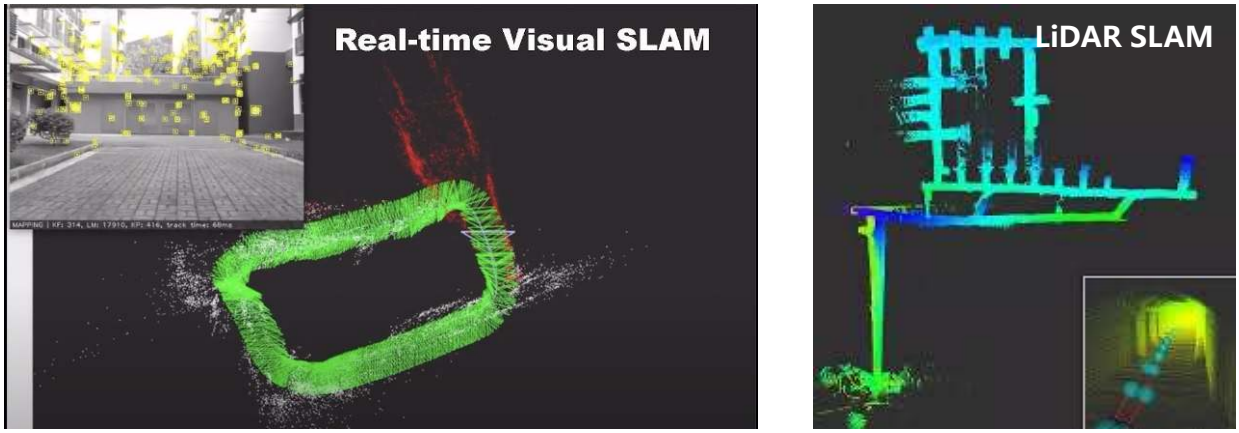


Figure 6.19: Example SLAM outputs: real-time visual SLAM (left), a keyframe trajectory over a sparse feature map with tracked image features in the inset, and LiDAR SLAM (right), a dense point-cloud map.

6.3.5.1 Task Definition

The input is a stream of exteroceptive observations (camera images, RGB-D frames, LiDAR, or radar), optionally fused with proprioceptive readings (IMU, wheel odometry, velocity and steering commands) and priors such as a coarse map, an initial pose, or GPS. The output is a map plus the estimated trajectory. Maps lie on an abstraction–detail spectrum: a *topological map* is a graph of distinctive places and their traversable connections, with no metric layout; a *sparse metric map* stores registered keypoints or landmarks and the camera-pose graph that observes them, enough for accurate localization but not for geometry; and a *dense metric map* reconstructs a complete surface or volume (point cloud, mesh, TSDF, or occupancy grid) for obstacle avoidance and 3D reconstruction. Two dense representations have become prominent in the

past three years: *neural implicit maps* that store geometry and appearance in network weights (an MLP in iMAP [923], a voxel-grid encoder in NICE-SLAM [924]) and *3D Gaussian Splatting (3DGS) maps* that represent the scene as explicit, differentiable 3D Gaussians [925, 926, 927].

6.3.5.2 Methodology

SLAM systems are classified by the exteroceptive sensor and by the density of the map they build. **LiDAR SLAM** measures range directly with a stable error model; KISS-ICP [928] and KISS-SLAM [929] show that a minimal pipeline delivers robust real-time 3D LiDAR odometry and mapping at city scale. **Visual SLAM** uses monocular, stereo, or RGB-D cameras, exploiting texture for place recognition, with PTAM [88] and the ORB-SLAM family [91, 736] as the foundational systems. **Multi-sensor SLAM** tightly fuses cameras with IMUs and/or LiDAR (VINS-Mono, OKVIS, FAST-LIO2), and **event-camera SLAM** uses asynchronous per-pixel brightness changes for high-speed and high-dynamic-range operation [930]. By map density, sparse keypoint-and-pose-graph systems (ORB-SLAM) sit next to semi-dense direct methods that track image gradients (LSD-SLAM [718], DSO [719]), dense geometric systems that fuse point clouds, meshes, or TSDFs (KinectFusion, ElasticFusion), neural implicit systems (iMAP [923], NICE-SLAM [924], Co-SLAM, NICER-SLAM), and 3DGS systems that render and optimize Gaussians jointly with poses (SplaTAM [925], MonoGS [927], SplatSLAM, MASt3R-SLAM [931]). Read historically, the field moved from feature-based geometry (PTAM, ORB-SLAM) to direct methods (LSD-SLAM, DSO), to tightly coupled multi-sensor systems (OKVIS, VINS-Mono, ORB-SLAM3), to learning-augmented pipelines (DROID-SLAM [95], SuperPoint and SuperGlue matching [539, 541]), to neural implicit and then 3DGS maps, and most recently to 3D geometry foundation models (DUSt3R [122], MASt3R [728], MASt3R-SLAM).

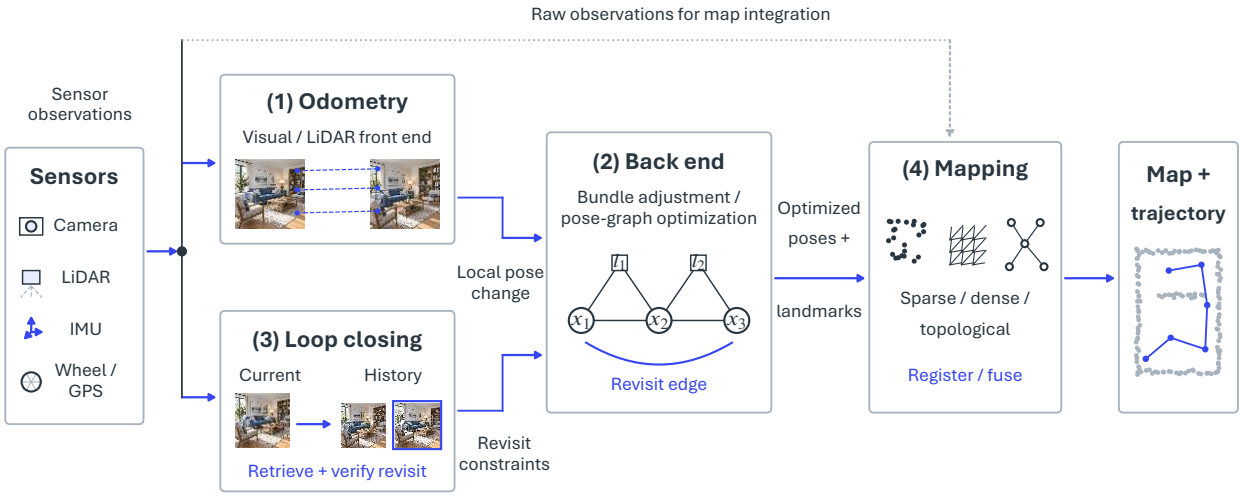


Figure 6.20: SLAM module data flow. Sensor observations feed odometry and place recognition; odometry constraints and verified revisit constraints enter back-end optimization. Mapping registers raw observations using optimized poses and landmarks, producing a map and trajectory. The blue edge in the pose graph is a loop-closure constraint between previously visited poses, not an execution-loop arrow. Map representations are alternatives.

Whatever the sensor or map, a SLAM system runs the four modules of Figure 6.20 in a loop. The **front end** is visual (or LiDAR) odometry, the incremental motion estimate between consecutive readings treated as a stand-alone task in §6.2.4; because each pose is estimated relative to the last, its errors accumulate as drift. **Back-end optimization** corrects that drift globally. Modern back ends pose the problem as factor-graph optimization: robot poses and landmarks are nodes, and measurements between them (visual reprojection, IMU pre-integration, GPS, loop-closure relative poses) are factors; solving for all nodes at once yields

a consistent trajectory and map, factor graphs absorb heterogeneous sensors naturally, and incremental solvers such as iSAM2 update the estimate in real time, with GTSAM, g^{2}o, and OKVIS as representative libraries. **Loop closing** supplies the back end with its strongest constraints by recognizing that the agent has returned to a previously visited place, comparing current observations against the history and, on a match, adding a constraint between two temporally distant poses that pulls the accumulated drift of the intervening trajectory back into alignment. **Mapping** finally places the sensor data in the global frame using the optimized trajectory, in whatever representation the application needs; for neural implicit and 3DGS maps it also minimizes a rendering loss (photometric or perceptual such as LPIPS) between observed images and views rendered from the current map, optimizing geometry, appearance, and poses jointly [925, 931].

Table 6.24: Mapping and SLAM benchmarks. Scale counts sequences unless noted.

Dataset	Task	Modality	Agent	Scene	Scale	Ground Truth
COLD [932]	VSLAM	mono + omni + odom	wheeled	lab	76	laser
TUM RGB-D [652]	VSLAM	RGB-D	handheld, wheeled	office, work shed	39	MoCap
KITTI [173]	VSLAM, VO	stereo + laser	car	suburban, highway	22	GPS/OXTS
ICL-NUIM [933]	VSLAM, VO	RGB-D	synthetic	home, office	8	simulator
EuRoC [733]	MSLAM	stereo + IMU	drone	work shed	11	Vicon MoCap/laser
SceneNet [934]	VSLAM	RGB-D	synthetic	home	15K	simulator
TUM VI [935]	MSLAM, VO	stereo + IMU	handheld	campus buildings	28	MoCap
Bonn RGB-D [936]	VSLAM	RGB-D	handheld	lab	24	MoCap
InteriorNet [937]	MSLAM	RGB-D + IMU	synthetic	home	15K	simulator
OpenLORIS-Scene [938]	MSLAM	RGB-D stereo + IMU + odom	wheeled	indoor	22	MoCap/laser
Replica [339]	dense	RGB-D	synthetic	indoor	18	simulator
ScanNet++ [939]	dense, sem	RGB-D	handheld	indoor	460	laser
TartanAir [666]	MSLAM, VO	stereo + IMU	synthetic	diverse	1M+ frames	simulator
ETH3D SLAM [940]	VSLAM, VO	stereo + global shutter	handheld	indoor/outdoor	–	LiDAR
DSEC [941]	event SLAM	stereo events	car	driving	53	LiDAR + GPS
CU-Multi [942]	collab. SLAM	RGB-D + IMU	multi-UAV	outdoor	–	MoCap/UWB
AirSwarm [943]	collab. SLAM	RGB-D + LiDAR	air-ground	outdoor	–	GPS

6.3.5.3 Datasets and Metrics

Table 6.24 groups the benchmarks by what they stress. The classical trajectory sets are TUM RGB-D [652] (handheld indoor RGB-D with motion-capture ground truth), KITTI [173] (stereo and LiDAR driving), and the synthetic ICL-NUIM [933] with exact ground truth, while COLD [932] targets topological localization under changing conditions. Visual-inertial systems are tested on EuRoC [733] (micro-aerial vehicle), TUM VI [935], and OpenLORIS-Scene [938], the last built for lifelong SLAM in changing indoor scenes, and Bonn RGB-D [936] isolates dynamic objects. Large synthetic corpora (SceneNet [934], InteriorNet [937], Replica [339], TartanAir [666]) supply photorealistic training and controlled evaluation, with Replica the standard testbed for neural implicit and 3DGS SLAM; ScanNet++ [939] and ETH3D SLAM [940] add laser-scanned geometry for dense reconstruction quality; and DSEC [941], CU-Multi [942], and AirSwarm [943] open the event-camera and multi-agent settings.

Evaluation covers trajectory, map, rendering, and system cost. Trajectory accuracy uses the absolute trajectory error and relative pose error defined in §6.2.4, after aligning the estimated trajectory to the ground truth by a rigid or, for monocular systems, similarity transform; RMSE is the usual summary, with mean, median, and standard deviation also reported. Dense maps are scored by *accuracy* (mean distance from reconstructed points to the nearest ground-truth point) and *completeness* (the reverse), summarized by an F-score at a distance threshold, with Chamfer distance common for neural and 3DGS maps. Neural implicit and 3DGS maps add novel-view-synthesis metrics (PSNR, SSIM, LPIPS) on held-out views of the same trajectory. Memory footprint, tracking and mapping frame rates, and end-to-end latency decide embedded deployment [929]. Mature evaluation tools such as EVO[2] and the RPG Trajectory Evaluation toolbox[3] implement these protocols.

6.3.5.4 Challenges

Deployment exposes SLAM to conditions its assumptions exclude: static scenes, stable illumination, abundant texture, and ample computation. Dynamic objects induce correspondences that drift with their own motion; illumination change, reflective and transparent surfaces, and repetitive texture create perceptual aliasing that confuses data association and place recognition alike; and resource-constrained platforms must run ever heavier front and back ends. Current work answers each directly. Moving features are detected and excluded through semantic segmentation, motion-consistency checks, or learned masks; learned matchers (SuperPoint [539], SuperGlue [541], LoFTR [542]) replace hand-crafted descriptors and tolerate larger viewpoint and illumination changes; neural implicit and 3DGS maps are optimized jointly with poses and yield photorealistic reconstructions usable for inspection, telepresence, and policy learning; and the two-view geometry priors of DUSt3R [122] and MASt3R [728] remove the fragile hand-engineered front end, already enabling real-time dense monocular SLAM at the robustness of MASt3R-SLAM [931].

Two broader shifts follow. Lifelong SLAM aims at months of continuous operation by combining relocalization, map update, and place re-identification against changing environments, and is now an evaluation track of its own; collaborative multi-agent SLAM fuses observations across robots for faster coverage and graceful degradation, increasingly together with event cameras and heterogeneous sensors for high-speed, low-light, and high-dynamic-range conditions. What remains open is concrete: 3DGS maps still do not scale to city-level memory and rendering budgets, learned components carry a sim-to-real gap, and verifying loop closures under strong perceptual aliasing is still the failure that breaks every back end.

6.4 Spatio-Temporal Perception



VERYTHING to this point has described a single instant. Spatio-temporal perception adds the missing axis, asking not only where entities are but how they move, when events happen, and which identities persist. Its three concerns rise from the pixel through the object to the event: scene flow recovers dense, per-point motion between frames; object tracking maintains the identity of an entity as it moves through space and time; and temporal localization fixes when an action or event occurs along a sequence. Together they give the otherwise static perception of the preceding sections a temporal thread, the continuity on which motion understanding and long-horizon perception depend.

6.4.1 Scene Flow Estimation

Scene flow is the dense three-dimensional motion of every visible surface point between two instants: optical flow lifted out of the image plane and joined to the change in depth. The image-based pipelines surveyed here take stereo video, monocular video, or a multi-view sequence and produce per-pixel 3D motion; some

[2] https://github.com/MichaelGrupp/evo
[3] https://github.com/uzh-rpg/rpg_trajectory_evaluation

refinement modules additionally consume estimated or measured depth. Dynamic scene reconstruction [299], motion-aware rendering [188], physical interaction modeling [680], and egocentric world understanding [681] all consume it.

6.4.1.1 Task Definition

Given consecutive frames I_t and I_{t+1} (optionally with a stereo partner or several views, and with intrinsics and relative poses when known), scene flow estimation returns a field $\mathbf{v}(\mathbf{x}) \in \mathbb{R}^3$ that displaces the surface point imaged at pixel $\mathbf{x}$ in frame t to its position at $t+1$, in camera or world coordinates. In the stereo setting this is equivalently a triple $(d_t, d_{t+1}, \mathbf{f})$ of disparity in both frames plus optical flow, which is how the KITTI benchmark represents and scores it. In image-only pipelines, structure and motion are inferred from appearance, parallax, and temporal correspondence. Geometry-aware refinement modules may instead take depth estimated upstream or measured by RGB-D sensors; point-cloud-only methods are outside the scope here.

6.4.1.2 Methodology

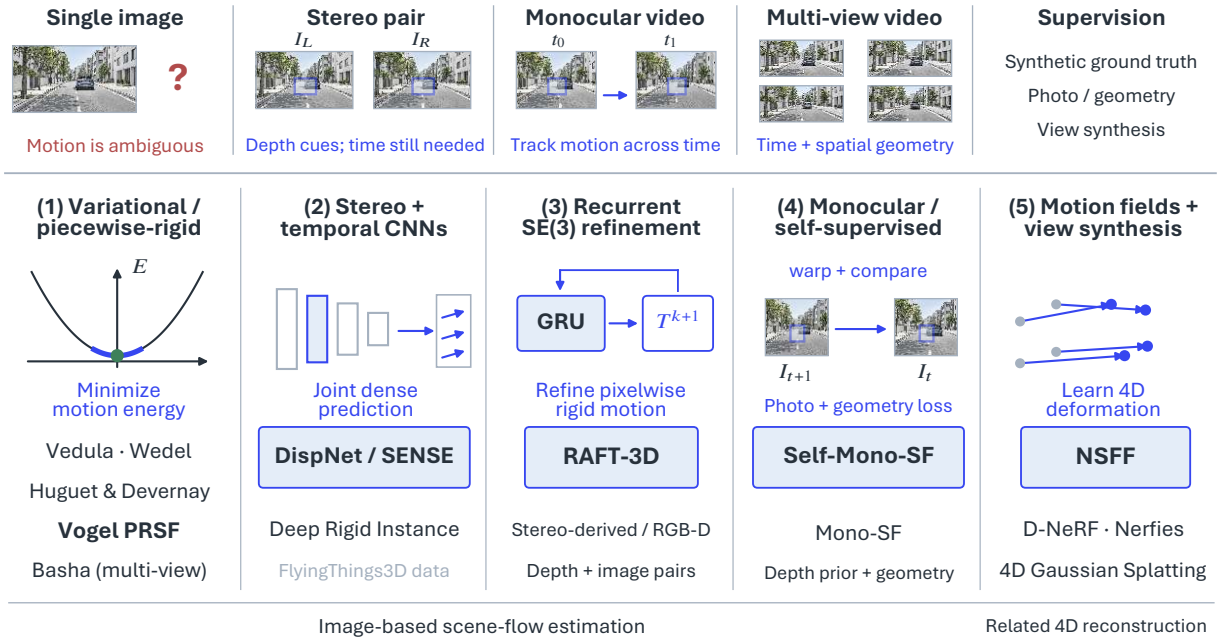


Figure 6.21: Image- and video-based scene flow. Top row: observation settings and supervision signals. Bottom row: (1) variational / piecewise-rigid energy minimization, (2) stereo and temporal CNNs, (3) recurrent SE(3) refinement with image and depth pairs, (4) monocular estimation using photometric and geometric constraints, and (5) related motion-field approaches to dynamic view synthesis. The columns illustrate distinct mechanisms rather than successive stages of one pipeline.

Methods differ along two axes: what they observe (stereo, the classical and still dominant setting; monocular video, the self-supervised frontier; multi-view rigs) and what supervises them (synthetic ground truth, photometric and geometric consistency, or view synthesis). Figure 6.21 groups them into five families along these axes.

Variational and energy-based methods. Vedula et al. [944] introduced the term and the problem: recover 3D structure and 3D motion jointly from a calibrated camera array. Huguet and Devernay [945] brought it to a single stereo pair, solving disparity and flow jointly in one variational energy; Wedel et al. [946] decoupled the two into a stereo stage and a flow stage, which made real-time estimation feasible; and Basha et al.

[947] returned to multi-view input under a view-centered energy. The strongest classical formulation is the piecewise-rigid scene model of Vogel et al. [948] (PRSF/PRSM), which segments the scene into rigidly moving planar patches and was the leading entry on KITTI Scene Flow at the benchmark's release.

End-to-end stereo and temporal CNNs. Mayer et al. [619] recast disparity and flow as supervised regression (DispNet, FlowNet) and released FlyingThings3D, Monkaa, and Driving, the synthetic corpora that made this training regime possible. SENSE [949] shares one encoder across disparity, optical flow, occlusion, and semantic heads; DRISF [950] keeps the piecewise-rigid prior of Vogel *et al*. but implements it as an instance-level decoder over learned features. Both surpassed the variational baselines on KITTI Scene Flow (Table 6.26), shifting the bottleneck from optimization to supervised data.

Recurrent geometric refinement. RAFT [255] introduced all-pairs correlation volumes with GRU-based iterative updates for 2D optical flow; RAFT-3D [951] uses image and depth pairs to iteratively refine a per-pixel SE(3) field with rigid-motion embeddings. Depth can be obtained from stereo or RGB-D input; it is an input to the refinement rather than necessarily inferred jointly from appearance. Its scene-flow results are included in Table 6.26.

Monocular and self-supervised methods. Mono-SF [952] was the first monocular entry on KITTI Scene Flow, resolving scale with a learned single-view depth prior. Self-Mono-SF [953] removes ground truth altogether, training depth, ego-motion, and 3D-motion heads on raw video through photometric and geometric consistency losses; the monocular depth priors of Section 6.2.1 and the ego-motion machinery of Section 6.2.4 are its two anchors. The same test-time-optimization idea applied to LiDAR point clouds (Neural Scene Flow Prior [954]) lies outside the image-based scope of this subsection.

Implicit-field and view-synthesis-coupled methods. A parallel line never targets scene flow as output but obtains a dense 4D motion field as a byproduct of dynamic view synthesis. Neural Scene Flow Fields [188] trains a dynamic NeRF together with an explicit scene-flow MLP; D-NeRF [187] represents motion as a deformation field into a canonical frame; Nerfies [299] extends this to casually captured monocular video; and 4D Gaussian Splatting [191] attaches time-parameterized motion to explicit Gaussian primitives. These methods are evaluated by rendering quality rather than by motion error, a split returned to under §Challenges.

6.4.1.3 Datasets and Metrics

Table 6.25 lists the image-based benchmarks. Synthetic sets provide dense, exact ground truth at scale (FlyingThings3D as the standard pretraining corpus; Sintel and Spring for film-grade motion, with Spring at HD resolution and super-resolved ground truth; TartanAir for environment diversity), while KITTI Scene Flow 2015 [955], built on the KITTI suite [173], is the only real-world benchmark and carries sparse LiDAR-derived labels on 200 training and 200 test scenes. LiDAR-based scene-flow splits of nuScenes [445], Waymo Open [446], and Argoverse 2 [661], and the event-camera flow of DSEC [941], serve the point-cloud and event communities and are not image-based benchmarks in the sense used here.

KITTI scores scene flow in image space. A pixel's disparity in frame t (D1), its disparity in frame $t{+}1$ (D2), and its optical flow (Fl) are each counted as outliers when the error exceeds both 3 px and 5% of the ground-truth value, and a scene-flow outlier (SF) is any pixel that fails at least one of the three; the benchmark reports outlier percentages over all pixels and separately over background and foreground. Point-level evaluation on synthetic data instead uses the 3D end-point error

$$\mathrm{EPE}_{3D} = \frac{1}{N}\sum_{i=1}^{N} \|\hat{\mathbf{v}}_i - \mathbf{v}_i\|_2, \tag{6.40}$$

Table 6.25: Image-based scene-flow benchmarks. Counts are indicative of the commonly used releases.

Dataset	Year	Type	Input	Scale	Ground truth
KITTI Scene Flow 2015 [955]	2015	Real driving	Stereo video	200 train / 200 test scenes	Sparse disparity (t, $t{+}1$) + flow from LiDAR and CAD fits
FlyingThings3D / Monkaa / Driving [619]	2016	Synthetic	Stereo video	∼25K / 8.6K / 4.4K frames	Dense disparity, flow, disparity change
MPI Sintel [956]	2012	Synthetic film	Stereo video	1,041 train frames, 23 seq.	Dense flow, depth, disparity
Spring [957]	2023	Synthetic film	Stereo video	∼6K frames, 47 seq., 1920×1080	Dense stereo, flow, scene flow (super-resolved)
TartanAir [666]	2020	Synthetic	Stereo video	>1M frames, 30 environments	Dense depth, flow, pose; indoor and outdoor, weather

Table 6.26: KITTI Scene Flow 2015 test set, outlier percentages (lower is better) over all pixels. Stereo rows from the RAFT-3D comparison [951]; monocular rows from the Mono-SF [952] and Self-Mono-SF [953] papers ("ft" = fine-tuned on KITTI). D1 / D2: disparity in frames t / $t{+}1$; Fl: optical flow; SF: scene flow.

Method	Family	D1-all	D2-all	Fl-all	SF-all
• *Stereo*					
OSF [955]	Energy-based, rigid objects	5.79	7.77	7.83	10.23
PRSM [948]	Energy-based, piecewise rigid	4.27	6.79	6.68	8.97
SENSE [949]	Shared-encoder CNN	2.22	5.89	7.64	9.55
DRISF [950]	CNN + rigid instances	2.55	4.04	4.73	6.31
RAFT-3D [951]	Recurrent SE(3) refinement	1.81	3.67	4.29	5.77
• *Monocular*					
Mono-SF [952]	Learned depth prior	16.32	19.59	12.77	23.08
Self-Mono-SF [953]	Self-supervised	34.02	36.34	23.54	49.54
Self-Mono-SF-ft [953]	Self-supervised, ft	22.16	25.24	15.91	33.88

with $\hat{\mathbf{v}}_i$ and $\mathbf{v}_i$ the predicted and true motion of point i, together with accuracy ratios at fixed thresholds (the fraction of points with error below 0.05 m or 5%, and below 0.1 m or 10%) and an outlier ratio (error above 0.3 m or 10%), conventions inherited from point-cloud scene flow.

Table 6.26 collects KITTI Scene Flow 2015 test results for the explicit-output families. RAFT-3D more than halves the SF-all error of the best classical method (PRSM), and the foreground column, where rigidly moving vehicles meet non-rigid motion, is where the learned methods gain most. The monocular rows are a different regime: Mono-SF, with a learned depth prior, sits at 23% SF-all, and Self-Mono-SF without any ground truth at 49.5% (33.9% after fine-tuning), the price of giving up stereo. The implicit-field family has no row because it reports rendering quality (PSNR, LPIPS) on its own scene sets rather than motion error.

6.4.1.4 Challenges

The supervision landscape is lopsided. Every dense-ground-truth benchmark in Table 6.25 is synthetic and stereo, so monocular methods are scored only against KITTI's sparse labels, and models pretrained on FlyingThings3D or Sintel still lose accuracy on real driving footage after fine-tuning, most visibly on foreground motion where synthetic random-object trajectories differ from vehicle kinematics. All current benchmarks also emphasize rigid or random-object motion; dense ground truth for non-rigid humans, hands, and animals does not exist at scale, which leaves the family of implicit-field methods that handle such motion best without a quantitative yardstick.

That yardstick problem is the second challenge. Explicit-output methods [948, 949, 951] are scored by KITTI's outlier rates; implicit-field methods [188, 187, 191] by PSNR and LPIPS of rendered views. No

protocol evaluates both on the same input, so the two paradigms are compared only qualitatively, and whether a motion field is good depends on whether scene flow is the end (the KITTI leaderboard) or a means (reconstruction, novel-view synthesis, physical reasoning). The cost profiles diverge as well: RAFT-3D needs several GRU iterations per frame, variational methods minutes per frame, and implicit fields minutes to hours per scene.

Foundation priors are reshaping the monocular line. Depth Anything [140] and DINOv2 [958] supply the per-pixel depth and semantic features that Self-Mono-SF had to learn from scratch, and SAM 2 [482] segmentation together with the object-pose estimators of Section 6.2.5 makes an object-centric formulation (a per-object SE(3) trajectory plus residual non-rigid motion) practical, a bridge to the tracking problem of Section 6.4.2. As implicit-field scene flow merges into 4D reconstruction, with 4D Gaussian Splatting [191] the early example, evaluation is likely to follow the downstream task rather than the KITTI leaderboard.

6.4.2 Object Tracking

Tracking localizes entities in every frame and associates the detections across frames so that each entity keeps one identity through the sequence. Where detection and segmentation describe a single instant, tracking yields trajectories, the representation that autonomous driving [959, 960], surveillance [961], AR/VR [220], and robotics [962] build on.

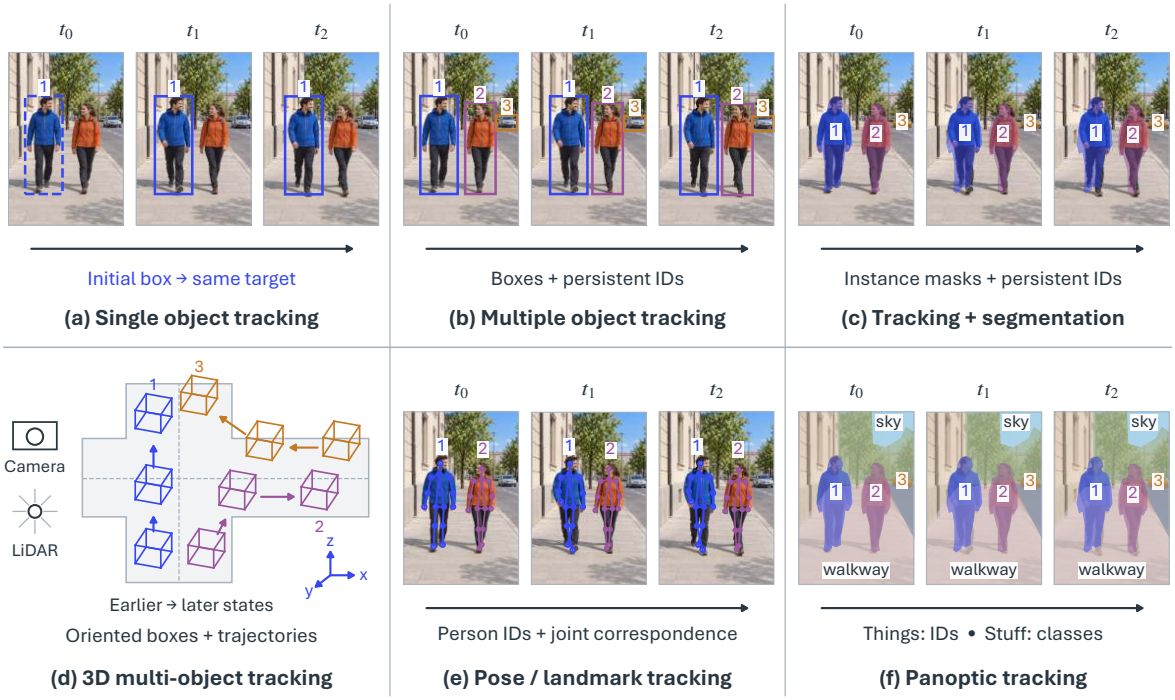


Figure 6.22: Object-tracking subtasks: (a) single object tracking (SOT) from an initial box, (b) multiple object tracking (MOT) with identity-consistent boxes, (c) multiple object tracking and segmentation (MOTS) with per-frame masks, (d) 3D multi-object tracking (3D-MOT) of oriented boxes from LiDAR or camera input, (e) pose and landmark tracking, and (f) panoptic tracking with semantic classes for stuff and persistent instance IDs for things. Repeated IDs denote the same entity across frames; arrows indicate increasing time. Scene images and overlaid states are illustrative.

6.4.2.1 Task Definition

Given a frame sequence $\{I_t\}_{t=1}^{T}$ (a video, or RGB-D, LiDAR, or multi-view streams), a tracker returns a set of trajectories $\{\mathcal{T}_k\}$, each $\mathcal{T}_k = \{(t, b_t^k)\}$ a time-indexed series of states b_t^k with one identity k held fixed over the

sequence. The state is a bounding box in the basic setting and a segmentation mask, a 3D box, or a set of keypoints in the variants.

The variants, illustrated in Figure 6.22, differ in how the targets are specified and what state is tracked. Single object tracking (SOT) [963, 964] follows one target given its first-frame box; multiple object tracking (MOT) [965, 966, 711] tracks every instance of given categories, typically pedestrians and vehicles, under identity constraints; MOTS [967] and video object segmentation replace boxes with masks; 3D-MOT [968, 429, 445, 446] outputs world-coordinate 3D boxes from LiDAR, RGB-D, or multi-view input; pose tracking [549, 969, 970] and landmark tracking [971] follow skeletal joints or facial keypoints; and panoptic tracking combines pixel-level semantic classes for stuff with temporally consistent instance identities for things. Category-specific trackers for pedestrians [972], vehicles [973], or animals [974] specialize any of these.

6.4.2.2 Methodology

Figure 6.23 organizes the methods into four families around image-plane box and mask tracking (SOT, MOT, MOTS, VOS). Pose, landmark, and panoptic tracking reuse the same propagate-or-associate machinery with a different state representation and are covered at the benchmark level in Table 6.27.

Propagation-based methods. Because a target moves little between consecutive frames, its previous state is a strong prior for the current one, and the oldest tracking paradigm exploits this by predicting the next state from the current one and correcting it against the new frame. Correlation filters realize this in the frequency domain: MOSSE [975] learns a minimum-output-sum-of-squared-error filter from a single frame and updates it online at hundreds of frames per second; KCF [976] and MKCF [977] add kernelization and multi-scale search; ECO [978] adds factorized convolution operators, a compact generative sample model, and conservative updates. These trackers are fast enough for embedded use but drift under large appearance change and clutter. MDNet [979] answered with online discriminative learning, a shared CNN with per-sequence domain branches that separate target from background, at a high per-frame cost. Siamese trackers moved the representation learning offline: SiamFC [980] correlates a template embedding with the search-region embedding in one forward pass, and SiamRPN [981], ATOM [982], and DiMP [983] add region-proposal regression, IoU-prediction refinement, and a discriminatively learned prediction head. Transformer trackers then replaced correlation with attention: TransT [984] fuses template and search features with self- and cross-attention, STARK [985] predicts the box with an encoder–decoder that also carries temporal state, and OSTrack [986] joins feature extraction and relation modeling in a single stream over concatenated template and search tokens, the current baseline design.

Mask propagation applies the same idea at the pixel level for video object segmentation [987, 988], with memory of past frames replacing a single template. XMem [989] keeps multi-scale spatiotemporal memories that remain stable over videos of more than 10,000 frames; AOT [990] and DeAOT [991] encode object identities as tokens so that many objects propagate in one pass; and SAM 2 [482] adds a streaming memory to the Segment Anything Model [138], giving promptable tracking that transfers across VOS benchmarks.

Detect-then-associate methods. The MOT standard pipeline detects objects in every frame and then solves an assignment problem between detections and existing tracks. SORT [992] predicts track positions with a Kalman filter [164] and matches by IoU with the Hungarian algorithm [993]; DeepSORT [994] adds an appearance embedding so that tracks survive occlusion. Track-RCNN [967], JDE [995], and FairMOT [996] train the re-identification embedding jointly with the detector; association remains a separate step, which keeps them in this family. All of these discard low-confidence detections, and since occluded targets are exactly those with low scores, trajectories fragment where they are hardest to keep. ByteTrack [997] associates nearly every detection and uses motion similarity to the existing tracks to reject the false positives among the low-score boxes, a change in association alone that raised MOT17 HOTA by several points (Table 6.28). In 3D, the same pipeline reappears: AB3DMOT [968] runs a 3D Kalman filter with Hungarian matching on LiDAR

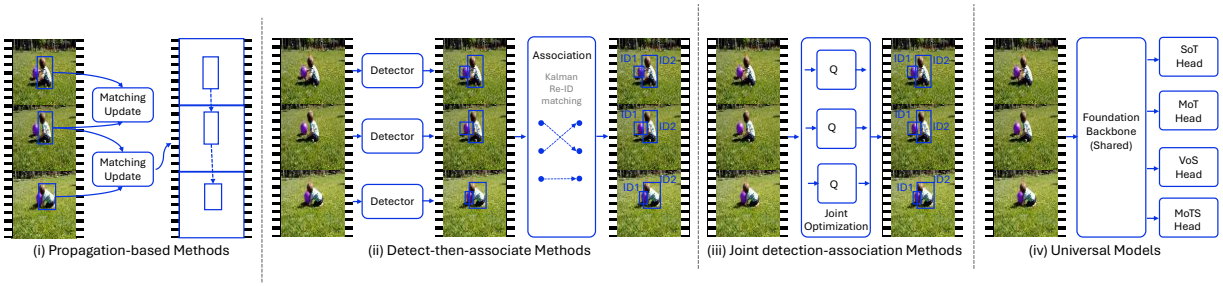


Figure 6.23: Four method families for object tracking: (i) propagation-based, (ii) detect-then-associate, (iii) joint detection–association, and (iv) universal models with a shared backbone and task heads for SOT, MOT, VOS, and MOTS.

detections, and CenterPoint [429] tracks 3D object centers through predicted velocities, with nuScenes [445] and Waymo [446] standardizing the AMOTA/AMOTP metrics.

Joint detection–association methods. End-to-end trackers fold association into the detector. CenterTrack [998] represents objects as center points and regresses their offset to the previous frame, so that association is a local lookup; TrackFormer [999] introduces tracking-by-attention, in which track queries carried from the previous frame and new object queries are decoded jointly by a Transformer; MOTR [1000] extends DETR [136] with track queries that persist across the whole sequence and are supervised with a sequence-level loss, which is what gives it its advantage on DanceTrack [1001], where appearance is uninformative and motion is non-linear. The price is training cost and data: these models need long annotated sequences and remain behind detect-then-associate trackers on MOT17 (Table 6.28).

Universal models. A single model for SOT, MOT, VOS, and MOTS is the most recent direction. UniTrack [1002] showed that one self-supervised appearance representation serves propagation and association tasks alike; Unicorn [1003] trains one network end to end for all four tasks; UTT [1004] unifies SOT and MOT in a Transformer; UniVS [1005] casts every video-segmentation task as prompt-driven segmentation; and OneTracker [1006] adapts a foundation tracker to RGB, depth, thermal, and event inputs with light adapters. In practice, SAM [138] combined with a memory tracker (XMem, DeAOT) already provides promptable multi-object segmentation and tracking from points, boxes, scribbles, or text, and benchmarks such as UASTrack [1007] and UNTrack [1008] extend the unified setting to RGB-X, multi-spectral, and UAV footage. Language enters as a specification signal as well: LaSOT [963] pairs every sequence with a sentence, TNL2K [1009] is built for text-initialized tracking, and LaMOT [1010] brings language-guided specification to the multi-object setting.

6.4.2.3 Datasets and Metrics

Table 6.27 groups the benchmarks by what is tracked. The SOT line runs from short, category-agnostic clips (OTB100, TC128, the VOT challenges) to large, long-term, category-diverse training sets (TrackingNet, LaSOT, GOT-10k, whose test classes are disjoint from training) and to condition-specific tests for aerial footage (UAV123), high frame rate (NFS), and language initialization (TNL2K). MOT benchmarks are pedestrian-only in MOT16/17 and MOT20 and multi-class driving footage in BDD100K; the 3D-MOT counterparts are the nuScenes and Waymo tracking tasks. DAVIS and YouTube-VOS supply mask-level ground truth, the latter with categories unseen at training time; the RGB-D sets add depth and have grown from tens of sequences (PTB, STC) to over a thousand (RGBD1K); and pose tracking moved from single-person upper-body clips (VideoPose2.0, Poses in the Wild, ICDPose) to multi-person full-body sequences with person identities (PoseTrack18, CMU Panoptic).

Table 6.27: Datasets for object tracking with bounding boxes (Bbox), segmentation masks (Mask), or human poses. "Frames / Boxes" gives annotated frames, or annotated boxes for multi-object datasets (MOT16/17, TrackingNet, YouTube-BB). For pose datasets, the annotation column gives the setting and the number of body landmarks.

Name	Videos	Frames / Boxes	Categories	Year	Attributes	Annotations
• *Single Object Tracking*						
VIVID [1011]	9	16,274	-	2005	-	Bbox
OTB100 [1012]	98	58,610	-	2013	11	Bbox
ALOV300 [1013]	314	89,364	64	2013	-	Bbox
TC128 [1014]	129	55,652	-	2015	11	Bbox
VOT16 [1015]	60	21,455	-	2016	-	Bbox
DAVIS-16 [244]	50	3,455	4	2016	15	Mask
UAV20L [1016]	20	58,670	-	2016	12	Bbox
UAV123 [1016]	123	113,476	9	2016	12	Bbox
NUS-PRO [1017]	365	135,305	8	2016	-	Bbox
VOT17 [1018]	60	21,356	24	2017	-	Bbox
YouTube-BB [1019]	380,000	5.6M	23	2017	-	Bbox
NFS [1020]	100	383,000	17	2017	9	Bbox
TrackingNet [964]	30,132	14.2M	27	2018	15	Bbox
LaSOT [963]	1,400	3.52M	70	2019	14	Bbox
TNL2K [1009]	2,000	1.24M	-	2021	17	Bbox
GOT-10k [1021]	10,000	1.5M	563	2021	6	Bbox
• *Multiple Object Tracking*						
MOT16 [965]	14	292,733	1	2016	-	Bbox
MOT17 [965]	42	564,228	1	2017	-	Bbox
BDD100K [711]	1,600	318K	8	2020	-	Bbox
• *Video Object Segmentation*						
DAVIS-17 [987]	150	10,459	-	2017	-	Mask
YouTube-VOS [988]	4,453	-	94	2018	-	Mask
• *RGB-D Object Tracking*						
PTB [1022]	100	21,542	-	2013	5	Bbox
STC [1023]	36	9,009	-	2017	12	Bbox
CDTB [1024]	80	101,956	-	2019	13	Bbox
DepthTrack [1025]	200	294,591	-	2021	15	Bbox
ARKitTrack [1026]	300	229.7K	144	2023	16	Mask
RGBD1K [1027]	1,050	717,900	-	2023	15	Bbox
• *Human Pose Tracking*						
VideoPose2.0 [1028]	44	1,286	-	2011	-	Single-person (-)
Poses in the Wild [1029]	30	830	-	2014	-	Single-person (7)
ICDPose [970]	60	2,700	-	2016	-	Single-person (-)
CMU Panoptic [1030]	65	297K	-	2017	-	Multi-person (-)
Multi-Person PoseTrack [969]	60	-	-	2017	-	Multi-person (14)
PoseTrack18 [549]	1,138	66,374	-	2018	-	Multi-person (15)

Each line has its own metrics. SOT uses the success plot: with $\mathrm{SR}(\tau)$ the fraction of frames whose predicted-box IoU exceeds τ, the headline number is its area under the curve,

$$\mathrm{AUC} = \int_0^1 \mathrm{SR}(\tau)\, d\tau, \tag{6.41}$$

reported with normalized precision (the fraction of frames whose center error, divided by the target size, is below a threshold) on LaSOT and TrackingNet, and with average overlap (mean IoU over frames) and $\mathrm{SR}_{0.5}$, $\mathrm{SR}_{0.75}$ on GOT-10k; the VOT challenges use expected average overlap (EAO), which folds accuracy

Table 6.28: Representative trackers on the standard benchmarks of each line. SOT: AUC on LaSOT and TrackingNet, AO on GOT-10k (from the OSTrack comparison [986]). MOT: MOT17 test, private detections (from the MOTR [1000] and ByteTrack [997] comparisons). VOS: $\mathcal{J}\&\mathcal{F}$ on DAVIS-17 val and $\mathcal{G}$ on YouTube-VOS 2019 val (from the SAM 2 comparison [482]).

• *Single object tracking*				
Method	**Design**	**LaSOT AUC**	**TrackingNet AUC**	**GOT-10k AO**
DiMP [983]	Discriminative prediction head	56.9	74.0	61.1
TransT [984]	Transformer feature fusion	64.9	81.4	67.1
STARK [985]	Encoder–decoder, temporal state	67.1	82.0	68.8
OSTrack-384 [986]	One-stream Transformer	71.1	83.9	73.7
• *Multiple object tracking (MOT17 test)*				
Method	**Family**	**HOTA**	**MOTA**	**IDF1**
CenterTrack [998]	Joint, center offsets	52.2	67.8	64.7
TrackFormer [999]	Joint, tracking-by-attention	–	74.1	68.0
MOTR [1000]	Joint, persistent queries	57.8	73.4	68.6
FairMOT [996]	Detect + joint Re-ID	59.3	73.7	72.3
ByteTrack [997]	Detect + low-score association	63.1	80.3	77.3
• *Video object segmentation*				
Method	**Design**	**DAVIS-17 $\mathcal{J}\&\mathcal{F}$**	**YT-VOS 2019 $\mathcal{G}$**	
XMem [989]	Multi-store memory	86.0	85.6	
DeAOT [991]	Identity tokens, decoupled propagation	86.2	86.1	
SAM 2 [482]	Promptable, streaming memory	90.7	89.3	

and failure rate into one expectation [1015]. MOT scores detection and identity jointly. MOTA [1031] charges false negatives, false positives, and identity switches against the number of ground-truth objects,

$$\mathrm{MOTA} = 1 - \frac{\mathrm{FN} + \mathrm{FP} + \mathrm{IDS}}{\mathrm{GT}}, \tag{6.42}$$

and so is dominated by detection; IDF1 is the F1 score of identity-level true positives, $\mathrm{IDF1} = 2\,\mathrm{IDTP}/(2\,\mathrm{IDTP} + \mathrm{IDFP} + \mathrm{IDFN})$, and emphasizes identity preservation; HOTA [1032], $\sqrt{\mathrm{DetA} \cdot \mathrm{AssA}}$ averaged over localization thresholds, balances the two and has become the primary MOT metric. VOS reports the Jaccard index $\mathcal{J} = |M \cap G|/|M \cup G|$ of predicted and true masks and the contour F-measure $\mathcal{F}$ from boundary precision and recall [244], averaged as $\mathcal{J}\&\mathcal{F}$.

Table 6.28 reports the methods of §Methodology on the benchmarks each line treats as standard. Within SOT the step from a discriminative head over correlation features (DiMP) to attention-based fusion (TransT, STARK) is worth about ten points of LaSOT AUC, and the one-stream design of OSTrack adds six more over TransT. On MOT17 the detect-then-associate trackers lead, and ByteTrack's gain over FairMOT comes from association alone; the end-to-end trackers (CenterTrack, TrackFormer, MOTR) trail on MOT17 but MOTR leads ByteTrack by 6.5 HOTA on DanceTrack [1000], where appearance cues are weak. VOS accuracy on DAVIS-17 has moved from the mid-80s (XMem, DeAOT) to above 90 $\mathcal{J}\&\mathcal{F}$ with SAM 2, which is also the first method trained on a corpus far larger than the benchmarks themselves.

6.4.2.4 Challenges

Identity under occlusion remains the central failure. Motion models and appearance embeddings bridge short occlusions, but long occlusions and dense crowds still produce identity switches and fragmented tracks [997, 1001], and memory-based propagation drifts or forgets over long videos even when each step is accurate [989]. HOTA and IDF1 measure the outcome but not robustness to occlusion length or the calibration of track confidence, so the metrics reward the easy part of the problem.

Trackers are also tuned to a few curated benchmarks and transfer poorly to UAV footage, thermal and event imagery, adverse weather, and multi-camera surveillance; foundation-model trackers such as OneTracker [1006] and the RGB-X benchmarks UASTrack [1007] and UNTrack [1008] address this directly but inherit the biases of their pretrained backbones. Long streams add memory and compute limits on top of drift.

The accuracy of Transformer trackers comes at a cost that keeps them off embedded hardware, while the lightweight correlation-filter line gives up robustness; and the unified models that would amortize that cost across tasks meet negative transfer between objectives with different demands, mask precision against identity association in particular [1003]. Language and segmentation priors widen what a tracker can be asked to do, at the price of prompt sensitivity that current benchmarks do not measure.

6.4.3 Temporal Localization

Temporal localization answers *when* something happens in a long, untrimmed video. The condition can be a category from a closed set, which gives temporal action localization (TAL) as treated in Section 6.3.4, or a free-form sentence, which gives temporal sentence grounding (TSG): return the segment that best matches "the person opens the microwave". This subsection sets the two side by side, with the weight on the text-conditioned problem, which turns localization from exhaustive detection into retrieval and is what video search and interactive navigation [1033, 1034, 1035] rely on.

6.4.3.1 Task Definition

Temporal Action Localization (TAL) takes an untrimmed video $V = \{x_t\}_{t=1}^{T}$, where x_t may be a visual, audio, or multimodal feature, and detects *all* instances of K predefined action classes, returning $\mathcal{A} = \{a_i\}_{i=1}^{N}$ with $a_i = (t_i^{\text{start}}, t_i^{\text{end}}, c_i, s_i)$: start and end times, a class label $c_i \in \{1, \dots, K\}$, and a confidence s_i (the (t_s, t_e, c) tuples of Section 6.3.4).

Temporal Sentence Grounding (TSG), also called natural language video localization or video moment retrieval, pairs V with a sentence q and returns a ranked list of moments $\mathcal{M}(q, V) = \{m_j\}_{j=1}^{M}$, $m_j = (\tau_j^{\text{start}}, \tau_j^{\text{end}}, r_j)$, with boundaries and a relevance score r_j; usually one segment is the answer. TAL is thus an exhaustive detection task over a closed label set, TSG a targeted retrieval task guided by open-ended language.

Both are studied under several settings, which Figure 6.25 lays side by side. Offline methods see the whole video; online methods infer causally over the streamed prefix and are judged on low-latency detection of action starts [1036, 1037]. Supervision ranges from frame-accurate boundaries to video-level labels only (weakly supervised TAL via multiple-instance learning, Section 6.3.4; weakly supervised TSG by aligning the two modalities during training [1038]), with self-supervised video pretraining [1039, 1040] and open-vocabulary or zero-shot variants [1041] reducing the labeled data needed. TSG additionally varies in scope, from a single video [1034, 1033] to corpus-level retrieval that first finds the video and then the moment (VCMR) [1042], and in output, with QVHighlights [1043] asking for a saliency ranking of highlight clips alongside the grounded moment.



Figure 6.24: Temporal localization methods. Top row (TAL, category-conditioned): (i) window-based, (ii) two-stage proposal-based, (iii) one-stage anchor-free, and (iv) Transformer set prediction. Bottom row (TSG, text-conditioned): (a) cross-modal matching, (b) 2D temporal-map reasoning with a proposal-graph alternative, and (c) Transformer set prediction. Hatched map cells are invalid start–end pairs; graph nodes represent candidate moments. Coarse-to-fine corpus retrieval is shown separately as a scope extension, not a mandatory stage of set prediction. Scene frames and scores are illustrative.

6.4.3.2 Methodology

Figure 6.24 traces both tasks along the same path from hand-designed pipelines to end-to-end Transformers.

The TAL row is the detection lineage of Section 6.3.4: sliding windows over hand-crafted features such as improved dense trajectories [1044], then two-stage proposal-plus-classification systems (R-C3D [1045], SSN [883], TAL-Net [1046]) with dedicated boundary-sensitive proposal generators (BSN [884], BMN [1047], G-TAD [1048]), then anchor-free dense detectors and query-based set prediction. Three developments go beyond that treatment. Dense single-stage detectors were scaled by ActionFormer [1049], a multi-scale Transformer encoder over the feature pyramid, and TriDet [1050], which replaces self-attention with a convolutional scalable-granularity-perception module and a trident boundary head; both still regress boundaries at every position and rely on non-maximum suppression. DETR-style set prediction removes anchors and suppression: RTD-Net [1051] applies a relaxed Transformer decoder to proposal generation and TadTR [1052] decodes the full action set end to end [1053, 1054]. And masked video pretraining [1039, 1040] now supplies the backbones that these heads are fine-tuned on.

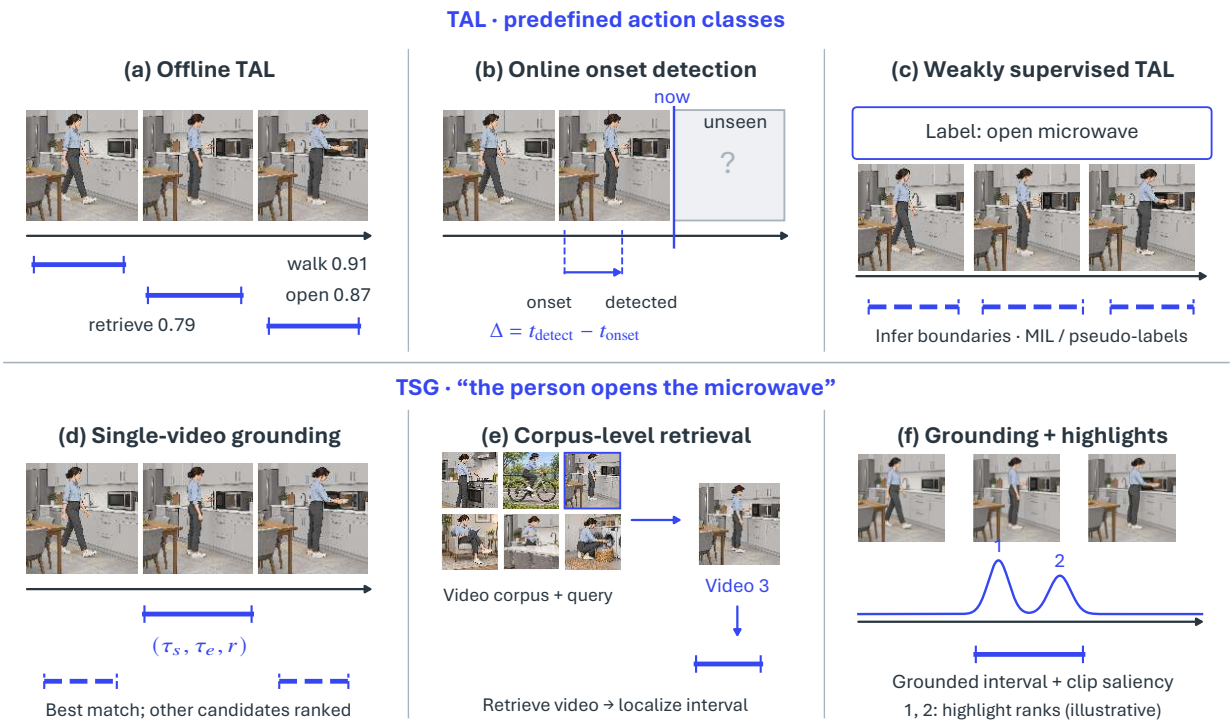


Figure 6.25: Six settings of temporal localization. Top row (TAL): (a) offline detection of K classes; (b) online detection of action starts [1036, 1037]; (c) weakly supervised training from video-level labels. Bottom row (TSG): (d) single-video sentence grounding [1034, 1033]; (e) corpus-level moment retrieval [1042]; (f) joint grounding and highlight detection [1043]. The online panel separates event onset, detection time, and the current observation boundary; future frames are unavailable. Dashed intervals in the weak-supervision panel denote inferred hypotheses, not boundary annotations. Scene examples, confidence values, and saliency ranks are illustrative.

TSG began with cross-modal matching. CTRL [1034] embeds the sentence and candidate clips in a shared space and ranks clips, with boundary regression on top; MCN [1033] does the same for DiDeMo with global video context. Later models reason over the candidates jointly: MAN [1055] refines moments iteratively through a graph over proposals, 2D-TAN [1056] scores every (start, end) pair on a two-dimensional temporal-adjacency map so that neighboring candidates inform each other, and VSLNet [1057] drops proposals altogether and predicts the start and end positions as span extraction, in analogy to reading comprehension. Moment-DETR [1058] brought set prediction to grounding, decoding moments and saliency scores directly from an encoder–decoder, while TVR [1042] extended grounding to corpus scale with a coarse-to-fine search over subtitles and frames.

6.4.3.3 Datasets and Metrics

Table 6.29 lists the benchmarks; the TAL rows are those of Section 6.3.4, retained for comparison. The TSG benchmarks move from a closed domain (TACoS [1059], cooking) to open-domain sentences over indoor activity (Charades-STA [1034]), Flickr videos with multiple references per query (DiDeMo [1033]), and dense paragraph-level captions (ActivityNet Captions [1035]); YouCook2 [1060] adds procedural structure, TVR [1042] multimodal (subtitle) context and corpus-level retrieval, and QVHighlights [1043] saliency labels for joint highlight detection.

Temporal IoU between a predicted and a ground-truth segment is the primitive for both tasks, and TAL is scored by mAP at fixed tIoU thresholds (per-threshold on THUMOS14, averaged over $[0.5{:}0.95]$ on ActivityNet) and by AR@AN for proposal quality [1062], as defined in Section 6.3.4. TSG uses retrieval

Table 6.29: Representative datasets for temporal localization; "Task" distinguishes category-conditioned TAL from text-conditioned TSG. Counts are indicative of commonly used versions and splits.

Task	Dataset	Year	Scale / Classes	Temporal annotations
TAL	THUMOS14 [905]	2014	200 train + 213 test videos / 20 cls	Frame-accurate, untrimmed
TAL	ActivityNet v1.3 [906]	2016	∼20k videos / 200 cls	Untrimmed, diverse
TAL	HACS Segments [907]	2019	50k videos / 200 cls; 139k segments	Clip+segment, dense
TAL	EPIC-KITCHENS-100 [1061]	2020	700 egocentric videos / verb-noun labels	Egocentric, narration-aligned
TAL	MultiTHUMOS [909]	2015	∼400 videos / 65 cls; ∼38.7k ann.	Dense, multi-label
TAL	FineAction [908]	2022	17k videos / 106 cls; 103k instances	Fine-grained, precise
TSG	TACoS [1059]	2013	127 cooking videos; ∼18.8k query-segment pairs	Cooking, aligned
TSG	Charades-STA [1034]	2017	∼6.6k videos; 16,128 pairs	Charades-based pairs
TSG	DiDeMo [1033]	2017	∼10k Flickr videos; ∼40k queries	Multi-reference, ranking
TSG	ActivityNet Captions [1035]	2017	20k videos; ∼100k sentences	Dense captions
TSG	YouCook2 [1060]	2018	2k untrimmed cooking videos / 89 recipes	Cooking procedures
TSG	TVR [1042]	2020	21.8k clips (60–90 s) from 6 TV shows; 109k queries	Subtitles, multi-modal
TSG	QVHighlights [1043]	2021	∼10k YouTube videos; query+saliency	Query-saliency, highlights

metrics instead. For queries $\mathcal{Q}$, ranked predictions $\hat{m}_j$, and reference segments $G(q)$,

$$\mathrm{R@}n,\mathrm{IoU}{=}\theta \;=\; \frac{1}{|\mathcal{Q}|}\sum_{q\in\mathcal{Q}} \mathbf{1}\Big(\max_{1\le j\le n}\ \max_{g\in G(q)} \mathrm{tIoU}(\hat{m}_j, g) \ge \theta\Big) \tag{6.43}$$

is the fraction of queries for which some top-n prediction overlaps a reference by at least θ (commonly $n \in \{1, 5\}$, $\theta \in \{0.3, 0.5, 0.7\}$), and $\mathrm{mIoU} = \frac{1}{|\mathcal{Q}|}\sum_q \max_{g\in G(q)} \mathrm{tIoU}(\hat{m}_1, g)$ scores the top prediction alone; QVHighlights adds mAP over tIoU thresholds [1043]. Recall at a threshold binarizes overlap and ignores the quality of the ranking, which motivated ranking-sensitive alternatives such as AxIoU [1063], and annotator disagreement about where a described moment begins and ends puts a ceiling on how precisely any of these can be measured.

Table 6.30 gives R@1 results for the TSG methods above; TAL numbers are in the table of Section 6.3.4. Relational reasoning over candidates (2D-TAN) and span prediction (VSLNet) roughly doubled CTRL's R@1 at tIoU 0.5 on Charades-STA and raised it by half again at 0.7, and Moment-DETR adds a further step on Charades-STA without proposals or suppression; its QVHighlights numbers are the reference point for joint grounding and highlight detection.

6.4.3.4 Challenges

Boundaries are soft. Actions span sub-second gestures to multi-minute activities, which no single temporal resolution serves, and annotators disagree about where an instance starts and ends; a sentence adds its own ambiguity about which part of an activity it describes. The tIoU-thresholded metrics treat these fuzzy boundaries as exact, so part of the measured error is irreducible [1063], and probabilistic boundary modeling with calibrated confidences is the natural response.

Table 6.30: Temporal sentence grounding, R@1 at tIoU 0.5 / 0.7 (%). Charades-STA and ActivityNet Captions values as reported in the 2D-TAN [1056], VSLNet [1057], and Moment-DETR [1058] papers (visual features differ across methods as in the originals; MAN uses I3D features); QVHighlights test values from Moment-DETR without pretraining, with mAP averaged over tIoU thresholds.

Method	Design	Charades-STA	ActivityNet Captions	QVHighlights
		R@1 0.5 / 0.7	R@1 0.5 / 0.7	R@1 0.5 / 0.7 (mAP)
CTRL [1034]	Clip–sentence matching	23.6 / 8.9	29.0 / 10.3	–
MAN [1055]	Iterative graph adjustment	46.5 / 22.7	–	–
2D-TAN [1056]	2D temporal-adjacency map	39.7 / 23.3	44.5 / 26.5	–
VSLNet [1057]	Span prediction	54.2 / 35.2	43.2 / 26.2	–
Moment-DETR [1058]	Set prediction	53.6 / 31.4	–	52.9 / 33.0 (30.7)

The two tasks are converging. A closed label set limits TAL, while TSG is bounded by the vocabulary of its query sets; text prompts in place of class labels [1041] and set prediction for moments [1058] point to a single architecture that grounds both categories and sentences. Doing this over minutes of video, across camera viewpoints (egocentric footage in particular), with audio, narration, or noisy subtitles as further evidence [1042, 1043], and causally for streaming use [1036] is where the current systems fall short: memory-efficient long-context encoders and corpus-scale coarse-to-fine retrieval are the two ingredients most actively developed.

6.5 Multimodal Spatial Perception



The preceding sections perceive space largely through vision. Yet vision alone is often incomplete: it is blind past occluders, silent about what lies outside the frame, and mute on which entity a person actually means. Multimodal spatial perception closes these gaps by grounding spatial understanding in signals beyond the image. The section takes two complementary couplings: audio-visual spatial localization, which uses sound to place events that vision may miss, and vision-language grounding, which ties linguistic reference to the specific entity or region it denotes. Neither adds a new spatial primitive; each supplies a channel of evidence that makes perception more complete where a single modality falls short.

6.5.1 Audio-Visual Spatial Localization

Humans localize a sound by combining auditory cues (interaural level and time differences, spectral shaping) with visual context (lip motion, object dynamics) [324]. Audio-visual spatial localization (AV-SL) asks a machine to do the same: find the pixel region, object, or 3D point that a sound comes from, under reverberation, noise, occlusion, and several simultaneous sources [398, 399]. The two modalities are complementary. Binaural or array audio supplies directional cues, while distance requires additional geometric, near-field, or learned evidence; vision says which entities are plausible sounders and separates acoustically confusable ones (two similar instruments, only one playing). Spatial audio for AR/VR [329], teleconferencing [1064], and audio-guided navigation [46] are the main consumers.

6.5.1.1 Task Definition

The input is an audio stream A (mono, binaural, or multichannel, optionally ambisonic [326]) time-aligned with a video V (RGB, depth, or multi-view) of the same interval; the audio may contain zero, one, or several sources, and the video may show silent objects that look like sounders. The output localizes the sounding entity in a form that depends on the variant: a 2D heatmap or mask, a face or body box with a speaking state,

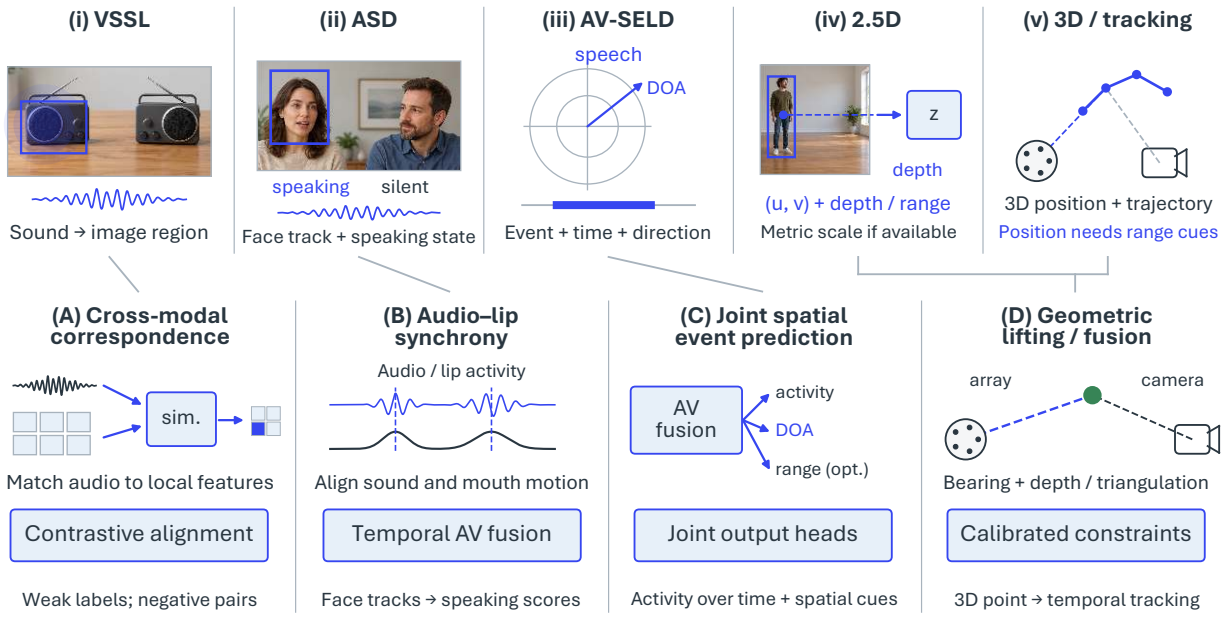


Figure 6.26: Audio-visual spatial localization. *Top*: task outputs for (i) VSSL, (ii) ASD, (iii) AV-SELD, (iv) 2.5D localization, and (v) 3D localization and tracking. *Bottom*: method families (A) cross-modal correspondence, (B) audio–lip synchrony, (C) joint spatial event prediction, and (D) geometric lifting or fusion; neutral connectors mark principal task–family associations, not computational flow. Directional arrows inside panels indicate prediction or information flow. The 2.5D panel links a selected image location to depth; 3D position additionally requires range or triangulation constraints. Scene imagery and overlays are illustrative.

an event class with its temporal extent and direction of arrival (DOA, azimuth and elevation) and optionally distance [1065], a 2D location with depth, or a 3D point or trajectory in camera or world coordinates. When the source is off screen or silent, the correct output is a null prediction rather than the nearest plausible object.

Five variants are in common use. Visual sound source localization (VSSL) highlights the sounding region of a frame or clip as a class-agnostic heatmap, in single- and multi-source settings and increasingly with negative or off-screen audio as a robustness test. Active speaker detection (ASD) decides who is speaking and localizes the talker as a face track, often under streaming constraints [1064]. Audio-visual sound event localization and detection (AV-SELD) couples what and when with where: event class, temporal extent, DOA, and in recent benchmarks distance [1065]. The 2.5D variant adds depth or range to a 2D localization from monocular depth [140], stereo, or echoes [328]; and 3D DOA estimation returns a spatial bearing, whereas source-position or trajectory estimation additionally uses range or triangulation constraints from calibrated cameras, arrays, or egocentric rigs [681].

6.5.1.2 Methodology

Figure 6.26 maps the five variants onto four method families.

Cross-Modal Correspondence (VSSL). VSSL is learned almost entirely without location labels, from the co-occurrence of sound and image. An audio encoder maps the clip to an embedding $\mathbf{a}$, a visual encoder maps the frame to a feature map $\mathbf{v}(\mathbf{x})$, and the localization map is the per-position similarity

$$S(\mathbf{x}) = \frac{\mathbf{a}^\top \mathbf{v}(\mathbf{x})}{\|\mathbf{a}\| \, \|\mathbf{v}(\mathbf{x})\|}, \tag{6.44}$$

trained so that matching audio-image pairs score higher than mismatched ones [1066, 399]; Sound of Pixels [400] instead mixes two clips and asks the network to separate them conditioned on pixel features,

which forces the same alignment. Later work targets the failure modes of co-occurrence. LVS [1067] mines hard negatives inside the same image (regions that look like sounders but do not match the audio) and introduces VGG-SS with explicit negative-audio evaluation; deep multimodal clustering [1068] and coarse-to-fine mixture objectives [1069] handle several sources at once; and SLAVC [1070] shows that earlier methods depend on early stopping and adds momentum encoders and extreme visual dropout so that the alignment does not collapse onto background.

Audio–Lip Alignment (ASD). ASD runs a face detector and tracker and then decides, per track and per frame, whether the visible mouth produced the audio. The decision rests on audio-visual synchrony: a visual encoder over the lip region and an audio encoder over the spectrogram are compared across time [330], and TalkNet [1064] adds long-term temporal modeling of both streams with cross-attention, which is what separates a brief pause from a non-speaker. Active Speakers in Context [1071] scores each candidate jointly with the other faces in the frame, and ASDNet [1072] makes the three stages explicit (per-frame audio-visual encoding, inter-speaker relation modeling, temporal modeling), with each stage trained for its role.

Joint Event and DOA Prediction (AV-SELD). SELDnet [1073] set the template: a convolutional recurrent network over multichannel spectrogram features with two heads, one classifying active events per frame and one regressing their DOA. ACCDOA [1074] folds the two into one output, a vector per class whose direction is the DOA and whose length is the activity, which removes the need to balance two losses; the DCASE challenge tracks [1075] standardized the protocol, and STARSS23 [1065] added video and source distance.

Geometric Lifting (2.5D and 3D). Lifting a 2D localization to metric space uses geometry rather than learned correspondence alone. VisualEchoes [328] shows that echoes of an emitted chirp carry depth: the time and spectrum of the return, fused with the image, improve monocular depth estimation and give the 2.5D variant its range. Audio-visual spatial alignment [331] learns representations by predicting whether 360° video and ambisonic audio are spatially consistent, a pretext that encodes direction, and self-supervised spatial audio generation [329] learns the inverse mapping from 360° video to ambisonics. Full 3D systems combine classical array processing (time differences of arrival from GCC-PHAT [1076], steered-response-power beamforming [1077]) with calibrated cameras and triangulation, and in simulation (SoundSpaces [46, 47]) with the agent's own motion.

6.5.1.3 Datasets and Metrics

Table 6.31 pairs each variant with its benchmarks and metrics; there is no shared benchmark across variants because their outputs differ.

Table 6.31: AV-SL task variants with representative datasets, outputs, and metrics. Newer VSSL test suites add negative or off-screen audio to measure robustness.

Task	Representative Datasets	Primary Outputs	Typical Metrics
VSSL (2D)	Flickr-SoundNet [1066], MUSIC [400], VGG-SS [1067] (from VGG-Sound [1078]), AVSBench [1079]	heatmap / mask	cIoU, AUC; AP with negatives
ASD	AVA-ActiveSpeaker [1080]	talker boxes + speaking state	mAP
AV-SELD	STARSS23 [1065], DCASE SELD tracks [1075]	event + time + DOA (+ distance)	F-score, error rate, DOA error, distance error
2.5D	Replica / Matterport3D echo splits [328]	2D + depth	AbsRel, RMSE
3D DOA / trajectory	STARSS23 positions [1065], SoundSpaces [46]	DOA / 3D point / trajectory	angular error, positional error

VSSL is scored by thresholding the localization map and computing its IoU with the annotated region (cIoU, the fraction of test images above 0.5) and the area under the IoU-threshold curve (AUC) [1066]; the

Table 6.32: Visual sound source localization and active speaker detection. VSSL: cIoU (%) on the Flickr-SoundNet and VGG-SS test sets, and AP (%) on their extended versions with negative audio, all models trained on VGG-Sound 144k, as reported in the SLAVC comparison [1070]. ASD: mAP (%) on AVA-ActiveSpeaker val, from the TalkNet [1064] and ASDNet [1072] papers.

• *Visual sound source localization*					
Method	**Design**	**Flickr cIoU**	**VGG-SS cIoU**	**Ext. Flickr AP**	**Ext. VGG-SS AP**
Attention10k [1066]	Audio-guided attention	42.3	18.5	–	–
DMC [1068]	Multimodal clustering	55.6	23.9	–	–
LVS [1067]	Hard negatives	71.9	34.4	9.8	5.2
SLAVC [1070]	Momentum encoder, dropout	83.2	37.2	51.6	33.0
• *Active speaker detection (AVA-ActiveSpeaker val, mAP)*					
Method	**Design**	**mAP**			
AVA baseline [1080]	Audio-visual fusion	79.2			
ASC [1071]	Inter-speaker context	87.1			
TalkNet [1064]	Long-term temporal, cross-attention	92.3			
ASDNet [1072]	Three-stage architecture	93.5			

extended VGG-SS and Flickr protocols add clips whose audio has no visible source and report average precision over the decision to localize at all [1067, 1070]. ASD uses detection-style mAP over face tracks with speaking labels [1080]. SELD combines detection (F-score, error rate) with localization (DOA error, localization recall) [1075], with distance error added in STARSS23 [1065]; 2.5D inherits the depth metrics of Section 6.2.1, and 3D settings report angular and positional error.

Table 6.32 collects the two variants with a stable leaderboard. On VSSL the negative-audio protocol changes the ranking: LVS leads the earlier methods on the standard Flickr test but drops to 9.8 AP when silent clips are mixed in, whereas SLAVC holds both. ASD on AVA-ActiveSpeaker has moved from 79.2 mAP for the dataset baseline to above 93 with explicit temporal and inter-speaker modeling, and the remaining differences among top systems are about latency and streaming rather than accuracy. AV-SELD has no such table: each DCASE edition changes the data and the metric, and the STARSS23 distance extension is not comparable to the DOA-only tracks that SELDnet and ACCDOA were evaluated on.

6.5.1.4 Challenges

Co-occurrence is not causation. A model that has learned which objects usually make which sounds will localize a guitar whether or not it is playing, and the negative-audio protocols [1067, 1070] exist because the standard tests rewarded exactly this; the open problem is grounding on the acoustic evidence itself, which matters most for off-screen and silent cases where the right answer is no localization.

Going from 2D saliency to metric 3D requires semantics and geometry together. Depth foundation models [140] (Section 6.2.1) and self-supervised visual features [958] now supply priors that AV-SL pipelines previously learned from scratch, and STARSS23's joint event, time, DOA, and distance protocol [1065] is the first benchmark that scores all four at once; multi-object objectives for overlapping events are the current frontier there.

Multiple simultaneous sources, identity through occlusion, and on-screen/off-screen transitions over long videos [1069, 1081] are the AV-SL form of the identity problem of Section 6.4.2 and the pose-continuity problem of Section 6.2.5. Streaming use adds causal encoders, low-latency fusion, calibration drift, and graceful fallback when one modality is missing [1064, 1072].

6.5.2 Vision-Language Grounding

Vision-language grounding ties a linguistic reference to the element of a scene it denotes. Given *"Put your hand on the small mug on the right side of the table behind the blue book"*, a grounded system must resolve the noun phrase, its attributes, and its spatial relations to one particular mug in the image; this referential link is what instruction following, multimodal dialogue, and human-robot interaction build on.

6.5.2.1 Task Definition

The input is a scene S (an image, a video, or a 3D environment) and a query Q, from a bare noun phrase ("the red cup") to a sentence with attributes, relations, actions, or an implicit goal ("the person who is waving", "something to open the box with"). The output $\mathcal{R}(Q, S)$ is the referred region in the form the variant requires: a box, a mask, a set of regions (possibly empty), a per-frame mask track, or a 3D box. Two formulations recur through the methods below: ranking a set of candidate regions r by a cross-modal score, or regressing the region directly,

$$\hat{r} = \arg\max_{r \in \mathcal{C}(S)} s\big(\phi(Q), \psi(S, r)\big) \qquad \text{or} \qquad \hat{r} = f_\theta(S, Q), \tag{6.45}$$

with ϕ and ψ the text and region encoders and $\mathcal{C}(S)$ a proposal set.

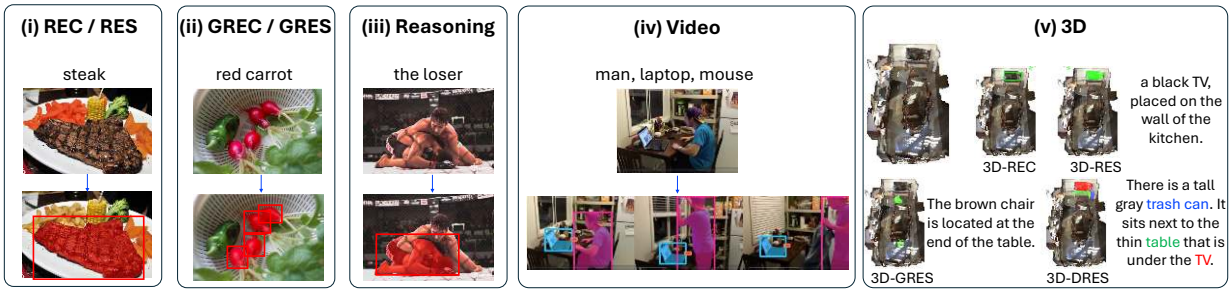


Figure 6.27: Vision-language grounding task variants: (i) REC / RES [1082, 1083]; (ii) generalized REC / RES with zero, one, or multiple targets [1084]; (iii) reasoning grounding [1085]; (iv) video referring and reasoning grounding [1086]; (v) 3D and embodied grounding [1087].

Figure 6.27 lays out the five variants, which differ in scene modality, output granularity, and query complexity.

Referring expression comprehension and segmentation (REC / RES). One image, one expression, one target, returned as a box (REC) or a mask (RES). RefCOCO and RefCOCO+ [1082] (short phrases, the latter without location words) and RefCOCOg [1083] (longer descriptions), all on MS COCO [440] images collected through ReferItGame [1088] or Mechanical Turk, have been the standard benchmarks for a decade; Flickr30k Entities [1089, 1090] grounds every entity mention of a caption at once.

Generalized REC / RES (GREC / GRES). An expression may match zero, one, or several objects ("the red carrots" versus "the unicorn"), which single-target methods cannot represent. gRefCOCO [1084], Ref-ZOM [1091], PhraseCut [1092], and OmniLabel [1093] evaluate this setting, and MaskGroups-HQ [1094] generalizes the prompt itself to arbitrary groups of masks specified by text and reference masks. The output is a set, so the metric must be as well (§Datasets and Metrics).

Reasoning grounding. Some queries require inference beyond noun matching: "the loser" needs a judgment about the people shown, "turn on the TV" needs the remote. CLEVR-Ref+ [1095] diagnoses this on rendered scenes with ground-truth reasoning programs; ReasonSeg [1085] poses implicit, multi-step queries on real images with mask output, extended by MMR [1096] (194K implicit queries, multi-target and part-level), ReasonSeg-Inst [1097] (instance level), and MC-Bench [1098] (instances across image pairs). Ref-Adv [1099] keeps the classic REC format but makes expressions minimally sufficient, negation-heavy, and paired with same-category distractors.

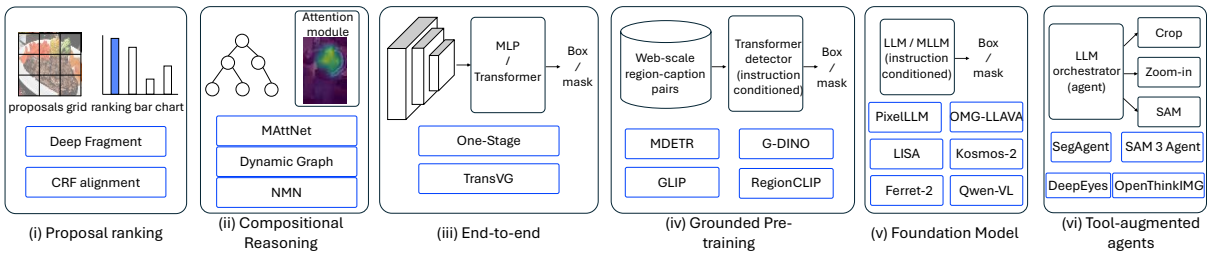


Figure 6.28: Vision-language grounding method families: (i) proposal ranking, (ii) compositional reasoning, (iii) end-to-end grounding, (iv) grounded pre-training, (v) foundation models, and (vi) tool-augmented agents. Panels (i)-(iv) correspond to families (A)-(D) in the text, (v) to (E)-(F), and (vi) to (G); family (H), generation for grounding, is not depicted.

Video referring and reasoning grounding. The referent must be found in every frame through motion, appearance change, and occlusion, and the query may name it by what it does ("the man waving his hand"). Refer-Youtube-VOS [246], Refer-DAVIS17 [1100], A2D- and JHMDB-Sentences [1101], MeViS [1102], ReVOS [1103], and ReasonVOS [1104] add language to existing segmentation and actor-action datasets; Ref-VPS [1105] extends the referent from objects to processes such as smoke spreading or waves breaking, which have no object boundary at all. Tracking machinery is usually folded in to carry identity across frames [1106].

3D and embodied grounding. The scene is a point cloud or a set of views, and the query refers to an object in 3D ("the chair next to the window"). ScanRefer [373] requires detecting the referent as a 3D box and Multi3DRefer [1107] allows zero to many targets; ReferIt3D [374] provides the object instances and scores selection among them; Instruct3D [1108] and Multi3DRes [1109] ask for point-level masks from instruction-style queries. Embodied grounding makes the output an action: an agent told "go pick up the red cup on the right" must identify the cup and reach it [201].

6.5.2.2 Methodology

Figure 6.28 organizes the methods into eight families, (A) to (H). They are trajectories rather than eras: the oldest paradigm has resurfaced at foundation-model scale.

(A) Proposal Ranking. The first grounding models instantiate the ranking form of Eq. (6.45): detect region proposals, embed them and the query in a joint space, and pick the best-scoring region. Deep Fragment Embeddings [1110] aligned sentence fragments with detected regions, and scene-graph CRFs [598] grounded structured queries to regions for retrieval. Proposal quality bounded these methods and context-dependent expressions defeated them. The same two-stage shape has returned with far stronger proposals: RAS [1094] has a segmentation foundation model propose candidate masks and a mask-centric multimodal LLM *select* the group that satisfies the prompt, and this non-autoregressive selection beats mask generation in both accuracy and latency.

(B) Compositional Reasoning. Modular networks parse the expression and route its parts to dedicated modules. CMN [1111] decomposes an expression into a subject–relationship–object triplet handled by localization and relationship modules; MAttNet [1112] soft-parses phrases into appearance, location, and relationship attention modules; dynamic graph attention [1113] reasons over the linguistic structure on a scene graph. These models disambiguate compositional queries ("the man behind the woman in red") but inherit every parsing error.

(C) End-to-End Grounding. The regression form of Eq. (6.45) removes proposals: a single network fuses the sentence embedding with the image feature map and predicts the box. One-Stage Grounding [1114] did this with a detector head, TransVG [1115] with a pure Transformer in which a learnable token attends over

visual and textual tokens and regresses coordinates. On the segmentation side, CRIS [245] transfers CLIP's image-text alignment to the pixel level with text-to-pixel contrastive learning and made end-to-end RES the standard formulation.

(D) Grounded Pre-Training. Phrase-region alignment learned at scale turned grounding into an open-vocabulary capability. MDETR [360] trains a DETR-style set predictor on text-aligned detection corpora so that every box is tied to a span of the caption; GLIP [361] reformulates detection as phrase grounding and scales to 27M region-caption pairs, giving zero-shot transfer; RegionCLIP [351] adds region-level contrastive alignment, and Grounding DINO [362] deep language fusion at every stage of a DINO detector, which made it the default open-vocabulary detector of later tool chains (family (G)). The residual weakness is the *entity shortcut*: models latch onto object names and ignore the description. DesCo [1116] counters it in the data, pre-training on LLM-generated rich descriptions with context-sensitive negative queries, which raises description-conditioned detection on LVIS and OmniLabel.

(E) Grounding MLLMs. The first multimodal LLMs built for grounding gave the language backbone a region interface. Kosmos-2 [1117] emits location tokens that hyperlink phrases to boxes inside the generated text; Ferret [1118] and Ferret-v2 [1119] encode boxes, points, and free-form regions as inputs at any resolution. Mask output comes from attaching a decoder: LISA [1085] decodes a dedicated segmentation token's embedding into a mask, which is what makes reasoning segmentation possible, and LISA++ [1097] extends it to instances within a dialogue; GLaMM [1120] unifies phrase grounding, referring segmentation, and free-form conversation under a grounded-conversation objective; Sa2VA [1121] couples a LLaVA-style backbone to SAM 2 [482] and carries dense grounding from images to video referring segmentation. Moondream 3's segmentation extension [1122] dispenses with the decoder and writes the mask autoregressively as a vector path, refined by a lightweight head and trained with reinforcement learning on mask overlap.

(F) Generalist MLLMs. A parallel line keeps no grounding component at all: the model writes coordinates as plain text, and grounding is one capability among many. Florence-2 [1123] is the compact case, a sequence-to-sequence model trained on the 5.4B-annotation FLD-5B corpus that casts captioning, detection, grounding, and polygon-vertex segmentation into one text interface, with mask accuracy below the decoder-based models of family (E). The frontier generalists, Gemini [1124], Qwen2.5-VL and Qwen3-VL [1125], InternVL 2.5 and InternVL 3 [1126, 1127], and Seed1.5-VL [1128], now match or exceed the specialists on the RefCOCO family from prompting alone (Table 6.34); Llama 3 [1129] folds phrase-to-box and phrase-to-mask annotations into its multimodal pre-training but reports no REC numbers.

(G) Tool-Augmented Agents. A third line delegates localization to specialist models under an outer control loop, which has moved from fixed pipelines to generated programs to learned tool use. Grounded SAM [1130] chains Grounding DINO boxes into SAM [138] masks without training; ViperGPT [1131] has a code LLM write a Python program that calls perception APIs, so that logic and arithmetic are symbolic and perception is delegated, with interpretable intermediate steps. SegAgent [1132] treats mask annotation as a sequential decision process, imitating the click sequences of human annotators on an interactive segmenter; DeepEyes [1133] internalizes the tool, using reinforcement learning to make the model emit coordinates that trigger zoom-in crops inside its own chain of thought. SAM 3 Agent [1134] wraps SAM 3, whose concept prompts return every instance matching a noun phrase, with an MLLM that proposes phrases, inspects the returned masks, and selects those satisfying the full expression, which gives zero-shot referring and reasoning segmentation without grounding-specific training.

(H) Generation for Grounding. The newest family treats a large text-to-image or text-to-video generator as a grounding model, on the premise that generative pre-training already induces a fine-grained map from language to spatial and temporal structure. VD-IT [1135] uses a frozen text-to-video diffusion model as the feature extractor for referring video segmentation and finds its features more temporally consistent and better aligned with language than discriminative video backbones; ReferEverything [1105] keeps the generator

intact and fine-tunes it to predict mask latents decoded by the frozen VAE, which transfers zero-shot to rare objects, "stuff", and the amorphous processes of Ref-VPS. Vision Banana [1136] instruction-tunes a state-of-the-art image generator to render the task output as an image, segmentation as painting the referred region a prescribed color, and matches specialist systems zero-shot while keeping its generative ability. Grounding quality in this family rises with the underlying generator.

6.5.2.3 Datasets and Metrics

Table 6.33 lists the benchmarks by variant. Annotation cost rises steeply from a 2D box to a per-frame mask to a 3D box, so image-level REC/RES corpora dominate in size and maturity, video referring sets are an order of magnitude smaller, and 3D corpora smaller still. Within the image benchmarks a generational split has opened. RefCOCO/+/g now serve mainly as training corpora: specialists and generalist MLLMs report Acc@0.5 above 90% on them, an audit found label errors in 14%, 24%, and 5% of RefCOCO, RefCOCO+, and RefCOCOg [1137], and much of the score survives shuffling the word order of the expression [1099]. The evaluation-oriented replacements are built to expose what saturation hides: Ref-L4 [1137] (cleaned labels, long expressions, a 365-category vocabulary), Ref-Adv [1099] (shortcut-suppressed, where the same models fall to roughly 60%), HC-RefLoCo [1138], GigaGrounding [1139], and GroundingME [1140], which scores discrimination, spatial reasoning, limited visibility, and the rejection of ungroundable queries, and on which the best model resolves 45% of samples while most score zero on rejection.

Table 6.33: Common vision-language grounding tasks with representative datasets and typical evaluation metrics; several tasks share datasets (*e.g.*, RefCOCO underlies both REC and RES).

Task	Representative Datasets	Primary Outputs	Typical Metrics
REC / RES	RefCOCO/+/g, Flickr30k Entities, Ref-L4, Ref-Adv, GroundingME	box / mask	Acc@0.5; gIoU / cIoU (RES)
GREC / GRES	gRefCOCO, OmniLabel, PhraseCut, Ref-ZOM, MaskGroups-HQ	multi-box / multi-mask	F1, N-acc, gIoU / cIoU, AP_{descr}
Reasoning grounding	CLEVR-Ref+, ReasonSeg, ReasonSeg-Inst, MMR, MC-Bench	mask (or box)	gIoU / cIoU, Acc@0.5
Video referring	MeViS, Refer-YT-VOS, Refer-DAVIS17, ReVOS, A2D-Sent., Ref-VPS	per-frame mask	$\mathcal{J}\&\mathcal{F}$; mAP (A2D-Sent.)
3D / Embodied	ScanRefer, ReferIt3D, Multi3DRefer, Instruct3D	3D box / 3D mask	Acc@0.25/0.5, F1@0.5, mIoU

Every metric here is built on the IoU of predicted and reference regions (Section 6.4.2 gives the mask form). REC reports Accuracy@0.5, the fraction of expressions whose box reaches IoU ≥ 0.5, with stricter thresholds and the threshold-averaged mAcc [1137] increasingly added; phrase grounding, where several proposals may be returned, reports Recall@k. For set-valued outputs (GREC / GRES) the prediction is compared with the ground-truth set through per-expression F1 over boxes or masks [1091] and a no-target accuracy for expressions with no referent [1084], and OmniLabel's AP_{descr} computes average precision over free-form descriptions including negative ones [1093]. Segmentation reports two aggregates that differ in weighting: gIoU (also called mIoU) averages per-sample IoU and so weights every expression equally, whereas cIoU accumulates intersection and union over the whole set before dividing and so weights large objects more; reasoning and referring segmentation papers report both [1085], often with Precision@X for the distribution of mask quality and a boundary IoU for edge accuracy [1122]. Video referring segmentation uses the $\mathcal{J}\&\mathcal{F}$ composite of Section 6.4.2.

Table 6.34 reports the two variants on which the families can be compared. On REC the step from proposal ranking and modular reasoning (MAttNet) to end-to-end regression (TransVG) and then to grounded pre-training (MDETR, Grounding DINO) is about 14 points of RefCOCO Acc@0.5, and the generalist MLLMs (Qwen2.5-VL, InternVL 2.5) sit at or above the best specialist without benchmark-specific training. On RES

the grounding MLLMs with a mask decoder have passed the end-to-end specialists (CRIS to Sa2VA is eleven cIoU points on RefCOCO), and Sa2VA's video columns show the same model carrying over to referring video segmentation; on ReasonSeg, LISA-7B reaches 52.9 gIoU after fine-tuning where open-vocabulary segmenters stay below 30 [1085]. The tool-augmented and generative families are evaluated zero-shot on subsets of these benchmarks and are not directly comparable here. What the table cannot show is cross-variant breadth: methods strong on image REC routinely degrade on video, reasoning, and 3D grounding, and no single architecture leads all five variants.

Table 6.34: Representative grounding methods. REC: Acc@0.5 (%) on RefCOCO val / testA / testB and RefCOCOg val, from the Grounding DINO [362], Ferret [1118], and Qwen2.5-VL [1125] comparisons. RES: cIoU (%) on the RefCOCO / RefCOCO+ / RefCOCOg val splits and $\mathcal{J}\&\mathcal{F}$ on MeViS and Ref-DAVIS17, from the Sa2VA comparison [1121] (CRIS from its own paper, ResNet-101).

• *Referring expression comprehension (Acc@0.5)*

Method	Family	RefCOCO			RefCOCOg
		val	testA	testB	val
MAttNet [1112]	(B) modular attention	76.7	81.1	70.0	66.6
TransVG [1115]	(C) end-to-end Transformer	81.0	82.7	78.4	68.7
MDETR [360]	(D) grounded pre-training	86.8	89.6	81.4	81.6
Grounding DINO-L [362]	(D) grounded pre-training	90.6	93.2	88.2	86.1
Ferret-13B [1118]	(E) grounding MLLM	89.5	92.4	84.4	85.8
Qwen2.5-VL-72B [1125]	(F) generalist MLLM	92.7	94.6	89.7	89.9
InternVL2.5-78B [1126]	(F) generalist MLLM	93.7	95.6	92.5	92.7

• *Referring segmentation (cIoU) and video referring segmentation ($\mathcal{J}\&\mathcal{F}$)*

Method	Family	RefCOCO	RefCOCO+	RefCOCOg	MeViS / Ref-DAVIS17
CRIS [245]	(C) end-to-end, CLIP	70.5	62.3	59.9	–
LISA-7B [1085]	(E) mask-decoder MLLM	74.9	65.1	67.9	–
GLaMM-7B [1120]	(E) grounded conversation	79.5	72.6	74.2	–
VISA-13B [1103]	(E) video reasoning MLLM	–	–	–	44.5 / 70.4
Sa2VA-8B [1121]	(E) MLLM + SAM 2	81.9	76.5	78.9	46.9 / 75.2

6.5.2.4 Challenges

Ambiguity and interaction. "The guy on the right" depends on the viewpoint, and "the red cup" presumes one salient red cup. People resolve this through shared context or a follow-up question; grounding models make an arbitrary choice or fall back on a dataset bias such as the largest instance. Recognizing that an expression matches several entities, abstaining under uncertainty, and asking for clarification are outside almost every current benchmark, which is one-shot (one query, one image), while real use is a dialogue with memory of what was grounded before.

Reasoning and commonsense. "The tool needed to open the box" cannot be resolved from the image alone, and adversarial evaluation shows that much of current REC accuracy rests on shortcut matching rather than reasoning [1099]. LLM-based models carry the knowledge but may answer "a key" when none is visible, and vision-only models lack it entirely; reasoning-driven segmentation [1085] and agentic active perception [1133] are the two bridges. Video adds temporal and causal structure ("the man who picks up the phone and then leaves", "the car that caused the accident") that current video grounding, which attends mostly to appearance and simple motion, does not capture.

Open world and 3D. Open-vocabulary detectors and CLIP-based models have blind spots on rare objects, fine-grained distinctions (viola versus violin), proper nouns, and figurative references ("the blue crown" for

a cap), and per-category analyses of large multimodal models show a strong bias toward common training categories [1137]. Description-conditioned pre-training [1116] and generative pre-training [1105, 1136] are the routes to long-tail coverage that do not require labeling it. Functional references ("the driver's seat") also imply spatial context that only 3D understanding resolves, and the 3D-language work surveyed above [1141, 201] will grow with augmented reality and robotics.

Efficiency and evaluation. The strongest systems are multi-model tool chains or LLMs with vision adapters and do not fit AR glasses or robots; compact models [1123, 1122] and non-autoregressive selection [1094] show that accuracy does not require scale, but the accuracy-efficiency frontier is unmapped. Evaluation has the saturation problem described under §Datasets and Metrics, and the new benchmarks, each with its own protocol, risk a cycle of leaderboard chasing; a single protocol spanning image, video, 3D, and dialogue, scoring rejection and clarification as well as localization, does not yet exist.

Chapter 7

Spatial Reasoning

Spatial reasoning is the step that turns perceived structure into decisions: given detections, depths, poses, and a task, it works out the relations, the causes, and the actions that follow, the why and how of spatial interaction rather than merely the what and where. It is where perception becomes goal-directed, supporting inference, planning, and problem-solving in physical and virtual environments.

The chapter is organized along the spectrum by which spatial reasoning has actually developed, from explicit symbolic models to learned and foundation-model-driven ones. At one end, reasoning runs over hand-built representations of geometry, physics, and logic, whose structure is inspectable and whose conclusions can be certified. In the middle, it emerges from data, as neural models acquire spatial competence directly rather than being told the rules. More recently, large pre-trained models supply reasoning as a general capability to be elicited and steered. Hybrid approaches recombine these, pairing the semantic reach of foundation models with the precision and verifiability of symbolic and learned components; a closing survey of benchmarks then records how the field measures spatial reasoning as it shifts from passive observation toward active, situated interaction.

7.1 Rule-based Symbolic Spatial Reasoning

Rule-based symbolic spatial reasoning covers the methods that operate over *explicit, model-based representations* of space (geometric primitives, physical laws, graph connectivity, formal constraints, action rules, or causal mechanisms) rather than over learned latent representations alone. What sets this family apart is that reasoning happens inside a model that can be inspected: its state and transitions are legible, so claims about feasibility, optimality, or safety can be backed by certificates rather than empirical averages, and when a plan fails it fails legibly, as an infeasible inverse kinematics, a violated constraint, a missing precondition, or an inconsistent causal attribution, not as an opaque correlation.

The section surveys six kinds of explicit structure, ordered from the most concrete to the most abstract: geometric and algebraic models of position, orientation, motion, and feasibility; physical models of contact, friction, and stability; topological and graph abstractions of space that turn planning into search; formal constraint systems whose solvers return certified plans or proofs of infeasibility; rule-based and programmatic logic that sequences and supervises actions; and causal models that explain, predict counterfactually, and prescribe interventions.

These paradigms are not isolated alternatives. Modern robotic systems routinely compose them: geometric solvers serve as feasibility oracles inside constraint-based search; behavior trees wrap symbolic plans for fault tolerance; causal action languages compile to SAT; topological maps anchor both classical and learning-based

navigation policies. The unifying thread is that each paradigm carries explicit structure (a state, an operator, a constraint, a rule, or a mechanism) that can be inspected, composed, and reasoned about.

7.1.1 Geometric and Algebraic Reasoning

Geometric and algebraic reasoning uses explicit mathematical models of position, orientation, shape, and motion to turn local measurements (image points, depths, joint angles, odometry) into quantities that can be analyzed and verified: where things are in a common frame, what a sensor can see, what a mechanism can reach, and whether a grasp will hold. The same models extend to time-varying settings, where relationships evolve under kinematic and differential constraints and the question becomes not only where entities can be but how they can move. Precision, interpretability, and domain independence have made this the common substrate of robotics [1142, 1143, 1144], AR/VR [112, 1145], and autonomous systems [54, 1146, 165], and the source of the hard constraints that learned components are checked against. Figure 7.1 gives the abstract formulation.

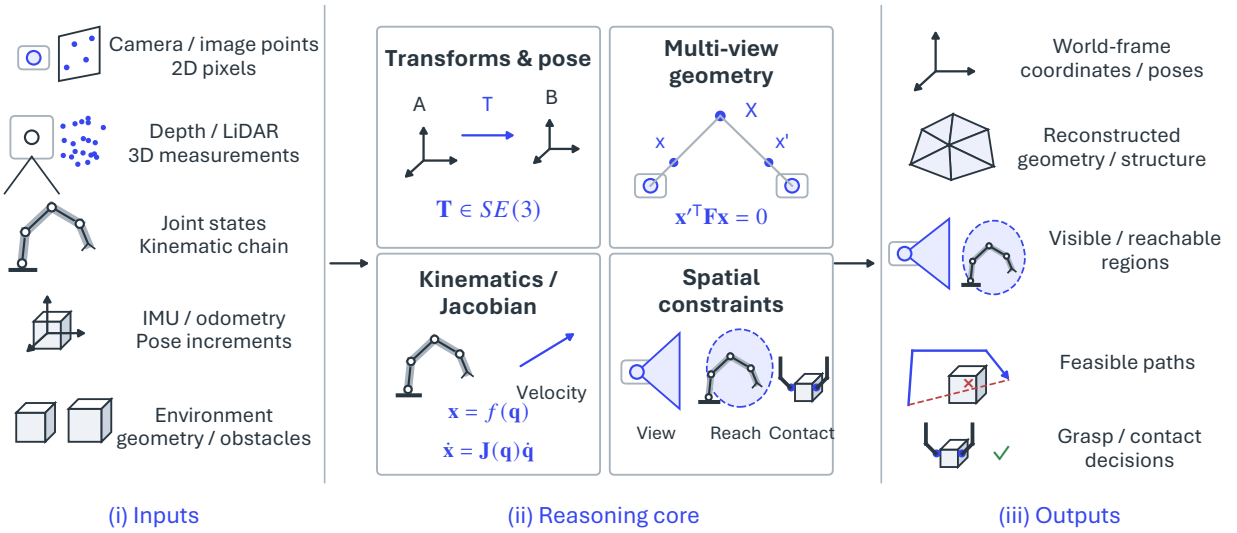


Figure 7.1: Abstract task formulation of geometric and algebraic reasoning: (i) local spatial measurements and system models are mapped by (ii) a core of reusable symbolic formalisms to (iii) analyzable, verifiable spatial outputs.

7.1.1.1 Task Definition

Given measurements $\{\mathbf{z}_k\}$ expressed in local frames (pixels, ranges, joint readings, pose increments) and a system model Θ (sensor calibration, kinematic parameters, environment geometry), geometric reasoning computes a world-frame quantity and a feasibility verdict,

$$\mathbf{x} = g(\{\mathbf{z}_k\}; \Theta), \qquad \psi(\mathbf{x}; \Theta) \in \{\text{reachable}, \text{visible}, \text{force-closed}, \ldots\}, \tag{7.1}$$

where $\mathbf{x}$ may be a pose, a reconstructed surface, a workspace or visibility region, or a trajectory, and ψ is a predicate that can be checked against the model. In the time-varying case the inputs are indexed over time and $\mathbf{x}$ becomes a trajectory with reachable sets and feasibility certificates along it.

7.1.1.2 Methodology

Transformation and Pose Representation. Rigid-body relationships between sensors, objects, and mechanisms are elements of the special Euclidean group

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix} \in SE(3), \tag{7.2}$$

where $\mathbf{R} \in SO(3)$ is a rotation and $\mathbf{t} \in \mathbb{R}^3$ a translation; composing them moves measurements across frames. Early robotics parameterized serial chains through the Denavit–Hartenberg convention [1147, 1148], which gives forward kinematics from a minimal set of link parameters. Lie-group formulations instead write motion as exponentials of screw axes, the product-of-exponentials representation [1142, 1143], which is coordinate-free and behaves well where the DH parameterization has discontinuities. Quaternions [1149, 1150] and dual quaternions [1151, 1152] give compact, numerically stable encodings of rotation and screw motion for calibration and sensor fusion.

Projective and Multi-View Geometry. Reasoning from images must recover 3D structure and camera motion from 2D measurements under perspective distortion and scale ambiguity. The constraint between corresponding points in two views is the epipolar condition

$$\mathbf{x}'^\top \mathbf{F} \mathbf{x} = 0, \tag{7.3}$$

with $\mathbf{F}$ the fundamental matrix. The eight-point algorithm [102] solves for $\mathbf{F}$ linearly, Hartley's normalization [1153] made it numerically stable, and Nistér's five-point solver [1154] reaches the minimal case for calibrated cameras. Zhang's homography-based calibration [692] removed the need for special rigs; the field is consolidated in [112], and structure-from-motion systems such as COLMAP [115] scale it to thousands of images, with RANSAC [101] enforcing the algebraic constraints against outliers throughout.

Jacobian and Differential Formulations. For motion rather than static configuration, the map from configuration to task space,

$$\mathbf{x} = f(\mathbf{q}), \tag{7.4}$$

is differentiated to relate velocities,

$$\dot{\mathbf{x}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}}, \tag{7.5}$$

and the Jacobian $\mathbf{J}(\mathbf{q})$ carries the information about feasibility, singularities, and controllability along a motion. Whitney's resolved motion rate control [1155] used it to command Cartesian velocities without closed-form inverse kinematics; Nakamura's damped least squares [1156] trades exactness for stability near singularities; redundancy resolution [1157, 1158] uses null-space motion for secondary objectives; and Khatib's operational-space formulation [1159] separates task and null-space dynamics so that objectives are written directly in task space.

Visibility and Coverage Geometry. What a sensor at pose p can observe is the set

$$\mathcal{V}(p) = \{x \in \mathcal{E} \mid \text{line}(p, x) \cap \mathcal{O} = \emptyset\}, \tag{7.6}$$

with $\mathcal{E}$ the environment and $\mathcal{O}$ the obstacles. The Art Gallery theorem [1160, 1161] bounds the number of viewpoints needed for full coverage; polygon visibility [1162] and shortest-path queries [1163] made the computations algorithmic; and Tarabanis *et al*. [1164] cast sensor planning as geometric optimization over fields of view modeled as cones, frusta, or polygons.

Workspace and Reachability Analysis. The reachable workspace of a mechanism is

$$\mathcal{R} = \{\mathbf{x} \mid \exists\, \mathbf{q} \in \mathcal{C}_{\text{free}} \text{ such that } \mathbf{x} = f(\mathbf{q})\}, \tag{7.7}$$

with $\mathcal{C}_{\text{free}}$ the feasible configuration space. Roth [1165] and Gosselin [1166] added performance and dexterity measures so that manipulability is characterized along with reachability; Merlet [1167] classified workspace types; Zlatanov *et al*. [1168] analyzed constraint-induced singularities; and sampling-based reachability maps [1169] trade closed-form guarantees for empirical coverage in high-dimensional systems.

Contact and Grasp Geometry. Whether a set of contacts can hold an object is decided in wrench space. Ferrari and Canny [1170] model the forces each contact can apply as a convex set of wrenches $\mathcal{W}$; the grasp is force-closed if

$$\mathbf{0} \in \operatorname{int}\operatorname{conv}(\mathcal{W}), \tag{7.8}$$

so that any external wrench can be resisted. GraspIt! [1171] made these metrics computable for arbitrary hands and objects; task-oriented measures [1172] weight the wrench space by what the task requires; and Dex-Net [336] samples analytic grasps at scale to train data-driven grasp predictors that keep the geometric criterion as their label.

7.1.1.3 Benchmarks

Geometric reasoning is evaluated against geometrically grounded ground truth: camera poses and intrinsics, depth maps and meshes, object models, grasp annotations. The pose, odometry, SLAM, and multi-view reconstruction benchmarks (TUM RGB-D [652], EuRoC [733], TUM-VI [935], KITTI [173], ETH3D [658], Tanks and Temples [1173], DTU [1174], Middlebury [694, 693], HPatches [1175]) and the long-term localization suites under seasonal and day-night change (Aachen, RobotCar Seasons, CMU Seasons [1176], InLoc [1177], 4Seasons [1178]) are those of Section 6.2.4 and Section 6.3.5, where their protocols and headline results are tabulated. Specific to the reasoning problems above are BAL [1179], which standardizes large bundle-adjustment problems for solver evaluation; the Active Vision Dataset [1180], ScanNet [448], and HM3D [205] for next-best-view, coverage, and exploration protocols; Cornell [1181], Jacquard [1182], and GraspNet-1Billion [1183] for grasp quality (rectangle labels, simulated grasp trials, and dense 6-DoF annotations with seen / unseen / novel splits respectively), with the YCB object set [1184] as the shared physical test set; and MoveIt [1185], whose reachability tooling is the de facto harness for inverse kinematics and motion-planning evaluation.

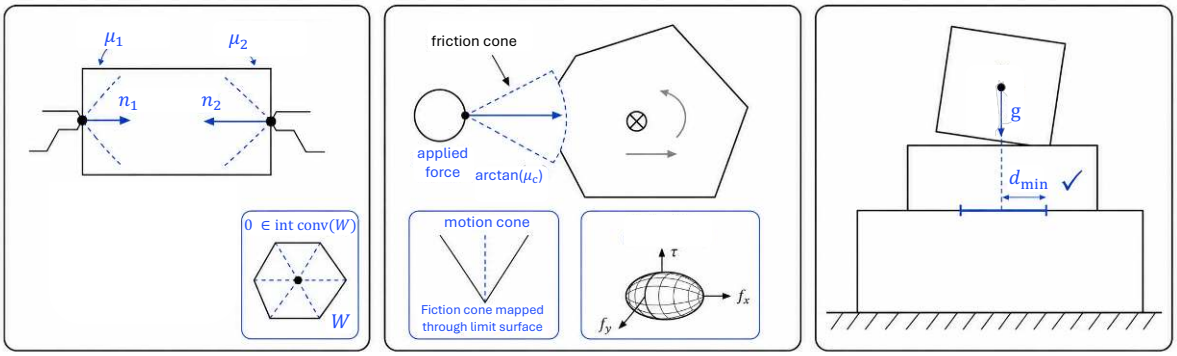
7.1.1.4 Challenges and Trends

The formulations above assume exact geometry, and real sensing is noisy, incomplete, and subject to calibration drift [652, 733]; bundle adjustment [1179] and reachability analysis scale poorly with problem size; many methods fail near singularities or degenerate feature configurations [1156, 1153]; the frameworks are strongest for static rigid bodies and approximate deformable and human-centric geometry [1186]; and evaluation is concentrated on a few canonical datasets and manipulators, which limits cross-domain comparison.

The response has been to keep the geometry and wrap it in probabilistic or learned machinery: factor-graph SLAM and robust bundle adjustment attach uncertainty to the symbolic model (Section 6.3.5); learned correspondence priors are refined by RANSAC [101]; and the long-term benchmarks [1178, 1176] push the same pipelines toward continual adaptation under changing conditions.

7.1.2 Physical Reasoning

Physically-aware spatial reasoning predicts how a spatial configuration evolves under physical law: rigid-body dynamics, contact and friction, and the conditions for static stability. Will a grasp hold? How will an object move when pushed? Will a stack of blocks stand or fall? Answering these from a model of mechanics, rather than from observed correlations, is what lets a robot anticipate the outcome of an interaction before committing to it; Figure 7.2 shows the three canonical cases.



(a) Grasp Stability (b) Planar Pushing (c) Stacking Stability

Figure 7.2: Physics-aware spatial reasoning in robotics: (a) grasp stability from contact forces and friction cones, (b) planar pushing under quasi-static assumptions, and (c) stacking stability under equilibrium and support constraints.

7.1.2.1 Task Definition

Let the environment at time t be the state

$$\mathbf{s}_t = (\mathbf{q}_t, \mathbf{v}_t, \theta), \tag{7.9}$$

with generalized coordinates $\mathbf{q}_t \in \mathbb{R}^n$, velocities $\mathbf{v}_t = \dot{\mathbf{q}}_t$, and physical parameters θ (masses and inertias in the mass matrix $M(\mathbf{q})$, friction coefficients $\{\mu_k\}$, restitution, geometry). An action $\mathbf{u}_t$ (a push, a grasp) enters as a generalized force $\boldsymbol{\tau}_{\text{ext},t+1}$, and the next state follows from a *contact-implicit time-stepping* update:

$$\begin{aligned} M(\mathbf{q}_t)\,\frac{\mathbf{v}_{t+1}-\mathbf{v}_t}{\Delta t} + \mathbf{h}(\mathbf{q}_t,\mathbf{v}_{t+1}) &= \boldsymbol{\tau}_{\text{ext},t+1} + \sum_{k=1}^{K}\Big(J_{n,k}^{\top}\lambda_{n,k} + J_{t,k}^{\top}\boldsymbol{\lambda}_{t,k}\Big), \\ \mathbf{q}_{t+1} &= \mathbf{q}_t + \Delta t\,\mathbf{v}_{t+1}, \end{aligned} \tag{7.10}$$

where $\mathbf{h}$ collects Coriolis, centrifugal, and gravitational terms, and for each potential contact k with gap function $\phi_k(\mathbf{q})$ and Jacobians $J_{n,k}, J_{t,k}$ the normal and tangential impulses satisfy non-penetration, complementarity, and Coulomb friction:

$$\begin{aligned} \phi_k(\mathbf{q}_{t+1}) \ge 0, \quad \lambda_{n,k} \ge 0, \quad \lambda_{n,k}\,\phi_k(\mathbf{q}_{t+1}) = 0, \\ \|\boldsymbol{\lambda}_{t,k}\| \le \mu_k\,\lambda_{n,k}. \end{aligned} \tag{7.11}$$

Physical reasoning is the computation, or approximation, of this transition $\mathbf{s}_{t+1} = \mathcal{T}(\mathbf{s}_t, \mathbf{u}_t)$ and of the queries derived from it (stability, collision, reachability). When motion is slow enough that inertial terms vanish, the *quasi-static* form

$$\sum_{k=1}^{K}\Big(J_{n,k}^{\top}\lambda_{n,k} + J_{t,k}^{\top}\boldsymbol{\lambda}_{t,k}\Big) + \boldsymbol{\tau}_{\text{ext}} \approx \mathbf{0}, \qquad \|\boldsymbol{\lambda}_{t,k}\| \le \mu_k\,\lambda_{n,k}, \tag{7.12}$$

reduces reasoning to an equilibrium feasibility test, which is the setting of the classical analytic models of grasping, pushing, and stacking.

7.1.2.2 Methodology

Model-based physical reasoning is carried out either numerically, in a physics engine that integrates the dynamics above, or analytically, with closed-form criteria for the quasi-static cases. Both rest on the same Newtonian foundation and differ in whether fidelity or interpretability is the priority.

Physics Engines. MuJoCo [1187], Bullet [1188], ODE [1189], and DART [1190] integrate rigid-body dynamics with contact by time stepping, and they resolve contact in one of two ways. Penalty methods treat a contact

as a compliant spring-damper that resists interpenetration,

$$f_{n,k} = k_p\,\delta_k + k_d\,\dot{\delta}_k, \tag{7.13}$$

with penetration depth δ_k, stiffness k_p, and damping k_d, and regularize friction similarly; they are cheap and simple, but stiff parameters destabilize the integration and soft ones introduce artificial compliance, which is acceptable in graphics and approximate planning but not in control. Constraint-based methods enforce Eq. (7.11) directly as complementarity constraints [1191], solved per step as a linear or nonlinear complementarity problem [1192], a convex quadratic program [1193], or a second-order cone program for exact friction cones [1194] (projected Gauss–Seidel and sequential-impulse schemes are the usual implementations). They give physically consistent, nearly penetration-free contact forces at higher cost and with harder scaling in the number of contacts, so engines in practice hybridize: hard constraints on normal forces, regularized tangential friction, warm starting, contact aggregation, and stabilization terms.

Analytic Quasi-Static Models. For the three settings of Figure 7.2, closed-form criteria replace simulation. At a point contact the admissible forces form the friction cone

$$\mathcal{F}_k = \{(f_{n,k}, \mathbf{f}_{t,k}) \mid f_{n,k} \geq 0,\ \|\mathbf{f}_{t,k}\| \leq \mu_k f_{n,k}\}, \tag{7.14}$$

which a polyhedral approximation turns into linear constraints. Mapping each contact's cone through its moment arm into wrench space and taking the convex hull gives the grasp wrench space, and the grasp is force-closed when the origin lies strictly inside it, Eq. (7.8) [265]; form closure is the stronger, purely geometric condition in which contact geometry alone prevents motion, and the Ferrari–Canny ϵ-metric (the distance from the origin to the hull boundary) grades grasp quality. For planar pushing the limit surface relates the applied wrench to the resulting slip, and its ellipsoidal approximation yields a motion cone of object velocities reachable from a given push, the basis of stable pushing and closed-loop push planning. For stacking, stability under gravity requires that the gravitational wrench be balanced by the supporting contacts [1195]: without friction, the center of mass must project into the support polygon; with friction, the contacts' friction cones must generate a wrench cone containing the gravitational wrench.

7.1.2.3 Benchmarks

PHYRE [1196] poses 2D physics puzzles in which an object must be placed so that simulated physics reaches a goal, scored by the number of attempts needed (AUCCESS) within and across puzzle templates; ShapeStacks [1195] labels 3D stacks as stable or unstable; PlasticineLab [1197] is a differentiable-physics suite for deformable manipulation; and the video suites IntPhys [1198], CLEVRER [1199], and Physion [680] test prediction and counterfactual judgment of physical outcomes, with ComPhy [1200] asking for hidden properties such as mass to be inferred from motion. Grasp quality under quasi-static analysis is evaluated on the grasping benchmarks of §7.1.1 (Cornell, YCB, GraspNet-1Billion). Metrics are predictive: position or trajectory error against the simulated outcome, success rate (stable stack, successful push or grasp), binary stability accuracy, and physical-plausibility terms such as momentum conservation and constraint violation.

7.1.2.4 Challenges and Trends

The parameters that matter most, mass, friction, elasticity, internal stability, are rarely observable. They must be inferred from interaction, from how far an object moves when pushed or how it responds to a collision; ComPhy [1200] tests exactly this, and most methods still either assume the parameters known, as classical simulation does, or approximate them in ways that fail outside the lab. Latent-dynamics learning [1201] and system identification [1202, 1203] are the two established routes.

The simulation-to-reality gap is the second limit. Small errors in restitution, micro-contact behavior, or unmodeled environmental factors compound once a policy leaves the engine, and fidelity trades against speed: high-fidelity simulation is too slow for large-scale training or real-time control, lightweight approximation too

inaccurate. Hybrid physics-plus-learning models, domain adaptation between simulated and real dynamics, and uncertainty-aware transfer each address one side of this trade-off; none closes it.

Perception is the third. Geometry, material, and object state come from noisy, incomplete sensing, and a small error in shape, pose, or contact location propagates into a large error in predicted force or stability. Systems that couple vision to a physics engine, Galileo [1204] and PhyNet [1203] among them, show that physical priors improve perception and that physical consistency can constrain it in return, but end-to-end pipelines in which the two regularize each other remain the exception.

7.1.3 Topological and Graph-based Reasoning

Topological and graph-based reasoning replaces continuous geometry with discrete connectivity, so that path planning, exploration, and coordination become combinatorial search over nodes and edges: places, corridors, and routes rather than metric trajectories. The abstraction is what makes large-scale, dynamic, or partially known environments tractable, where maintaining globally consistent geometry is hard but connectivity and reachability are stable. Figure 7.3 gives the abstract formulation.

7.1.3.1 Task Definition

Given a configuration space $\mathcal{C}$ with free subset $\mathcal{C}_{\text{free}} \subseteq \mathcal{C}$ (known a priori or inferred online from sensor streams), robot constraints, and optional priors such as floor plans or road networks, most methods build a planning graph $G = (V, E, \mathbf{w})$ in which vertices V are places, sampled configurations, or abstract states, edges E are feasible transitions, and $\mathbf{w} : E \to \mathbb{R}_{\geq 0}$ assigns metric or semantic costs (distance, time, energy, risk); in some methods the graph is implicit and expanded locally during search. The output is a collision-free path $\pi = \langle v_0 = s, \ldots, v_k = g \rangle$ of minimal accumulated cost, a topological map of place connectivity (often linked to local metric submaps), or, with several agents, conflict-free trajectories $\{\pi_i(t)\}_{i=1}^{N}$ that minimize makespan or total cost under vertex and edge conflict constraints.

7.1.3.2 Methodology

Visibility and Line-of-Sight Graphs. Connecting configurations that can see each other reduces shortest-path planning to graph search with interpretable waypoints. The visibility graph of Lozano-Pérez and Wesley [1205] links obstacle, start, and goal vertices and recovers exact Euclidean shortest paths in polygonal environments by Dijkstra or A*; the configuration-space construction via Minkowski sums [1206] extends it to robots of finite size by growing the obstacles instead of shrinking the robot; continuous-Dijkstra algorithms settle the complexity at $O(n \log n)$ [1207, 1208]. Grid planners bring line-of-sight into the search itself: Field D* interpolates across cell faces for smooth, replannable paths [1209], Theta* updates parents by visibility to produce near-Euclidean paths on the fly [1210], and Jump Point Search prunes symmetric expansions by jumping between visibility events [1211].

Clearance-Oriented Skeletons. Where safety matters more than length, free space is retracted to a skeleton that keeps connectivity and maximizes distance to obstacles. Voronoi diagrams and medial axes give maximal-clearance paths, with Fortune's sweepline [1212] for efficient planar construction; the generalized and hierarchical Voronoi graphs [1213, 1214] let a robot follow ridges with its sensors and build the clearance roadmap incrementally while detecting junctions and loops; and distance-transform updates [1215, 1216] maintain the skeleton in dynamic environments for real-time replanning.

Decomposition Graphs and Coverage. Discretizing space into cells trades geometric fidelity for simplicity and completeness. Occupancy grids [79, 1217] are probabilistic cell maps updated from range data on which planning is grid search; exact decompositions serve coverage, with boustrophedon decomposition [1218] partitioning free space into monotone cells and ordering them through an adjacency graph for complete, low-redundancy sweeps; and the coverage literature is systematized in [1219].

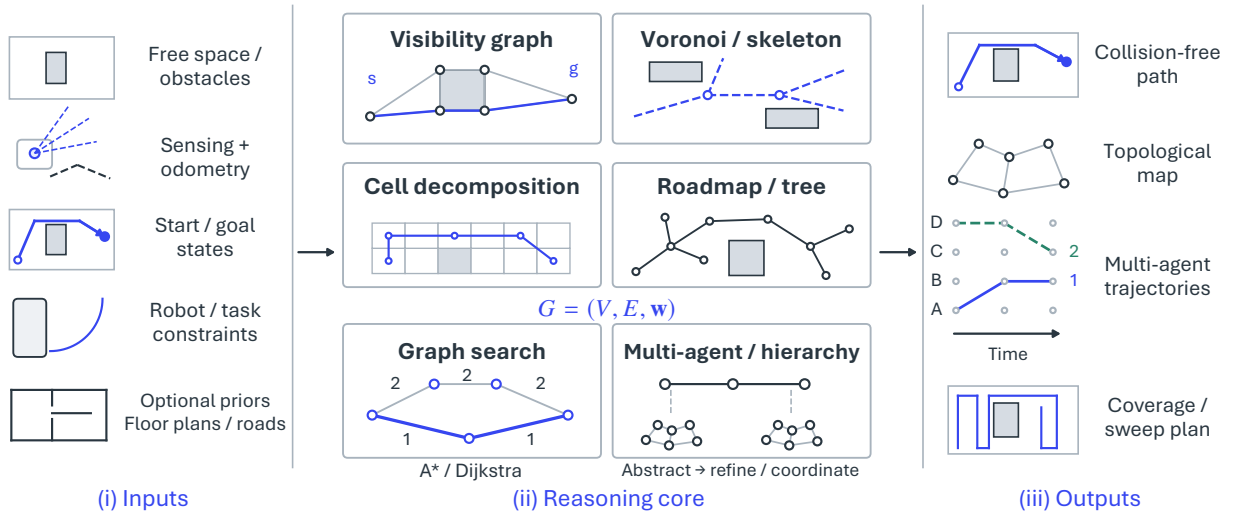


Figure 7.3: Abstract task formulation of topological and graph-based reasoning: (i) free space, sensing, start and goal states, and constraints are turned by (ii) construction and search over a graph $G = (V, E, w)$ into (iii) decision-ready graph artifacts such as collision-free paths, topological maps, coordinated multi-agent trajectories, and coverage plans. These are alternative task outputs. In the multi-agent example, rows denote vertices and columns denote successive time steps; horizontal segments represent waiting.

Sampling-Based Roadmaps and Trees. In high-dimensional configuration spaces, the graph is discovered by sampling. Probabilistic roadmaps [1220] sample and connect configurations in a preprocessing phase amortized over many queries and are probabilistically complete; RRT and RRT-Connect [1221, 1222] grow trees toward unexplored regions for single-query planning under kinematic and kinodynamic constraints; PRM* and RRT* [1223] add connection-radius and rewiring rules that make them asymptotically optimal; and FMT* and BIT* [1224, 1225] combine batch sampling with dynamic programming and heuristic search over increasingly dense implicit graphs for faster, anytime convergence.

Optimal, Any-Angle, and Incremental Graph Search. On a given graph, Dijkstra's algorithm and A* [70, 66] are the optimal single-source and heuristic searches; LPA* [1226] and D* Lite [1227] repair a shortest path incrementally as edge costs change; and the any-angle planners above (Field D*, Theta*, JPS) improve path quality without giving up the discrete search.

Multi-Agent and Hierarchical Graph Reasoning. With several agents on one graph, coordination is search over a time-expanded network. Conflict-Based Search [1228] splits the problem into a high-level search over conflict constraints and low-level single-agent planning and is complete and optimal; ICTS [1229] searches joint cost allocations backed by individual searches; ECBS and EECBS [1230, 1231] are bounded-suboptimal with focal heuristics and better conflict handling; and SAT or ILP compilations [1232, 1233] reach global optimality on moderate fleets at the cost of solver scalability. Hierarchical and multi-resolution planning [1234] and contraction hierarchies [1235] precompute abstraction layers that answer shortest-path queries on continental road networks in well under a millisecond, with standardized maps and instances [1236, 1237] for comparison.

7.1.3.3 Benchmarks

The MovingAI suite [1236, 1237] supplies standardized 2D maps from games, cities, and synthetic generators for single-agent search (path cost, runtime, memory, node expansions) and the MAPF instances on which CBS, ICTS, ECBS, and compilation-based solvers are compared by makespan, sum of costs, and success under time and memory limits, with the ICAPS MAPF tracker [1238] as the community leaderboard. OMPL's Planner Arena [1239] compares PRM, RRT, FMT, and BIT variants on success rate, planning time, path

quality, and anytime behavior, and the DIMACS shortest-path challenge [1240] evaluates query latency, preprocessing cost, and optimality on continental road graphs. Topological mapping and pose-graph SLAM are stressed by place recognition under appearance change: RADISH [1241] provides classic multi-sensor logs, and Oxford RobotCar [1242], NCLT [1243], and Nordland [1244] span seasons and weather, with KITTI and EuRoC (Section 6.2.4) for urban and indoor pose graphs.

7.1.3.4 Challenges and Trends

Graph density sets the trade-off: sparse graphs lose connectivity or optimality, dense ones lose real-time response, and the problem sharpens in high-dimensional and lifelong settings [1223, 1227]. Perceptual aliasing, drifting pose graphs, and dynamic obstacles complicate planning under partial information [54, 1245], and on the topological SLAM benchmarks the binding difficulty is not search but appearance robustness, with cross-season and cross-weather queries far harder than same-condition retrieval. Multi-agent coordination strains conflict resolution and time-expanded representations as fleets grow, while fairness and deadlock avoidance must be kept [1228, 1233, 1238].

The directions are structured rather than end-to-end: hierarchical and multi-resolution planning that abstracts and refines [1234, 1235], risk-aware and formally verified planning under uncertainty [1246, 1247], bounded-suboptimal and distributed coordination for responsiveness [1230, 1231], and neuro-symbolic methods that enrich the graph with learned perception, heuristics, and semantics [1248].

7.1.4 Constraint-based Spatial Solving

Many robotics problems are not about finding *a* plan but one that *must* satisfy strict requirements: no collisions, no joint-limit violations, stable contacts, correct timing, sometimes guaranteed dynamics or safety margins. Constraint-based spatial solving states the problem as a collection of constraints, what must hold and what must not, and hands it to a solver that either returns a satisfying solution or proves that none exists. Compared with sampling- and heuristic-driven planning, this buys certificates of feasibility, an explanation of why a candidate fails, and, under an explicit model, optimality or bounded-suboptimality guarantees, which is what manipulation in clutter, multi-robot coordination in shared space, and assembly demand. Figure 7.4 gives the abstract formulation.

7.1.4.1 Task Definition

Let $x = (\mathbf{q}_{1:T}, \mathbf{u}_{1:T}, \mathbf{z})$ collect continuous states and controls together with discrete mode and logic variables $\mathbf{z}$. Constraint-based solving seeks

$$\text{find } x \in \mathcal{X} \quad \text{s.t.} \quad g_i(x) \leq 0, \;\; h_j(x) = 0, \;\; \phi(\mathbf{z}, x) = \top, \tag{7.15}$$

where the inequalities g_i encode geometric and dynamic feasibility (collision, joint, and dynamics limits), the equalities h_j encode kinematics and contact conditions, and ϕ encodes task logic such as preconditions and effects. The optimization variant minimizes an objective $f(x)$ as well. The output is a feasible x (a collision-free and dynamically admissible trajectory, a task-robot schedule, an assembly sequence), with an objective value or optimality gap when optimizing, or a certificate that no such x exists.

7.1.4.2 Methodology

The solver families differ in the variables and constraints they handle and the guarantees they return; practical systems are rarely pure, combining discrete reasoning over actions and modes with continuous optimization over motions and contacts, and calling fast geometric oracles (collision checking, inverse kinematics) inside the solver loop.

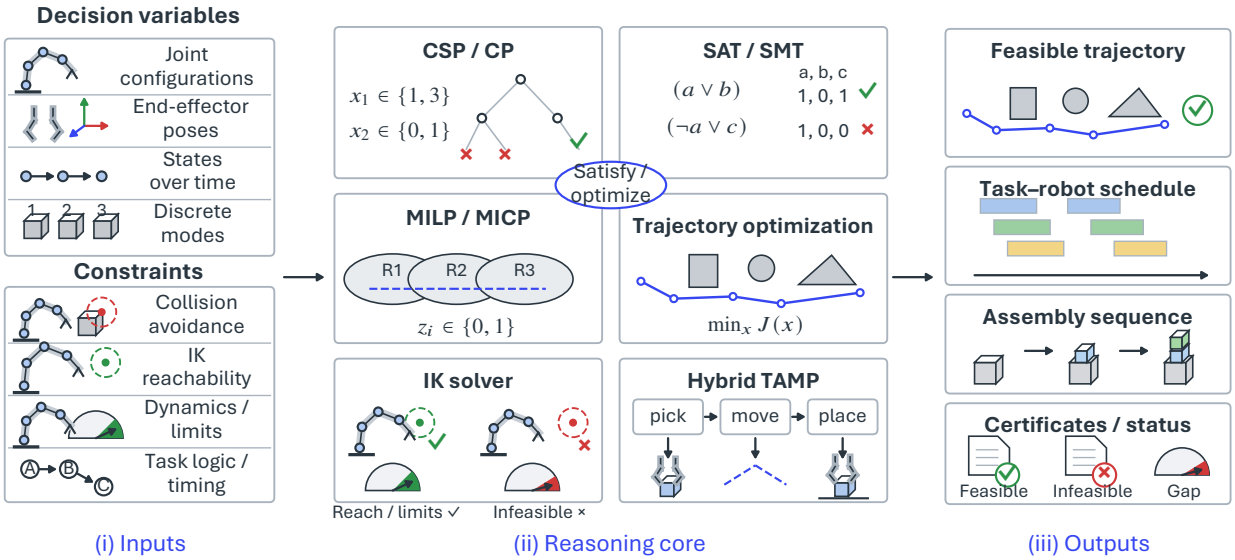


Figure 7.4: Abstract task formulation of constraint-based spatial solving: (i) decision variables and constraints, (ii) a solver core that satisfies or optimizes them, and (iii) feasible trajectories, schedules, and assembly sequences, with feasibility or infeasibility certificates and optimality-gap bounds where supported by the solver and problem formulation. The output types are alternatives; failure of a local solver alone is not an infeasibility proof.

Constraint Programming (CSP/CP). When the decisions are combinatorial (which object, which placement, which grasp) but each must pass a geometric test, constraint programming encodes the choices as variables under constraints and interleaves symbolic branching with propagation, pruning, and geometric checks. Lozano-Pérez and Kaelbling [1249] cast task-and-motion planning as constraint satisfaction over action skeletons, pruning symbolic branches as soon as a geometric feasibility test fails; Lagriffoul *et al.* [1250] propagate and filter continuous domains before any expensive collision or IK call, so that many failures become cheap inconsistencies; and Dantam *et al.* [1251] interleave discrete search with incremental feasibility queries that reuse information across related constraints.

SAT/SMT Compilations. Compiling planning and feasibility into Boolean formulas, or into formulas over richer theories such as real arithmetic, puts clause learning, incremental solving, and conflict-driven pruning to work. dReal [1252] decides nonlinear real arithmetic up to a numerical tolerance (δ-completeness), which gives certified answers where exact decidability is impossible; SMT-based planning for hybrid systems [1253] reasons first over an abstraction and adds constraints as counterexamples appear; SAT encodings of multi-agent pathfinding [1232] obtain optimal schedules on time-expanded graphs (the MAPF problem itself is treated in §7.1.3); and SMT over linear real arithmetic extends the encodings to continuous time and to robustness against execution delays [1254, 1255].

Mixed-Integer Formulations (MICP/MILP). When discrete mode choices (which region, which contact mode, which precondition) interact with continuous geometry and dynamics, binary variables encode the nonconvex choice and convex constraints capture feasibility within a mode. IRIS [1256] inflates large convex obstacle-free regions that serve as certified safe sets, turning collision avoidance into membership constraints with explicit margins; Graphs of Convex Sets [1257] compile motion planning through clutter into a shortest-path problem over convex nodes and edges that convex optimization, with mixed-integer selection where needed, solves with certificates; and later work constructs convex sets directly in configuration space and in space-time [1258, 1259]. In each case a globally nonconvex feasibility problem is replaced by a sequence of convex ones without giving up the guarantee.

Trajectory Optimization. Even with the task fixed, an executable motion must satisfy kinematic limits, collision avoidance, smoothness, and often dynamics, which is a continuous, usually nonconvex, constrained optimization whose local optimum comes with optimality conditions and sensitivities. CHOMP [1260] deforms an initial path by functional gradient descent on a signed-distance field; STOMP [1261] replaces the gradient with stochastic perturbations to handle non-smooth costs and local minima; TrajOpt [1262] solves a sequence of convex programs with hinge-loss collision penalties; and GPMP2 [1263] writes planning as inference on a sparse factor graph with Gaussian-process priors, which gives scalability and incremental replanning. Increasingly these optimizers run inside the safe sets of IRIS and GCS, which supply initialization, constrain the search, or verify the result.

Inverse Kinematics as a Constraint Subproblem. "Is there a joint configuration that reaches this pose under limits and collisions?" is the most frequent feasibility query in any hybrid planner, so IK solvers are built as fast oracles. IKFast [1264] derives closed-form solutions and generates optimized code for repeated real-time queries; TRAC-IK [1265] combines complementary numerical strategies to raise the solve rate and avoid stalling the planner above it; and global IK by mixed-integer programming [1266] returns certifiable solutions for full chains when a feasibility proof is needed.

Hybrid Logic–Geometric Planning. In long-horizon problems the discrete decision (what to do next) and the continuous one (how to do it) cannot be separated, and task-and-motion planners close the loop between them. Logic-Geometric Programming [1267, 1268] couples symbolic action sequences to bound-constrained nonlinear programs through a multi-bound tree search, so that continuous feasibility and cost decide which symbolic branches are worth expanding; PDDLStream [1269] integrates black-box geometric generators (grasps, placements, motions) into symbolic planning by optimistic refinement, requesting continuous witnesses only when needed while remaining complete under suitable stream assumptions; R-LGP [1270] biases the search with reachability guidance; and COAST [1271] uses constraints to communicate between task planner and stream generators, cutting wasted geometric proposals by an order of magnitude in planning time.

7.1.4.3 Benchmarks

Evaluation asks whether a method returns solutions that satisfy strict feasibility conditions and, increasingly, whether it can show that it did (margins, certificates, optimality gaps) under a comparable oracle budget of collision checks and IK calls. OMPL's benchmark suite and Planner Arena [1272] score planners on standardized problems with feasibility diagnostics and solution cost; the platform-independent TAMP benchmarks of Lagriffoul *et al*. [1273] specify symbolic operators together with IK and collision feasibility so that a jointly feasible task-motion plan is required; and the assembly benchmarks RAMP [1274] (real-world assemblies requiring geometric fit, stability under gravity, and grasp or fixture feasibility), ASAP [1275] (large-scale 3D assembly sequences validated by physics-based feasibility checks), and FurnitureBench [1276] (long-horizon real-world furniture assembly scored by success rate and completion time) stress feasibility across geometry, stability, and sequencing at once. Multi-robot coordination under space-time collision constraints uses the MAPF benchmarks of §7.1.3.

7.1.4.4 Challenges and Trends

The central difficulty is integrating a discrete task structure with continuous, highly nonconvex feasibility at scale and with certificates. Task-level choices (modes, grasps, placements, contact sets, schedules) interact combinatorially with geometric and dynamic constraints and fragment the feasible set; mixed-integer formulations grow expensive with the number of binary decisions [1256, 1258], trajectory optimizers return locally feasible motions that may violate a symbolic assumption such as the intended contact mode [1260, 1262], and symbolic planners propose sequences that are geometrically unrealizable. Information flows poorly in both directions: continuous solvers rarely return explanations (active sets, separating hyperplanes, δ-unsat

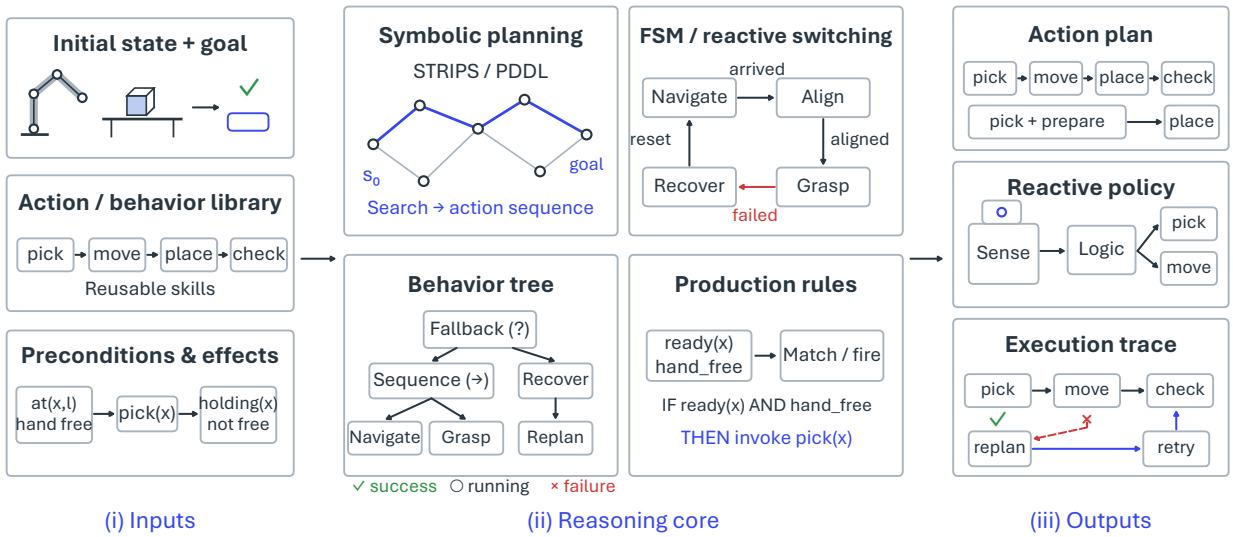


Figure 7.5: Abstract task formulation of rule-based and programmatic reasoning: (i) symbolic task structure, (ii) four representations for executing decision logic, and (iii) executable plans, reactive policies, and monitored execution traces with replanning after a disturbance.

cores) that the symbolic layer could use to prune, and symbolic abstractions are often too coarse to keep the continuous subproblems tractable [1269, 1267]. On the hybrid and assembly benchmarks this coupling, not raw search speed, is what separates planners: those that interleave symbolic branching with continuous oracles waste fewer feasibility checks and degrade more gracefully across domains.

Assurance under uncertainty is the second difficulty. Real systems need machine-checkable feasibility and fast infeasibility proofs, but nonlinear dynamics, contact, and hybrid interactions make strong certificates hard; convex surrogates and bounded-horizon relaxations buy speed at the price of completeness or tightness; execution requires real-time replanning with expensive oracles in the loop; and robustness to noise and model error calls for explicit risk and temporal-logic constraints [1277, 1278, 1279]. Evaluations still under-report certificate quality, oracle counts, and failure explanations.

The trend is toward structure with certificates: graphs of convex or geodesically convex safe sets that reduce global nonconvex feasibility to sequences of convex subproblems with explicit proofs [1257, 1259], coupled to trajectory-optimization and MPC back-ends that return sensitivities and constraint activations for the search above them; chance constraints, temporal logic, and control barrier functions for persistent safety [1277, 1279]; GPU acceleration for real-time replanning; and learning that guides branching, cutting, or warm starts without discarding the guarantee [1270, 1271].

7.1.5 Rule-based and Programmatic Reasoning

Rule-based and programmatic reasoning is the explicit, human-auditable layer that decides *when* an action is allowed, *in what order* actions happen, and *what to do next* when conditions change, using symbolic operators, transition rules, or program-like control structures. In a robot it is the supervisory layer that sequences reusable skills (navigate, grasp, insert) and invokes geometric solvers or learned policies only when their preconditions hold, which is what makes a decision explainable (why the detour when the corridor was blocked), verifiable, and able to carry commonsense and safety constraints through a long task. Figure 7.5 gives the abstract formulation.

7.1.5.1 Task Definition

A symbolic planning problem is a tuple $\Pi = \langle \mathcal{S}, \mathcal{A}, s_0, G \rangle$ with symbolic states $\mathcal{S}$, an operator library $\mathcal{A}$, an initial state s_0, and a goal G; each operator $a \in \mathcal{A}$ carries a precondition and effects,

$$a = \big(\mathrm{pre}(a),\ \mathrm{eff}^{+}(a),\ \mathrm{eff}^{-}(a)\big), \qquad s' = \big(s \setminus \mathrm{eff}^{-}(a)\big) \cup \mathrm{eff}^{+}(a), \tag{7.16}$$

and is applicable in s iff $\mathrm{pre}(a) \subseteq s$. The planning output is a sequence $\pi = \langle a_1, \ldots, a_k \rangle$ whose successive application to s_0 reaches a state satisfying G; the reactive variant instead returns a policy μ that maps the current observation and symbolic state to a behavior. Depending on the formalism, the inputs also carry resources, temporal constraints, and exogenous events.

7.1.5.2 Methodology

The formalisms split by whether they *plan* (search for a correct sequence before acting) or *react* (choose a behavior continually at run time); deployed systems usually do both, with a planner proposing the sequence and an execution structure (FSM, behavior tree, rule engine) monitoring conditions and triggering recovery.

Classical Symbolic Planning (STRIPS/PDDL and Heuristic Search). STRIPS [67] fixed the representation: planning as search over symbolic states with action schemas, so that a plan can be checked against the model and the continuous motion layer sits behind a clean abstraction boundary. PDDL [1280] standardized domain and problem specification so that planners could be compared across navigation, logistics, and assembly, with extensions (ADL, numeric fluents, temporal constructs) that capture robotic requirements without abandoning explicit semantics. Graphplan [1281] builds a planning graph whose mutual-exclusion constraints prune incompatible action combinations early, and FF [1282] made forward search fast with relaxed-plan heuristics that ignore delete effects, which capture ordering constraints such as unstacking before stacking in object-rich domains. Task-and-motion planning (§7.1.4) extends these planners with geometric feasibility checks [1283, 1284, 1285]; the common pattern is to keep the symbolic reasoning explicit, delegate feasibility to motion and IK solvers, and wrap the plan in an execution structure with monitors and recovery.

Finite-State Machines and Reactive Mode Switching. An FSM switches among a few well-defined behaviors (navigate, align, grasp) with explicit transition guards, which makes it transparent and fast but prone to state explosion as contingencies multiply. Statecharts [1286] add hierarchy and concurrency, so that complex controllers are nested reusable submachines with precise semantics; Brooks' subsumption architecture [81] layers behaviors so that higher ones override lower ones, a blueprint for composing navigation and avoidance without global planning, systematized as behavior-based robotics [62]; teleo-reactive programs [1287] evaluate conditions continuously and fire the highest-priority satisfied rule; and ROS tooling such as FlexBE [1288] provides hierarchical, event-driven state machines at deployment scale.

Behavior Trees. A behavior tree re-evaluates its conditions on every tick and chooses among structured alternatives (sequence, fallback, parallel), so that long-horizon execution survives disturbances without committing to one fixed sequence; subtrees encapsulate reusable routines, guards and fallbacks trigger immediate recovery, and the composition semantics fit perception and motion skills. BTs originated in game AI [1289]; Marzinotto *et al.* [1290] formalized their semantics and their relation to other switching structures, and Colledanchise and Ögren [1291, 1292] showed which compositions preserve safety and robustness and consolidated the design principles. Compiling or wrapping a symbolic plan into a BT keeps the nominal task structure while exposing runtime guards and recovery branches [1293]; ARIAC systems pair PDDL planning with BT execution monitors to handle failures, occlusions, and defective parts [1294]; and event-driven BTs [1295] react to non-periodic stimuli rather than waiting for the next tick.

Logic-based Execution and Production Rule Systems. A production system keeps a working memory of facts and fires forward-chaining rules when their conditions hold, which suits event-condition-action policies, mode switching, and safety enforcement (no-go zones, mutual exclusions, priorities) and depends on fast incremental matching. OPS5 [1296] set the model; the Rete algorithm [1297] caches partial matches so that large rule sets update in milliseconds as facts change; CLIPS [1298] made the toolchain portable; and Soar [1299] integrated large rule bases with planning and execution in one architecture. In robotics, KnowRob [1300] and CRAM [1301, 1302] join symbolic knowledge to rule-driven execution for reasoning about actions, effects, and everyday manipulation, and LTLMoP [1303] compiles temporal-logic specifications into correct-by-construction reactive controllers whose safety and sequencing constraints are explicit at the program level.

7.1.5.3 Benchmarks

The International Planning Competition supplies the canonical PDDL domains and problems across classical, temporal and numeric, probabilistic, and HTN tracks, compared by plan quality, makespan, and runtime [1280], with `planning.domains` [1304] mirroring and extending the sets for reproducible evaluation. Long-horizon embodied suites test execution under partial observability and variation: BEHAVIOR-100 and BEHAVIOR-1K on OmniGibson [1305, 1306] specify household activities at the predicate level, ALFRED [1307] and TEACh [1308] score instruction-conditioned action sequences, and RoboTHOR [962] and the Habitat challenge [204] pair high-level decision logic with navigation policies under SPL. ARIAC [1294] injects faulty parts, occlusions, and failures into kitting and assembly, and FurnitureBench [1276] and RoboCup@Home [1309] stress recovery and explainability on real tasks; qualitative solvers over RCC-8 [76] and Allen relations [1310] are tested on consistency problems, and composite pipelines reuse the TAMP and OMPL harnesses of §7.1.4.

7.1.5.4 Challenges and Trends

The results on these benchmarks divide along the closed-world boundary. On IPC-style domains with fully specified preconditions and effects, classical planners remain strong; on long-horizon embodied tasks the limit is perception noise and grounding ambiguity rather than search, and a logically applicable action may be geometrically unrealizable because of a collision, an IK failure, or a missing grasp. Hand-authored operators, rules, and recovery logic also grow combinatorially with object count and contingency. Integrated task-and-motion planning (§7.1.4) answers the coupling problem by interleaving symbolic search with feasibility checks or compiling symbolic commitments into optimization back-ends [1249, 1269, 1267], and hierarchical abstraction and compositional skill models contain the search while keeping it auditable [1284, 1285].

Robustness to noise and nonstationarity is why the industrial and instruction-conditioned benchmarks reward compositions rather than single formalisms: a behavior tree around a symbolic plan makes disturbances and recoveries explicit through runtime guards, fallbacks, and safety-preserving composition [1290, 1291, 1292], and Rete-style engines [1297] supply fast event-condition-action policies over streaming facts, the pattern ARIAC entries converge on [1294].

Learned perception and language models increasingly propose the high-level procedure or heuristic while the symbolic structure stays as the safety shell that enforces preconditions, effects, and constraints: SayCan [1311] grounds language in robot skills by value-based scoring, and plan-to-BT compilation [1293] exposes parallelism, guards, and recovery at run time. Domain modeling, heuristic guidance, and recovery synthesis are being automated piece by piece while the programmatic structure is retained for auditing, debugging, and verification.

7.1.6 Causal Reasoning

Causal reasoning moves from describing *what* happened to explaining *why* it happened and predicting *what would have happened* under a different action. For a robot the relevant outcomes are chains of spatial events (a collision, a delay, a deadlock, an unstable grasp, a near-miss in shared space), and making the causal assumptions explicit, rather than leaving them implicit in "actions cause state changes", is what lets the system attribute a failure to its causes rather than its symptoms, evaluate counterfactual alternatives such as a different approach direction or timing, and, with several agents, separate primary causes from downstream effects when choosing an intervention. Figure 7.6 gives the abstract formulation.

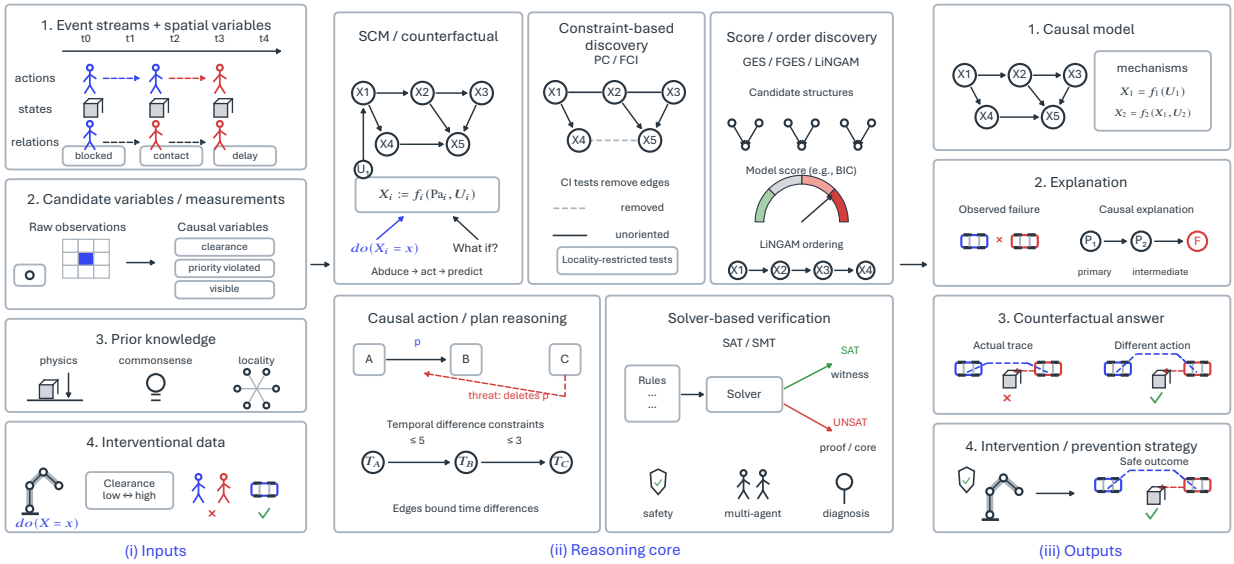


Figure 7.6: Abstract task formulation of causal reasoning for spatial intelligence: (i) event streams, candidate variables, prior knowledge, and interventional data feed (ii) methods that discover and verify causal structure, yielding (iii) causal models, explanations, counterfactual answers, and intervention or prevention strategies. The method families serve different roles rather than forming a serial pipeline. Verification returns a satisfying witness or an unsatisfiability proof/core where supported; counterfactual scenes keep the objects fixed while changing the blue agent's action.

7.1.6.1 Task Definition

The inputs are spatiotemporal event streams (agent actions, object states, contact and visibility relations, environment variables) mapped to causal variables by measurement functions (`in_contact`, `blocked`, `priority_violated`), optional prior knowledge from physics or commonsense, and, when available, interventional data in which a controllable variable was set deliberately. A structural causal model over these variables,

$$\mathcal{M} = \langle \mathcal{G}, \{f_i\}, P(U) \rangle, \qquad X_i := f_i(\mathrm{Pa}(X_i), U_i), \tag{7.17}$$

consists of a directed acyclic graph $\mathcal{G}$, a mechanism f_i that generates each variable from its parents $\mathrm{Pa}(X_i)$ and exogenous noise U_i, and a noise distribution $P(U)$ [1312]. The outputs are the model itself when it must be learned, the explanation of an observed outcome (which variables caused it), the answer to an interventional query $P(Y \mid do(X = x))$ or a counterfactual one (Y_x for a unit whose actual trace was observed), and an intervention or prevention strategy that changes controllable causes to reach a desired spatial outcome.

7.1.6.2 Methodology

The methods differ in where the causal structure comes from (assumed, learned from data, or written as domain knowledge) and in what they are asked (intervention effects, counterfactuals, responsibility,

verification); in spatial domains they also differ in how they use locality, temporal precedence, and network connectivity as inductive bias.

Structural Causal Models and Counterfactuals. An SCM makes mechanisms explicit ("clearance determines collision risk", "traffic density sets delay") and separates intervention from observation: the $do(\cdot)$ operator replaces a structural equation instead of conditioning on a value, which is what is needed when, say, congestion and delay are both driven by time of day. Spatial pipelines typically start from a graph fixed by temporal precedence and locality priors (edges among nearby regions, interacting agents, adjacent cells), estimate the mechanisms from data, and validate them against implied conditional independences and refutation tests [1313]. Counterfactuals follow the abduction-action-prediction recipe: infer the exogenous noise consistent with the observed trace, apply the hypothetical intervention, predict the outcome [1312]. DoWhy and DoWhy-GCM [1314, 1315] implement identification, estimation, and refutation for this workflow on real spatiotemporal data.

Causal Discovery: Constraint- and Score-Based. When the graph is unknown but observational data are plentiful, constraint-based methods read structure off conditional independences: the PC algorithm builds the skeleton by independence tests and orients what it can, returning a Markov equivalence class, and FCI [1316] handles latent confounding and selection bias by returning a partial ancestral graph. Score-based methods instead search graph space under an objective such as BIC: GES [1317] greedily adds and then removes edges over equivalence classes, and its forward-backward structure often recovers mesoscale organization such as bottlenecks that drive downstream congestion. Spatial data strain both. Large conditioning sets are expensive and statistically weak, so practice restricts conditioning to a radius or k-hop neighborhood, forbids edges that violate adjacency or reachability priors [1318], and uses order-independent variants (PC-stable [1319]) whose high-dimensional consistency is understood [1320]; ambiguous orientations are then resolved by targeted interventions. causal-learn [1321] and the TETRAD / Py-Tetrad implementations of FGES [1322, 1323] provide these pipelines at facility and city scale.

Independent-Noise Discovery (LiNGAM). In linear-Gaussian settings many edge directions are unidentifiable from independences alone. LiNGAM [1324] assumes a linear structural model with independent non-Gaussian noise, under which the causal ordering is identifiable, and DirectLiNGAM [1325] estimates it by regression rather than ICA. Spatiotemporal signals need care because autocorrelation and smoothing push the noise toward Gaussian; temporal blocking, prewhitening, and locality constraints are the usual remedies, and the recovered ordering then scaffolds an SCM for intervention analysis.

Causal Action Languages and Plan-Space Causal Links. Much causal structure in robotics is designed knowledge rather than discovered: actions have direct and indirect effects, and the system must reason about change under incomplete information. The action language C$^+$ [1326] writes causal theories of action with inertia and nonmonotonic semantics, compactly describing how fluents such as `occupied(c)` or `visible(a,o)` evolve; CCALC [1327] compiles them to SAT or ASP so that prediction, diagnosis, and plan synthesis use the same solvers; and multi-valued fluents and external geometric predicates (collision checks, IK) are being integrated without abandoning the formal semantics [1328]. In plan space, partial-order causal-link planning [1329] represents a plan as partially ordered actions plus links $A \xrightarrow{p} B$ (action A establishes precondition p for B), which makes threats explicit and repairable by promotion, demotion, or binding constraints; UCPOP [1330] extended the representation to ADL and conditional effects and VHPOP [1331] added heuristics and reachability analysis. In spatial domains a causal link is often a physical dependency, "unblock door" enabling "move cart", which bridges symbolic causality and spatial feasibility.

Execution and Verification. A causal plan must be realized in time and checked. Simple temporal networks [1332] enforce difference constraints between events with polynomial consistency and dispatchable forms [1333], and STNUs with dynamic-controllability analysis [1334] handle contingent durations, so that ordering and clearance constraints stay satisfiable under bounded disturbance. The SAT/SMT machinery of

§7.1.4 enters here as an explanation device: when a schedule is unsatisfiable, the unsat core is a minimal set of conflicting commitments, a concrete statement of which decisions caused the failure [1232, 1233]; and for hybrid dynamics δ-complete SMT (dReal) [1252] and dReach [1335] verify safety and synthesize interventions with numerically sound proofs or counterexamples.

7.1.6.3 Benchmarks

No single benchmark covers the causal pipeline, and the suites stress different links of it. CLEVRER [1199] pairs synthetic collision videos with descriptive, explanatory, predictive, and counterfactual questions, and CLEVRER-Humans [1336] adds human-annotated causal event graphs against which model attributions can be compared; the video suites of §7.1.2 (PHYRE [1196], Physion and Physion++ [680, 1337]) score intervention selection and latent-property inference, and CRaFT [1338], ContPhy [1339], NExT-QA [1340], AGQA [1341, 1342], and SUTD-TrafficQA [1343] ask causal questions in natural language about real and traffic video. Embodied platforms with predicate-level state (BEHAVIOR-1K, ALFRED, iGibson 2.0 [1344], ManiSkill2 [1345], RLBench [1346]; §7.1.5) test whether inferred dependencies improve intervention selection and recovery; the MAPF suites and Flatland [1347] (§7.1.3) test reasoning over space-time constraints in which one agent's decision causes another's delay; and ARCH-COMP [1348] with the dReal / dReach problem sets [1252, 1335] evaluate causal safety reasoning on hybrid dynamics by proof or counterexample and solve time. Discovery itself is evaluated on recovery of the Markov equivalence class under faithfulness, with scalability bounded by conditioning-set size.

7.1.6.4 Challenges and Trends

Identification is brittle. Latent confounding, selection bias, and temporal aggregation can make a query non-identifiable from logs, and estimates become unstable when locality and precedence assumptions fail [1312, 1313]; constraint- and score-based discovery is expensive on dense telemetry with strong short-range dependence and sensitive to test strength, ordering, and regularization [1316, 1319, 1320, 1317]; and LiNGAM's non-Gaussianity is exactly what smoothed trajectories and aggregated traffic signals lack [1324, 1325]. The response is structure-aware discovery that constrains search by topology and locality and then resolves the remaining ambiguity by targeted robot interventions, connecting learning to actuation [1316, 1317].

Causal reasoning is only as good as its variables, and in embodied settings contact, occlusion, blockage, and object state must be grounded from noisy perception. The errors of that mapping propagate into SCM estimates and counterfactual conclusions [1313, 1314]. Knowledge-enabled frameworks such as KnowRob and CRAM (§7.1.5) show the shape of the interface, perception populating an explicit, interpretable model whose parameters are then estimated statistically, and the trend is toward such perception-to-symbol interfaces that tolerate sensor noise [1300, 1301, 1302].

Between a causal recommendation and a safe execution there is a verification-synthesis gap: the recommendation must become a temporally controllable schedule and a certified controller, and execution uncertainty invalidates a symbolic causal plan unless temporal networks and hybrid verification are integrated with it [1332, 1334]. SCM toolchains are therefore being embedded in decision stacks so that counterfactual policies compile into action-language or plan-space structures and then into dispatchable networks [1315, 1326, 1334], with SAT/SMT certificates and unsat-core explanations supplying the accountability [1232, 1252, 1335]; evaluation that scores explanation quality, interventional effectiveness, and certification cost on the same task is what the field still lacks.

7.2 Learning-based Spatial Reasoning



EARNING-BASED SPATIAL REASONING refers to the class of methods that acquire and perform spatial reasoning from data, rather than from the hand-authored geometric, mechanical, or symbolic

models surveyed in §7.1. The six subsections that follow are ordered by the *locus of the structural prior*: it begins inside the learned representation (latent feature interaction, explicit programs, hard-coded symmetries, optimization layers in the forward pass), moves to the agent's action-conditioned behavior (reinforcement learning with spatial inductive biases), and ends with joint decision-making across coordinated agents.

7.2.1 Implicit Neural Spatial Reasoning

We define *implicit neural spatial reasoning* as the family of neural approaches in which spatial relations are encoded in learned feature representations, attention patterns, latent object interactions, or architectural inductive biases, rather than specified through hand-written rules, symbolic programs, or fully explicit relational graphs. *Implicit* does not mean that these models never use boxes, masks, object proposals, or task-level labels; it means that the reasoning procedure itself is never given as a symbolic program, so the model must learn spatial dependencies from end-to-end or weakly structured supervision. Several families below (latent scene graphs, capsule networks) expose partly inspectable structure, yet remain implicit in that their relational structure is learned or emergent rather than supervised. Figure 7.7 shows the representative tasks.

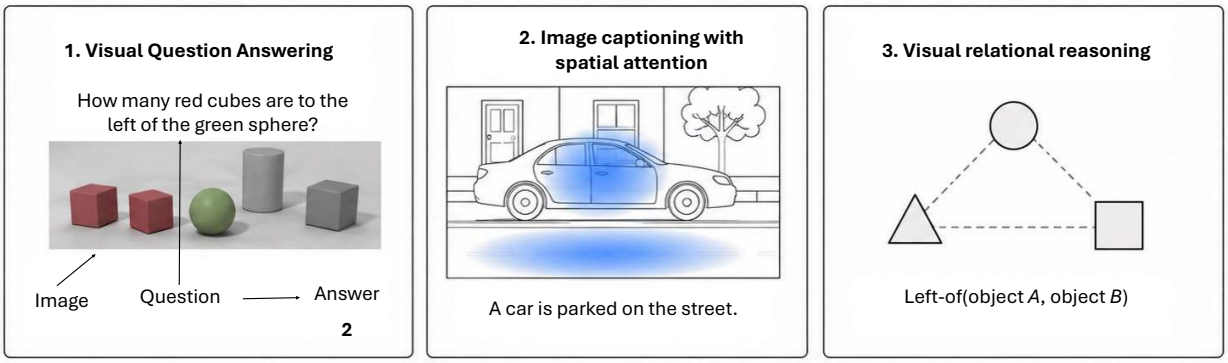


Figure 7.7: Representative tasks in which spatial relations must be inferred without explicit symbolic supervision: (1) visual question answering over compositional scenes, (2) image captioning with spatial attention, and (3) visual relational reasoning.

7.2.1.1 Task Definition

Given one or more images I, optionally paired with a language query or task goal q (and possibly boxes, masks, or proposals as features), an implicit reasoner computes a latent spatial representation and a task output,

$$z = f_\theta(I, q), \qquad y = h_\psi(z), \tag{7.18}$$

where z encodes the relations relevant to the task and y is an answer label, a caption, or a relation label. Supervision reaches the model only through y; the relational structure inside z is never annotated, so it must emerge from end-to-end training. The methods below differ in the mechanism by which z is formed (Figure 7.8).

7.2.1.2 Methodology

Relational Aggregation over Latent Objects. The most direct approach treats a set of feature vectors as latent "objects" and learns to aggregate their interactions. **Relation Networks (RN)** [1349] score every pair: given

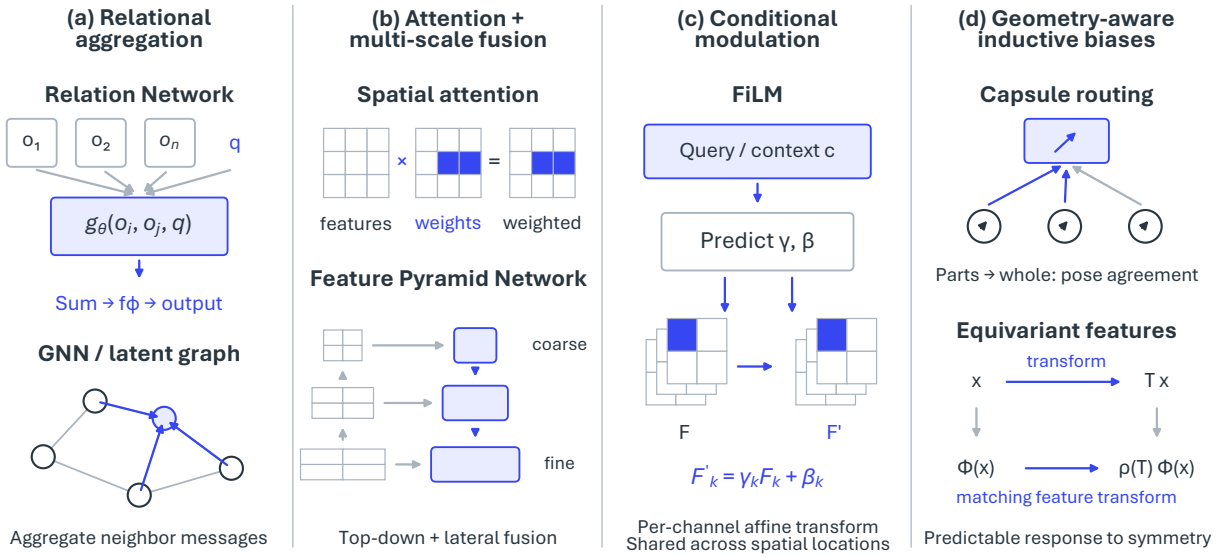


Figure 7.8: Four mechanisms for learning spatial representations: (a) relational aggregation (Relation Network, GNN message passing), (b) spatial attention with multi-scale fusion (FPN), (c) conditional modulation (FiLM), and (d) geometry-aware inductive biases (capsule routing, equivariant features). Blue marks selected features, messages, and operations. The mechanisms can be combined in one model, and attention and FPN support spatial representation rather than forming complete reasoning procedures.

objects $O = \{o_1, \ldots, o_n\}$ (*e.g.*, feature vectors from spatial regions of a CNN map) and an optional query q,

$$\mathrm{RN}(O, q) = f_\phi\left(\sum_{i,j} g_\theta(o_i, o_j, q)\right), \tag{7.19}$$

where g_θ produces a query-conditioned relation embedding for each pair and f_ϕ maps the sum to the task output. Considering all pairs forces the network to learn pairwise spatial and semantic relations, at $O(n^2)$ cost in the number of objects. **Graph neural networks and latent scene graphs** generalize this from a complete graph to a learned, sparser topology, with proposals or feature regions as nodes and learned edges as spatial dependencies; iterative message passing for scene-graph generation [798] alternates between refining node features and updating relation embeddings, learning the topology end-to-end. Such models are the clearest *semi-explicit* case in this subsection: the graph is inspectable and often trained with relation annotations (*e.g.*, from Visual Genome), yet the intermediate relational representation is learned rather than specified by a program, and its quality is bounded by the upstream proposal stage.

Spatial Attention and Multi-Scale Feature Fusion. A second family controls *where* and *how strongly* the model attends across the spatial layout, and fuses evidence across scales, rather than scoring all pairs. CNN spatial attention weights locations in a feature grid, from the hard and soft attention of Xu et al. [1350] for captioning and grounded VQA [1351] to SCA-CNN's joint spatial and channel attention [1352]; vision transformers make relating image parts intrinsic through self-attention over patches [142], with locality-aware variants (Swin [469], PVT [1353], Twins [1354]) trading global receptive field for cost; and Feature Pyramid Networks [597] add a top-down pathway so that a relation coupling a fine object with a coarse region (a car "on" a street) can be read from one map. These designs are representation backbones that preserve and aggregate spatial information for the reasoning built on top of them, not reasoning modules in themselves.

Conditional Feature Modulation. A third family conditions one representation on another rather than aggregating or attending. In **FiLM (Feature-wise Linear Modulation)** [1355], a conditioning vector (*e.g.*, an encoded query or global context) predicts affine parameters (γ, β) that are applied feature-wise to a target

map, so the same backbone can be steered per task. For a query like "What is to the left of the green cube?", a FiLM layer can learn to amplify features to the left of regions identified as the green cube, making reasoning adaptive and context-dependent.

Geometry-Aware Inductive Biases. A fourth family bakes geometric regularities into the architecture, trading data-driven flexibility for sample efficiency where the relevant geometry is known. **Capsule Networks** [1356] replace scalar activations with vector-valued capsules whose orientation encodes instantiation parameters (pose, deformation) and whose magnitude encodes presence; *dynamic routing* decides how lower-level capsules compose into higher-level ones by agreement in predicted pose, a stronger part-whole bias than translation-only convolution, though capsules have been hard to scale beyond small digit and shape datasets. Equivariant and steerable convolutions [1357], which commute with a chosen transformation group, are the subject of §7.2.3; both share the cost that the symmetry must be chosen in advance.

7.2.1.3 Benchmarks

CLEVR [1358] is the canonical benchmark, pairing rendered scenes with multi-step relational questions whose ground-truth program structure is known; Sort-of-CLEVR [1349] isolates a controlled relational vocabulary. The decisive stress tests are the *systematic-generalization* splits: CLEVR-CoGenT [1358] holds out attribute-composition combinations, and GQA [839] scales compositional questions to real images. Open-domain VQA v2 [1359] and Visual7W [1351] probe whether implicit reasoning transfers beyond synthetic templates to natural images and free-form language. On CLEVR, RN reaches 95.5% and FiLM 97.7%; Table 7.1 in §7.2.2 places these against the program-structured reasoners, which close the remaining gap only on in-distribution questions.

7.2.1.4 Challenges and Trends

Interpretability is intrinsic: because relations live in distributed activations rather than named symbols, it is hard to audit which relations a model actually exploits, and attention-map visualizations are only an indirect proxy. *Compositional generalization* is the most striking failure: even at 97–98% CLEVR accuracy, the same models drop sharply on CLEVR-CoGenT [1358] and GQA [839], revealing dataset-specific shortcuts rather than transferable spatial concepts; the related *shortcut-learning* concern in open-domain VQA, where language priors and object co-occurrence dominate the signal, is the imbalance VQA v2's balanced design targets [1359]. *Scalability* divides the families: RN-style pairwise modeling is $O(n^2)$ in the number of objects, attention backbones trade receptive field against compute [469, 1353], and GNNs inherit the brittleness of the proposal stage [798]. The more fundamental limit is the *2D-to-3D* gap: these methods reason over 2D feature grids and image-plane relations, whereas much spatial reasoning depends on depth ordering, occlusion, viewpoint, and metric layout, and strong 2D backbones and vision–language models degrade sharply once relations hinge on 3D geometry [612, 611].

Two responses run through the rest of this section. The explicit-structured methods of §7.2.2 reintroduce programs and graphs over learned features, recovering inspectability while keeping the representational power of deep networks; the geometry-aware methods of §7.2.3 answer the 2D-to-3D gap directly by building the symmetry group into the architecture, where it is known. Whether web-scale pretraining alone closes the systematic-generalization gap, or explicit structure remains necessary, is the open question that §7.3 takes up.

7.2.2 Explicit Structured Neural Reasoning

Explicit structured neural reasoning decomposes a spatial problem into an inspectable intermediate structure, a program, a scene graph, a sequence of reasoning states, or executable code that calls vision tools, and obtains the answer by executing or traversing that structure over visual representations. Unlike the rule-based reasoning of §7.1, the structure is learned or inferred from data rather than written by hand; unlike the

implicit reasoners of §7.2.1, it exposes spatial predicates directly, so that an inspectable edge or a named operation can represent containment, adjacency, support, or relative position, making them auditable and composable instead of entangled in a feature space. Figure 7.9 previews the representative tasks.

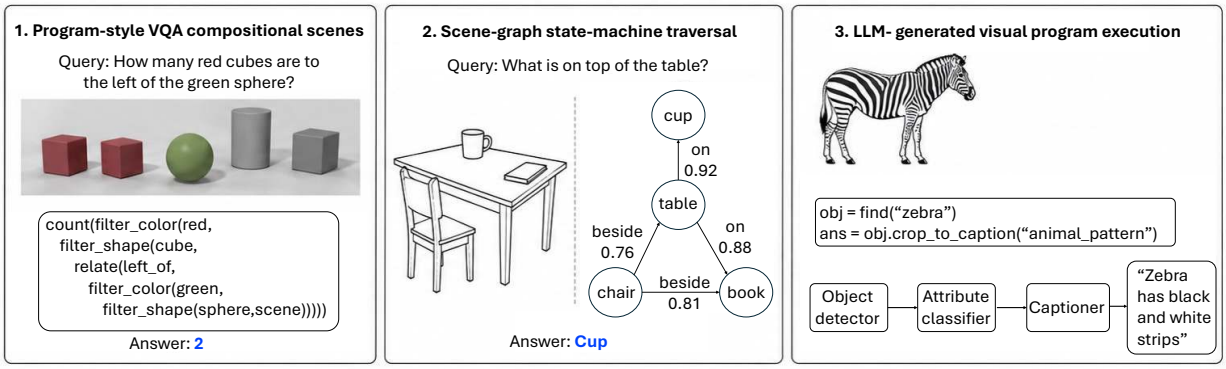


Figure 7.9: Representative tasks for explicit structured neural reasoning, in which the reasoning trace is inspectable: (1) program-style VQA on compositional scenes, (2) state-machine traversal of a probabilistic scene graph, and (3) execution of an LLM-generated visual program.

7.2.2.1 Task Definition

Given a visual observation $\mathbf{I}$ (a single image or a set of views) and a query or goal q, the system constructs an intermediate structure and executes it,

$$\mathcal{S} = \mathrm{Parse}_\phi(\mathbf{I}, q), \qquad y = \mathcal{E}(\mathcal{S}; \mathbf{I}), \tag{7.20}$$

where $\mathcal{S}$ may be a program layout $P = (m_1, \ldots, m_k)$, a recurrent sequence of reasoning states, a scene graph $G = (V, E)$ with a finite-state traversal over its nodes, or a code trace that calls external tools, and the executor $\mathcal{E}$ realizes $\mathcal{S}$ through neural modules, symbolic operators, graph transitions, or tool calls. The output y may be an answer, a localized region, a predicted relation, or an action. The defining property is therefore not a specific program syntax but the presence of an inspectable $\mathcal{S}$ that mediates between perception and prediction, which is what makes the reasoning path explicit and controllable.

7.2.2.2 Methodology

The four families in Figure 7.10 differ in how much structure they expose, enough to make reasoning inspectable and compositional, but not so rigid that a perception error, a parser error, or a missing primitive causes deterministic failure. Each exposes a different facet of space: recurrent architectures a temporal trace of attentional focus, scene-graph methods object-centric relations, program-based methods named operations such as filtering, relating, counting, and comparing, and LLM-generated programs tool-call traces that combine detection, segmentation, localization, and external knowledge.

Structured Recurrent Reasoning Architectures. This family imposes a step-wise reasoning structure in the architecture itself, so the model is fully differentiable and trained end-to-end from input–output pairs while still exposing a step-by-step trace. The Memory, Attention, and Composition (MAC) network [1360] chains p recurrent cells, each of which attends to the question to choose the operation for the current step (control unit), extracts the relevant visual information by spatial attention guided by that control signal (read unit), and integrates it into a reasoning state passed to the next cell (write unit); the attention maps of successive read units provide a transparent trace of where the model is looking, without a rigid symbolic program. MAC is thus a boundary case: explicit at the level of reasoning dynamics, implicit at the level of spatial representation, since the operations run over dense feature maps rather than named objects or

relations. Because the spatial logic stays in the learned weights, it is prone to statistical shortcuts (if "red spheres" co-occur with "cubes" in training, the model may learn that correlation rather than "left of"), and soft attention maps are often diffuse, so "to the left" bleeds into irrelevant regions in cluttered scenes.

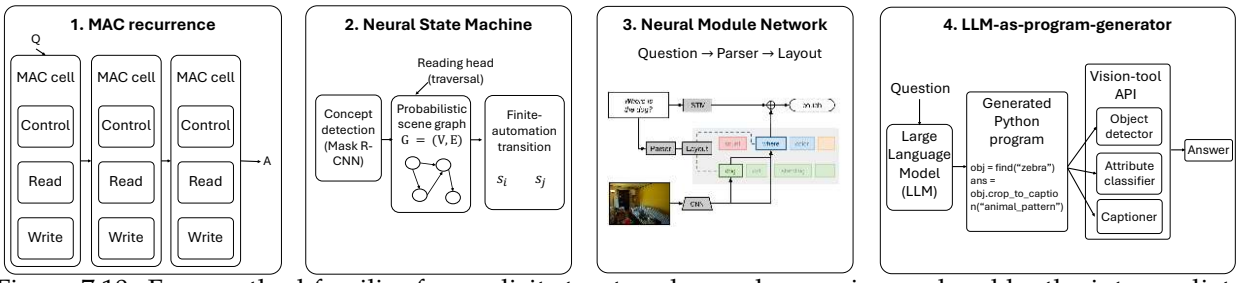


Figure 7.10: Four method families for explicit structured neural reasoning, ordered by the intermediate structure $\mathcal{S}$ each exposes: (A) the MAC recurrent reasoning cell [1360]; (B) the Neural State Machine [1361]; (C) program-based reasoning, a parse-to-execute pipeline whose executor ranges from neural modules (NMN [1362], IEP [1363]) through the symbolic executor of NS-VQA [1364] to the quasi-symbolic executor of NS-CL [347]; and (D) LLM-as-program-generator pipelines that call vision tools (*e.g.*, ViperGPT [1131], VisProg [1365], Code-as-Policies [1366]).

Structured Scene Representations. A related line strengthens the structural prior on the reasoning substrate. Where MAC reasons over a soft grid of features, the Neural State Machine (NSM) [1361] reasons over a discrete semantic scene graph and simulates a finite automaton over it: a concept-detection module (Mask R-CNN) identifies entities, each node v_i is initialized with semantic properties over a learned concept vocabulary, each edge e_{ij} carries a distribution over spatial and semantic relation types (left of, behind, on top of, holding), and the query is answered by sequentially traversing nodes along valid transitions. More broadly, scene graphs act as a generic structured substrate: a downstream reasoner can traverse the graph as a hard environment (the NSM stance), attend over it as a soft relational prior, or search it for graph-based question answering, in each case obtaining an inspectable cut-point between perception and reasoning. The price is that reasoning quality becomes tightly coupled to object detection, attribute grounding, and relation prediction, a dependency we return to in the challenges.

Modular and Program-Based Reasoning. A third family translates the query itself into an executable program, and its members differ mainly in the nature of the *executor*. The Neural Module Network (NMN) [1362] is the canonical *neural-executor* instance: an external semantic parser converts a question such as "What color is the llama?" into a layout, *e.g.*, `classify[color](attend[llama])`, whose slots are instantiated by distinct learnable modules (attention, re-attention, classification, combination, measurement) trained jointly to maximize the likelihood of the gold answer. Because NMN relies on an off-the-shelf parser, IEP [1363] replaces it with a sequence-to-sequence program generator that emits the program as a sequence of functions, trained jointly with the execution modules via REINFORCE [1367] or by maximizing answer likelihood over a distribution of programs. Moving further toward symbols, Neuro-Symbolic VQA (NS-VQA) [1364] adopts a *symbolic executor*: a scene parser turns the image into a symbolic table, after which the program reduces to deterministic functional operators, yielding fully interpretable execution at the cost of a near-perfect-perception assumption. The Neuro-Symbolic Concept Learner (NS-CL) [347] occupies the middle ground with a *quasi-symbolic* executor: a neural perception module produces object-centric representations and a semantic parser emits symbolic programs, but the executor relaxes discrete operations such as filtering and relating into differentiable ones, so the whole system trains end-to-end from answers alone.

LLM-Generated Visual Programs. The most active recent extension lets a large language model emit the program directly, replacing the seq2seq parser of IEP or the hand-crafted parser of NMN with a *tool-based executor*. ViperGPT [1131] and VisProg [1365] pair a frozen LLM with a Python-style API of vision

tools (detectors, captioners, attribute classifiers) that the generated code invokes, making the approach compositional and zero-shot without per-task training. Code-as-Policies [1366] pushes the same program-synthesis paradigm into embodied control, treating the LLM-emitted program as a closed-loop policy over perception and actuation primitives; it is included here as an embodied extension of the paradigm rather than as a VQA method. The interpretability gain is real, since the emitted code is human-readable, but these systems re-inherit the brittleness of the underlying LLM (hallucinated API calls, ungrounded tool routing) as a failure mode the earlier hand-coded DSLs were designed to avoid.

Unifying View: Parser, Representation, Executor, and Supervision. The four families are points in a shared design space with four axes. The *parser* that maps a query to structure may be hand-built (NMN), learned end-to-end (IEP, NS-CL), or a frozen LLM (ViperGPT, VisProg); the *representation* over which reasoning runs may be a dense feature map (MAC), an object-centric scene graph (NSM, and the scene-graph-generation literature [798] supervised at scale by Visual Genome [796]), or a symbolic scene table (NS-VQA); the *executor* may be neural, symbolic, quasi-symbolic, or a code interpreter; and the *supervision* may be answer-only, program-supervised, or, in tool-based systems, augmented with execution traces and tool feedback. The inspectable structure between perception and answer is the common thread: it provides a natural cut-point at which symbolic and learned components can be swapped without retraining the rest of the stack, and the same four axes reappear in the foundation-model reasoners of §7.3.

Beyond VQA: 3D, Temporal, and Embodied Extensions. The same design pattern extends beyond static images: referring-expression grounding has been lifted from 2D images to 3D scenes [374], where object-centric and spatial-relation graphs supply the substrate for reasoning about support, containment, proximity, and navigability, and in video and embodied settings the intermediate structure becomes a temporal scene graph or a task plan that links perception to action (§7.3, Part IV). These extensions also expose the central limitation of current explicit methods: most available program primitives and graph predicates are still designed for static 2D relations, whereas viewpoint change, object permanence, physical interaction, and multi-step navigation need 3D, temporal, and causal operators.

7.2.2.3 Benchmarks and Performance

The CLEVR family [1358] of §7.2.1 is the canonical testbed here as well, with the added property that each question is generated alongside a ground-truth functional program, which gives both program-supervised and answer-only training signals; MAC, IEP, NS-CL, and NS-VQA all converge to the high-90s on it, making it attractive but saturated. GQA [839] is the dominant open-domain benchmark: it pairs natural-image scenes with compositional questions whose ground-truth scene graphs (from Visual Genome [796]) make scene-graph-based reasoners directly applicable, and its out-of-distribution splits expose the systematic-generalization gap hidden by saturated CLEVR scores. VQA v2 [1359] remains the reference point for visual-program systems such as ViperGPT and VisProg, which are evaluated zero-shot on GQA and VQA v2. Referring-expression grounding benchmarks (RefCOCO and RefCOCO+ [1082], RefCOCOg [1083], and the earlier ReferItGame [1088]) are equally relevant, since they test whether a system can bind spatial language ("the object left of / behind / next to . . . ") to a localized region even when the output is a box or mask rather than an answer. Embodied program-as-policy systems such as Code-as-Policies are evaluated by task success in robotic or simulated manipulation environments and should not be compared against VQA accuracy. Table 7.1 collects the headline numbers.

7.2.2.4 Challenges and Trends

Exposing programs, graphs, state transitions, or tool calls buys interpretability but creates new failure modes, and the numbers above locate them. The first bottleneck is **parsing**: if a question is mapped to the wrong program or traversal path, even a correct executor produces the wrong answer. The second is **symbol grounding and underspecification**: spatial predicates such as `Left`, `Behind`, `Near`, or `Between` depend on

Table 7.1: Implicit and explicit structured reasoners on CLEVR and GQA (overall accuracy, %, higher is better). CLEVR numbers are the standard test split from the original papers; GQA numbers are test-dev or test-standard as reported. ViperGPT and VisProg run zero-shot with a frozen LLM, and VisProg is evaluated on a curated test-dev subset, so its number is not directly comparable. RN and FiLM are the implicit reference points of §7.2.1.

Method	Structure exposed	Supervision	CLEVR	GQA
RN [1349]	none (pairwise latent)	answers	95.5	–
FiLM [1355]	none (feature modulation)	answers	97.7	–
MAC [1360]	recurrent control / read / write	answers	98.9	54.1
IEP [1363]	generated program, neural executor	programs (700k) + answers	96.9	–
NS-CL [347]	program, quasi-symbolic executor	answers (curriculum)	98.9	–
NS-VQA [1364]	program, symbolic executor	scene tables + 270 programs	99.8	–
NSM [1361]	probabilistic scene graph, automaton	answers	–	63.2
ViperGPT [1131]	LLM-generated code, tool executor	none (zero-shot)	–	48.1
VisProg [1365]	LLM-generated code, tool executor	none (zero-shot)	–	50.5

viewpoint, scale, object extent, and task context, so many spatial queries do not determine a unique program, a single deterministic DSL operator is often too rigid, and new predicates demand manual re-engineering of the executor. The third is **error propagation**: mistakes by the detector, scene parser, relation classifier, or program generator are inherited by the executor and can cause deterministic failure, which is why NSM reaches 63% on GQA while the same family saturates CLEVR; on natural images the bottleneck has moved from reasoning execution to perception, scene-graph construction, language grounding, and long-tail relation coverage. A fourth, more structural tension is the **differentiability–interpretability trade-off**: the more symbolic the executor, the more auditable but the harder to train end-to-end; the more differentiable, the more robust but the less transparent. Because the value of explicit structure lies in the intermediate trace, **evaluation** should move beyond final-answer accuracy to whether the parsed program is faithful to the query, whether graph edges correspond to real relations, whether tool calls are necessary and grounded, and whether the execution trace actually supports the answer.

Most explicit systems also reason over static 2D layouts or object graphs rather than 3D geometry, temporal dynamics, or physical causality: they can answer whether A is left of B, but rarely simulate how a scene changes under motion, contact, containment, support, or occlusion, which is exactly the mental-simulation capacity human-level spatial reasoning requires. LLM-generated programs widen the scope by replacing fixed parsers with flexible code generation, trading a small accuracy gap on saturated benchmarks for zero-shot generality across VQA, grounding, and knowledge-grounded VQA, but they add hallucinated tool calls, invalid APIs, ungrounded intermediate variables, and the absence of a verifiable execution loop. The emerging direction is therefore not a return to purely symbolic reasoning but *verifiable hybrid* systems, in which learned perception produces object-centric representations, structured programs or graphs provide an inspectable scaffold, and neural or tool-based executors absorb the variability of real visual inputs, the direction of §7.3 and §7.4.

7.2.3 Geometry-aware Neural Spatial Reasoning

Where the implicit and explicit methods above learn spatial relationships from data, geometry-aware approaches build the relevant symmetries into the network itself: an object rotated, translated, or re-indexed keeps its identity, and the architecture is constructed so that this holds by design rather than being

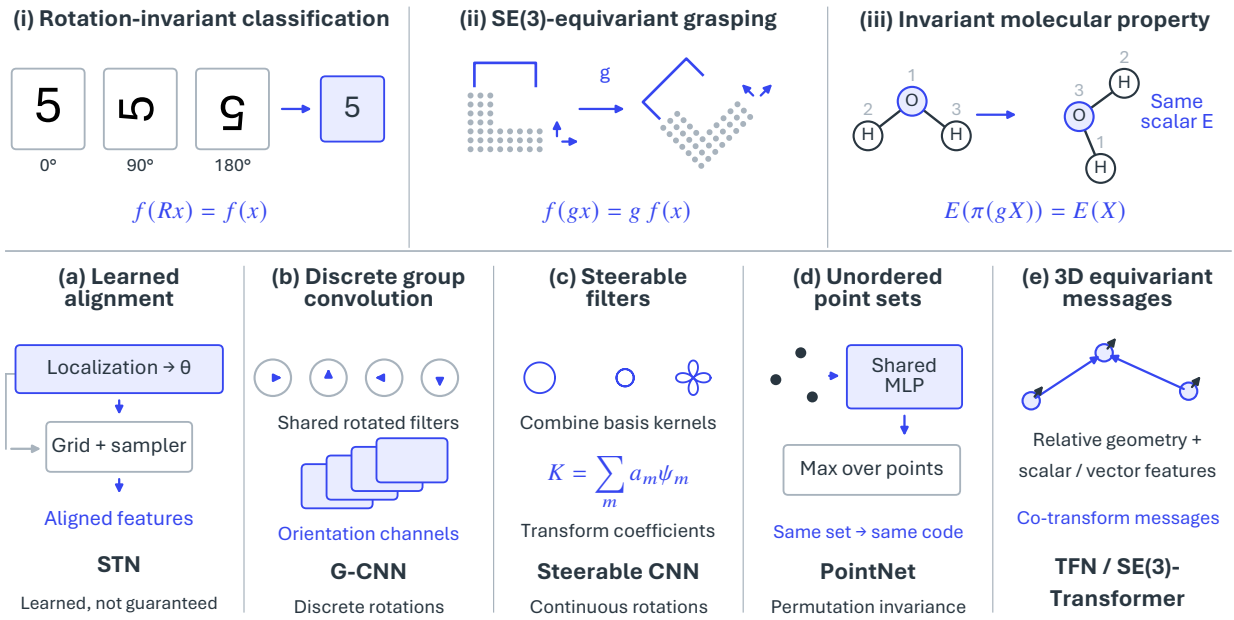


Figure 7.11: Symmetry-driven tasks (top) and geometric neural mechanisms (bottom). Top: (i) rotation-invariant classification, (ii) SE(3)-equivariant grasping, shown with a planar rotation, and (iii) a molecular scalar invariant to rigid motion and atom reindexing. Bottom: (a) STN [1368], (b) G-CNN [1357], (c) steerable CNN [1369], (d) PointNet [132], and (e) TFN / SE(3)-Transformer [1370, 379]. Blue marks transformations, geometric features, and preserved outputs; all examples are schematic.

learned from many transformed examples. The gain is sample efficiency and robustness under the chosen transformations; the cost is that the symmetry group has to be chosen in advance. Figure 7.11 pairs three symmetry-driven task settings with the method families that supply the matching inductive bias.

7.2.3.1 Task Definition

The input x lives in a space with a known geometric structure (a 2D image, a 3D point cloud, a molecular graph), optionally with a query q, and the network $M_{\theta,G}$ is designed to respect a chosen group G of transformations: translations for standard CNNs, rotations SO(3), rigid motions SE(3), or permutations of set elements. Its output $\mathbf{z}$ is either invariant or equivariant to the action of G.

Invariance vs. equivariance. An *invariant* representation stays fixed under the group action, $M_{\theta,G}(g \cdot x) = M_{\theta,G}(x)$, and deliberately discards pose; an *equivariant* representation co-transforms with the input, $M_{\theta,G}(g \cdot x) = \rho(g) \cdot M_{\theta,G}(x)$, where $\rho(g)$ is the representation of the group action on the feature space, and thereby preserves it. In either case geometric consistency is a native constraint that the architecture guarantees by construction.

7.2.3.2 Methodology

Invariance is the right target when the output should not depend on orientation or position (classification), equivariance when pose and relative arrangement are the point (manipulation, keypoints, instance segmentation, molecular modeling); the layers below realize each.

Invariance by Pooling and Augmentation. The baseline strategies work without changing the architecture. Augmenting the training set with rotated, flipped, or scaled copies pushes a standard CNN toward assigning

the same label to all variants, but buys only an approximate invariance and requires examples of every transformation to be tolerated; global average pooling [1371] at the end of a feature extractor discards the "where" of each feature and keeps only its presence, which is the main reason classification CNNs are robust to translation. Neither guarantees invariance to anything beyond translation.

Active Alignment with Spatial Transformer Networks. Rather than ignoring transformations, a Spatial Transformer Network (STN) [1368] learns to undo them: a small localization network regresses the parameters of an affine transformation that it predicts will bring the object into a canonical pose, a grid generator turns these parameters into a sampling grid, and a differentiable sampler produces the rectified feature map, so the localization network is trained by backpropagation from the task loss alone. The module can be inserted at any point in a network and acts as learned attentional alignment, normalizing a region of interest for the layers downstream; the invariance it provides is learned, not guaranteed.

Permutation Invariance over Point Sets. A separate symmetry arises when the input is an unordered set rather than a grid: the model must be invariant to the ordering of its elements. DeepSets [1372] shows that any permutation-invariant function over a set can be written as $f(X) = \phi\big(\sum_{x \in X} \psi(x)\big)$ for suitable ϕ and ψ, encode each element independently, sum, then decode, which turns set-level invariance from an ad-hoc choice into a constructive recipe and makes explicit which aggregators preserve invariance and which silently break it; the same framework yields permutation-*equivariant* layers for per-element outputs (per-point labels, per-atom properties). PointNet [132] is the canonical instance on raw point clouds, a shared multilayer perceptron per point followed by symmetric max pooling, and PointNet++ [270] adds locality by repeatedly grouping points within a metric ball and applying the same shared-MLP-plus-pool pattern at multiple scales. Permutation invariance comes for free in this family; rigid-motion invariance does not, which is the gap the equivariant methods below address.

From Translation to Group Equivariance. A standard convolution is translation-equivariant (shifting the image shifts the feature map) but not rotation-equivariant: rotating the input changes the feature map unpredictably. Group equivariant convolutions (G-CNNs) [1357] generalize the operation to a larger discrete group such as the four planar rotations: a single filter is rotated into a set of transformed copies, all are correlated with the input, and the feature map gains an orientation dimension, so a feature that reads "edge at $45°$" becomes "edge at $135°$" at the corresponding location when the input rotates by $90°$. Equivariance is therefore not learned but imposed, layer by layer.

Steerable Filters and Equivariance in Every Layer. Steerable CNNs [1369] replace the group-indexed filter copies by filters expressed in a steerable basis, so that the response at a transformed orientation follows by linear combination of basis responses rather than by explicit rotation; the general E(2) framework [1373] extends this to continuous rotations and reflections with circular-harmonic bases under one configurable interface, and library efforts such as e3nn [1374] do the same for E(3) / SE(3). Full equivariance also requires that pooling, nonlinearities, and normalization respect the group: a standard ReLU applied to the components of a geometric vector breaks it (activations are therefore applied to invariant quantities such as the vector norm), and channel-wise batch normalization breaks the coupling between channels, which motivates equivariant normalization [1375]. The guarantee is only as strong as its weakest layer; one non-equivariant operation anywhere in the stack silently undoes the symmetry the rest preserve.

Equivariant Attention and Continuous SE(3) Symmetries. A more recent line lifts the treatment from convolutions to attention, with queries, keys, and values that transform predictably under a chosen Lie group. The SE(3)-Transformer [379] is the canonical instance, building each attention layer from tensor-field representations so that rotating the input scene rotates the whole stack of outputs consistently; Tensor Field Networks [1370] use the same machinery without an attention head and have been especially influential in molecular modeling, where SE(3) equivariance is a problem constraint rather than an optional bias. Both parameterize the kernel as a combination of spherical-harmonic basis functions, which is computationally

heavy, so a parallel line pursues lighter equivariance: E(n)-equivariant graph networks (EGNN) [1376] achieve E(n)-equivariant message passing from relative distances and coordinate updates alone, and Vector Neurons [380] lift scalar point-cloud features to 3D vectors, obtaining SO(3) equivariance as a drop-in primitive, each trading representational richness for scalability. With the libraries above, trying equivariance on a new task now costs about as much engineering as a convolutional baseline; the choice of group still has to be made by the designer.

7.2.3.3 Benchmarks

Benchmarks in this area are constructed to expose behavior under symmetry transformations. MNIST-rot is the standard 2D testbed, separating methods that learn rotation robustness through augmentation from those that build it in; ModelNet40 [1377] evaluates point-cloud classification under three protocols, aligned training and test (z/z), training with azimuthal rotations and testing under arbitrary SO(3) rotations (z/SO(3)), and arbitrary rotations in both (SO(3)/SO(3)), which is where augmentation, STN-style alignment, and PointNet's permutation-invariant pooling are separated from genuine rotation handling; ScanObjectNN [1378] adds real scan noise and occlusion. For equivariance as a hard constraint, QM9 [1379] (small-molecule property prediction) and interatomic-potential tasks are the reference, where the SE(3)-equivariant models NequIP [1380] and MACE [1381] set the current points for data efficiency and accuracy. Table 7.2 reports the two ladders on which the method families can be compared directly.

Table 7.2: Symmetry benchmarks. MNIST-rot: test error (%, lower is better); plain CNN and P4CNN figures from Cohen and Welling [1357], E(2)-steerable figure from Weiler and Cesa [1373]. ModelNet40: classification accuracy (%, higher is better) under the train/test rotation protocols z/z, z/SO(3), and SO(3)/SO(3), all as re-evaluated by Deng et al. [380].

Method	MNIST-rot err.	z/z	z/SO(3)	SO(3)/SO(3)
Plain CNN (Z2CNN) [1357]	5.03	–	–	–
P4CNN (G-CNN) [1357]	2.28	–	–	–
E(2)-steerable CNN [1373]	0.68	–	–	–
PointNet [132]	–	85.9	19.6	74.7
PointNet++ [270]	–	91.8	28.4	85.0
VN-PointNet [380]	–	77.5	77.5	77.2
VN-DGCNN [380]	–	89.5	89.5	90.2

7.2.3.4 Challenges and Trends

The most fundamental difficulty is choosing the symmetry group: invariance to a group that is too large (SO(3) when only SO(2) is task-relevant) destroys discriminative information, invariance to one too small leaves the model brittle to the symmetries the task actually exhibits, and the trade-off has no purely data-driven solution [1357, 1369]. Cost scales with the group: classical equivariant architectures index filters by group elements, so parameters and memory grow with the size of the chosen group, limiting how freely reflections, multi-scale rotations, or higher-order tensor representations can be included [1373]. The assumption of *exact* symmetry is also brittle: point clouds with self-occlusion, partial-view RGB, and noisy LiDAR scans do not realize an SO(3)-orbit, so an architecture that is mathematically equivariant can still fail when the input violates the assumed structure. Table 7.2 shows where the two targets diverge: on 2D classification both hand-coded invariance and learned equivariance saturate, so data efficiency rather than accuracy becomes the discriminating axis, while on ModelNet40 the invariance-only PointNet family collapses under unseen rotations and the equivariant family holds its accuracy, because the task requires geometric information to be *preserved*, not discarded. Finally, hard equivariance sits uneasily with the attention-based representations that dominate the rest of the stack (§7.2.1, §7.2.2, vision–language pretraining): self-attention is permutation-equivariant but not rotation-equivariant, and bolting an equivariant layer onto an attention pipeline often degrades the inductive bias that motivated the choice.

Equivariant attention is the main compromise on offer, and frame averaging [1382] moves in the opposite direction, turning any off-the-shelf backbone into an invariant or equivariant one by symmetrizing its outputs over a small input-dependent set of group frames, which decouples the choice of symmetry from the choice of architecture and sidesteps the parameter cost of group-indexed filters. These threads are pulling geometry-aware reasoning back toward the rest of the chapter: vision–language and vision–language–action models attach equivariant heads where 3D pose and geometric consistency matter, as in grasp policies and pose-conditioned generation, and primitives first devised for object classification are finding second lives in world models, embodied control, and 3D generation.

7.2.4 Differentiable Optimization Layers for Spatial Reasoning

Where the three preceding subsections put structure into the learned representation, differentiable optimization layers put it into the computation: selected feed-forward operations are replaced by the solution of a constrained optimization problem, so that hard logical, geometric, or physical constraints are enforced inside the forward pass while the whole system stays end-to-end differentiable [1383]. In spatial reasoning this is how a network enforces distances, collision avoidance, or alignment, or solves a sub-problem such as pose estimation, within a learnable pipeline. Figure 7.12 previews the representative tasks (top row) and the method families (bottom row).

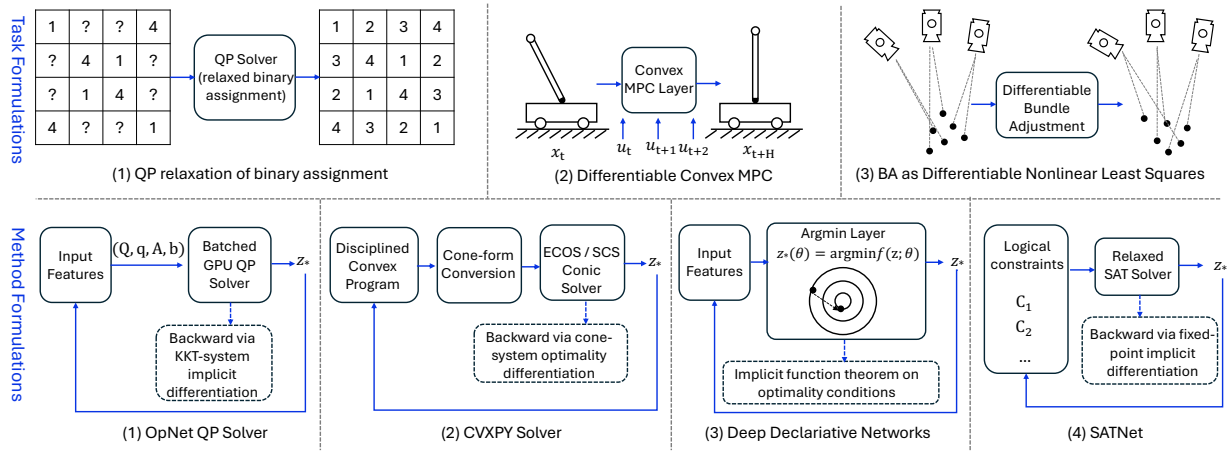


Figure 7.12: Differentiable optimization layers for spatial reasoning. Top, tasks: (1) QP relaxation of binary assignment for Sudoku [1384, 1385], (2) differentiable convex MPC [1386, 1387], and (3) bundle adjustment as differentiable nonlinear least squares for end-to-end SLAM [95]. Bottom, methods: (1) OptNet [1384], (2) CVXPY Layers [1387, 1388], (3) Deep Declarative Networks and differentiable geometry solvers [1389, 1390], and (4) SATNet-style differentiable SAT layers [1385, 1391].

7.2.4.1 Task Definition

Given an input parameter vector θ produced by the preceding layers, a differentiable optimization layer computes

$$z^\star = \arg\min_{z \in \mathcal{Z}} f(z;\theta) \;\; \text{s.t.} \;\; g(z;\theta) \le 0, \;\; h(z;\theta) = 0, \tag{7.21}$$

where f is an objective function, g, h denote inequality and equality constraints, and $\mathcal{Z}$ is a feasible domain. The forward pass computes the optimal solution $z^\star(\theta)$, while the backward pass relies on implicit differentiation of the optimality conditions to obtain $\frac{\partial z^\star}{\partial \theta}$; the layer is declarative, in that the model specifies what to solve rather than how to compute it. Under this view, QP solvers [1384], convex optimization layers [1387], differentiable geometry solvers [95], and differentiable logic program solvers [1385, 1391] are all instantiations of the same layer, tailored to different domains of spatial or symbolic reasoning.

7.2.4.2 Methodology

QP and Convex Optimization Layers. Amos and Kolter [1384] introduced OptNet, which treats the solution of a quadratic program as a layer: the forward pass solves

$$z^\star = \arg\min_z \tfrac{1}{2} z^\top Q z + q^\top z$$

subject to affine constraints whose matrices can depend on previous network outputs, and the backward pass differentiates the KKT optimality conditions exactly, with a custom GPU interior-point solver that handles many small QPs in parallel and reuses its factorization for the backward pass. OptNet learned to solve Sudoku from data where conventional architectures could not, which showed that implicit layers can handle discrete or constrained reasoning problems. Agrawal et al. [1387] generalized the idea to any disciplined convex program: CVXPY Layers [1388] canonicalize the problem into conic form, solve it, and differentiate through the cone program's optimality conditions, so a constrained least-squares, a constrained logistic regression, or a robot-motion-limit constraint can be inserted as a layer without deriving gradients by hand.

Implicit Differentiation. All of these layers share one mechanism. Rather than unrolling an iterative solver, one differentiates the closed-form optimality (KKT) conditions, which for a convex program comprise primal feasibility, dual feasibility, complementarity, and stationarity [1392]; differentiating this system and solving one linear system yields $\frac{\partial z^\star}{\partial \theta}$ exactly, without back-propagating through solver iterations. Where OptNet hand-codes this backward pass, Deep Declarative Networks [1389] apply automatic differentiation to the KKT system itself. The same principle extends to root-finding, fixed-point, and equilibrium layers via the implicit function theorem; deep equilibrium models [1393] treat an entire weight-tied network as a single fixed-point layer differentiated this way, and differentiable physics engines and planners propagate gradients through the equilibrium or optimum conditions of a physical system.

Differentiable Geometry Solvers. A subclass is tailored to geometric problems that are themselves argmin problems, typically minimizing a reprojection error: camera pose from 2D–3D correspondences, bundle adjustment, inverse kinematics. Campbell et al. [1390] introduced a differentiable blind PnP solver that estimates 3D pose as a layer; Teed and Deng [95] built a differentiable bundle-adjustment module into a SLAM system, solving for camera poses and 3D points by iterative optimization while letting gradients flow; EPro-PnP [744] recasts PnP as a probabilistic layer with a learnable distribution over poses; and Theseus [1394] provides a reusable library for differentiable nonlinear least squares (Gauss–Newton, Levenberg–Marquardt) that underlies bundle adjustment, pose-graph optimization, and inverse kinematics. These layers keep the domain-specific optimizer but make it differentiable, so the whole vision or robotics system trains end-to-end while the geometry stays correct and constrained, for instance so that estimated object poses agree across views; Gould et al. [1389] refer to such constructs as declarative nodes and demonstrate them on visual Sudoku and robust geometry estimation.

Differentiable Logic Layers. Beyond continuous optimization, the same construction extends to symbolic reasoning. SATNet [1385] integrates a differentiable satisfiability solver as a layer: the forward pass solves a relaxed SAT problem (a quadratic unconstrained binary optimization) to produce assignments consistent with logical constraints, and the backward pass differentiates implicitly through the solver's fixed-point conditions; Shindo et al. [1391] extend differentiable logic programs to abstract visual reasoning benchmarks such as Raven's Progressive Matrices. These layers belong to the same class of constraint-based implicit modules as the QP and convex layers, specialized to combinatorial and relational dependencies.

7.2.4.3 Benchmarks

Differentiable optimization layers have been validated across symbolic, geometric, and control settings. Sudoku [1384, 1385] and its visual variant [1383] test whether a layer models constraint satisfaction and

Table 7.3: Representative differentiable-optimization layers, organized by solver / layer type (*cf*. Figure 7.12).

Method	Year	Solver / layer type	Task / benchmark
OptNet [1384]	2017	QP layer, KKT implicit diff.	Sudoku from data
Differentiable MPC [1386]	2018	convex MPC layer	planning / control
CVXPY Layers [1387]	2019	disciplined convex / cone layer	control, geometry
SATNet [1385]	2019	differentiable MAXSAT relaxation	visual Sudoku
Deep Declarative Nets [1389]	2021	implicit-function-theorem nodes	robust geometry, Sudoku
DROID-SLAM [95]	2021	differentiable bundle adjustment	monocular / stereo SLAM
Theseus [1394]	2022	differentiable nonlinear least squares	pose-graph / BA / IK
EPro-PnP [744]	2022	probabilistic PnP layer	monocular 6-DoF pose

propagates gradients through discrete reasoning; Raven's Progressive Matrices (RPM / I-RAVEN) [1391] test whether differentiable logic captures relational structure in visual analogy puzzles; cart-pole, inverted-pendulum, and reaching tasks in MuJoCo and OpenAI Gym [1387] test whether differentiable MPC [1386] or convex layers improve control while enforcing physical feasibility; and maze navigation and obstacle avoidance test QP-based collision-avoidance layers by whether the agent reaches its target without violating safety constraints. Evaluation combines task metrics with criteria specific to the embedded solver: geometric accuracy (translation / rotation error in pose estimation, trajectory or map error for a bundle-adjustment layer), task success rate (puzzles solved, feasible plans produced) together with the count or magnitude of violated constraints and, for physics-informed layers, residual energy or momentum drift, and solver cost and gradient stability (forward / backward runtime, iterations to converge, and sensitivity of training to approximate versus exact implicit gradients). Table 7.3 lists the representative layers by solver type.

7.2.4.4 Challenges and Trends

Solving an optimization problem inside a layer is more expensive than a convolution or a matrix product, and the forward pass becomes the bottleneck as problem size grows, which is exactly the regime of SLAM, bundle adjustment, and motion planning. OptNet's batched GPU solver shows how problem structure buys tractability for many small problems, but global route planning or large-scale 3D reconstruction still overwhelms existing solvers, forcing approximate solves, warm starts, or hierarchical decomposition to keep each optimization manageable.

Gradient stability is the second difficulty. If the solution is non-unique or depends discontinuously on the input, as at boundary optima, at active-set changes in a linear program, or in the presence of discrete variables, the implicit gradient is unstable or undefined; strict convexity guarantees a well-defined Jacobian, and real problems frequently violate it. Entropy smoothing, small regularizers, and perturbation methods restore usable gradients in many cases, and in others the backward pass has to be designed by hand.

The third is physical consistency. An optimization layer enforces only the constraints it is given: if the formulation omits a friction constraint in a contact model, the network exploits the gap and converges to implausible solutions, and because the layer sits inside a trainable system, the surrounding network can learn to feed it inputs that satisfy the constraints superficially while violating their intended meaning. The benefits of interpretability and correctness therefore depend on the optimization model reflecting the real dynamics, which in practice means careful constraint modeling, additional penalties, or iterative refinement of the solver design; current work refines solvers, numerical stability, and integration with deep-learning frameworks toward layers that are both theoretically grounded and deployable in spatial systems.

7.2.5 Reinforcement Learning with Spatial Inductive Biases

Where the previous four subsections constrain the learned representation, reinforcement learning shifts the locus of structure to the *action-conditioned* setting, in which an agent must reason about spatial relationships as a consequence of its own behavior over a long horizon. Many spatial tasks, such as navigation, exploration, manipulation, and embodied interaction, cannot be reduced to static perception or single-step prediction; they require reasoning about how actions transform the agent's position, orientation, and relations to surrounding objects over time, under geometric constraints, partial observability, and delayed reward. Generic RL policies built on unstructured function approximators struggle to generalize across unseen layouts, so a growing body of work introduces *spatial inductive biases*, explicit planning iterations, map-like latent representations, differentiable search operators, or relational state abstractions, that constrain the hypothesis space of policies or value functions to reflect the compositional and geometric structure of spatial decision-making.

7.2.5.1 Task Definition

At each step t the agent receives an observation o_t from the environment $\mathcal{E}$ (RGB / RGB-D frames, egocentric maps, point clouds, language goals) together with its action history $a_{<t}$, chooses an action a_t from a continuous control space or a discrete set $\mathcal{A} = \{1, \dots, K\}$, and receives a scalar reward r_t. The policy or value function is parameterized with a spatial structural prior S (a differentiable planning module, a learned map or belief representation, a relational or graph encoder, an equivariant policy head),

$$\pi_\theta(a_t \mid o_t, a_{<t};\, S), \qquad \max_\theta \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right], \tag{7.22}$$

so that S shapes *which* policies and value functions the agent considers in the first place, not only how it generalizes after the fact; the optimization itself remains a standard RL objective (Bellman, policy gradient, actor–critic, or preference-based).

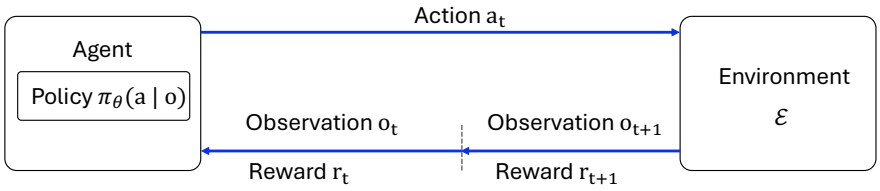


Figure 7.13: The general framework in RL: the agent acts on the environment, receives an observation and a reward, and updates its policy from this feedback.

7.2.5.2 Methodology

Figure 7.13 shows the loop every method below shares: the agent observes, acts, receives a reward and a new state, and updates its policy to maximize long-term return. Learning a good policy requires balancing **exploration** of untried actions against **exploitation** of the actions currently believed best, the exploration–exploitation trade-off that underlies all of RL. The subsections that follow walk through the optimization backbones in the order the field developed them, value-based, policy-based, actor–critic, PPO, and the recent preference-based and group-relative objectives, and note at each step where spatial structure enters.

Value-Based RL. Value-based methods learn a value function that estimates the expected cumulative return of acting under a policy π. The optimal state-action value function is

$$Q^*(s,a) = \max_\pi \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s, a_0 = a\right], \tag{7.23}$$

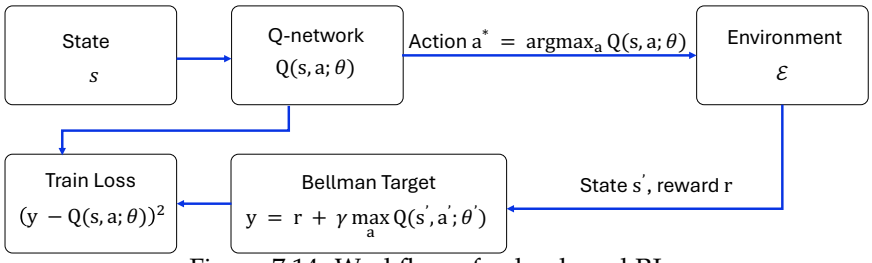


Figure 7.14: Workflow of value-based RL.

where $\gamma \in (0, 1)$ is the discount factor, and it satisfies the *Bellman equation*: if the optimal values $Q^*(s', a')$ of the next state are known for every action a', the best action now is the one that maximizes $r + \gamma Q^*(s', a')$,

$$Q^*(s, a) = \mathbb{E}_{s' \sim \mathcal{E}}[r + \gamma \max_{a'} Q^*(s', a') | s, a]. \tag{7.24}$$

Iterating this identity as an update, $Q_{i+1}(s, a) = \mathbb{E}_{s' \sim \mathcal{E}}[r + \gamma \ \max_{a'} \ Q_i(s', a') | s, a]$, converges to Q^*, but a table of values indexed by every state is useless in high-dimensional observation spaces, so a function approximator $Q(s, a; \theta) \approx Q^*(s, a)$ is learned instead, a neural network in the deep-RL setting (a Q-network). Training minimizes a sequence of regression losses whose target is the bootstrapped Bellman estimate,

$$L_i(\theta_i) = \mathbb{E}_{s, a \sim \rho(\cdot)}[(y_i - Q(s, a; \theta_i))^2], \tag{7.25}$$

with $y_i = \mathbb{E}_{s' \sim \mathcal{E}}[r + \gamma \ \max_{a'} Q(s', a'; \theta_{i-1}) | s, a]$ computed from the previous parameters θ_{i-1}, held fixed, and $\rho(s, a)$ the distribution of visited state–action pairs; the greedy policy $\pi(s; \theta) = \arg\max_a Q(s, a; \theta)$ then follows from the learned values.

Deep Q-networks (DQN) [1395] made this scale to high-dimensional observations with experience replay and a slowly updated target network; their convergence is analyzed theoretically by Fan et al. [1396], and Figure 7.14 shows the resulting workflow. Beyond control, value-based RL drives *active visual perception*: Caicedo and Lazebnik [1397] cast object localization as a sequential decision process in which a Q-network iteratively transforms an attention window until it tightly bounds the target, turning detection into a learned spatial search policy, a formulation since applied to active detection, visual search, and region selection for fine-grained recognition, where the agent learns *where to look* rather than scanning exhaustively. In spatial domains the limitation of vanilla DQN is that any planning behavior must be learned implicitly from data, which generalizes poorly to novel layouts, so several works embed planning structure in the value estimate itself. Value Iteration Networks (VIN) [1398] unroll Bellman updates as convolutions over a spatial grid, a differentiable approximation of value iteration; Gated Path Planning Networks (GPPN) [1399] replace the recurrence with gated units to improve optimization stability and long-range dependencies; and under partial observability QMDP-Net [1400] integrates belief updates and approximate POMDP planning (QMDP) [1401] into one differentiable network. In each case the learned value function is constrained to follow spatially structured dynamic programming rather than arbitrary function approximation.

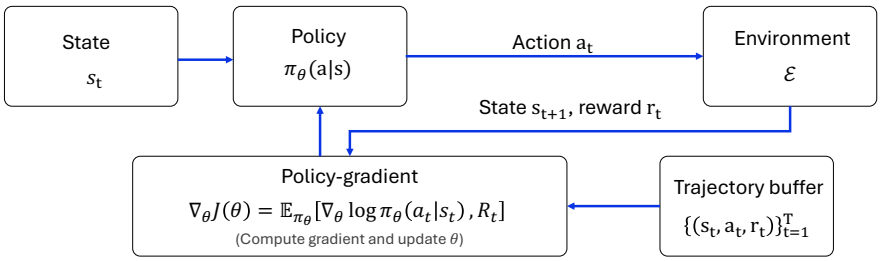


Figure 7.15: Workflow of policy-based RL.

Policy-Based RL. Policy-based methods [1367, 1402] parameterize a policy $\pi_\theta(a|s)$ directly and maximize expected return,

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right], \tag{7.26}$$

by gradient ascent on

$$\nabla_\theta J(\theta) = \mathbb{E}_{\pi_\theta}\left[\nabla_\theta \log \pi_\theta(a_t|s_t)\, R_t\right], \tag{7.27}$$

where $R_t = \sum_{k=t}^{\infty} \gamma^{k-t} r_k$ is the return from time-step t (Figure 7.15). Optimizing action distributions directly suits spatial decision problems in which exploration and long-horizon credit assignment are central [1403], and policy gradients train embodied agents to actively gather visual evidence: Embodied Question Answering [1404] optimizes a navigation policy so that the agent moves to vantage points from which it can answer questions about a 3D scene. The spatial competence, however, comes from the architecture rather than the objective; policy-gradient agents generalize across layouts when the policy carries a learned map [1405], a grounded spatial representation [1406], memory, or a relational encoder [1407].

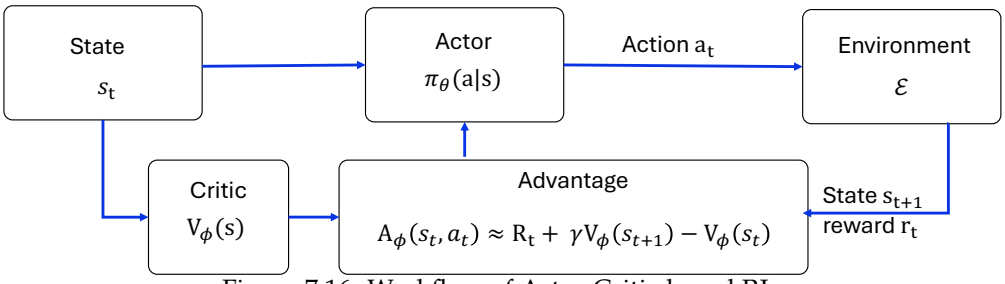


Figure 7.16: Workflow of Actor-Critic-based RL.

Actor-Critic RL. Actor–critic methods [1408] combine the two by jointly learning a policy $\pi_\theta(a|s)$ (the actor, which controls behavior) and a value function $V_\phi(s)$ (the critic, which scores the actions taken), as in Figure 7.16. The critic provides a low-variance estimate of the advantage,

$$A_\phi(s_t, a_t) \approx r_t + \gamma V_\phi(s_{t+1}) - V_\phi(s_t), \tag{7.28}$$

or, when a Q-function is learned as well, $A_\phi(s_t, a_t) = Q_\phi(s_t, a_t) - V_\phi(s_t)$, and the advantage replaces the raw return R_t in the policy-gradient update. The asynchronous advantage actor-critic (A3C) [1409] made this stable and scalable and became a standard backbone for embodied navigation and spatial control: target-driven visual navigation [1410] conditions an A3C agent on an image of the goal and reaches semantically specified targets in unseen indoor scenes, and Neural SLAM [1411] gives an A3C agent a differentiable map memory. The separation between actor and critic matches the structure of spatial tasks, with the actor responsible for geometry-aware decisions and the critic supplying global, long-horizon value signals, which is why actor–critic methods dominate navigation benchmarks that require memory, planning, and partial observability [1412].

PPO-Style Policy Optimization. Proximal Policy Optimization (PPO) [1413] is an actor–critic algorithm that stabilizes training by constraining each policy update through a clipped surrogate objective,

$$L^{\text{PPO}}(\theta) = \mathbb{E}\left[\min\left(r_t(\theta)A_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t\right)\right], \tag{7.29}$$

where $r_t(\theta)$ is the probability ratio between the new and old policies (Figure 7.17). Its robustness and sample efficiency made PPO the de facto backbone for training agents in large-scale photorealistic 3D simulators such as Gibson [93] and Matterport3D [340], and in manipulation suites such as panda-gym[1]. Scaled to 2.5 billion frames with DD-PPO [1414], it essentially solves PointGoal navigation on Habitat (99.7% success and 0.97 SPL

[1] https://github.com/qgallouedec/panda-gym

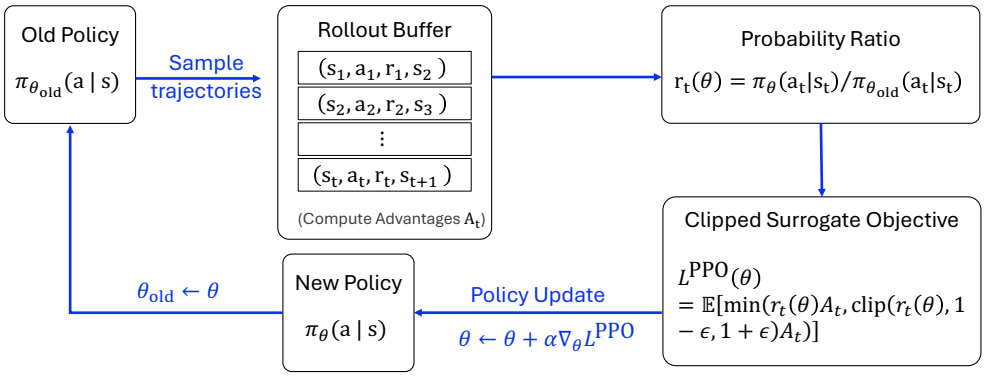


Figure 7.17: Workflow of PPO-based RL.

on Gibson validation), and the same backbone underpins vision-and-language navigation [1415], object-goal navigation [1416], audio-visual navigation [46], and planning over uncertain topological maps [1417].

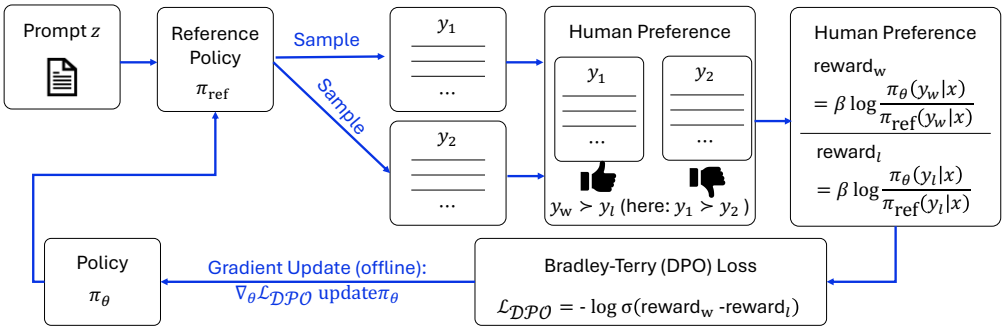


Figure 7.18: Workflow of DPO-based RL.

Preference-Based Policy Optimization (DPO). Preference-based learning replaces the reward signal with comparisons. Direct Preference Optimization (DPO) [1418] learns a policy from preference pairs without an explicit reward model or online rollouts (Figure 7.18): for two completions (y_1, y_2) of an input x, a labeler marks the preferred one, $y_w \succ y_l \mid x$, and by expressing the implied reward of a Bradley–Terry preference model [1419] through the optimal policy itself, DPO collapses the reward-learning-then-RL pipeline into a single maximum-likelihood objective on π_θ:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\big[\log \sigma\big(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\big)\big], \tag{7.30}$$

where π_{ref} is a fixed reference policy and β controls the deviation from it. Developed for language-model alignment, DPO applies to spatial reasoning whenever supervision takes the form of trajectory preferences, plan rankings, or human judgments of spatial behavior [1420, 1421, 1422], which arise naturally in navigation, manipulation, and interactive tasks where dense rewards are hard to specify.

Group-Based Policy Optimization (GRPO). Group Relative Policy Optimization (GRPO) [1423] removes the learned critic and replaces it with a group-relative baseline computed from multiple sampled trajectories (Figure 7.19). For each question q, it samples a group of outputs $\{o_1, o_2, \cdots, o_G\}$ from the old policy $\pi_{\theta_{\text{old}}}$ and

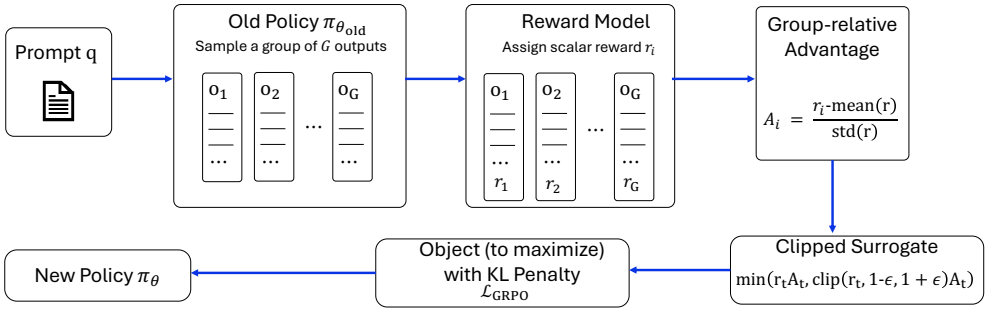


Figure 7.19: Workflow of GRPO-based RL.

optimizes π_θ by maximizing

$$\max_{\pi_\theta} \mathbb{E}_{q\sim D_{\text{rl}},\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(O|q)} \left[\frac{1}{G}\sum_{i=1}^{G}\min\left(\frac{\pi_\theta(o_i)}{\pi_{\theta_{\text{old}}}(o_i)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i)}{\pi_{\theta_{\text{old}}}(o_i)}, 1-\epsilon, 1+\epsilon\right)A_i\right) - \beta\,\mathbb{D}_{\text{KL}}\left(\pi_\theta\|\pi_{\text{ref}}\right)\right], \tag{7.31}$$

where ϵ and β are hyperparameters and the advantage A_i is computed from the rewards $\mathbf{r} = \{r_1, \ldots, r_G\}$ that a reward model assigns to the group, normalized within the group only:

$$A_i = \frac{r_i - \text{mean}(\mathbf{r})}{\text{std}(\mathbf{r})}. \tag{7.32}$$

Eliminating the value network reduces memory and stabilizes training. Although explored first for language models, the formulation transfers to spatial decision-making wherever several candidate plans or trajectories can be scored against one another, and it has become the standard recipe for reinforcement fine-tuning of spatial vision–language models on visual reasoning, navigation, multi-step assembly, and video spatial reasoning [1424, 1425, 1426, 1427].

Architectural Spatial Inductive Biases. The backbones above are domain-agnostic; what makes a policy *spatial* is the prior S in its architecture, and three families recur across the examples already met. **Map-based representations** give the agent an explicit allocentric memory: cognitive maps [1405] and Neural Map [1428] write observations into a spatially indexed latent grid, while Neural SLAM [1411] and Active Neural SLAM [1429] couple differentiable mapping with a hierarchical exploration policy. **Differentiable planning modules** embed dynamic programming in the network, as in the VIN / GPPN / QMDP-Net line discussed under value-based RL, or learn a value-equivalent model and plan by lookahead search, as in MuZero [148]. **Relational abstractions** replace flat feature maps with object- or graph-structured state, letting attention or message passing capture relations that transfer across layouts [1407]. In each case the RL objective is unchanged; the spatial competence comes from the prior, which is why these architectures generalize across geometries where unstructured policies overfit.

7.2.5.3 Benchmarks

Two complementary benchmark families shape the empirical landscape. Embodied navigation and exploration benchmarks stress long-horizon planning under partial observability in photorealistic 3D scenes: Habitat [94] is the dominant platform, paired with real-world scans from Matterport3D [340] and Gibson [93] for large, scene-diverse PointGoal and ObjectGoal navigation that exercises geometry-aware exploration and cross-scene generalization; SoundSpaces [46] adds audio-visual cues as a multimodal test of spatial state estimation; and SOON [1430] extends vision-and-language navigation to scenario-oriented object goals in unseen environments. Symbolic and relational benchmarks test whether structured priors yield combinatorial and

compositional generalization: PDDLGym[2] builds Gym environments from PDDL domains with relational observations and actions, and lightweight gridworld suites such as MiniGrid[3] support controlled evaluation of planning, memory, and compositional generalization; the differentiable planners (VIN, GPPN, QMDP-Net) are evaluated on procedurally generated gridworld mazes and POMDPs of increasing size, where the question is whether the architectural prior generalizes to unseen layouts rather than memorizing training mazes. Across both families the choice of RL algorithm matters less than the choice of prior: PPO has converged as the standard optimizer for embodied navigation, and differences between agents mostly reflect the map, planning module, or relational encoder on the policy, while on gridworld benchmarks the methods that embed planning structure consistently dominate unstructured DQN, with the largest gains where the task is most planning-shaped.

7.2.5.4 Challenges and Trends

Sample inefficiency comes first. Spatial tasks involve long horizons, sparse rewards, and combinatorial state spaces, so naïve exploration is prohibitively expensive; architectural priors such as learned maps, planning modules, and relational representations improve efficiency [1398, 1402], but current RL systems still need orders of magnitude more interaction than humans or classical planners to acquire robust spatial competence.

Generalization and transfer come second. Many agents perform well in their training environments and degrade sharply in unseen layouts, novel geometries, or altered dynamics, which suggests that much of the learned behavior rests on superficial correlations rather than abstract spatial structure; structured state representations, world models, and modular policy design [1403, 1405] help, but how to enforce spatial invariances and compositional generalization in RL in a principled way remains open.

Reward specification is the third. Spatial tasks often lack dense, well-defined rewards, especially when success depends on long-term plan quality, safety, or efficiency rather than immediate outcomes. Preference-based methods such as DPO [1418] and group-relative objectives such as GRPO [1423] learn from comparative feedback instead, at the price of reward-model bias, preference inconsistency, and the difficulty of aligning learned preferences with the true task objective in complex spatial domains.

The directions that follow from these three are hybrid methods that integrate learning with explicit planning or search [1398, 1400], structure-aware representations (object-centric, relational, graph-based) that scale spatial reasoning beyond grid-based and navigation-centric benchmarks [1407], and evaluation protocols that weight out-of-distribution generalization, robustness to environmental variation, and reasoning quality over performance on fixed benchmarks.

7.2.6 Multi-agent Spatial Reasoning

Where the previous five subsections study a single learning agent, multi-agent spatial reasoning (MASR) moves to the *joint* regime: multiple embodied or abstract agents perceive, communicate, and act within a shared environment to accomplish cooperative or competitive goals, and the structural prior now governs decentralized decisions over a shared spatial state. MASR links low-level sensorimotor streams (proprioception, occupancy grids, trajectories) to relational decision processes (joint planning, role assignment, coalition formation); its core problem is *credit assignment and coordination* under spatial constraints, that is, how agents with partial, decentralized observations reason over nearby entities, infer latent intent, and synthesize actions that produce coherent team-level behavior. Deep learning has moved the field from rule-driven coordination to end-to-end policies that integrate perception, communication, and control under partial observability and non-stationarity [1431, 1432, 1433], with autonomous driving and interactive motion forecasting over road geometry (WOMD [218], nuScenes [445]), multi-robot exploration, formation control,

[2] https://github.com/tomsilver/pddlgym
[3] https://github.com/Farama-Foundation/Minigrid

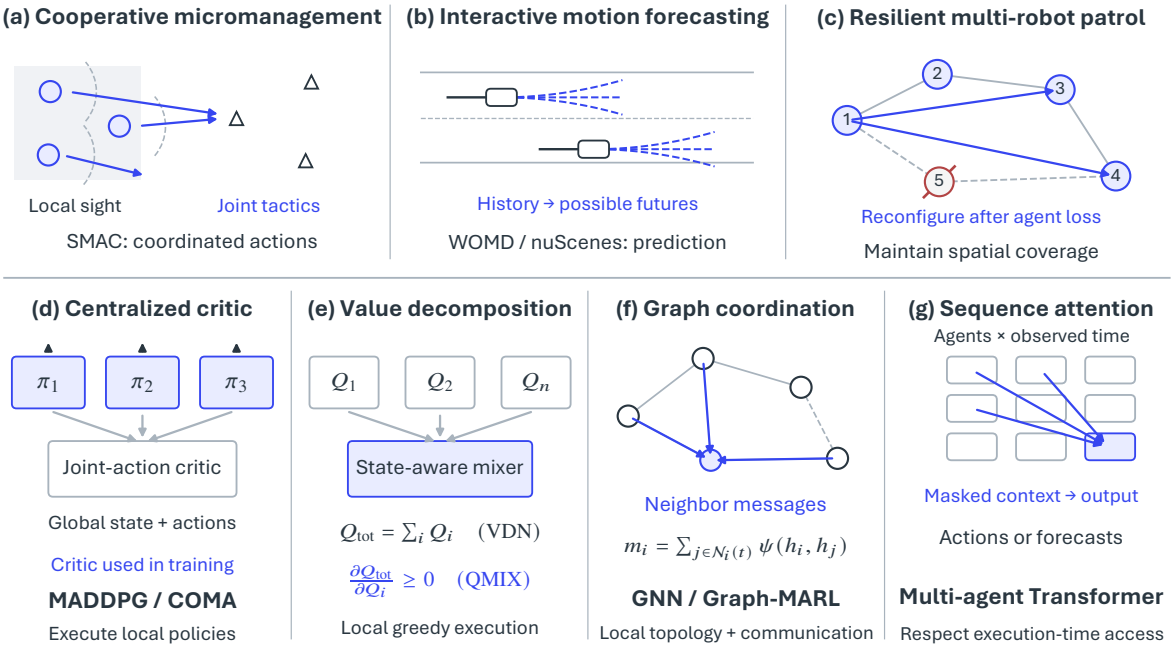


Figure 7.20: Multi-agent spatial reasoning tasks (top) and mechanisms (bottom). Top: (a) cooperative micromanagement under partial observation (SMAC [1435]), (b) interactive motion forecasting (WOMD, nuScenes [218, 445]), and (c) multi-robot patrol after agent loss [1434]. Bottom: (d) centralized critic (MADDPG, COMA [1431, 1432]), (e) value decomposition (VDN, QMIX [1437, 1433]), (f) graph coordination, and (g) attention across agents and time. Graph and sequence architectures serve prediction or control and do not by themselves imply decentralized execution. Red marks the lost agent; dashed trajectories are candidate futures and dashed graph links are intermittent interactions. Examples are schematic.

and patrol under attrition [1434], and real-time strategy and team-sports simulation [1435, 1436] as its representative domains. We stress at the outset that the canonical algorithms surveyed below (VDN, QMIX, COMA, MADDPG) are themselves spatially agnostic: they solve a generic cooperative Dec-POMDP and see geometry only through the environment and the reward. Explicit spatial structure enters chiefly through the benchmarks and through the graph-based family, where the interaction graph $\mathcal{G}_t$ makes neighborhood relations a first-class part of the policy.

7.2.6.1 Task Definition

We cast cooperative multi-agent spatial reasoning as a graph-structured Dec-POMDP

$$\mathcal{M} = (\mathcal{V}, \{\mathcal{A}_i\}, \mathcal{S}, P, R, \{\mathcal{O}_i\}, O, \rho_0, \gamma, T), \tag{7.33}$$

with a (possibly time-varying) interaction graph $\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t)$ derived from state (*e.g.*, proximity or visibility) or estimated locally. At time t, agent i receives a partial egocentric observation $o_t^i \sim O(\cdot \mid s_t, i)$ (RGB-D, LiDAR, occupancy, kinematics) and optional neighbor messages m_t^i; we treat $\tilde{o}_t^i = (o_t^i, m_t^i, \mathcal{G}_t^i)$ as the input available to agent i, with any bandwidth cost included in R. Agents act with decentralized, history-dependent policies

$$a_t^i \sim \pi_\theta^i(\cdot \mid h_t^i), \quad h_t^i = \mathrm{upd}(h_{t-1}^i, \tilde{o}_t^i), \tag{7.34}$$

the joint policy factorizes as $\pi_\theta(\mathbf{a}_t \mid \mathbf{h}_t) = \prod_i \pi_\theta^i(a_t^i \mid h_t^i)$, and the objective is

$$\max_\theta \ \mathbb{E}_{s_1 \sim \rho_0,\, \pi_\theta,\, P}\Big[\sum_{t=1}^{T} \gamma^{t-1} R(s_t, \mathbf{a}_t)\Big], \qquad \mathbf{a}_t = (a_t^1, \ldots, a_t^n). \tag{7.35}$$

In predictive variants, auxiliary heads output multi-agent forecasts or spatial fields with confidences for planning or losses; these are *not* environment actions. Settings differ along two further axes: online methods act in streaming settings under tight latency, whereas offline methods learn from logged interaction data (*e.g.*, driving datasets) with batch updates and conservative behavior constraints; and centralized training with decentralized execution (CTDE) exploits global state to train joint value functions or critics while enforcing decentralized policies at test time [1432, 1433], whereas fully decentralized learning never accesses global state, trading sample efficiency for deployability.

7.2.6.2 Methodology

Figure 7.20 summarizes the tasks above and the method families discussed below.

Classical Paradigms. Before learning, coordination relied on potential fields, formation keeping via graph Laplacians, and optimal multi-agent pathfinding with conflict-based search [1228], treated in §7.1.3; these methods offer strong guarantees but struggle with perception noise, combinatorial scaling, and model mismatch in open worlds.

CTDE and Value Decomposition. Two lines made CTDE work. Centralized critics resolve credit assignment and non-stationarity while keeping execution decentralized: COMA uses a counterfactual baseline computed by a centralized critic for decentralized actors [1432], and MADDPG extends deterministic policy gradients with agent-specific centralized critics [1431]. Value factorization instead decomposes the joint value: VDN sums per-agent utilities to approximate Q_{tot} for cooperative tasks [1437], and QMIX generalizes the sum to a monotonic mixing network that preserves individual-global argmax consistency with more representational capacity [1433]. These approaches made MARL scalable on SMAC [1435], where the value-decomposition line dominated the early cooperative leaderboard while the centralized-critic methods anchored other regimes (MADDPG on the particle environments, COMA on StarCraft micromanagement); the monotonic constraint and homogeneous-role assumptions limit expressivity, as the challenges below detail.

Graph Neural Networks for Coordination. Graph-based MARL encodes agents as nodes and interactions as edges, and message passing or attention over the neighborhood learns role-specialized coordination [1438]; this family improves sample efficiency and robustness under topology changes and communication limits, and MAGEC [1434] shows resilience to agent attrition and communication disturbance in multi-robot patrol by combining GNN encoders with PPO-style multi-agent optimization. Graph MARL targets a different axis from the leaderboard methods, robustness to topology disturbances rather than headline return, which exposes a gap between coordination-quality benchmarks (SMAC, MPE) and deployment-quality ones (multi-robot patrol, driving forecasting on WOMD and nuScenes) that standardized suites still under-represent.

Sequence Models and Transformers. Casting MARL as sequence modeling applies attention over multi-entity, multi-horizon contexts: the Multi-Agent Transformer [1439] maps the sequence of agents' observations to a sequence of actions with an encoder–decoder, using the multi-agent advantage decomposition to turn joint policy search into sequential decisions with linear complexity in the number of agents. Transformer controllers, often hybridized with graph layers, address long-range temporal and relational dependencies, unify perception and control, and support offline RL through trajectory modeling; learned memory alleviates partial observability, while quadratic attention and credit assignment over long horizons remain open.

7.2.6.3 Benchmarks

Benchmarks span two regimes: simulation suites that stress cooperation, partial observability, and communication learning under controlled physics and reward shaping, and real-world multi-agent trajectory datasets that test interactive prediction and motion forecasting at deployment scale. Table 7.4 summarizes the most widely used resources.

Table 7.4: Common MASR benchmarks spanning simulated coordination to real-world multi-agent trajectory datasets.

Dataset / Suite	Year	Scale	Action/Task Space
SMAC [1435]	2019	22+ micro scenarios	Discrete, decentralized
Multi-Agent Particle Env. (MPE) [1431]	2017	10+ 2D templates	Continuous/discrete
GRF [1436]	2020	Dozens of tasks / full-game	Continuous/discrete
nuScenes [445]	2020	1,000 scenes $\times$ 20 s @ 2 Hz	Trajectory prediction
WOMD [218]	2021	10^5 scenes $\times$ 20 s @ 10 Hz	Trajectory prediction
Neural MMO [1440, 1441]	2020	Persistent worlds; $O(10^2)$ agents	Long-horizon survival

Evaluation depends on whether the setting is control or prediction. Control is scored by mean episodic return $\mathbb{E}_\pi[\sum_t r_t]$, success or win rate, time-to-goal, collision and rule-violation rates, fairness or load-balancing indices for cooperative logistics, and communication overhead; prediction and planning are scored by spatial accuracy, from displacement errors over the forecast horizon (minADE, minFDE, miss rate on WOMD and nuScenes) to intersection-over-union between predicted and ground-truth occupied regions or trajectory tubes.

7.2.6.4 Challenges and Trends

Multi-agent spatial reasoning faces structural difficulties that distinguish it from the single-agent settings of §7.2.1–§7.2.5, organized here along four axes within the classical multi-agent paradigm.

Credit Assignment under Partial Observability and Long Horizons. The decentralized-execution constraint of the Dec-POMDP forces each agent to act on its own history h_t^i alone, so relating a local action a_t^i to a delayed team-level reward r_T across a horizon T produces an irreducibly noisy policy gradient. Counterfactual baselines such as COMA [1432] marginalize over alternative actions of agent i while holding the rest of $\mathbf{a}_t$ fixed and reduce gradient variance substantially, but they still rely on a centralized critic that, at scale, becomes a moving target for itself. Value decomposition (VDN, QMIX) [1437, 1433] side-steps the centralized critic by factorizing Q_{tot} across agents, yet the monotonicity constraint $\partial Q_{\text{tot}}/\partial Q_i \geq 0$ that guarantees decentralized *argmax* consistency cannot represent strictly non-monotonic cooperation games, a representational ceiling made explicit by the later non-monotonic factorizations QTRAN, weighted QMIX, and QPLEX, and visible on cooperative matrix games and SMAC's hardest scenarios [1442, 1443, 1444, 1435]. Under sparse reward shaping, policies trained against these signals tend to chase short-term spatial cues ("move toward the closest enemy", "follow the nearest teammate") rather than team-level objectives.

Scaling Laws of Joint Reasoning and Communication. Sequence-model multi-agent Transformers and dense graph message passing both incur $\mathcal{O}(n^2)$ complexity in the number of agents, plus additional quadratic terms across temporal contexts, which collides with the real-time latency budgets of multi-robot [1434] and autonomous-driving deployments [218, 445]. Sparse, learned interaction graphs are the natural mitigation, but they introduce their own credit-assignment difficulty: the edges $\mathcal{E}_t$ are themselves part of the learned policy, and gradients flowing through hard edge selection are biased or high-variance. Communication bandwidth is a related gap: in the formulation above any message m_t^i should carry a cost that appears in R, but the standard simulators (SMAC [1435], MPE [1431], GRF [1436]) idealize communication as free and noiseless, leaving little benchmark surface against which to push bandwidth-aware coordination.

Non-stationarity, Heterogeneous Roles, and Partner Generalization. Even in fully cooperative settings, intra-team policies co-evolve, so from the perspective of any single agent the environment is non-stationary [1431], the failure mode the centralized-critic family of MADDPG / COMA [1431, 1432] was introduced to address. The problem sharpens when teammates are heterogeneous, may drop out (the patrol-with-attrition setting of MAGEC [1434]), or are drawn from an unseen partner distribution at evaluation time, the zero-shot

coordination problem. Persistent open-world simulators such as Neural MMO [1440, 1441] push along this axis with procedurally generated maps and long-horizon survival, but standardized protocols for partner generalization and role specialization remain immature compared with the single-agent supervised benchmarks of §7.2.1 and §7.2.3.

Ambiguous Spatial Semantics and Sim-to-Real Transfer. The egocentric observations o_t^i that drive decentralized policies are partial and noisy: occluded teammates, uncertain affordances, and depth-discontinuity-induced map errors distort both prediction and control. Driving benchmarks such as WOMD [218] and nuScenes [445] expose the prediction side of this gap, but the closed-loop control side (whether the same forecasts survive when an agent acts on its own predictions) is comparatively under-evaluated. The sim-to-real gap on multi-robot systems is wider still: perception noise and actuation latency interact non-linearly with the coordination signal in ways that simulators rarely model, and resilience under realistic communication disturbance and agent attrition has only recently received first-class benchmark coverage [1434].

Three routes follow from these axes. Topology-aware sparse interaction (learned edges, hierarchical groupings, or attention masks parameterized by $\mathcal{G}_t$) preserves long-range coordination while scaling sub-quadratically, with the GNN+PPO patrol architecture of MAGEC [1434] as an early instantiation and graph encoders combined with sequence-model attention over short temporal windows as the natural next step. The monotonicity ceiling of VDN / QMIX [1437, 1433] can be relaxed by attention-weighted mixers, weighted projection schemes, or dueling-style decompositions that keep individual-global *argmax* consistency while admitting non-monotonic cooperation, so that COMA-style counterfactual credit assignment [1432] no longer depends on a joint-action critic that becomes the bottleneck. And explicit objectives for agent attrition and communication disturbance [1434], paired with failure-centric curricula and hooks for formal verification, move resilience from an emergent property of training to a first-class objective, decisive in safety-critical multi-robot and driving deployments [218, 445] where worst-case performance dominates the deployment decision; closed-loop evaluation suites that jointly report team return, safety, calibration, and latency are the missing measurement substrate, and the benchmarks for partner generalization, bandwidth-aware coordination, and sim-to-real transfer that would play the role SMAC and GRF played for CTDE are still under construction.

7.3 Foundation Model-driven Spatial Reasoning



OUNDATION model-driven spatial reasoning turns to large pre-trained models (LLMs, vision-language models, vision-language-action models, and large vision models) as the engine of spatial inference. The motivation is empirical: such models display a striking facility with spatial concepts and relations even without task-specific fine-tuning, and a large body of work now recruits them as the core of systems for spatial question answering, planning, and explanation.

What separates these approaches is the locus and granularity at which the foundation model intervenes. At one end the model is the reasoning core itself, answering spatial queries directly (LLM/VLM-centric reasoning) or additionally emitting low-level actions (VLA-based reasoning). Its reasoning can be drawn out step by step, as intermediate deductions in language (chain-of-thought) or as an executable program synthesized from the task (code and DSL generation). It can be coupled to external machinery such as planners, renderers, simulators, and perception modules (tool-augmented reasoning), or grounded in external spatial memory (retrieval- and memory-augmented reasoning). And it can be handed control of a closed interaction loop with the environment (agentic reasoning) or paired with a learned neural simulator of dynamics that serves as the spatial substrate (world-model-based reasoning).

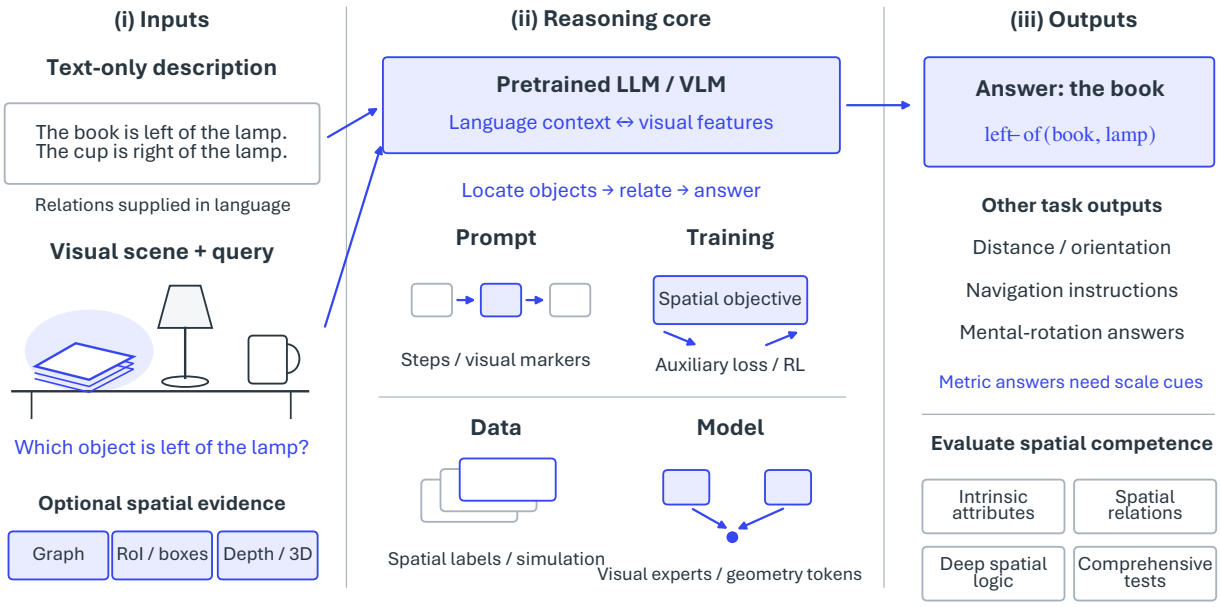


Figure 7.21: LLM/VLM-centric spatial reasoning, illustrated by the query "Which object is left of the lamp?": (i) text-only or visual inputs with optional spatial evidence, (ii) a pretrained LLM/VLM with four complementary, not sequential, intervention strategies (prompt, training, data, model), and (iii) the answer, other task outputs, and evaluation categories. Shading marks the queried region, not a measured attention map.

7.3.1 LLM/VLM-centric Spatial Reasoning

Large language and vision-language models have become a default substrate for spatial reasoning, not because they compute geometry explicitly, but because large-scale pretraining leaves them with a usable, if imperfect, prior over how objects are arranged, oriented, and related. This subsection examines that prior directly: what spatial competence text-only LLMs acquire from language alone, how visual grounding sharpens it in VLMs, and where both still fall short of the metric, viewpoint-consistent reasoning that physical tasks demand. The methods are organized by the intervention used to elicit or strengthen this competence, from prompting to architectural change.

7.3.1.1 Task Definition

Spatial reasoning, which embodied AI [56, 1445], autonomous driving [1446, 790], and spatial content generation [1447, 1448] all depend on, is cast here as inference over learned representations rather than explicit geometric computation: a pre-trained model $\mathcal{M}_\theta$ answers a spatial query q from context c,

$$y = \mathcal{M}_\theta(q, c), \qquad c \in \{\varnothing,\ I,\ I + \text{aux}\}, \tag{7.36}$$

where the context may be absent (a text-only LLM), an image or image set I (a VLM), or an image together with auxiliary evidence such as a scene graph, region features, or depth, and the output y is a relation, a metric estimate, an orientation, or a navigation plan. For text-only LLMs the spatial prior is built from statistical patterns in massive corpora [15, 1449, 1450]: such models infer that "sky" is above "ground", but the reasoning is *ungrounded*, with no mapping to physical reality or visual signal, and therefore prone to physically impossible inferences. VLMs ground the same prior by aligning the LLM with a visual encoder, and spatial competence then extends beyond recognizing objects and their intrinsic attributes to reasoning about relations among objects and to *spatial extrapolation*, inferring unseen states or running mental simulations from what is perceived [1451, 229].

7.3.1.2 Methodology

Interventions against the spatial deficiencies of foundation models fall into four levels, in increasing order of invasiveness: prompt engineering leaves the model untouched, training strategies update its weights, data-centric augmentation changes what it learns from, and model-centric enhancement alters the architecture.

Prompt Engineering. Prompting elicits latent spatial ability at inference time without weight updates. Textual prompting applies chain-of-thought reasoning [1452] so that the model decomposes a spatial task into intermediate steps (identifying reference objects before deducing relative positions), and inserting scene graphs or coordinates extracted by expert models into the prompt raises accuracy further [1453]. Visual prompting guides attention through markers superimposed on the image itself (masks, coordinate axes, keypoints), and in zero-shot settings such explicit visual cues can outperform fine-tuned models [1454, 1455].

Training Strategies. Contrastive pretraining objectives induce a "bag-of-concepts" behavior that under-supervises spatial structure, so stronger supervisory signals are introduced to enforce spatial alignment. Auxiliary losses such as image reconstruction or coordinate prediction compel the model to attend to pixel-level spatial detail rather than global semantics alone [1456]. Reinforcement learning with spatial reward functions (PPO or GRPO [1457, 1458]; view-consistency rewards [1459]) trains models to perform mental visual simulation or to produce more precise spatial descriptions, moving spatial training from imitation toward preference-based optimization.

Data-Centric Augmentation. Because high-quality spatially annotated data is scarce, augmentation drives much of the progress. Existing 2D/3D datasets are re-annotated with explicit spatial relations, metric distances, or orientations [1460], and simulation engines synthesize data with perfect geometric ground truth, which is especially effective for extrapolation tasks such as mental rotation and path planning where real-world annotation is impractical [1461, 1462].

Model-Centric Enhancement. Architectural changes target the fine-grained visual-spatial features that mainstream encoders miss. Because CLIP-style encoders [143] prioritize global semantics over local geometry, higher-resolution encoders or fused features from multiple visual experts (SAM [138, 482, 1134], DINOv2 [958]) sharpen the perception of shape and position; region-of-interest features or scene graphs supplied as additional input tokens decouple objects from their relationships and tighten fine-grained alignment [1463]; and for 3D reasoning, models ingest point clouds, voxel grids, or depth maps directly [1464] or lift 2D features into implicit 3D representations from multi-view images [1465]. A recent alternative internalizes 3D awareness through auxiliary supervision instead of extra inputs: SpatialSV [1466] forces hidden visual features to predict explicit 3D targets (depth, ray maps, point clouds) during training, injecting geometric priors at no inference cost.

7.3.1.3 Benchmarks

Evaluation has moved from general VQA to benchmarks that probe spatial capability along four levels [1451]. Intrinsic-attribute benchmarks test perceptual precision on single objects, with EgoOrientBench [1467] evaluating object orientation from an egocentric viewpoint and GeoMeter [1468] metric distance and dimension, and they show that even state-of-the-art models are weak at reference-free perception. Relational benchmarks, the most populated class, test the parsing of spatial relations through vision-centric questions such as relative position (CV-Bench [1469], What's Up [1470]), and this is where general-purpose VLMs peak. Deep-spatial-logic benchmarks go beyond visible relations: SRBench [1471] evaluates mental rotation and navigation, and MindCube [1472] the construction of an internal 3D mental model from multiple views. Comprehensive suites such as RealWorldQA [1473] and OmniSpatial [1474] cover all levels at once; a wider coverage map of these static probes appears in Table 7.5 of §7.3.7.

7.3.1.4 Challenges and Trends

Three obstacles limit what foundation models can be made to do spatially. Mainstream vision encoders such as CLIP optimize for global semantic alignment and capture coarse semantics at the expense of fine-grained geometric detail [1475, 1476], and the contrastive objective compounds this by inducing the "bag-of-concepts" behavior that under-supervises structural relations among objects [1477, 1478]. Training data with precise metric annotation (quantitative distance, 3D coordinates) is scarce, and existing datasets rarely support the extrapolation tasks that require reasoning beyond direct observation [1479, 1480]. Spatial reference is ambiguous: "to the left" may mean the viewer's left (egocentric), the object's left (allocentric), or another agent's left, and VLMs seldom switch frame of reference from context, which misaligns interpretation in embodied use [1479, 1481].

Two directions respond to these. Reinforcement learning with process rewards such as GRPO moves training from static VQA toward interaction, letting the model acquire spatial logic by trial and error in simulation or the real world [1457]. World-model-informed pretraining attacks the 2D-to-3D gap directly, learning physical regularities (gravity, collision) alongside image-text correspondence from synthetic corpora with exact geometric ground truth [202, 1461, 229]; the use of such world models at inference time is the subject of §7.3.8.

7.3.2 VLA-based Spatial Reasoning

Vision-language-action (VLA) models push spatial reasoning past description and into execution: the same latent competence that lets a model answer where an object sits must now drive an end effector toward it without violating depth, contact, or stability constraints. Spatial reasoning here is therefore inseparable from the agent's body and the physics of its workspace, and a plan that reads correctly on paper still fails if it ignores reachability or collision. The methods below are organized by their action-generation paradigm (autoregressive, diffusion-based, reinforcement-learning-driven, and hybrid).

7.3.2.1 Task Definition

A VLA model extends the reasoning core of §7.3.1 with an action head: given a visual observation o_t, a language instruction ℓ, and the robot's proprioceptive state s_t, it emits an executable action chunk

$$a_{t:t+H} \sim \pi_\theta(\cdot \mid o_t,\ \ell,\ s_t), \tag{7.37}$$

where each action encodes end-effector increments, gripper commands, joint targets, or pose waypoints that a controller converts into motion [1482, 1483]. Spatial reasoning under this paradigm is *embodied grounding*: language tokens must be anchored not only to visual features but to the three-dimensional geometry, affordances, and constraints of the workspace, so that the generated actions respect depth, stability, and contact rather than merely naming where objects are.

7.3.2.2 Methodology

VLA methods for spatial reasoning fall into four classes by their action-generation and reasoning paradigm.

Autoregression-Based Methods. Autoregressive methods cast spatial manipulation as next-token prediction over a unified token stream of discrete action tokens. Gato [149] showed that heterogeneous modalities can be jointly tokenized and trained in one autoregressive model; PointVLA [1484] injects point-cloud representations into a pretrained VLA for lightweight 3D reasoning without redesigning the architecture; and TraceVLA [1485] adds visual trace prompting, explicit spatiotemporal cues that improve long-horizon spatial awareness and task consistency during generation.

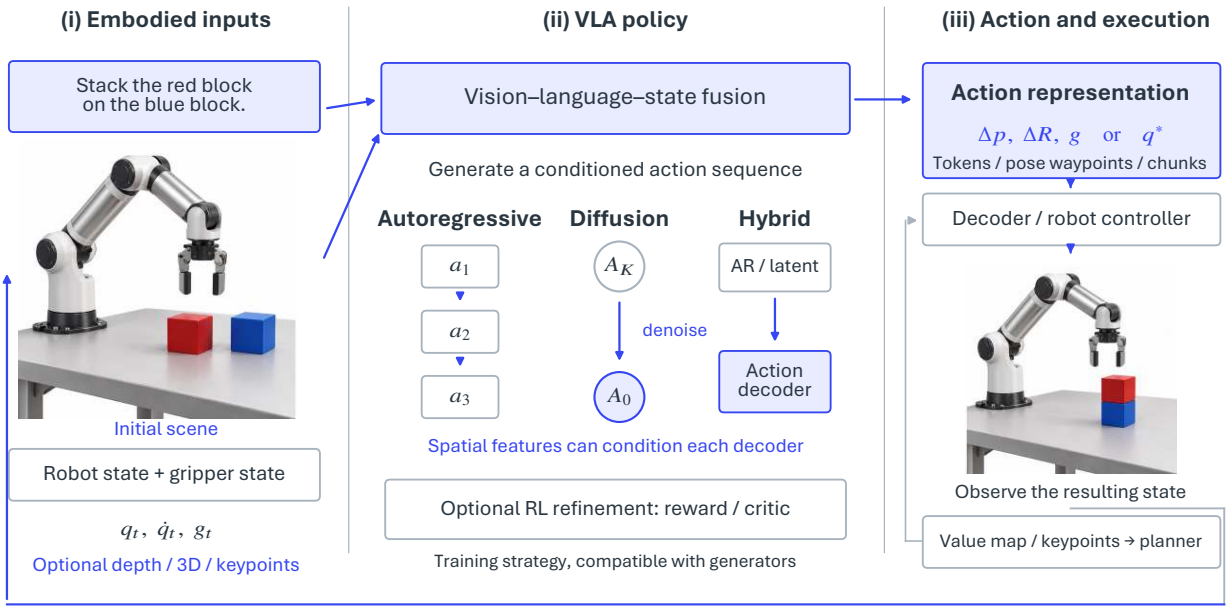


Figure 7.22: VLA-based spatial reasoning as a perception–action loop: (i) embodied inputs, (ii) a VLA policy that generates actions autoregressively, by diffusion (denoising A_K to A_0), or with a hybrid, plus optional RL refinement, and (iii) an action representation executed by a controller, with environment feedback closing the loop. $\Delta p, \Delta R, g, q^*$ are end-effector translation and rotation increments, gripper command, and joint targets. The robot scenes illustrate the initial state and intended outcome, not experimental observations.

Diffusion-Based Methods. Diffusion methods model action generation as conditional denoising, which suits trajectories that must respect physical constraints and uncertainty. SE(3)-DiffusionFields [1486] runs the diffusion process in $\mathrm{SE}(3)$ pose space and jointly optimizes grasping and motion planning; 3D Diffuser Actor [1487] conditions the denoiser on structured 3D scene representations so that manipulation trajectories stay geometrically consistent; and ForceVLA [1488] brings force and tactile feedback in through a mixture-of-experts design for contact-rich manipulation.

Reinforcement Learning-Driven Methods. Reinforcement learning refines VLA policies through interaction with the environment, particularly on long-horizon tasks (the broader treatment of RL with spatial inductive biases is §7.2.5). VIP [1489] learns a goal-conditioned, action-independent value function that implicitly encodes spatial distance to assess task progress; NaVILA [1490] couples VLA representations with RL policies so that quadruped robots navigate and avoid dynamic obstacles on difficult terrain; and SafeVLA [1491] adds safety critics that constrain high-risk behavior during manipulation.

Hybrid Architectures. Hybrids combine generative mechanisms and add explicit spatial representations to strengthen 3D reasoning. 3D-VLA [201] joins autoregressive language modeling with diffusion-based action prediction inside a generative 3D world model; VoxPoser [1492] has an LLM compose 3D voxel value maps, separating goal interpretation from geometric planning; ReKep [1493] models spatiotemporal dependencies as relational keypoint constraints for precise manipulation; and RoboPoint [1494] predicts affordance maps that encode where actions are feasible, as a prior for downstream planning and control.

7.3.2.3 Benchmarks

Evaluation rests on two complementary resources: generalization-oriented benchmarks in simulation and large-scale real-world embodied datasets. VIMA-Bench [1495] measures spatial generalization through a four-level protocol that runs from random object placement to fully novel task compositions; ALFRED [1307] poses

long-horizon household tasks that combine high-level goals with low-level instructions; and CALVIN [1496] and LOTA-Bench [1497] test language-conditioned manipulation across environments, scoring generalization to unseen objects and configurations. Among datasets, Open X-Embodiment [1498] is the most comprehensive, integrating 22 robot datasets from 21 institutions and 527 skills to enable cross-platform generalization; BridgeData [1499] spans 71 tasks in 10 environments and shows that joint training on limited out-of-domain data raises success rates substantially; and RH20T [1500] adds vision, force, and audio across 147 tasks, suited to one-shot learning and experience-based policy transfer. Across these suites performance falls sharply along every generalization axis (L1 to L4 in VIMA-Bench, seen to unseen splits in ALFRED, single to multi-step chains in CALVIN), while data scale and embodiment diversity are the most reliable lever for cross-platform transfer and multimodal sensing (force, audio) becomes decisive where vision alone underdetermines the contact configuration.

7.3.2.4 Challenges and Trends

The generalization drop above reflects a **semantic–physical misalignment**: VLA systems inherit their competence from LLMs and VLMs that excel at semantic abstraction and cross-modal alignment but have had little exposure to physical interaction, so instructions are often interpreted correctly and executed incorrectly. The same gap appears as **pseudo-interaction**: actions are generated from statistical co-occurrence in the training data rather than from an understanding of environment dynamics or causality, so when the environment is unfamiliar or changes quickly the model cannot revise its strategy from sensory feedback.

Two further limits are practical. **Long-horizon consistency** is unsolved for autoregressive and diffusion policies alike: over extended action sequences the spatial representation drifts or hallucinates, and existing embodied datasets, including Open X-Embodiment, are dominated by tabletop manipulation and grasping, so environmental diversity and large unstructured scenes are under-represented. **Inference latency** compounds across autoregressive decoding of large Transformers, and for high-frequency tasks such as dynamic grasping or mobile navigation the compute and memory demands remain prohibitive on embedded platforms.

The directions that follow are a unified token stream over vision, language, and action that lets VLA systems grow into proto world models (the world-model view is developed in §7.3.8); active perception and causal reasoning, in which the agent probes its environment and updates its policy from feedback, as the answer to pseudo-interaction; and virtual-real data synthesis (high-fidelity simulation, synthetic generation, multi-robot data sharing) as the route to embodied data at the scale that text corpora provided for language models.

7.3.3 Chain-of-Thought for Spatial Reasoning

Spatial reasoning poses unique challenges for large language models (LLMs), as it requires the manipulation of geometric relations, perspective-taking, and dynamic transformations: skills that are not purely symbolic but grounded in embodied cognition. Chain-of-Thought (CoT) prompting [78], originally developed as a mechanism to elicit intermediate reasoning steps in arithmetic and commonsense tasks, has therefore been imported into spatial reasoning not as a generic recipe for longer answers, but as a way to externalize the intermediate spatial commitments that a model must keep consistent: which object is left of which, which frame of reference is active, which action changes the scene, and which hypothesized 3D configuration explains the observation. The evidence is mixed. In some settings, explicit rationales help decompose a spatial query into auditable subgoals; in others, they induce a fluent but ungrounded verbal trace that makes spatial performance worse [1501].

7.3.3.1 Task Definition

The task is to answer a spatial query q about a scene or environment, given context c that may be purely textual, visual, 3D, temporal, or embodied. A CoT-style reasoner augments the answer with an intermediate

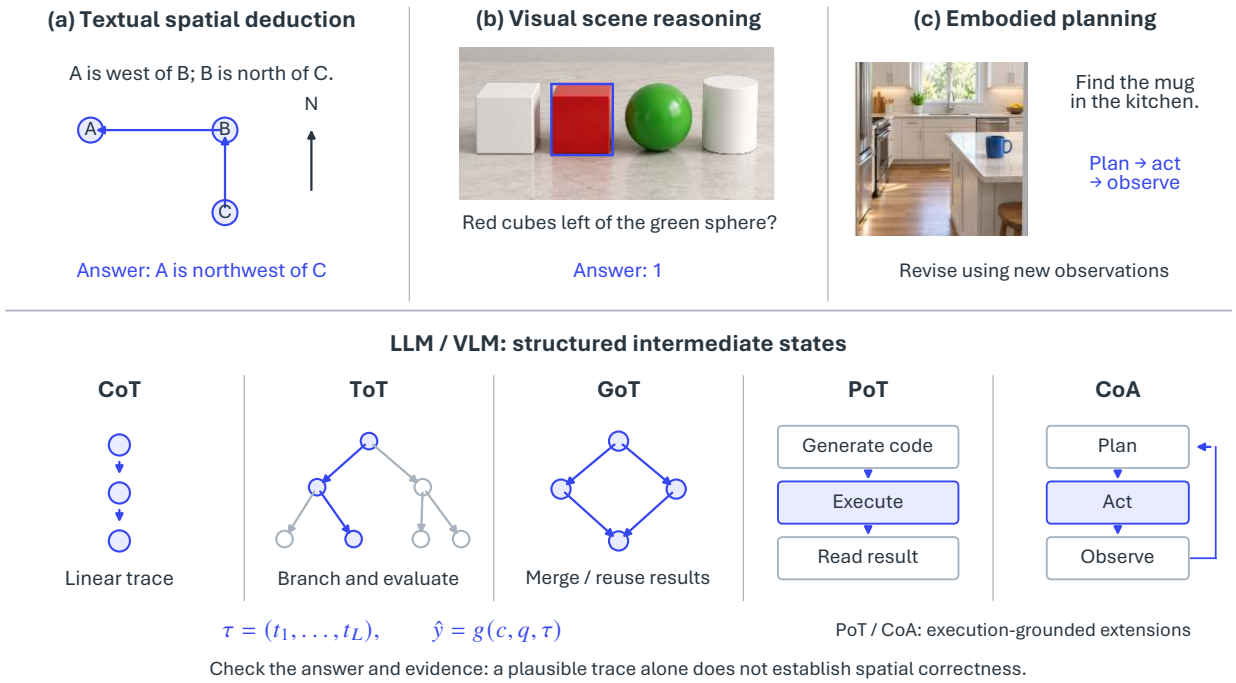


Figure 7.23: Spatial reasoning tasks (top) and intermediate-trace structures (bottom). Top: (a) textual deduction in a north-up frame, (b) visual counting, where the boxed red cube is the only one left of the green sphere, and (c) embodied planning revised by new observations. Bottom: CoT, ToT, and GoT differ in trace topology; PoT and CoA ground the trace in program execution and environment interaction. $\tau = (t_1, \ldots, t_L)$ is the intermediate trace and $\hat{y} = g(c, q, \tau)$ the final answer. Scenes and trace diagrams are illustrative, not recorded model computation.

trace

$$\tau = (t_1, \ldots, t_L), \qquad \hat{y} = g(c, q, \tau), \tag{7.38}$$

where each t_ℓ records a partial spatial commitment: a relation, coordinate estimate, subgoal, viewpoint change, object binding, or predicted action effect. The trace is useful only insofar as it improves the grounded answer $\hat{y}$ or makes the failure mode inspectable; many apparent "reasoning steps" are not grounded in metric or perceptual evidence and merely rationalize a final guess, so the central question is not whether a model can produce more text, but whether the trace helps preserve spatial invariants across transformations, viewpoints, time steps, and actions.

7.3.3.2 Methodology

CoT, Tree-of-Thought (ToT) [1502], and Graph-of-Thought (GoT) [1503] differ mainly in the topology of the intermediate trace: CoT is a single path through partial spatial commitments; ToT branches over alternative hypotheses or subplans and uses a value function to prune; GoT allows shared sub-derivations, such as reusing a computed room layout or object graph across several queried relations. All three search over partial reasoning states with a proposal mechanism, an evaluator V, and a halting rule H,

$$\tau^\star = \arg\max_{\tau} V(\tau) \quad \text{s.t.} \quad t_{\ell+1} \in \mathrm{Propose}(t_{\le \ell}), \;\; H(\tau) = 1, \tag{7.39}$$

and differ in whether the search structure is a path, a tree of alternatives, or a graph of reusable sub-results, and in whether the trace stays verbal or becomes executable through programs (Program-of-Thought, PoT [1504]) or environment-changing actions (Chain-of-Action, CoA [1505]). The use cases below organize how these traces are deployed.

Text-Only Spatial Deduction. In linguistic spatial puzzles, route descriptions, and relative-position QA, the model must maintain a latent arrangement without visual feedback. CoT can expose intermediate bindings (*e.g.*, "A is north of B; C is west of A") and reduce one-shot relational confusion [1506], while ToT-style branching can preserve competing layouts until enough constraints have been applied. The persistent failure mode is coordinate drift: even with explicit steps, LLMs lose track of the origin, orientation, or accumulated displacement in textual mazes and grid-navigation tasks [1507], so text-only CoT is most reliable when the trace is tied to a compact symbolic artifact (a table, coordinate sketch, or adjacency list) rather than to unconstrained prose.

Multimodal Scene Reasoning. In image- or video-grounded QA, a CoT trace must bind language to perceptual evidence: which object instance is referred to, what relation is visible, which is occluded, and which viewpoint assumption is made. Free-form rationales can verbalize a relation that the image does not support, so recent work moves toward grounded intermediate artifacts (structured layouts, object bindings, diagrams, or executable module calls): SCoT-style planners use layout-coordinate traces to guide diffusion models on the generation side [1508], and program-mediated systems call perception modules rather than describe them [1509], the paradigm of §7.3.4. Here ToT and GoT serve less as generic search algorithms than as hypothesis-management devices that let a model compare object correspondences, camera frames, or depth-order explanations before committing to an answer.

Embodied Planning and VLA Reasoning. In navigation, manipulation, and vision-language-action tasks, spatial reasoning unfolds over time, and the trace must connect perceptual state, subgoal choice, action feasibility, and feedback from the environment. CoT-style subgoals support long-horizon planning in navigation [1510, 1511], embodied instruction following [1512, 1513], and VLA models that expose intermediate visual or action commitments: CoT-VLA pairs reasoning with predicted visual states [1514], ACoT-VLA formulates the trace in action space through coarse action intents and reference trajectories [1515], and scene graphs stabilize the trace by giving the model a persistent substrate for object identities and relations [1516].

7.3.3.3 Benchmarks

Evaluation mirrors the three use cases. Spatial puzzles, relative-position QA, and textual maze navigation are evaluated with prompt-based CoT on the suites of [1506, 1507]; visual question answering and scene-grounded compositional queries are evaluated under CoT with program execution (§7.3.4), where the rationale's correctness is mediated by program output rather than by the rationale string; and long-horizon planning with sensory feedback is evaluated through CoT-augmented navigation (NavCoT [1510], NavR1 [1511]), CoT-augmented instruction following (ThinkBot [1512], EmbodiedGPT [1513]), visual- and action-trace supervision for VLA models [1514, 1515], and scene-graph-augmented embodied reasoning [1516].

7.3.3.4 Challenges and Trends

The transfer of CoT to spatial intelligence is uneven. **Coordinate hallucination** persists under simple textual CoT: even with ToT-style beam expansion, prompt-based planners mistrack absolute positions across multi-step chains in maze and grid-navigation problems [1507]. **Ungrounded supervision** can make things worse: textual CoT applied without a structured or rationale-grounded training signal degrades spatial VLM performance, because the model is forced to verbalize a reasoning trace it does not possess [1517, 1501], especially when the prompt rewards plausible explanation length over geometric correctness. **Search cost** scales super-linearly in solution depth for ToT and GoT (and in the branching factor for ToT), which limits real-time or on-device deployment. **Trace–action mismatch** arises in embodied settings when a VLA model produces a coherent verbal rationale while its action head follows a different latent policy; action-aware CoT aligns rationales with action tokens and visual predictions [1515], but reliable attribution between the visible trace and the executed behavior remains unresolved.

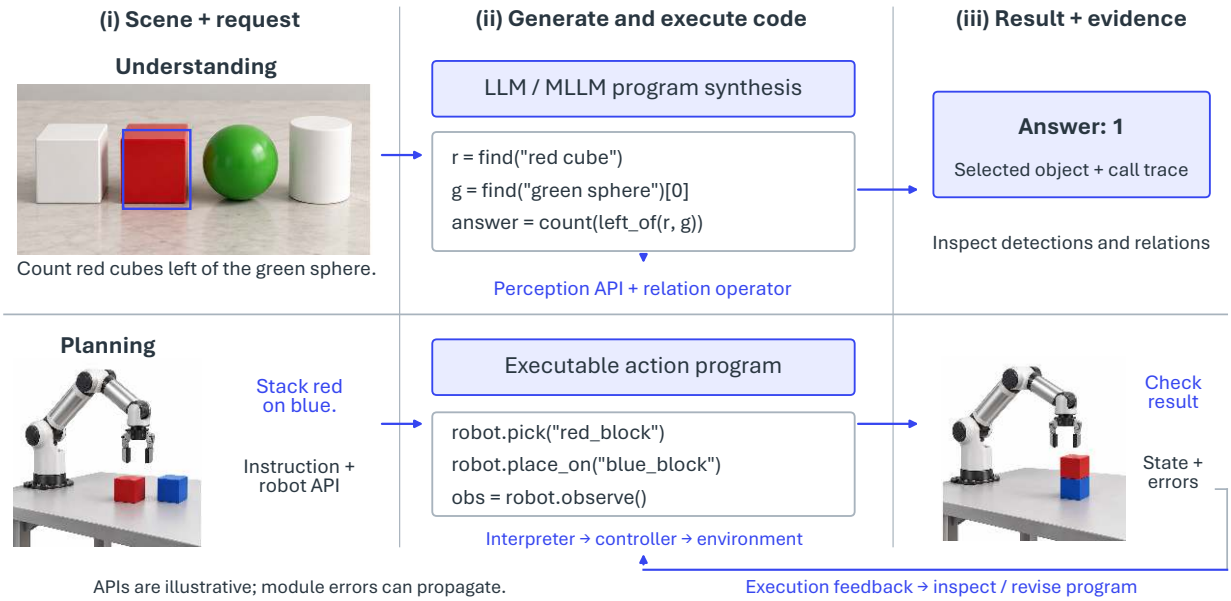


Figure 7.24: Code/DSL generation for spatial reasoning in two modes: understanding (top), where a counting query becomes a program over perception calls whose call trace is inspectable, and planning (bottom), where an instruction becomes robot API calls whose execution feedback drives program revision. Function names are illustrative pseudocode, not a specific system's API, and the scenes do not show experimental execution.

Grounded rationale supervision pairs CoT with explicit spatial signals (scene graphs [1516], depth maps, 3D coordinate frames, action traces, or layout-coordinate traces for spatial generation [1508]) so the rationale is verifiable rather than free-form. A complementary line makes the trace *execution-grounded*: PoT-style traces compile and run against a geometric engine or vision–language tool, with execution outputs serving as a truth signal, the natural bridge from CoT to code / DSL generation in §7.3.4; for embodied agents the parallel bridge runs from CoT to CoA and the agentic orchestration loops of §7.3.7.

7.3.4 Code/DSL Generation for Spatial Reasoning

Code/DSL generation treats spatial reasoning as program synthesis: an LLM or MLLM translates a natural-language task into an executable program, and a more deterministic system (a robot control API, a physics simulator, a library of vision functions) executes it. Relative to the explicit structured reasoners of §7.2.2, the specialized parser and fixed module set are replaced by the world knowledge and reasoning ability of the LLM, so the operation set is no longer confined to a predefined vocabulary; relative to the chain-of-thought traces of §7.3.3, the trace is a program whose execution against an external environment supplies a truth signal that a free-form rationale cannot. Figure 7.24 shows the two operating modes.

7.3.4.1 Task Definition

Given a high-level, often ambiguous, natural-language query q and access to an API $\mathcal{A}$ (a robot's control interface, a programming-language interpreter, or a set of vision-language functions), the LLM acts as a high-level planner that emits a program and offloads execution to the external system $\mathcal{E}$,

$$P = \mathrm{LLM}(q, \mathcal{A}), \qquad y = \mathrm{Exec}(P; \mathcal{E}), \tag{7.40}$$

where P is written in a general-purpose language such as Python or in a domain-specific language and composed of calls that manipulate objects, query spatial relations, or control an agent, and the output y is a change in the environment's state (a robot moving an object), an answer to the query, or both.

7.3.4.2 Methodology

Code/DSL for Understanding. For static scenes the LLM generates code that dissects a visual query by composing calls to perception modules, so the program is itself an explicit, verifiable reasoning chain. VisProg [1365] prompts an LLM to produce a program in a high-level DSL that sequentially invokes modules for image understanding, image manipulation, and knowledge retrieval, implemented either heuristically in Python or by neural models such as OWL-ViT [422] and Stable Diffusion [185]. ViperGPT [1131] instead exposes two classes, `ImagePatch` and `VideoSegment`, whose methods (`find`, `exists`, `compute_depth`, `verify_property`) each call a pretrained model, and generates a Python script that chains these calls to answer a complex query about an image or video, composing existing vision tools zero-shot. Both depend on a static, human-defined DSL; VADAR [1518] removes that limit by synthesizing the API implementations dynamically, in an API-generation stage followed by program synthesis, and LEFT [348] makes the approach learnable by having the LLM generate differentiable logic programs that chain trainable perception networks, so concept grounding can be learned where no pretrained module exists. The same recipe lifts to 3D: LLM-Grounder [1519] decomposes a grounding query into Python code that calls 3D grounding tools such as OpenScene [39] or LERF [369] to locate objects in a scene.

Code/DSL for Planning. Beyond static understanding, the generated code can serve as a policy that acts in an environment, with the LLM as high-level planner and the interpreter handling deterministic low-level execution. Code as Policies [1366] has the LLM write Python that calls a robot's control API, so that "put the apple in the bowl" becomes *robot.pick('apple')* followed by *robot.place('bowl')*, with implicit steps and preconditions supplied by the model's common sense. In simulation, Voyager [150] explores Minecraft by writing, executing, and refining its own code: it prompts the LLM for code to achieve a goal, receives environment feedback (success, failure, new inventory items), debugs accordingly, and adds successful routines to a growing library of reusable skills.

7.3.4.3 Benchmarks

Code/DSL generation for spatial reasoning is typically evaluated under three benchmark families that mirror the two-mode split of the §Methodology above. For 2D scene understanding, methods are evaluated on compositional visual question answering (GQA [839]), referring expression comprehension (RefCOCO / RefCOCO+ / RefCOCOg [1082, 1083, 1088]), and image-editing benchmarks. For 3D scene understanding, evaluation has converged on 3D referring expression and 3D visual grounding benchmarks (*e.g.*, ScanRefer [373], Multi3DRefer [1107], ScanQA [41]), which exercise both the perception-API composition and the long-tail of object-attribute reasoning. For embodied planning, the canonical evaluation suites are ALFWorld [1520] and VirtualHome [1521] (long-horizon household tasks) and Minecraft (open-world goal acquisition, as exemplified by Voyager [150]); the headline metric in this family is task-success rate over multi-step goals rather than single-step accuracy. Across all three families, the recurring methodological observation is that code-generation methods inherit the accuracy of their underlying perception or control modules, so benchmark performance tracks the module suite more tightly than it tracks the LLM backbone.

7.3.4.4 Challenges and Trends

The interpretable trace comes with three recurring limits. A **fixed, human-authored DSL** (ViperGPT's `find` / `compute_depth` / `verify_property`) restricts the model to a predefined operation set, so spatial concepts outside the API are systematically out of reach; VADAR's dynamic API synthesis [1518] is a response but does not solve the coverage problem. **Brittle execution with silent failure** follows from the dependence on perceptual modules (OWL-ViT [422], OpenScene [39], LERF [369]): an upstream error propagates through the program and yields a plausible but wrong answer with no diagnostic signal. **Sample-inefficient debug loops** limit self-refinement agents such as Voyager [150] and Code as Policies [1366], whose LLM calls per task scale poorly in high-dimensional or stochastic environments.

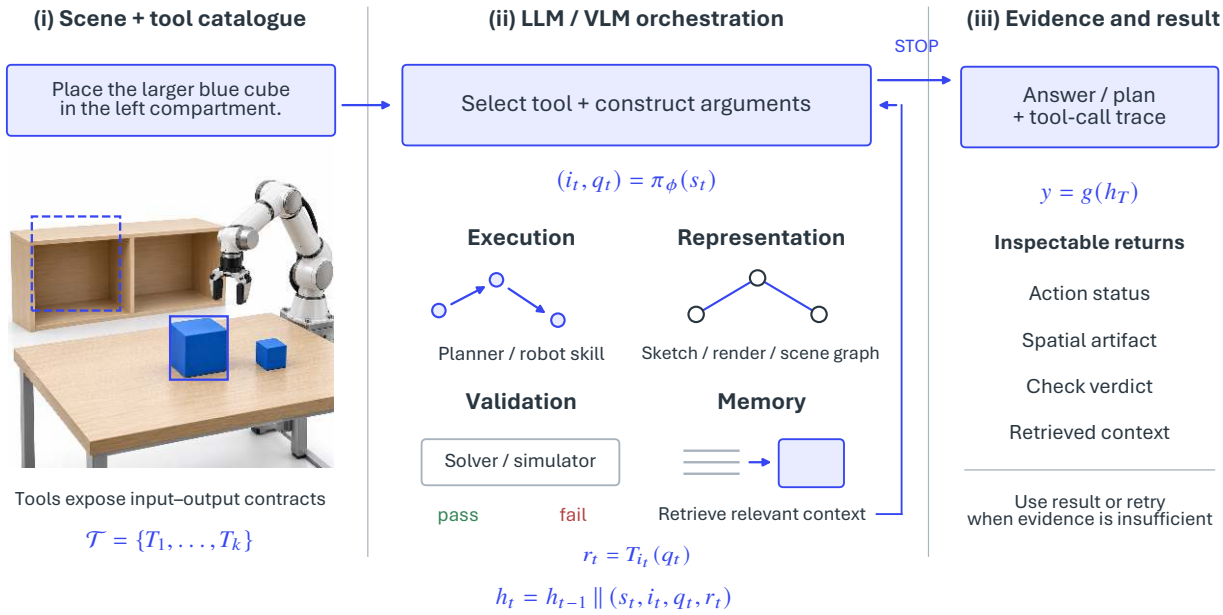


Figure 7.25: Tool-augmented spatial reasoning as iterative orchestration over black-box tools: (i) scene, instruction, and tool catalogue $\mathcal{T}$, (ii) an LLM/VLM that selects tool i_t with arguments q_t, receives result r_t, and appends to history h_t until STOP, and (iii) the answer or plan with its tool-call trace. Execution, representation, validation, and memory are tool roles, not sequential stages. Solid and dashed boxes mark the selected cube and target compartment; the scene is a generated illustration with annotated, not detected, overlays.

Two directions follow. PoT-style execution-grounded rationales are being folded back into chain-of-thought training, with the generated program serving as both rationale and verification signal, and *learnable DSLs* such as LEFT [348] replace fixed perception with differentiable logic programs whose components are trainable, pointing toward a DSL that adapts under task supervision rather than being hand-engineered.

7.3.5 Tool-augmented Reasoning

Tool-augmented reasoning in spatial intelligence is the paradigm in which an LLM or VLM acts not as an isolated solver but as an orchestrator over a catalogue of external operators. These operators may be planners, renderers, CAD systems, physics simulators, retrieval engines, perception modules, or metric estimators. The section boundary is deliberately operational: tools are treated as black boxes with input–output contracts, and the central research problem is the orchestration policy that decides when to call them, how to parameterize the call, and how much to trust the returned result. When the executable program itself becomes the reasoning trace, the detailed treatment belongs to code / DSL generation (§7.3.4); when persistent external memory becomes the learned object, it belongs to retrieval- and memory-augmented reasoning (§7.3.6); when the loop becomes closed-loop control, it becomes agentic spatial reasoning (§7.3.7); and when the tool is a learned predictive simulator, the specialized discussion is world-model-based reasoning (§7.3.8).

7.3.5.1 Task Definition

A tool-augmented reasoning system is formally defined as a tuple $(\mathcal{M}_\theta, \mathcal{T}, \pi_\phi)$, where $\mathcal{M}_\theta$ denotes the reasoning core, $\mathcal{T} = \{T_1, \ldots, T_k\}$ is a catalogue of external tools, and each tool $T_i : \mathcal{I}_i \to \mathcal{O}_i$ is exposed only through a structured input–output interface. The orchestration policy π_ϕ governs tool selection, argument construction, response parsing, retry / fallback behavior, and feedback integration. Given an input $x \in \mathcal{X}$, the system maintains a time-ordered reasoning history h_t (initially $h_0 = \varnothing$) and proceeds by the iterative

update

$$\begin{aligned} s_t &= \mathcal{M}_\theta(x, h_{t-1}), \quad (i_t, q_t) = \pi_\phi(s_t), \\ r_t &= T_{i_t}(q_t), \qquad\qquad h_t = h_{t-1} \,\|\, (s_t, i_t, q_t, r_t), \end{aligned} \tag{7.41}$$

where s_t is the intermediate reasoning state, (i_t, q_t) is the selected tool-and-query pair, r_t is the returned result, and $\|$ denotes appending to the interaction history. The loop halts when the policy or model emits a STOP decision, after which an output decoder g produces the final answer $y = g(h_T)$. Both the reasoning core and the tool internals are black-box; what is exposed, optimized, and evaluated is the interface history and the policy that controls it.

7.3.5.2 Methodology

Two design choices interact: the *tool catalogue* determines which black-box capabilities are available, and the *orchestration policy* determines how they are sequenced, checked, and folded back into the model's spatial answer. The catalogue roles below are complementary, not sequential.

Execution-Augmented Reasoning. Execution tools bridge the semantic–geometric gap between high-level language and low-level feasible action [1522]: they expose callable primitives such as motion planners, robot skills, affordance estimators, navigation controllers, or simulator actions, and the orchestrator translates a spatial intention into a sequence of calls, monitors whether the result satisfies the goal, and recovers when a tool fails. Representative systems include affordance-grounded planning, where language-model outputs are modulated by value functions for skill feasibility [1311]; API- or code-mediated manipulation, where perception and control primitives are composed through generated calls [1366, 1492]; and action-token models that fold robot actions into the model vocabulary [1523, 97, 56]. SpatialClaw [1524] studies the action interface itself: a stateful Python kernel pre-loaded with perception and geometry primitives lets the agent execute one analysis step, observe intermediate outputs, and adapt the next call rather than committing to a full static program. Closed-loop feedback and hierarchical decomposition [1525, 1526, 1527, 1528, 1529] push this family toward agentic orchestration.

Representation-Augmented Reasoning. Representation tools externalize spatial working memory through sketches, 3D renders, CAD systems, scene graphs, game engines, or visualization buffers, motivated by the observation that transformers struggle to manipulate fine-grained geometry internally [1530]. Externalization reduces hallucination by moving spatial computation out of parametric memory [1530] and enables iterative self-correction through visual feedback loops [1457]; the claim is that spatial reasoning improves when models can *see what they think*, as in Vision-Augmented Prompting [1531], which interleaves image synthesis with chain-of-thought so that textual and visual reasoning can be reconciled. Recent spatial-tool agents make the evidence acquisition active: PERIA [1532] pairs perception tools that expose symbolic or spatial evidence with interaction tools that manipulate the visual context, learning multi-tool behavior from synthesized trajectories and reinforcement learning, and S-Agent [1533] casts the VLM as a semantic planner over a hierarchy of spatial tools that lift 2D object evidence into 3D geometric support while accumulating scene and agent memory across multi-view images or video.

Validation-Augmented Reasoning. Validation tools check whether intermediate outputs are logically, causally, geometrically, or physically consistent: the orchestrator translates a candidate answer or plan into a verifiable representation and queries a planner, solver, judge, or simulator. Generated spatial plans can be encoded as constraint-satisfaction problems or PDDL specifications and validated by symbolic planners or SMT solvers [1534, 22, 1535], which checks preconditions, geometric relations, and causal effects; physics simulators such as MuJoCo [1187] and Isaac Sim[4] act as physical oracles for the feasibility of proposed movements, collisions, and trajectories [1536, 1492, 229]; neuro-symbolic systems integrate Answer Set Programming solvers that translate natural language into formally verifiable logical rules for relational

[4] https://developer.nvidia.com/isaac/sim

and topological coherence [1537, 1538]; and plan-critic mechanisms [1539, 1540] or an LLM-as-a-Judge that evaluates and refines the plans of a primary planner close a self-correcting loop [1541].

Memory-Augmented Reasoning. Memory tools expose external storage and retrieval through a black-box query interface: the orchestrator asks for relevant prior observations, spatial subgraphs, object histories, or task episodes and decides how to incorporate what is returned, which addresses the *"crisis of context"* in long-horizon robotic tasks whose sensory streams exceed the finite context window of an LLM. At this level the memory may be a hierarchical cognitive architecture such as RoboMemory or an open-vocabulary 3D scene graph such as ConceptGraphs (both treated in §7.3.6), a classical 3D or dynamic scene graph [341, 1542], or a system-level memory manager: MemOS [1543] introduces paging and scheduling that swap spatial contexts in and out of the reasoning core, treating the LLM as a CPU and the tool as RAM, and spatio-temporal retrieval engines such as ReMEmbR [1544] and MemoTime [1545] index video history in vector databases so that an agent can answer "where did I leave the keys five minutes ago?".

Predictive Tools and World-Model Feedback. Some catalogues include tools that simulate counterfactual or future spatial states rather than observe or verify the present one: generative video models, learned dynamics models, and predictive latent models are called as expensive but powerful instruments for "what would happen if" or "what would this look like from another viewpoint" [1536, 1546]. Here they are one class of callable tool; their feedback forms, imagination cost, and fidelity are the subject of §7.3.8.

7.3.5.3 Benchmarks

Evaluation asks whether the orchestrator selects useful tools and converts their outputs into better spatial decisions. For execution-augmented tool use the canonical suites are SayCan [1311] (LLM-grounded household manipulation), ALFWorld [1520] and VirtualHome [1521] (long-horizon text-and-embodied environments), and Voyager-on-Minecraft [150] (open-world skill accretion through code-as-policy). For spatial-reasoning-specific tool use, SpaceTools [1547] targets the metric gap between VLM perception and physical manipulation, and Cambrian-S [1546] introduces the spatial-supersensing framework that exercises tool augmentation from perception (Stage I) to predictive world modeling (Stage IV). Validation- and memory-augmented reasoning are evaluated on the memory and agentic benchmarks of §7.3.6 and §7.3.7, and predictive tools on those of §7.3.8, where the central question is whether imagined evidence improves the final answer enough to justify its cost.

7.3.5.4 Challenges and Trends

The recurring limitations originate in the LLM–tool interface rather than in any individual tool. The **exploration bottleneck** is combinatorial: each tool adds an action dimension and tool composition multiplies them, so naive reinforcement learning over tool use is inefficient; Double Interactive Reinforcement Learning (DIRL) [1547] warm-starts from expert demonstrations before open exploration, but the scaling of the action space itself is unresolved. The **metric gap** moves rather than disappears: VLMs excel at semantic perception but fail at the precise metric estimation manipulation requires, and offloading geometry to depth estimators and pose solvers lets the LLM reason over measurements instead of hallucinations [1547, 1522], yet query formulation, output parsing, and error attribution at the interface become the new precision bottleneck. **Interface calibration** compounds this: tool outputs arrive with heterogeneous units, confidence scores, coordinate frames, and failure modes, and an orchestrator that cannot judge whether a returned depth map, solver result, retrieved memory, or imagined rollout is trustworthy can perform worse than a direct model response.

Learned orchestration is one direction: policies are moving from hand-written call rules toward imitation- or reinforcement-learned controllers that decide which tool is worth its latency and uncertainty budget [1547]. A parallel shift is the *System-1 / System-2 architectural split*, in which the neural backbone handles fast, intuitive

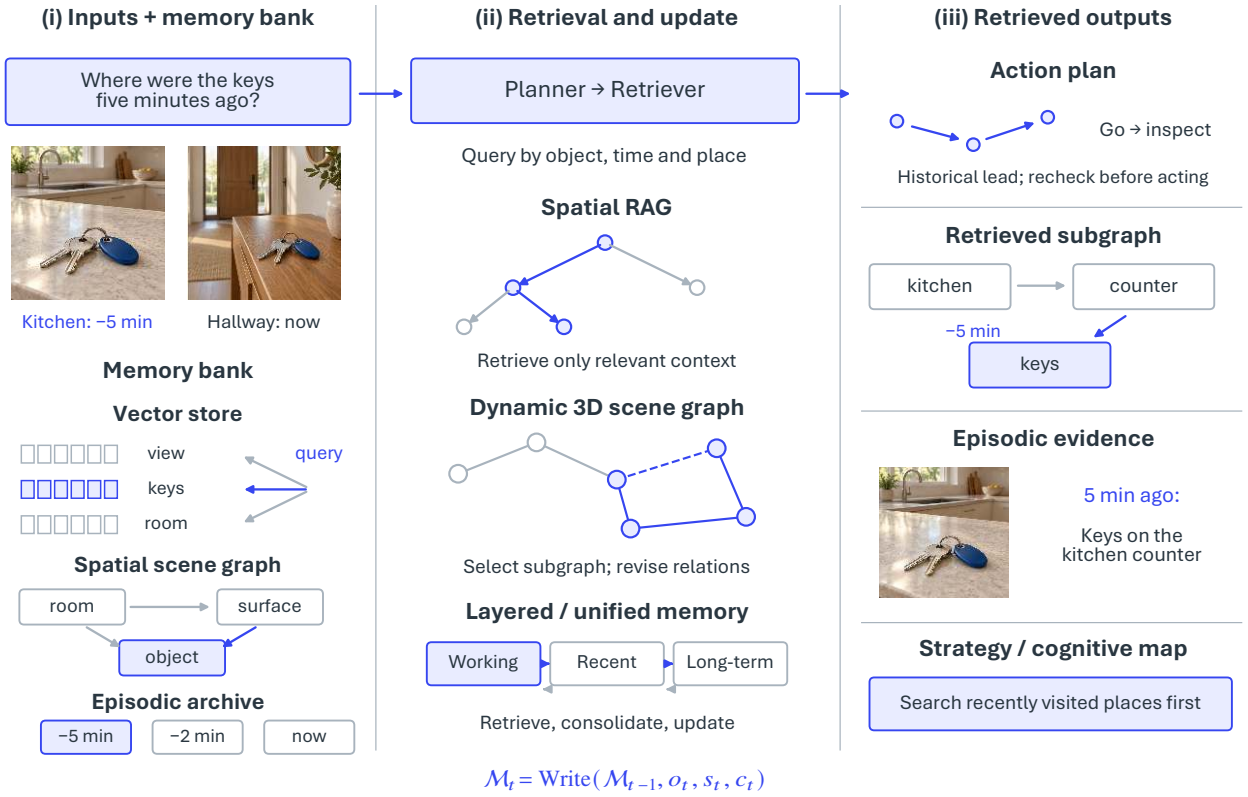


Figure 7.26: Retrieval- and memory-augmented spatial reasoning: (i) a query and a memory bank of vector records, spatial scene graphs, and episodic archives, (ii) retrieval and update via spatial RAG, dynamic 3D scene graphs, or layered memory, and (iii) alternative output forms rather than joint outputs of one query. In the example, the keys were on the kitchen counter five minutes ago and are in the hallway now. Scenes and retrieval diagrams are illustrative, not measured similarity scores.

perception (System 1) while the toolset handles slow, deliberative, verifiable reasoning (System 2) [1548], achieving both flexibility and rigor.

7.3.6 Retrieval- & Memory-augmented Spatial Reasoning

Retrieval- and memory-augmented spatial reasoning is best understood through a *Planner-Memory-Retriever* decomposition. The Planner $\mathcal{P}$ decides what spatial information is needed for the current instruction; the Memory Bank $\mathcal{M}$ stores an external, updateable state of the world; and the Retriever $\mathcal{R}$ selects the subset of that memory that should condition the next reasoning or action step. Unlike one-shot retrieval-augmented generation, spatial memory must also support *write* and *update*: an agent may discover that a door is blocked, that an object has moved, or that a previous topological belief was wrong, and the memory state must change accordingly.

The memory bank is therefore not merely a text corpus. In spatial domains, $\mathcal{M}$ may contain vector embeddings of views and instructions, metric maps and voxel belief states, dynamic 3D scene graphs, frontier snapshots, past trajectories, user preferences, and brain-inspired episodic archives. The central problem is to make these heterogeneous memories queryable: semantic retrieval must locate relevant objects and events, while geometric retrieval must preserve relations such as containment, support, reachability, and path connectivity.

7.3.6.1 Task Definition

Where §7.3.5 treats memory as one black-box tool among others, here the *external memory itself* is the central learning target. Formally, a retrieval- & memory-augmented system is the triple $(\mathcal{P}, \mathcal{M}, \mathcal{R})$, where $\mathcal{P}$ is a reasoning core (an LLM, VLM, or embodied planner), $\mathcal{M}$ is an external memory store, and $\mathcal{R} : \mathcal{Q} \times \mathcal{M} \to \mathcal{C}$ maps a query q_t and the current memory state to a relevant context fragment c_t. Given an instruction x, an observation stream o_t, and a reasoning history h_t (initially $h_0 = \varnothing$), the system proceeds by the iterative update

$$\begin{aligned} q_t &= \mathrm{Query}_{\mathcal{P}}(x, o_t, h_{t-1}), \qquad & c_t &= \mathcal{R}(q_t, \mathcal{M}_{t-1}), \qquad & s_t &= \mathcal{P}(x, o_t, c_t, h_{t-1}), \\ \mathcal{M}_t &= \mathrm{Write}(\mathcal{M}_{t-1}, o_t, s_t, c_t), & h_t &= h_{t-1} \cup \{(o_t, q_t, c_t, s_t)\}, \end{aligned} \tag{7.42}$$

where $\mathrm{Query}_{\mathcal{P}}$ extracts the planner's information need, $\mathcal{R}$ returns the retrieved context, and Write inserts, revises, or invalidates memory entries using the newest observation and decision state. The decomposition brings three concrete advantages: *scalability*, by letting agents operate over large environments while fetching only task-relevant subgraphs or vectors; *personalization*, through targeted retrieval of user profiles, historical preferences, and episodic interactions; and *explainability*, since decisions can be traced to the specific memories, rules, or map fragments that were retrieved.

7.3.6.2 Methodology

Three families of memory architecture instantiate the $(\mathcal{P}, \mathcal{M}, \mathcal{R})$ triple at different levels of abstraction and scale.

Retrieval-Augmented Generation for Navigation and Instruction. Vision-and-Language Navigation (VLN) demands a tight coupling of visual perception, language understanding, and action generation, and retrieval-augmented generation (RAG) bridges high-level planning and local observation by externalizing spatial context; the design question is what the memory bank should store and how the retriever should index it.

Textual abstraction vs. visual fidelity. NavRAG [1549] argues for hierarchical abstraction: a "Hierarchical Scene Description Tree" converts the 3D environment into structured text (Region → Object → Attribute), so the retriever can access information at the granularity the query requires, at the cost of visual nuance. VisRAG [1550] takes the opposite stance with a multimodal pipeline that bypasses text parsing entirely: a VLM retrieves images directly from the query, preserving texture, lighting, and style that text-only retrieval discards.

Privileged maps vs. egocentric mental simulation. A second line replaces privileged global maps with human-like episodic memory. GoViG [1551] generates instructions solely from egocentric observations of start and goal states, alternating between predicting the next visual view and generating the corresponding instruction, so that the navigational narrative is built on a hallucinated path; retrieval shifts from static facts to potential futures.

Hybridizing spatial and semantic logic. Retrieval in physical spaces requires logic that pure semantic search cannot provide (*e.g.*, "near", "inside", "along"). Spatial-RAG [1552] combines sparse spatial filtering (SQL-based constraints) with dense semantic matching, formulating retrieval as multi-objective optimization, which is what grounds CoT reasoning [1553] so that multi-step plans respect geometric feasibility.

Dynamic 3D Scene Graphs and Subgraph Retrieval. As environments grow, flat memory representations such as point clouds become intractable, and hierarchical abstraction through 3D scene graphs (3DSGs) has taken over: Hydra [1554] builds metric-semantic scene graphs in real time, and open-vocabulary successors embed vision–language features directly into graph nodes [1555], turning scene graphs from rigid structural maps into queryable vector stores. Because global graphs overwhelm planner context windows, processing has moved from holistic graph analysis to sparse subgraph retrieval, through semantic pruning [1556] or RAG-based mechanisms [1557] that isolate task-relevant topological neighborhoods. The convergence of

graph and vector retrieval matters for spatial memory in particular: graph edges preserve physical relations, while vector neighborhoods preserve open-vocabulary semantic similarity.

Unified and Brain-Inspired Memory Architectures. RAG provides access to information; the organization of that information is equally critical, and recent architectures borrow the brain's division of labor. RoboMemory [1558] integrates working, short-term, and long-term memory in one cognitive framework so that the agent iteratively refines its model of the environment, and 3D-Mem [1559] takes a visual-centric route, using "frontier snapshots" to manage exploration and hallucinate potential future states. 3DLLM-Mem [1560] grounds the integration of retrieved memory into the inference stream with a memory fusion layer that cross-attends from the current instruction into the stored 3D memory bank:

$$f^{Q}_{\text{fuse}} = \text{Softmax}\left(\frac{f^{Q}_{t}(f^{K})^{\top}}{\sqrt{C}}\right) f^{V}, \tag{7.43}$$

where f^{Q}_{t} are the query features of the current instruction and f^{K}, f^{V} the keys and values derived from the memory bank. Addressing specific failure modes, VLM^2 [1561] introduces a dual-memory module for the semantic–geometric misalignment of video-based spatial reasoning, and Vision-to-Geometry [1562] incorporates geometric cues such as field-of-view coverage to prevent redundant re-exploration.

7.3.6.3 Benchmarks

Progress in memory-augmented spatial reasoning requires separating *memory-specific* tests from broad spatial probes.

Memory-specific evaluation. 3DMem-Bench [1560] serves as a standard for assessing long-term persistence: comprising over 26,000 trajectories, it challenges agents with Spatial–Temporal QA (*e.g.*, "What color was the bed in the room you visited five minutes ago?"), requiring the retrieval of past observations that are no longer in the active context. Moving beyond static environments, EmbodiedBench [208] tests long-horizon planning where objects are moved or manipulated during the task, benchmarking the *write/update* capability of memory systems, their ability to overwrite obsolete data without catastrophic forgetting. OpenEQA [1563] offers a complementary episodic-memory setting by asking questions about an environment the agent has previously explored, making it useful for measuring whether retrieved experience rather than immediate perception supports the answer.

Broad spatial probes as diagnostic complements. Benchmarks such as SpaCE-10 [1564], SITE [1565], and the static probes collected in Table 7.5 (OmniSpatial [1474], Spatial-DISE [1566], Open3DVQA [1567]) evaluate spatial orientation, perspective taking, metric estimation, open-space 3D relations, and intrinsic/extrinsic static or dynamic reasoning. They are valuable for diagnosing what information a memory system should retain (viewpoint-dependent relations, object-centric attributes, multi-step multi-view dependencies), but they are not memory benchmarks unless the protocol hides necessary information in a prior episode and requires retrieval from $\mathcal{M}$: a model can fail a broad spatial probe because it lacks geometric perception, because it retrieved the wrong memory, or because the memory was never written in the first place.

7.3.6.4 Challenges and Trends

The recurring limitations originate in the Planner-Memory-Retriever interface rather than in any individual component. **Query formation under unbounded context** is the central bottleneck, not memory size alone: the act of deciding what to retrieve requires compressing the current task, viewpoint, uncertainty, and history into a finite representation, and memory paging and scheduling alleviate but do not remove the need for a learned retrieval policy. **Write/update consistency** is where append-only memories become dangerous: agents need mechanisms for reconciling conflicting observations, invalidating stale object locations, and representing uncertainty when the world may have changed since the last visit; EmbodiedBench-style protocols make the

issue visible, but robust update semantics remain open. **Integration mechanics** are unresolved between in-context learning, which concatenates retrieved text to the prompt (simple but token-inefficient), and cross-attention fusion, which injects retrieved 3D features into attention layers (3DLLM-Mem [1560]; expressive but architecture-coupled), with no clear winner on the fidelity–modularity trade-off. **Modality mismatch** has no general theory: RAG-over-text (NavRAG [1549]) discards visual nuance, RAG-over-images (VisRAG [1550]) loses structural priors, and when to retrieve text, images, or 3D representations remains task-dependent.

Memory is shifting *from static maps to dynamic semantic belief states*: systems store probabilities, semantic labels, and interaction histories rather than coordinates, enabling agents to reason under uncertainty about where an object *likely* resides given commonsense and past observation, as RoboMemory [1558] typifies. Graph- and vector-based retrieval are converging into hybrid architectures that use vector search to identify graph entry nodes and topological traversal for neighborhood reasoning. Most abstractly, *meta-cognitive memory* such as Meta-Memory [1568] generates task-specific "cognitive maps" and retrieves *strategies* rather than raw data, pointing toward learnable retrieval policies and bridging into the agentic orchestration of §7.3.7.

7.3.7 Agentic Spatial Reasoning

Agentic spatial reasoning asks how a model organizes *interaction* with space. The core object is not a single chain-of-thought prompt, tool call, memory store, VLA policy, or world model, but an orchestration policy that repeatedly decides what to perceive, what to infer, what to do, and how to revise its state after feedback. In this closed-loop view, a wall is not only a geometric plane but also a constraint on motion; a cup is not only an object category but also a potential affordance; and a failed action is not only an error but also evidence that should reshape the next plan.

"Agentic" is therefore a composition principle: the rationales, programs, tools, memory, VLA policies, and world models of the preceding subsections become resources inside a perceive–reason–act–reflect loop, and the research question is how the loop is scheduled and corrected: when should the agent gather more evidence, retrieve memory, call a geometric tool, simulate a future, ask another agent, or execute an action?

7.3.7.1 Task Definition

Whereas §7.3.5 treats tools as black-box operators and §7.3.6 treats external memory as the central learning target, agentic spatial reasoning composes both into a closed-loop cycle in which the orchestration policy itself becomes the learning target. Formally, an agentic spatial reasoning system is the tuple $(\mathcal{P}, \mathcal{S}, \mathcal{A}, \mathcal{F}, \pi)$, where $\mathcal{P}$ is a planning core (an LLM or VLM), $\mathcal{S}$ is the spatial state space (perception, internal map, history), $\mathcal{A}$ is the action space (sensorimotor commands or sub-goals), $\mathcal{F}$ is a reflection / critic function that scores proposed actions or rationales, and π is the orchestration policy. Given an instruction x, the system maintains a history h_t (initially $h_0 = \varnothing$) and proceeds by the iterative perceive–reason–act–reflect cycle

$$\begin{aligned} o_t &= \mathrm{Perceive}(\mathrm{env}), & s_t &= \mathcal{P}(o_t, h_{t-1}), \\ a_t &\sim \pi(\cdot \mid s_t), & r_t &= \mathcal{F}(s_t, a_t, o_{t+1}), \\ h_t &= h_{t-1} \cup \{(o_t, s_t, a_t, r_t)\}, \end{aligned} \tag{7.44}$$

where $\mathrm{Perceive}$ extracts the current spatial observation, π samples a proposed action, $\mathcal{F}$ supplies a reflection signal (verbal critique, planner-critic verdict, or value estimate), and the history accumulates all four. This formulation makes explicit that "agentic" is not a new capability but rather a structural promotion: the orchestration policy π becomes the central learning target rather than the planner $\mathcal{P}$ or any individual tool.

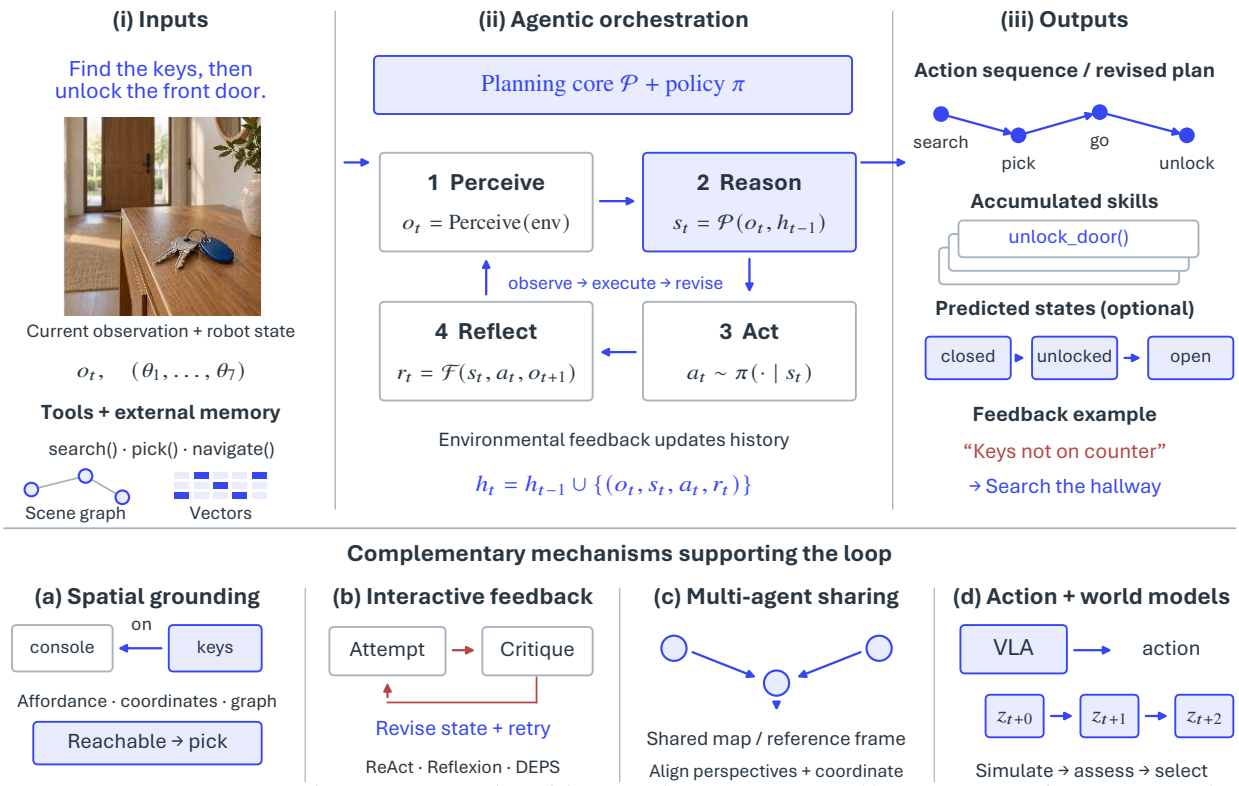


Figure 7.27: Agentic spatial reasoning as closed-loop orchestration. Top: (i) instruction, observation, robot state, tools, and memory, (ii) the perceive–reason–act–reflect loop under planning core $\mathcal{P}$ and policy π, and (iii) complementary, not simultaneous, output forms; the door-state sequence is a prediction and the failed counter search is feedback from a different step. Bottom: (a)–(d) mechanisms that support the loop. The scene is a generated illustration, not an experimental observation.

7.3.7.2 Methodology

Four orchestration problems organize the methods: grounding observations into executable state, closing the feedback loop, coordinating across agents, and using action and world-model modules without reducing agency to any one of them.

Grounding Observations into Executable State. The first responsibility of an agentic loop is to convert an instruction and observation into a state that can support action, preserving referential consistency between the agent's internal model and the physical environment. Probabilistic affordance alignment, as in SayCan [1311], PaLM-E [56], and RT-2 [97], weights the likelihood of an instruction ("pick up the apple") by the geometric feasibility of the action in the current scene; spatial reasoning stays implicit, and the orchestration layer must decide whether that is sufficient or whether to request an explicit representation. Explicit intermediate representations supply the precision implicit alignment lacks: SpatialCoT [1569] compels the VLM to output metric coordinates inside its chain of thought, a "think-coordinate-act" cycle that creates a verifiable spatial bottleneck; Embodied-R1 [1570] uses reinforcement learning to refine a pointing representation so that the model outputs 3D reference points (spatial anchors) that decouple high-level planning from motor control; and procedural querying treats the environment as a queryable database, through dynamically synthesized APIs (VADAR, §7.3.4) or a stateful code kernel that executes one cell at a time and revises the next from the observed outputs (SpatialClaw, §7.3.5). Beyond the immediate percept, agents ground reasoning in persistent structured abstractions: object-centric 3D scene graphs such as ConceptGraphs (§7.3.6) let the agent reason over topological relations ("behind", "supported by") rather than raw geometry, and MapGPT [1571] converts

topological maps into textual prompts that give the LLM a global view of connectivity for long-horizon planning.

Closing the Feedback Loop. Agentic spatial reasoning is a temporal *process* rather than a single inference, and the methodological shift is from producing one rationale to maintaining a loop that observes the consequences of its own choices. The chain-of-thought paradigm (§7.3.3) improved reasoning depth but remained open-loop; ReAct [21] closed the loop by interleaving reasoning traces with executable actions so that spatial outcomes are observed before the next step is planned. Observation alone is often insufficient for correction, so Reflexion [1572] introduced "verbal reinforcement learning", in which the agent writes explicit critiques of its failures into short-term context, and DEPS [1573] forces the agent to *explain* failures in geometric terms before choosing new sub-goals, turning retrying into debugging. For coherence across long horizons, agents need memory beyond the context window: Voyager (§7.3.4) stores successful spatial behaviors as executable code in a vector database and retrieves them when a similar context recurs, consolidating transient experience into procedural knowledge. Reflection can also precede execution. Reasoning via Planning (RAP) [22] and Tree of Thoughts [1502] explore branching spatial possibilities in a latent space; STMA [1539] ties such simulation to dynamic knowledge graphs with planner–critic loops that validate whether a proposed path is topologically valid; and vision-language world models such as VLWM [1574] roll out semantically described future states so that the orchestration policy can compare hypothetical futures before committing.

Coordinating Multiple Spatial Agents. In shared environments spatial reasoning becomes distributed, and movement has to be negotiated across perspectives and intents. Multi-agent reinforcement learning induces spatial cooperation without explicit communication, as in the SMAC formations learned by decentralized actor–critic and value-decomposition methods [1435] (§7.2.6), but there each agent treats the others as dynamic obstacles and has no symbolic model of its partners. A *spatial theory of mind* adds the ability to model another agent's geometric viewpoint, transforming egocentric perception into an allocentric frame ("my left" is "your right"): RoCo [1575] uses foundation models for embodied dialog in which agents disambiguate spatial references by simulating the partner's visual field, and ProAgent [1576] shows agents communicating to update a shared map and synchronize movement, converging on a common "language of space".

Orchestrating Action and World-Model Modules. Generalist VLA policies and world models are components of agentic systems, but they do not by themselves define agency; they become agentic when an orchestration policy uses them inside the loop, with a VLA module executing a sub-goal, a world model predicting the consequence of a candidate action, and a reflection module deciding whether to retry, replan, or gather more evidence. What makes an action representation useful to a loop is that it exposes intermediate commitments the loop can inspect, revise, or hand to a controller: beyond the unified action interfaces of RT-2 [97], Gato [149], and Octo [1577], world-language-action models [1578] jointly predict textual subtasks, subgoal images, and robot actions, and ACoT-VLA [1515] reasons through coarse action intents before fine-grained control. Cross-embodiment transfer (§7.3.2; RoboCat's self-improvement cycles adapt to new bodies [1579]) becomes an orchestration problem too, since the same high-level plan may require different perception checks, affordance tests, and recovery routines on different bodies. World models (§7.3.8), from DreamerV3 [152] and Genie [157] to WorldPilot-style prediction inside planning loops [1580], serve the loop as cognitive simulators: the agentic use is not video generation for its own sake but closed-loop decision support, in which the agent imagines, critiques, and then acts.

7.3.7.3 Benchmarks

Evaluating agentic spatial reasoning requires benchmarks that go beyond static visual question answering: a useful protocol tests whether the agent can gather missing observations, choose tools or actions, use memory across long horizons, and revise its plan after feedback, and static spatial probes should be read as component diagnostics rather than direct evidence of closed-loop agency. Two protocols exercise the

loop directly: iVISPAR [1581] evaluates VLMs as agents in an interactive sliding-tile setting with visual 2D, visual 3D, and text inputs, where the answer depends on multi-step state transitions rather than one-shot recognition, and TIR-Bench [1582] requires agents to use tools (zoom, crop, rotate) to manipulate images while solving spatial puzzles. Memory-centered settings (OpenEQA [1563], §7.3.6) and open-3D probes such as Open3DVQA [1567] test what the loop must retain, and the broad visual-spatial probes (SpatialEval [1450], VSI-Bench [227], OmniSpatial [1474], Spatial-DISE [1566]) identify weak spatial primitives along the axes of static vs. dynamic content, single vs. multi-view, ego vs. exo perspective, and seen vs. unseen queried information; Table 7.5 maps the coverage.

Table 7.5: Representative spatial-reasoning benchmarks discussed in §7.3.7. Perspective marks whether ego and exo viewpoints are explicitly represented; Interactive marks protocols that exercise action or tool selection rather than only answer generation; Query Target is whether the queried information is visible (Seen), hidden (Unseen), or an interactive state (State). *Undeclared* marks attributes the source paper does not state.

Benchmark	QA / Tasks	Modality	Scene	View Type		Perspective		Interactive	Query Target
				Single	Multi	Ego	Exo		
3DSRBench [1583]	2,772	Image	Static	✓	✗	✗	✓	✗	Seen
InternSpatial-Bench [1584]	6,008	Image	Static	✓	✗	✗	✓	✗	Seen
OmniSpatial [1474]	∼8,400	Image	Static	✓	✗	✓	✓	✗	Seen
Spatial-DISE [1566]	559	Image	Mixed	✓	✓	*Taxonomy*		✗	Seen
Open3DVQA [1567]	9,000	3D / Sim	Static	✓	✗	✓	✓	✗	Seen
iVISPAR [1581]	Puzzles	2D/3D/Text	Dynamic	✓	✗	*N/A*		✓	State
Spatial457 [1585]	23,752	Image	Static	✓	✗	*Undeclared*		✗	Seen
All-Angles Bench [1586]	∼2,100	Image	Static	✗	✓	*Undeclared*		✗	Seen
EmbSpatial-Bench [1587]	3,640	Image	Static	✓	✗	✓	✗	✗	Seen
MMSI-Bench [1588]	1,000	Image	Static	✗	✓	*Undeclared*		✗	Seen
MindCube [1472]	21,154	Image	Static	✗	✓	*Undeclared*		✗	Unseen
VSI-Bench [227]	5,130	Video	Static	✓	✗	✓	✗	✗	Seen
VLM4D [1589]	∼1,800	Video	Dynamic	✓	✗	✓	✓	✗	Seen
STI-Bench [1590]	2,064	Video	Dynamic	✓	✗	✓	✓	✗	Seen
DSI-Bench [1591]	∼1,700	Video	Dynamic	✓	✗	*Undeclared*		✗	Seen

Table 7.5 should be read as a coverage map, not as a leaderboard for agentic systems. Most entries evaluate static answer generation or localized visual-spatial primitives; only interactive protocols such as iVISPAR and tool-use settings directly exercise action selection and feedback. Claims about agentic spatial reasoning should therefore report loop-level metrics (observation gathering, action choice, recovery after failure, memory update, and final task success) rather than single-turn accuracy alone.

7.3.7.4 Challenges and Trends

Boundaries are collapsing between language and action (VLAs), between robot bodies (cross-embodiment), and between perception and imagination (world models), and two difficulties gate progress toward generalist spatial agents. The **sim-to-real fidelity gap**: high-fidelity simulators (predictive world models, generative video) accelerate training but introduce systematic biases that physical execution exposes, and the cost–precision frontier for transfer remains open. **Long-horizon spatiotemporal coherence**: across multi-hour episodes the agent's internal map drifts, and episodic memory and reflection reduce but do not eliminate the drift, because the bottleneck is the finite context window of the planning core.

Explicit orchestration over observation–state–action–reflection tuples is one direction: VLA and WLA systems are beginning to expose intermediate subgoals, action intents, and predicted states that a loop can inspect rather than hiding all control inside a single forward pass. A second is *mental simulation via generative world models*: DreamerV3-style, Genie-style, VLWM-style, and WorldPilot-style models let agents verify spatial logic in imagination before acting, shifting the paradigm from "trial-and-error" toward "trial-in-imagination" while keeping the agentic focus on when to simulate, when to act, and when to revise.

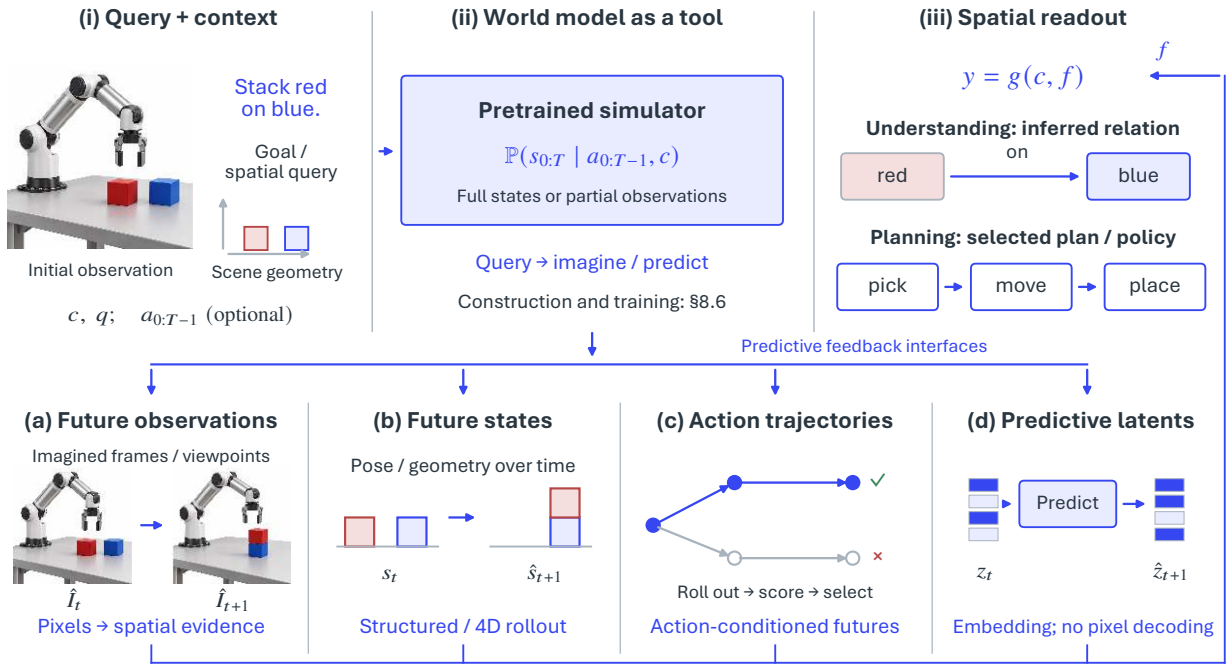


Figure 7.28: World-model-based spatial reasoning read as tool use: (i) a spatial query q with context c and optional actions $a_{0:T-1}$; (ii) a pretrained world model (constructed and trained as in §8.6) whose predictive feedback f takes four possibly overlapping forms, namely (a) future observations, (b) future states, (c) action-conditioned trajectories, and (d) predictive latents; and (iii) a readout $y = g(c, f)$ that gives an inferred relation (understanding) or a selected plan (planning). The robot scenes are illustrative, not experimental observations.

7.3.8 World-Model-based Spatial Reasoning

A world model is a neural simulator of how world states transition, conditioned on an agent's actions and/or external multi-modal context. Trained on large-scale episode trajectories, it acquires a generalizable, latent model of environment dynamics. This subsection treats that model as a *tool*: the object of study is the spatial-reasoning task, and the world model is the instrument used to solve it. Its defining capability is that it *simulates* the world and can therefore return a predictive feedback signal, a rollout of imagined futures or a dynamics-aware latent, that a reasoner consumes to answer a spatial query or to plan. We accordingly organize the discussion by *how the world model is queried and what feedback it returns*, rather than by how it is built. The complementary question of how world models are themselves constructed, trained, and scaled is the subject of §8.6 (World Modeling); here we take a pretrained world model as given. Figure 7.28 schematizes this tool-use formulation; its two usage modes, understanding-oriented and planning-oriented, cut across the feedback forms surveyed below.

7.3.8.1 Task Definition

The task addressed in this subsection is spatial reasoning itself: given a query q about a scene together with the available context c (a textual description, one or more observations, or a scene specification), a reasoner must produce a spatial answer y, either an inferred spatial property o (*e.g.*, a relation, a structure, or a physical attribute) or a plan / policy π that acts on the scene. What distinguishes the methods surveyed here is *how* they arrive at y: rather than mapping c to y in a single feed-forward pass, the reasoner consults a *world model* as an auxiliary instrument that can imagine how the scene would look or evolve, and reasons over that imagined evidence. Concretely, the world model is a pretrained simulator exposing a distribution $\mathbb{P}(s_0, s_1, \ldots, s_T \mid a_0, a_1, \ldots, a_{T-1}, c)$ over world states s_t, or over their partial observations $\mathrm{proj}(s_t)$ when

full states are unavailable, optionally conditioned on an action sequence $\{a_t\}_{t=0}^{T-1}$ (null when the model is action-free). A world-model-based reasoner treats $\mathbb{P}$ as a queryable tool: given the query, it (i) conditions $\mathbb{P}$ on the context c and, when planning, on a candidate action sequence; (ii) draws predictive feedback f from $\mathbb{P}$, a rollout of future observations, future states, or action-conditioned trajectories, or a dynamics-aware latent; and (iii) applies a readout g that maps the feedback back to the spatial answer, $y = g(c, f)$. The world model supplies the imagined evidence f, but producing the spatial answer y remains the objective, and the fidelity of f bounds how much the world model can help.

7.3.8.2 Methodology

We treat the world model as a pretrained instrument and organize methods by *the form of predictive feedback it returns* to the reasoner. Two families emerge: *generative feedback*, in which the model is rolled out to synthesize explicit futures that the reasoner inspects, and *latent-representation feedback*, in which the model's dynamics-aware embedding is consumed directly without decoding; the understanding-versus-planning distinction cuts across both.

Generative Feedback via Future Observations. The most direct use of a world model is to synthesize future or counterfactual *pixels* and read spatial evidence off them, an analysis-by-synthesis strategy in which the reasoning outcome o (a spatial relationship, structure, or physical property) is whatever best reproduces the observed or imagined imagery under $\mathbb{P}$. Predating deep generative models, Galileo [1204] instantiates this idea with a physics engine in place of a neural simulator: it tracks object velocities across raw videos and recovers 3D shape, mass, and friction as the parameters that best reproduce the observed motion, so that the projected state $\mathrm{proj}(s_t)$ is the object velocity v_t and the videos themselves define the transition distribution $\mathbb{P}(v_0, \ldots, v_T \mid c)$. Modern instances replace the engine with an Internet-scale image or video diffusion prior: DragAPart [1592], Puppet-Master [1593], ATOP [1594], and DreamArt [1595] fine-tune such priors into articulation simulators $\mathbb{P}(I_0, \ldots, I_T \mid c)$ whose frames $I_t = \mathrm{proj}(s_t)$ render an articulated object under a hypothesized configuration, and recover the 3D articulation (*e.g.*, its motion axes) by aligning the rendered hypothesis with the synthesized video. A second, inference-time use synthesizes *unobserved viewpoints* to resolve spatial questions that a single view underdetermines: MindJourney [202] lets a vision-language model "imagine" the scene from a new camera pose with a view-conditioned world model before answering, turning the world model into a test-time reasoning aid rather than a training-time simulator. Because such imagination is expensive and not always helpful, AVIC [1596] learns *when* and *how much* to imagine, gating world-model invocation on the sufficiency of the current visual evidence and matching or exceeding fixed-imagination strategies at a fraction of the world-model calls. Pushing imagination from a fixed sweep to an *active* policy, SpatialDreamer [1597] closes the loop: a reinforcement-learned agent chooses which viewpoints to imagine through the world model and then grounds its answer in the imagined evidence, under a geometric policy-optimization objective that rewards geometrically consistent rollouts.

Generative Feedback via Future States. Rather than decode pixels, a second line rolls the world model out in a *structured or higher-dimensional state space*, trading photorealism for geometric consistency. TesserAct [192] learns a 4D (3D-plus-time) world model whose rollouts stay spatially coherent enough to support robotic manipulation, and persistent-3D world models [1598] maintain an explicit 3D memory so that revisited regions remain consistent across long horizons. State-space feedback is attractive when the downstream query is itself geometric, since it sidesteps the appearance-modeling burden that pixel rollouts carry.

Generative Feedback via Action-Conditioned Trajectories. When the goal is to *act*, the world model is queried as a simulator: candidate action sequences are rolled out and scored by the favorability of the resulting states, yielding a policy $\pi : (s_{\le t}, a_{<t}) \mapsto a_t$ that steers the episode toward a preferred terminal state $s_T \in S_{\text{good}}$ (the model-based-RL view; the broader treatment of RL with spatial inductive biases is §7.2.5). The paradigm originates with Ha and Schmidhuber [4], who introduced the term "world model" in its current sense, training a generative recurrent model of future observations and using it as an internal simulator for

policy search, and with Dreamer [147], which learned the simulator in a compact latent space and optimized the policy purely on imagined rollouts. Scaling this recipe, UniSim [1599] builds a video-diffusion simulator from which real-world manipulation policies can be extracted, and DreamGen [1600] uses video world models as trajectory generators, pseudo-labeling the synthesized rollouts with an inverse-dynamics model to train policies that generalize to new behaviors and environments. Most recently, VAGEN [1601] turns this feedback back onto the reasoner itself, using reinforcement learning to *supervise* a vision-language agent's internal world-model reasoning: it decomposes that reasoning into state estimation ("what is the current state?") and transition modeling ("what comes next?") and rewards accurate latent prediction.

Latent-Representation Feedback. A final family dispenses with generation entirely and consumes the world model's *predictive latent* as the spatial signal. Joint-embedding predictive architectures learn to predict the representation of a future or masked observation rather than its pixels: I-JEPA [151] establishes the recipe for images, and V-JEPA 2 [155] extends it to video, yielding dynamics-aware embeddings that support understanding, prediction, and planning without ever rendering a frame, while VL-JEPA [200] ports the idea to vision-language inputs. A related line keeps the latent explicitly *geometric*: 3DThinker [1602] performs 3D mental imagery during reasoning by generating a 3D latent and aligning it with that of a 3D foundation model, letting a vision-language model reason about spatial relations from only a few views without any 3D input or explicit rendering. Because the feedback is an embedding rather than an image, latent world models avoid the appearance-synthesis cost and the pixel-space memory drift that limit generative rollouts, at the price of a signal that is harder to inspect directly; they are an increasingly prominent, query-efficient alternative for spatial reasoning.

7.3.8.3 Benchmarks

Evaluation follows the usage mode. Planning-oriented use is scored on the embodied suites of §7.3.2 and on long-horizon household environments such as ALFWorld [1520] and VirtualHome [1521], where the question is whether policies extracted from imagined rollouts transfer to the real environment. Understanding-oriented use is scored on the static spatial probes that single views underdetermine, VSI-Bench [227] and MindCube [1472] among them, under test-time-scaling protocols that vary how many imagined views the reasoner is allowed and whether a verifier selects among them [1603, 1596]. The world model itself is assessed on video-generation quality (FVD, FID) together with task-specific physics-consistency probes, and surveys of general world models track the field's shift from 2D toward 3D-consistent simulation [1604], while recent analyses argue that advancing MLLM spatial reasoning demands purpose-built recipes and data rather than scale alone [1522]. No single benchmark currently exercises a world model along perception fidelity, action conditioning, and long-horizon spatial memory at once, a gap the challenges below make explicit.

7.3.8.4 Challenges and Trends

Reading the world model as a tool clarifies where the difficulties lie: some are inherited from the instrument itself, and are properly the concern of §8.6, while others are specific to how a reasoner queries it. The most active questions of 2025–2026 are no longer whether a world model *can* imagine, but whether its imagined evidence can be *trusted* and whether consulting it is worth the cost.

Imagination is not a free lunch. Systematic probes report that world-model imagination improves spatial reasoning in only a minority of cases, and that adding imagined views yields non-monotonic, sometimes negative accuracy gains at steep token and latency cost [1596, 1603]; the sheer amount of imagination is an unreliable proxy for accuracy [1522], which recasts "when and how much to imagine" as a first-class test-time-scaling problem, with adaptive gating [1596] an early answer and a predictive account of *which* queries need the world model still missing. Faithfulness is the companion problem: a world model can render fluent futures that are physically or relationally wrong, so likelihood under $\mathbb{P}$ is necessary but not sufficient, and when the imagined evidence lacks fidelity verification buys little, since heuristic verifiers give

limited calibration gains and an information bottleneck sets in, as a reasoner cannot recover a signal the imagination never contained [1603]. Grounding imagined evidence to the actual scene and calibrating trust in it remain open; physics-consistency objectives and geometry-grounded latents [1602] are partial responses.

The reasoner can be no more reliable than the simulator it queries. Even state-of-the-art models such as Genie 3 [157], despite markedly longer-horizon consistency, still drift in spatial memory, and consistency mechanisms [1598, 1605] narrow but do not close this gap, with hallucination and long-horizon error compounding over rollout length; raising this ceiling is the province of world-model construction (§8.6), and the reasoning methods surveyed here can only mitigate its symptoms.

The feedback forms also trade off along different axes, with no principled account of which a given query should use: pixel rollouts are inspectable but expensive and prone to appearance drift; state or 4D rollouts are geometrically consistent but require structured simulators; latent feedback is cheap but opaque; action rollouts commit early to a policy; and most systems commit to one form a priori. Extending world-model-based reasoning beyond pixels (*e.g.*, to forces [1606], tactile, or audio modalities) is gated by the absence of internet-scale pre-training data in those modalities, a constraint that also surfaces in the multi-modal aspects of §7.2.6.

Two shifts define the current frontier. *From external tool to trained-in reasoning*: rather than invoking a frozen simulator at inference time, recent work *trains* the reasoner's use of the world model, whether by reinforcement learning that supervises an internal state-estimation and transition-modeling loop [1601] or that rewards geometrically consistent active imagination [1597], or by distilling the world model's imagined dynamics directly into a vision-language model so that no simulator need be queried at test time [1607]. *From pixels to predictive latents*: the imagined branch is increasingly consumed as a latent process whose hidden states guide reasoning, rather than as a faithfully rendered video [151, 155, 1602]. Both shifts move the field away from ever-larger, faithfully-rendered pixel simulators and toward cheaper, more trustworthy, query-efficient feedback that a reasoner can afford to consult, and can be trained to consult well.

7.4 Hybrid Spatial Reasoning



Conventional approaches to spatial reasoning exhibit fundamental limitations. Systems that rely solely on linguistic or symbolic reasoning typically lack perceptual grounding in real-world visual and physical structures (*e.g.*, geometry, physics, spatial continuity), making them inadequate for handling cross-view transformations, temporal evolution, or physically coherent scenarios. Conversely, purely perceptual or vision-based systems, such as those combining visual recognition with rule-based geometric reasoning, often fall short in performing high-level abstract reasoning, multi-step planning, long-term memory integration, or interactive decision-making. These limitations have motivated a growing body of work advocating for a hybrid paradigm of spatial reasoning: integrating heterogeneous modules with distinct capabilities (perception, world modeling, memory, and action) so as to jointly leverage low-level spatial and physical cues together with high-level cognitive reasoning capabilities. A perceptual or physics-informed module supplies geometry, relations, affordances, and dynamics; a symbolic or neural module plans, infers, and abstracts over them; memory and history extend the pipeline to multi-step, multi-view, and temporally extended tasks; and because no single modality, task, or dataset carries the whole system, the composition transfers across vision-only, language-only, embodied, and multimodal settings.

7.4.1 Task Definition

A hybrid spatial reasoning system is a pair $(\{M_i\}_{i=1}^{k}, \pi)$: a set of k heterogeneous modules $M_i \in \{M_{\text{percept}}, M_{\text{world}}, M_{\text{symb}}, M_{\text{mem}}, M_{\text{plan}}, M_{\text{act}}\}$, each contributing a distinct inductive bias (perception, world model, symbolic constraints, memory, planning, or action), and an orchestration function $\pi : \mathcal{S} \to \{1, \ldots, k\}$ that selects which module acts on the current system state $s_t \in \mathcal{S}$ (perception, internal map, history). Starting

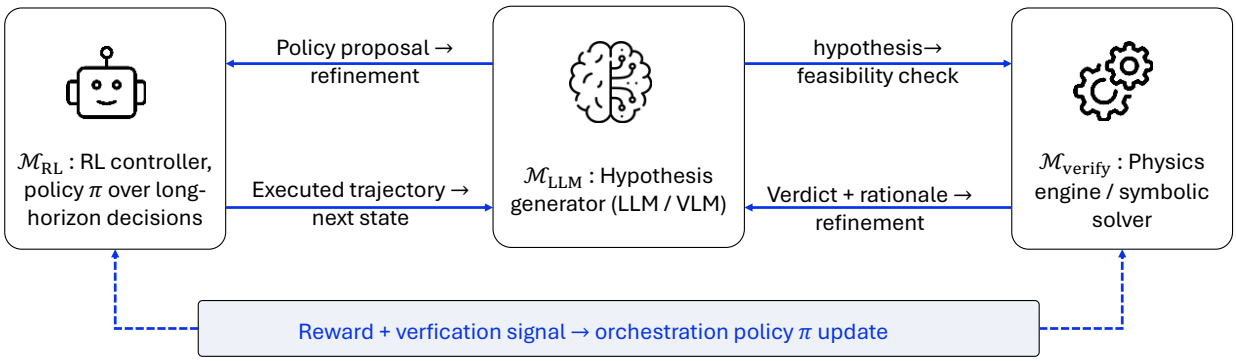


Figure 7.29: The foundation-model + reinforcement-learning hybrid pattern: an LLM/VLM hypothesis generator coupled with an RL controller and with a physics engine or symbolic solver for verification. The RL controller's policy (written π in the figure) is π_{RL} in the text; the bottom bar is the orchestration function π of Eq. (7.45).

from the input $s_0 = x$, the system evolves as

$$s_{t+1} = M_{\pi(s_t)}(s_t), \qquad y = \text{Read}(s_T), \tag{7.45}$$

and the three composition styles found in practice are special cases of Eq. (7.45): a *pipeline* fixes π to a constant module order, an *iterative* system lets π revisit modules until a verifier accepts, and an *end-to-end* system makes every M_i differentiable so that the whole trajectory is trained jointly. Hybrid spatial reasoning is the paradigm in which π and the module set $\{M_i\}$ are designed together to minimize the precision-grounding asymmetry: the foundation models of §7.3 supply semantic competence, and the symbolic and learning-based components of §7.1–§7.2 supply metric precision and verifiable structure. Existing systems therefore differ along three axes: *which* modules they compose, *how* they compose them (pipeline, iterative, or end-to-end), and how the orchestration function is realized (hand-coded rules or a learned policy).

7.4.2 Methodology

Five hybridization patterns recur in the literature, distinguished by which module the foundation model is paired with and by what supplies the truth signal.

Foundation Models with Reinforcement Learning. A prominent hybrid couples a foundation model's semantic competence with reinforcement learning's long-horizon decision-making, often closed by a verifier that supplies executable structure (Figure 7.29). The foundation model proposes hypotheses, sub-goals, or candidate action sequences; an RL policy π_{RL} commits to and executes them over extended horizons; and the resulting reward, optionally augmented by a physics engine or symbolic checker, updates both the policy and the orchestration. Embodied-R [1608] instantiates this division of labor: a large vision–language model handles perception, while a small language model trained by RL, with a reward that scores the logical consistency between its reasoning trace and its answer, performs the slow, deliberative spatial reasoning, evaluated on UrbanVideo-Bench and VSI-Bench. The group-relative fine-tuning recipes of §7.2.5 (Reason-RFT, OctoNav, SpaceR [1424, 1425, 1427]) are the same hybrid with the verifier folded into a verifiable reward and are not repeated here. The foundation model contributes broad priors and RL supplies credit assignment and exploration; the cost is the sample inefficiency and reward-specification difficulty inherited from RL (§7.2.5).

Foundation Models with Explicit Physical Simulation. A second pattern pairs the foundation model with a physics engine, so that the truth signal is a simulated trajectory rather than a learned score; the engine side is the physical reasoning of §7.1.2. The LLM proposes hypotheses, structural descriptions, or programs, the simulator executes them under conservation laws, kinematic constraints, contact dynamics, and reachability limits, and the discrepancy between simulation and observation is fed back for another

round. LLMPhy [1609] is the clearest instance: the LLM writes Python programs that embed estimates of latent physical parameters (friction, stiffness, mass) and of the scene layout, a physics engine executes them, and the reconstruction error against the observed video is returned as feedback, so that building the digital twin becomes a black-box optimization driven by the LLM. DiffGen [1610] makes the loop differentiable, generating robot demonstrations by descending gradients through a differentiable simulator, a differentiable renderer, and a vision–language model until the rendered outcome matches the language instruction in embedding space. PhysVLM [1611] moves the physics into the input instead: a robot's reachable workspace is abstracted into a robot-agnostic space-physical reachability map that the VLM consumes alongside the image, trained on Phys100K and evaluated on the EQA-phys benchmark. At the far end, ChronoLLM [1612] fine-tunes LLMs to write PyChrono multibody-simulation scripts, from a single pendulum to a full vehicle on deformable terrain; here the model authors the simulation rather than reasoning inside it, which places it at the boundary with the code-generation paradigm of §7.3.4. What the simulator contributes is an inductive bias that stays valid for unseen objects and configurations, so the hybrid extrapolates where a purely statistical model would not.

Foundation Models with Symbolic Reasoning. A third pattern inserts an explicit symbolic layer (logical constraints, relational predicates, or formal planning specifications) between natural-language reasoning and executable spatial inference, replacing the hand-crafted parsers of §7.1 with foundation-model-generated symbolic structure. Two sub-patterns differ in when the symbolic machinery is used. In the first it is called at inference time: NS3D [377] has an LLM translate a 3D referring expression into a first-order logic program that a neural executor grounds against the scene's objects and relations on ReferIt3D, and LEFT [348] (§7.3.4) makes the same programs differentiable so that concept grounding is learned rather than hand-specified. In the second, symbolic knowledge enters only at training time as a constraint on a neural model: Premsri and Kordjamshidi [1538] train language models for reasoning over spatial language with spatial logic rules as auxiliary losses, and RelateSeg [1613] extracts relations such as $\langle \text{cat}, \text{to-right-of}, \text{person} \rangle$ from open-vocabulary segmentation outputs, writes them as first-order logic, and relaxes them with fuzzy logic into a single auxiliary loss that improves mIoU on four benchmarks without adding parameters. The Scene Language [1614] (§8.2.5) is the representation such hybrids increasingly target: programs for structure, words for semantics, embeddings for appearance. The symbolic layer buys consistency and compositional guarantees; its price is a closed predicate vocabulary that the neural side constantly exceeds, a point taken up under §Challenges and Trends.

Foundation Models with Spatial Memory and World Models. A fourth pattern gives the foundation model a persistent spatial state that it does not have to carry in its context window: an explicit memory (episodic trajectories, topological maps, 3D layouts) or a learned world model that predicts how the scene changes under action. The hybrid question is the interface between the two. MSNav [1615] splits zero-shot vision-and-language navigation into a memory module, a spatial module, and a decision module, with a small fine-tuned LLM extracting the landmarks and relations the spatial module needs, and reaches state-of-the-art zero-shot results on R2R and REVERIE. MindExplore [1616] stacks a reasoning layer (task-specific chains of thought that emit meta-actions) over an acting layer (a mixture of policy experts with a multimodal diffusion policy) and uses a memory mechanism as the feedback channel between the two, running at 30 FPS in dynamic sand environments. The 3D spatial memories of §7.3.6 (3DSPMR [1562], Embodied-RAG [1557], Meta-Memory [1568]) and the world-model coupling of §7.3.8 are the memory- and simulator-side components these hybrids draw on; MindJourney's test-time gain on SAT [202] (about eight points of top-1 accuracy from imagined viewpoints alone, with the VLM untouched) is the cleanest quantitative evidence that a world model used as a tool closes part of the gap. In each case the foundation model keeps high-level inference, instruction following, and semantic grounding, and offloads persistence, geometry, and dynamics to the dedicated module.

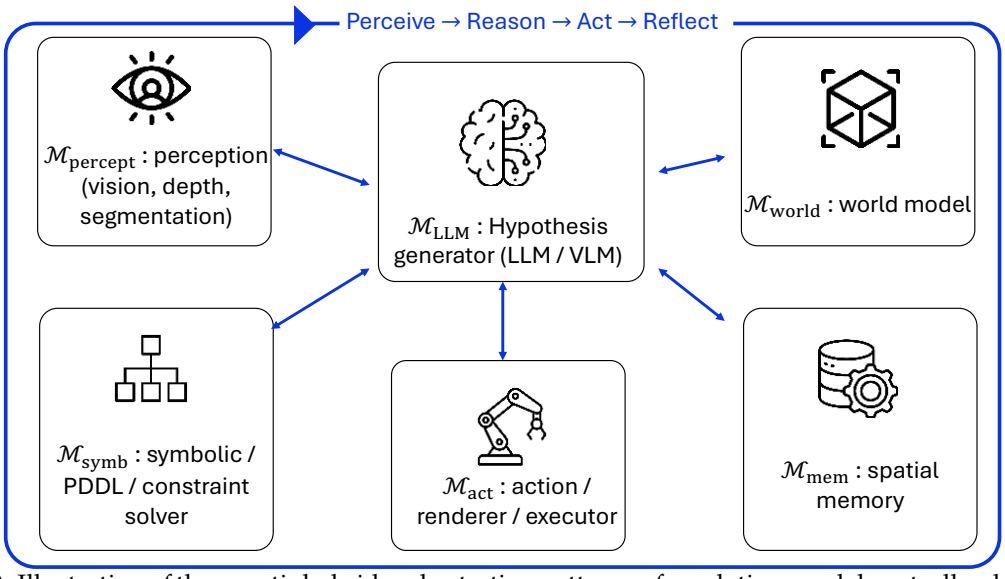


Figure 7.30: Illustration of the agentic hybrid orchestration pattern: a foundation-model controller dynamically selects perception, world-modeling, symbolic-reasoning, memory, and action tools within a closed-loop perceive–reason–act–reflect cycle.

Agentic Orchestration of Hybrid Components. The fifth pattern makes the orchestration function itself the object of design: a foundation model acts as a deliberative controller that decides, at each step, which of the perception, world-model, symbolic, memory, and action modules to invoke, within a closed perceive–reason–act–reflect loop (Figure 7.30). It differs from the agentic reasoning of §7.3.7 in what is being scheduled: there, tools and memory; here, additionally symbolic engines and physics simulators, with the scheduler π of Eq. (7.45) itself a candidate for learning. VADAR [1518] (§7.3.4) is the program-synthesis version of such a controller, generating the API it then calls. "See it. Say it. Sorted" [1617] is a training-free version for sketch-to-diagram generation: a critic VLM proposes a few qualitative relational edits, several LLMs synthesize candidate SVG programs, and a judge VLM selects among them, which keeps global constraints such as alignment and connectivity intact where numerical estimation would drift. Manh et al. [1618] arrive at the same module inventory from neuroscience, arguing that such a controller should sit over multimodal sensing, multisensory integration, egocentric–allocentric conversion, a cognitive map, spatial memory, and spatial reasoning. The gain is compositionality and long-horizon consistency that no static architecture provides; the cost is the latency and attribution problems discussed below.

7.4.3 Benchmarks

Hybrid systems are evaluated end to end on the benchmark of their target task rather than module by module, and the benchmark is often one the paper introduces. Among the systems above, Embodied-R reports on UrbanVideo-Bench and VSI-Bench, MindJourney on SAT, NS3D on ReferIt3D, MSNav on R2R and REVERIE, and PhysVLM on RoboVQA and OpenEQA alongside its own EQA-phys; LLMPhy ships TraySim, CLEVRER-LLMPhy, and Real-TraySim, MindExplore ships SandGo-1k and SandThink-21k, and DrawingBench [1619] is built specifically for agentic hybrids, scoring an LLM's sequence of low-level mouse actions against eight rule-based criteria for the spatial configuration it was asked to draw. Physics-side components are still checked against PHYRE and ShapeStacks (§7.1.2) and symbolic components against the constraint- and rule-based suites of §7.1.4–§7.1.5, but no current benchmark attributes an end-to-end score to the modules that produced it; that gap is taken up under §Challenges and Trends and in the cross-paradigm survey of §7.5 (Table 7.6, whose 15-row agentic subset is Table 7.5 of §7.3.7).

7.4.4 Challenges and Trends

Hybrid spatial reasoning has produced architectures that combine complementary inductive biases in principled ways, but several recurring limitations originate in the module-composition framework itself rather than in any individual module.

Frame, Scale, and Geometric Consistency. Unlike generic modular systems, spatial hybrids must reconcile heterogeneous *spatial* representations across modules: perception emits camera or egocentric frames, map and memory modules operate in world or allocentric coordinates, symbolic reasoners abstract to topological relations, and physics engines expect metric units. Misaligned reference frames, scales, or conventions silently corrupt downstream reasoning, and because each module reports confidence in its own terms, small upstream pose, depth, or scale errors propagate and compound across the pipeline without a coherent notion of uncertainty. Maintaining frame, scale, and geometric consistency is thus a first-order obstacle with no analogue in non-spatial hybrid systems.

Interfaces, Latency, and Attribution. The strength of a hybrid system is bounded by its worst module interface. Symbolic-to-neural translation is particularly brittle when the symbolic side uses closed-vocabulary predicates while the neural side proposes open-vocabulary ones, and the interface design becomes a hidden first-class hyperparameter that current literature seldom isolates. Each module invocation also adds wall-clock latency that compounds across a perceive–reason–act–reflect loop, so high-frequency manipulation and navigation may not be reachable through orchestration alone. And when a hybrid fails, attributing the failure to a specific module is non-trivial; this complicates iteration and is often cited as a practical reason why end-to-end models can outperform hybrid stacks on aggregate benchmarks despite offering weaker per-module guarantees.

The orchestration function π is migrating from hand-coded rules to *learnable orchestration policies* trained with reinforcement or imitation learning, which blurs the boundary with §7.3.7 in exactly the way the §7.3 Summary anticipated. *Differentiable hybrid stacks*, built from differentiable physics simulators, differentiable optimization layers (§7.2.4), and differentiable retrieval, make end-to-end training of a hybrid viable and remove the historical interface penalty, offering a path through the precision–grounding asymmetry that motivates this section. Evaluation is changing too, toward the *holistic and diagnostic*: in response to the attribution problem, emerging practice pairs aggregate task metrics with per-module diagnostic probes and counterfactual module-swap ablations that isolate where a hybrid system succeeds or fails, a direction consolidated in the cross-paradigm benchmark survey of §7.5.

7.5 Spatial Reasoning Benchmarks

Early spatial-reasoning benchmarks positioned the model as a detached observer. CLEVR [1358] and VSR [1479] ask for relative geometric relations from a static third-person viewpoint, and their 2D inputs leave depth ambiguous, a weakness that 3DSRBench [1583], MMSI-Bench [1588], DSI-Bench [1591], and STI-Bench [1590] now probe directly. SQA3D [1620], OpenEQA [1563], and EmbSpatial-Bench [1587] instead place the model inside the scene, so that it must self-localize before it can answer, and ScanQA [41], ScanRefer [373], and Multi3DRefer [1107] ground queries in point clouds and RGB-D data, where the answer is a coordinate rather than a relation. The reasoning demanded has deepened in parallel: IntPhys [1198] and Physion [680] test whether a model expects gravity, collision, and friction to hold over time, and OpenEQA and RealWorldQA [1473] ask functional questions ("Can I navigate through this gap?", "Where would I find a cloth to clean a spill?") in which geometry matters only through what it affords.

Data provenance splits the field. Simulators, the sources behind SpatialVLM [229], AI2-THOR [1621], and Habitat [204], yield unlimited, perfectly annotated data; real-world collections such as ScanNet [448] and OpenEQA supply the clutter, occlusion, and sensor noise that simulation does not reproduce. The resulting practice is sim-to-real: pre-train on large synthetic corpora for general spatial logic, then fine-tune on the

scarcer real data. Annotated 3D data remain scarce and fine-grained metric estimation remains hard for large models; both limits recur throughout the table below.

7.5.1 Comparative Analysis

Table 7.6 compiles 61 spatial-reasoning benchmarks across four categories (physical-focused, rooted in §7.1.2; vision-based, overlapping with §7.3.1; 3D-focused, rooted in §7.3.6 and §7.3.8; and robot-environment, rooted in §7.3.7 and §7.4). A narrower 15-row subset focused specifically on agentic spatial reasoning is provided in Table 7.5 of §7.3.7; this subset filters Table 7.6 to image-and-video benchmarks that exercise agentic deployments.

A modality migration runs from the top of the table to the bottom: the physical-focused benchmarks rely on synthetic 3D renderings, the vision-based block is image- and video-heavy with growing 3D extensions, the 3D-focused block is built on point clouds and RGB-D scans, and the robot block has moved entirely to 3D environments. *The Perspective axis is imbalanced*: a large fraction of vision-based benchmarks carry *Undeclared* ego/exo annotations, the physical-focused benchmarks lean exo (a third-person observer judging physical plausibility), and the robot block is entirely ego; the field is migrating from spectator-style observation to situated cognition faster than benchmark annotations record it. *The Data-Source spectrum is bifurcated* between synthetic or simulated (top block and robot block) and real (middle blocks), with the Real/Sim benchmarks (SpaRE, 3DSRBench, RoboSpatial, SpatialVLM, SITE) as the bridge that the sim-to-real training practice of the opening relies on.

7.5.2 Future Directions

Unified Spatial-Cognition Benchmark. No single benchmark in Table 7.6 exercises a model along all of perception fidelity, symbolic precision, memory persistence, and long-horizon action conditioning simultaneously. A unified benchmark covering all four §7 paradigms (§7.1 rule-based, §7.2 learning-based, §7.3 foundation-model-driven, §7.4 hybrid) would close the evaluation-attribution challenge flagged in §7.4 §Challenges and Trends.

Cross-Modal Evaluation. The current axes (Modality, View, Perspective, Data Source, Task Type) underspecify the cross-modal demand of contemporary spatial reasoning. Future benchmarks should explicitly probe modality-to-modality transfer (*e.g.*, 2D-trained models evaluated on 3D queries; text-conditioned policies evaluated on video-grounded tasks), so that the modality-mismatch challenge flagged in §7.3.6 (text-vs.-image-vs.-3D retrieval) becomes a first-class evaluation axis rather than a per-paper case study.

Embodied Long-Horizon Coverage. The Robot block in Table 7.6 covers navigation thoroughly but underrepresents long-horizon manipulation and multi-room interactive tasks. The §7.3.7 15-row subset (Table 7.5) further restricts attention to image-and-video benchmarks; the embodied long-horizon suites ALFWorld and VirtualHome (§7.3.4) and cross-embodiment evaluation on Open X-Embodiment (§7.3.2) are the nearest existing coverage, and standardizing on them would close the long-horizon-spatiotemporal-coherence challenge flagged in §7.3.7 §Challenges and Trends.

From Descriptive to Interactive Spatial Reasoning. The Task Type column of Table 7.6 is dominated by visual question answering: the overwhelming majority of benchmarks ask a model to *describe* static spatial relations ("where is the object"), and only the small Robot block evaluates reasoning that is conditioned on the agent's own actions. Yet the trajectory traced in this chapter, from passive observation to situated interaction, calls for benchmarks that test *interactive* spatial reasoning: predicting how an action changes the spatial state, judging affordances and reachability ("can I fit, reach, or grasp this"), and reasoning counterfactually about manipulation outcomes. Such evaluation couples the action-conditioned reasoning of §7.2.5 with

Table 7.6: Spatial-reasoning benchmarks surveyed in this chapter, grouped by what they test. `Number` is the evaluation-set size where the source paper documents one (train / test when both are given, scene counts for simulators, ∼ for approximate figures). `Perspective` marks explicitly designed ego (first-person) and exo (third-person) settings; *Undeclared* marks attributes the source paper does not state. `Data Source` and `Task Type` follow the source papers.

Benchmark	Number	Modality	View Type		Perspective		Data Source	Task Type
			Single	Multi	Ego	Exo		
• *Physical Focused Spatial Reasoning*								
IntPhys [1198]	15k / 4.3k	3D	✓	✗	✗	✓	Synthetic	VoE plausibility
IntPhys 2 [1622]	1,416	3D	✓	✗	✗	✓	Synthetic	VoE plausibility
Physion [680]	∼1.2k	3D	✓	✗	✗	✓	Synthetic	Physical prediction / choice
Physion++ [1337]	∼10k	3D	✓	✗	✗	✓	Synthetic	Physical prediction / choice
Physical Concepts Dataset [1623]	300k	3D	✓	✗	✗	✓	Synthetic	classification
Physics-IQ [1624]	396	Video	✓	✗	✗	✓	Real	Generative physical prediction
SPhyR [1625]	1,296	Image	✓	✗	*Undeclared*		Synthetic	Physical prediction / infill
• *Vision-Based Spatial Reasoning*								
CLEVR [1358]	853k	Image	✓	✗	✗	✓	Simulation	VQA
VSR [1479]	10,972	Image	✓	✗	✗	✓	Real	VQA
What's Up [1470]	∼8,000	Image	✓	✗	✗	✓	Real	VQA
RocketScience [1626]	482	Image	✓	✗	✗	✓	Real	VQA
SpaRE [1627]	3.4M	Image	✓	✗	✗	✓	Real/Sim	VQA
RealWorldQA [1473]	765	Image	✓	✗	✓	✗	Real	VQA
3DSRBench [1583]	2,772	Image	✓	✗	✗	✓	Real/Sim	VQA
InternSpatial-Bench [1584]	6,008	Image	✓	✗	✗	✓	Real	VQA
OmniSpatial [1474]	∼8,400	Image	✓	✗	✓	✓	Real	VQA
Spatial457 [1585]	23,752	Image	✓	✗	*Undeclared*		Real	VQA
All-Angles Bench [1586]	∼2,100	Image	✗	✓	*Undeclared*		Real	VQA
EmbSpatial-Bench [1587]	3,640	Image	✓	✗	✓	✗	Real	VQA
MMSI-Bench [1588]	1,000	Image	✗	✓	*Undeclared*		Real	VQA
MindCube [1472]	21,154	Image	✗	✓	*Undeclared*		Simulation	VQA
BLINK [1628]	3,807	Image	✓	✗	✗	✓	Real	VQA
NuScenes-SpatialQA [1629]	3.3M+	Video	✓	✓	✓	✓	Real	VQA
VSI-Bench [227]	5,130	Video	✓	✗	✓	✗	Real	VQA
VLM4D [1589]	∼1,800	Video	✓	✗	✓	✓	Real	VQA
STI-Bench [1590]	2,064	Video	✓	✗	✓	✓	Real	VQA
DSI-Bench [1591]	∼1,700	Video	✓	✗	*Undeclared*		Real	VQA
SITE [1565]	8,068	Video	✓	✗	*Undeclared*		Real/Sim	VQA
RoboSpatial [783]	3M	Image/3D	✓	✗	✓	✗	Real/Sim	VQA
OSD [789]	*Undeclared*	Image	✓	✗	✗	✓	Real	VQA
SpatialVLM [229]	2B	Image	✓	✗	✗	✓	Real/Sim	VQA
ProVision [1630]	65k	Image	✓	✗	✗	✓	Simulation	VQA
WhatIsWhere [1631]	6,000	Image	✓	✗	✗	✓	Real	VQA
SURDS [1632]	9,250	Image	✓	✗	✓	✗	Real	VQA
EarthSpatialBench [1633]	325k+	Image	✓	✗	✗	✓	Real	VQA
PureSpace [1634]	1,200	Image	✓	✗	*Undeclared*		Simulation	VQA
SAT [1635]	175k / 150	Image	✗	✓	✓	✓	Simulation	VQA
SpatiaLQA [1636]	9,605	Image	✓	✗	✗	✓	Real	VQA
Q-Spatial++ [1459]	101	Image	✓	✗	✗	✓	Real	VQA
• *3D Focused Spatial Reasoning*								
ScanQA [41]	41,363	3D	✗	✓	✗	✓	Real (ScanNet)	3D VQA
SQA3D [1620]	33,460	3D	✓	✗	✓	✗	Real (ScanNet)	3D VQA
MMScan QA [1637]	295K	3D+Image	✓	✓	✗	✓	Real (EmbodiedScan)	3D VQA
OpenEQA [1563]	1,632	Video/3D	✓	✗	✓	✗	Real (HM3D, ScanNet)	Embodied QA (EM / A splits)
Space3D-Bench [1638]	1,000	3D	✓	✗	✓	✗	Real(Replica)	3D QA
Scan2Cap [1639]	39,775	3D	✗	✓	✗	✓	Real (ScanNet)	3D Captioning
MMScan Captioning [1637]	172k	3D+Image	✓	✓	✗	✓	Real (EmbodiedScan)	3D Captioning
ScanRefer [373]	51,583	3D	✗	✓	✗	✓	Real (ScanNet)	3D Grounding
Multi3DRefer [1107]	42,735	3D	✗	✓	✗	✓	Real (ScanNet)	3D Grounding
SR3D [374]	83,572	3D	✗	✓	✗	✓	Real (ScanNet)	3D Grounding
NR3D [374]	41,503	3D	✗	✓	✗	✓	Real (ScanNet)	3D Grounding
SpaceEra++ [1640]	925k	Video/3D	✗	✓	✓	✗	Synthetic	3D VQA
Ego3D-Bench [1641]	8,600+	Image	✗	✓	✓	✗	Real	VQA
SpatialScore [1642]	28,093 (Hard: 1,400)	Image/3D	✗	✓	*Undeclared*		Real/Sim	VQA
iVISPAR [1581]	300	3D Env	✓	✗	✗	✓	Simulation	Interactive reasoning
• *Robot*								
AI2-THOR [1621]	120 (Scenes)	3D Env	✗	✓	✓	✗	Simulation	Nav/Manip
TDW [1643]	*Infinite*	3D Env	✗	✓	✓	✓	Simulation	Physical Sim
Habitat 2.0 [204]	100 (Scenes)	3D Env	✗	✓	✓	✗	Simulation	Navigation
iGibson 2.0 [1344]	15 (Scenes)	3D Env	✗	✓	✓	✗	Simulation	Interactive Nav
Habitat 3.0 [98]	211 (Scenes)	3D Env	✗	✓	✓	✗	Simulation	Social Nav
R2R [1415]	21,567	Image/Path	✗	✓	✓	✗	Real	Navigation
RxR [1644]	126,069	Image/Path	✗	✓	✓	✗	Real	Navigation

the embodied hybrid pipelines of §7.4, extending the descriptive benchmarks surveyed here toward the manipulation- and affordance-grounded settings that embodied deployment ultimately demands.

Chapter 8

Spatial Reconstruction and Generation

Spatial reconstruction and generation is the productive counterpart to spatial perception and reasoning: rather than analyzing a scene that is given, it synthesizes spatial content, recovering or creating the geometry, appearance, and dynamics of objects, scenes, and worlds from conditions such as text, images, partial observations, or explicit constraints. Where perception (Chapter 6) maps observations into structured spatial representations and reasoning (Chapter 7) draws inferences over them, generation closes the loop by mapping intent back into concrete spatial artifacts, a capability central to embodied agents, content creation, simulation, and world modeling.

The chapter is organized along an axis of increasing output structure and dimensionality, from flat images to volumetric objects and scenes, to video, embodied motion, and full four-dimensional space-time content, and finally to world models and to programmatic generation that emits interpretable, executable structure. What holds this range together is a single recurring backbone: an encoder maps conditions into a latent, a generator samples structure, and a decoder or renderer produces the final artifact, specialized to each output domain. Read this way, the subfields are less seven separate problems than one conditional-generation abstraction instantiated against progressively richer codomains.

8.1 2D Spatial Generation

The 2D spatial generation encompasses a broad spectrum of tasks that require synthesizing or editing images under explicit spatial constraints. These constraints may be *user-specified* (bounding boxes, masks, sketches, keypoints, depth maps) or *derived* from scene understanding (layout graphs, semantic maps, relational structures), and the overarching goal is to produce images that are not only visually realistic but also *faithful to the specified spatial configuration*: objects appearing in the correct locations, at the correct scales and orientations, while remaining globally coherent in lighting, perspective, and style. 2D generation is where the most mature *semantic-to-pixel prior* in contemporary AI is taught to honor *spatial structure*: it is the proving ground on which large pre-trained visual priors are forced to localize objects, follow geometric scaffolds, respect occlusion ordering, and obey region- or instruction-level edits, all without falling out of the implicit grammar of natural images.

The section follows the 2D pipeline in three steps that share one latent-generator framework. §8.1.1 reviews foundational text-to-image generation, the semantic-to-pixel prior on which everything else here is built, and the three paradigms (adversarial, autoregressive, and diffusion with its flow-matching generalization) that parameterize it. §8.1.2 adds a structural condition (layouts, masks, edges, depth, pose, scene graphs, region phrases) and asks how it is injected into a frozen or co-trained generator. §8.1.3 replaces the empty

canvas with an existing image and a mask, so that the generator must change what lies inside the mask while preserving what lies outside it.

8.1.1 Text-to-image Generation

Text-to-image (T2I) generation synthesizes an image from a natural-language description. For the rest of this chapter its importance lies less in the images themselves than in the prior it provides: a high-capacity mapping from compositional language to the visual space of objects, surfaces, layouts, and viewpoints, learned once at web scale and then reused. The spatial controllers of §8.1.2 and the editors of §8.1.3 impose constraints on this prior rather than replacing it, and the text-to-3D pipelines of §8.2 distill it through score distillation or multi-view consistency objectives; whatever relational competence the prior has, or lacks, therefore propagates downstream. The field has passed through three architectural paradigms, from adversarial training to autoregressive token modeling to iterative denoising, with diffusion and its flow-matching generalization now setting the standard for quality and prompt fidelity [185, 1645].

8.1.1.1 Task Definition

We formulate text-to-image generation in a paradigm-agnostic form that exposes the structural choices shared by modern systems. Let $\mathcal{X}$ denote the space of tokenized text prompts and $\mathcal{Y} = [0,1]^{H\times W\times 3}$ the pixel-space image domain. Almost all modern T2I systems operate not directly on $\mathcal{Y}$ but on a compressed *latent space* $\mathcal{Z} \subset \mathbb{R}^d$ provided by an autoencoder pair

$$E : \mathcal{Y} \to \mathcal{Z}, \qquad D : \mathcal{Z} \to \mathcal{Y}, \qquad D \circ E \approx \mathrm{id}_{\mathcal{Y}}, \tag{8.1}$$

where E encodes a training image into a low-dimensional code and D decodes a code back into a pixel image. A text encoder $\phi : \mathcal{X} \to \mathcal{E}$ maps the prompt into a conditioning embedding $\phi(\mathbf{x}) \in \mathcal{E}$. The core learning problem is then to train a conditional latent sampler

$$G_\theta : \mathcal{E} \to \mathcal{Z}, \qquad \hat{\mathbf{z}} \sim p_\theta\big(\mathbf{z} \,\big|\, \phi(\mathbf{x})\big), \tag{8.2}$$

so that, after decoding, the resulting image $\hat{\mathbf{y}} = D(\hat{\mathbf{z}})$ is both photorealistic and semantically aligned with the prompt $\mathbf{x}$. The three paradigms covered below differ chiefly in how the conditional distribution $p_\theta(\mathbf{z} \mid \phi(\mathbf{x}))$ is parameterized and optimized; the autoencoder pair (E, D) and the text encoder ϕ are largely shared design choices across paradigms. Figure 8.1 schematizes this paradigm-agnostic pipeline and the substrate that the three paradigms share.

8.1.1.2 Methodology

Three generative paradigms, taken in the order in which each dominated T2I, parameterize the sampler G_θ of Eq. (8.2) differently; the bottom row of Figure 8.1 draws them against the common pipeline. A second set of advances refines the sampler without redefining it: backbone evolution, sampling acceleration, and the migration of leading systems to proprietary stacks. Structural conditioning interfaces, the fourth such refinement, are the subject of §8.1.2 and are not repeated here.

Generative Adversarial Networks. GAN-based T2I models instantiate G_θ as a feed-forward generator $G_\theta(\mathbf{n}, \phi(\mathbf{x})) \to \hat{\mathbf{z}}$ driven by an auxiliary noise vector $\mathbf{n} \sim \mathcal{N}(0, I)$; the conditional distribution $p_\theta(\mathbf{z} \mid \phi(\mathbf{x}))$ is defined only implicitly, as the pushforward of $\mathbf{n}$ through G_θ. Training follows the two-player game of Goodfellow et al. [1646], pitting the generator against a discriminator D_ψ,

$$\min_\theta \max_\psi \; \mathbb{E}_{\mathbf{y}\sim p_{\text{data}}}\big[\log D_\psi(\mathbf{y}, \phi(\mathbf{x}))\big] + \mathbb{E}_{\mathbf{n}}\big[\log\big(1 - D_\psi(D(G_\theta(\mathbf{n}, \phi(\mathbf{x}))), \phi(\mathbf{x}))\big)\big], \tag{8.3}$$

in which D_ψ scores both photorealism and text-image consistency. The result is sharp high-frequency detail and single-pass inference, bought at the price of unstable training and mode collapse. Within this framework,

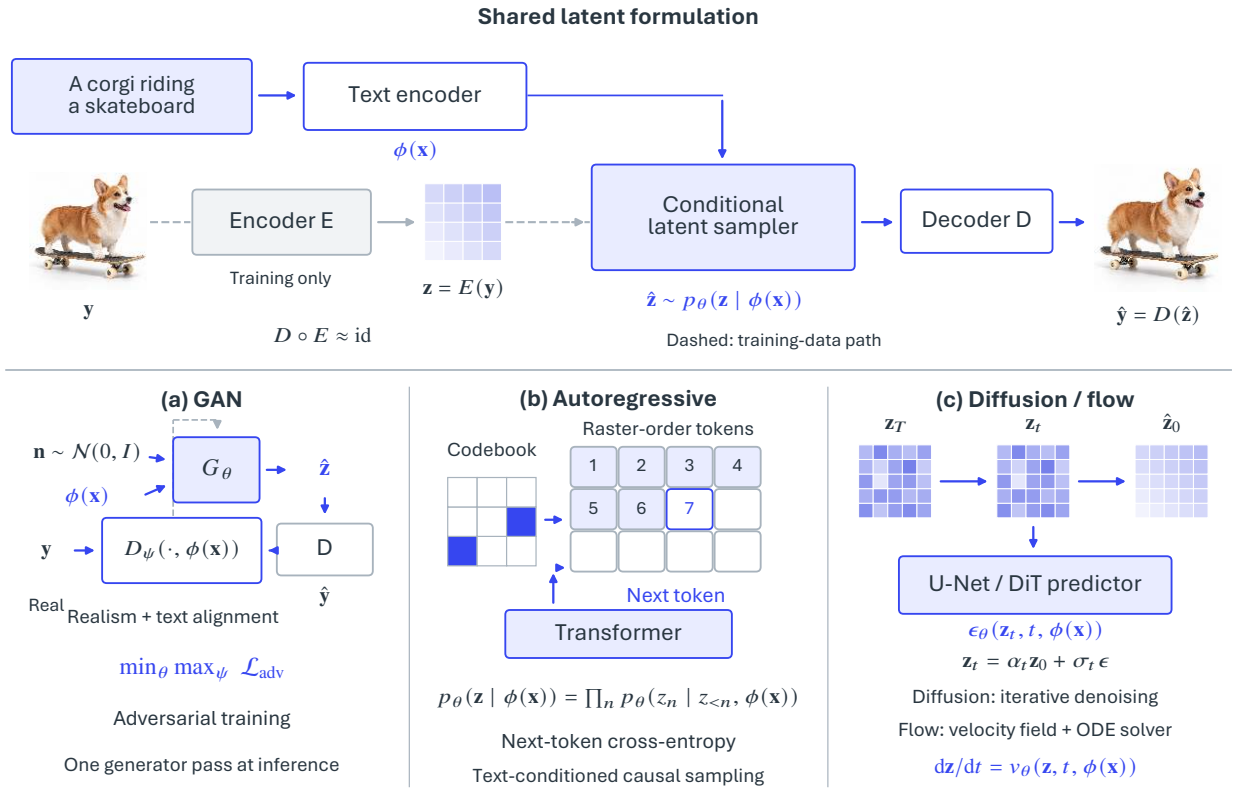


Figure 8.1: The latent-generator view of text-to-image generation. Top: the shared pipeline of Eqs. (8.1) and (8.2); the encoder E and the dashed paths are used only in training. Bottom: the sampler G_θ as (a) a GAN, (b) an autoregressive token model, and (c) diffusion or flow matching. Not every architecture uses a separate autoencoder; the corgi is a schematic illustration.

StackGAN [1647] generated a low-resolution image first and refined it with a second GAN conditioned on both the text and the coarse image; AttnGAN [1648] let the generator attend to individual words while rendering different regions, with an auxiliary image-text matching loss (DAMSM) that reduced attribute errors such as wrongly colored objects; XMC-GAN [1649] replaced that loss with cross-modal contrastive learning and improved multi-object prompts. By 2021 diffusion models had overtaken these systems on COCO, chiefly because they train stably and cover more of the image distribution, and research attention moved on; the GAN emphasis on high-frequency detail survives in hybrid and adversarially fine-tuned samplers.

Autoregressive Models. Autoregressive (AR) models discretize the latent space: the autoencoder E is realized as a vector-quantized variational autoencoder (VQ-VAE) whose output is a sequence of discrete tokens $\mathbf{z} = (z_1, \ldots, z_N) \in \mathcal{V}^N$ over a learned codebook $\mathcal{V}$, and G_θ is factorized in causal raster order,

$$p_\theta\big(\mathbf{z} \,\big|\, \phi(\mathbf{x})\big) \;=\; \prod_{n=1}^{N} p_\theta\big(z_n \,\big|\, z_{<n},\, \phi(\mathbf{x})\big), \tag{8.4}$$

trained by maximum likelihood on the next-token cross-entropy. T2I thus becomes sequence-to-sequence translation, and the transformer that performs it inherits the compositional competence of language models: DALL-E [1650] modeled text and image tokens as a single sequence and showed zero-shot generalization across prompt types. The cost is decoding latency, one forward pass per token, which grows with resolution. Parti [1651] scaled the transformer to 20B parameters, generated at 256×256, and reached 1024×1024 with a convolutional super-resolution stage; Muse [1652] abandoned strict sequential order for masked token

prediction, filling in many tokens per step over a handful of rounds, and matched diffusion quality on several benchmarks at a fraction of the sampling time. AR generation remains attractive where images are produced jointly with text (interleaved documents, storyboards) and a single sequence model handles both.

Diffusion Models. Diffusion models realize G_θ as iterative denoising rather than a single feed-forward pass. A fixed *forward* process corrupts the clean latent $\mathbf{z}_0 = E(\mathbf{y})$ along a schedule $(\alpha_t, \sigma_t)_{t=0}^{T}$, $\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \epsilon$ with $\epsilon \sim \mathcal{N}(0, I)$ and $\mathbf{z}_T \approx \mathcal{N}(0, I)$; a noise-prediction network ϵ_θ learns to invert it under the denoising score-matching objective

$$\min_\theta \; \mathbb{E}_{t, \mathbf{z}_0, \epsilon} \Big[\big\| \epsilon - \epsilon_\theta\big(\mathbf{z}_t,\, t,\, \phi(\mathbf{x})\big) \big\|_2^2 \Big], \tag{8.5}$$

which is the diffusion-specific realization of the training objective implicit in Eq. (8.2). At inference the model starts from $\mathbf{z}_T \sim \mathcal{N}(0, I)$, applies ϵ_θ for $t = T, \ldots, 1$, and decodes $\hat{\mathbf{y}} = D(\hat{\mathbf{z}}_0)$; each step corrects a small error, which is why the final samples are both detailed and faithful. Running this process in the latent space of Eq. (8.1) rather than in pixels, as Latent Diffusion Models [185] did (*e.g.*, $4 \times 64 \times 64$ instead of $3 \times 512 \times 512$ for a 512px image), cut memory and compute enough to train on commodity hardware and to release Stable Diffusion openly. Conditioning the denoiser on embeddings from a pretrained language-image model, as in GLIDE [1653] and DALL·E 2 [1654], improved prompt alignment, and classifier-free guidance [1655] supplied a single knob that trades diversity against fidelity by extrapolating the conditional prediction away from the unconditional one. The remaining weakness is sampling cost, dozens of sequential network evaluations per image; higher-order ODE samplers and the distillation methods below address it. Diffusion or flow-matching cores now underlie most leading systems, with the native multimodal autoregressive stacks discussed under proprietary systems the main exception.

Flow Matching and Rectified Flow. Rather than a fourth paradigm, flow matching is diffusion reformulated in continuous time. It parameterizes G_θ as a learned *velocity field* $v_\theta(\mathbf{z}_t, t, \phi(\mathbf{x}))$ instead of a noise predictor: training picks the straight interpolation $\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\mathbf{z}_1$ between a clean latent $\mathbf{z}_0 = E(\mathbf{y})$ and a Gaussian sample $\mathbf{z}_1 \sim \mathcal{N}(0, I)$ and regresses the velocity,

$$\min_\theta \; \mathbb{E}_{t, (\mathbf{z}_0, \mathbf{z}_1)} \Big[\big\| v_\theta(\mathbf{z}_t, t, \phi(\mathbf{x})) - (\mathbf{z}_1 - \mathbf{z}_0) \big\|_2^2 \Big], \tag{8.6}$$

and inference integrates the probability-flow ODE $\mathrm{d}\mathbf{z}_t/\mathrm{d}t = v_\theta(\mathbf{z}_t, t, \phi(\mathbf{x}))$ from $t = 1$ back to $t = 0$. The formulation was introduced concurrently by Lipman et al. [1656] (Flow Matching) and Liu et al. [1657] (Rectified Flow); Albergo and Vanden-Eijnden [1658] gave the unifying stochastic-interpolant view in which diffusion is the special case of a noisier interpolation schedule. Three consequences matter in practice. Straight-line interpolations yield nearly straight ODE trajectories, so matched quality is reached in noticeably fewer steps (genuine one-step generation still needs a reflow or distillation stage); the objective is a single regression with no noise-schedule design and no per-schedule guidance recalibration; and the loss scales smoothly with model size and resolution. Stable Diffusion 3 [1659] scaled the recipe to T2I on a DiT-style transformer backbone and reported the best GenEval scores among open models together with favorable human preference on Parti-prompts; the Flux family followed the same recipe. Flow matching composes with every refinement below as readily as a noise-predictor core does and has become the default continuous-time formulation in 2024-onwards systems.

Backbone Evolution: From U-Net to Diffusion Transformers. Early diffusion models used the U-Net for its multiscale feature fusion and locality bias. The Diffusion Transformer (DiT) of Peebles and Xie [1645] replaced it with a stack of transformer layers over latent patch tokens, giving every denoising step a global receptive field, and showed that FID improves monotonically with depth, width, and token count along predictable scaling laws, reaching FID 2.27 on class-conditional ImageNet at 256^2. Attention cost grows quadratically with resolution, so hybrid CNN-transformer designs and efficient attention remain active; the DiT lineage nonetheless underlies SD3 and most 2024-onwards systems, and it points toward a single transformer that generates several modalities, differing only in tokenization and training data.

Sampling Acceleration and Distillation. Fifty or more denoising steps rule out interactive use and low-power deployment. Salimans and Ho [1660] distilled a T-step teacher into a $T/2$-step student and iterated the halving to reach 1–4 steps. Consistency Models [1661] train a network to map a noisy latent at any timestep directly to the clean image, either from scratch with a consistency loss or by distilling a pretrained diffusion model, and Latent Consistency Models [1662] applied the idea to Stable Diffusion's latent space, matching 50-step quality in 2–4 steps. All of these are post-training procedures that leave the architecture untouched, so any strong backbone can be made fast after the fact.

Proprietary and Commercial Systems. Leading systems have migrated to proprietary, industrially scaled stacks of three kinds. Native multimodal models such as OpenAI's GPT Image 1.5 [1663] and Google's Gemini 3 Pro Image (Nano Banana Pro) [1664] run a unified transformer over interleaved text and image tokens and emphasize semantic reasoning and multi-turn editing; latent-diffusion iterations such as Imagen 4 [1665] scale the LDM recipe with stronger text encoders and curated data; preference-tuned systems such as Midjourney v7 [1666] and FLUX.2 [max] [1667] fine-tune strong priors on human-preference data (FLUX.2 on the rectified-flow recipe of §8.1.1.2; Midjourney's architecture is undisclosed). Each reduces to one of the three paradigms plus some of the refinements just listed; because architectural and training detail is withheld, public benchmarking and reproduction still rest on the open SD / DiT stack.

8.1.1.3 Benchmarks

Three independent things are usually conflated under "benchmark": the training corpora on which models are fit, the evaluation suites of prompts that probe particular capabilities, and the metrics that score the resulting images. A model trained on LAION-5B can be evaluated on GenEval prompts and scored by FID, CLIPScore, or an LLM judge, and disagreement between those scores is itself informative.

Training Corpora. Corpus scale has driven much of T2I progress. Early work fit fine-grained sets with ten captions per image, CUB-200-2011 (11,788 bird images) [1668] and Oxford-102 Flowers (8,189 images) [1669]; MS-COCO Captions [440] (123,287 images, five captions each) became the standard general-scene corpus and, for years, the evaluation set as well, a conflation that the suites below were designed to end. Current models train on web-scale image-text pairs: LAION-5B [1670] releases 5.85 billion CLIP-filtered pairs, and proprietary corpora of similar scale, less documented, underlie DALL-E 2 and Parti. Scale brought duplication, demographic bias, unsafe content, and licensing disputes with it, and filtered subsets such as LAION-Aesthetics and curation pipelines are now part of every training recipe.

Evaluation Benchmark Suites. An evaluation suite is a curated prompt set plus a verification protocol: prompts are adversarial or capability-stratified, reference images are usually absent, and scoring is automatic (an object detector, a VQA model, or an LLM/MLLM judge) rather than reconstruction-based. Table 8.1 lists the main suites since 2022. The earliest, DrawBench [1671] and PartiPrompts [1651], are human-pairwise prompt sets released with Imagen and Parti; HRS-Bench [1672] and HEIM [1673] are holistic, scoring 13 and 12 axes respectively; TIFA [1674], GenEval [1675], and T2I-CompBench [1676] introduced automatic verification by VQA, object detection, and BLIP-VQA plus UniDet, and the latter two are now reported alongside FID and CLIPScore in most papers. For spatial intelligence the salient suite is VISOR [1677], which isolates two-object relations (left/right, above/below) over 25,280 prompts and finds that otherwise strong models frequently render the wrong configuration. DPG-Bench [1678] targets dense, long compositional prompts, GenAI-Bench [1679] pairs compositional prompts with VQAScore judging, and WISE [1680] asks whether generated images agree with real-world facts (flag colors, landmarks, anatomy), a knowledge axis orthogonal to compositional binding.

Evaluation Metrics. Metrics fall into four tiers. Distributional metrics compare the generated and real image populations and say nothing about any individual prompt: the Fréchet Inception Distance (FID) [1681] is the Wasserstein-2 distance between Gaussian fits of Inception-V3 features of generated and real images,

Table 8.1: Representative evaluation benchmark suites for text-to-image generation, ordered approximately by year of release. "Scoring" indicates the dominant verification mechanism: detector- or VQA-based automatic checking, learned preference models, MLLM/LLM-as-judge rubric scoring, or aggregated human preference. Several suites combine multiple scoring channels.

Benchmark	Year	# Prompts	Evaluation Axis	Scoring
DrawBench [1671]	2022	200	Broad capability + alignment (11 categories)	Human pairwise
PartiPrompts [1651]	2022	1,600	Capability spectrum (12 categories)	Human pairwise
VISOR [1677]	2022	25,280	Two-object spatial relations (left/right, above/below)	Object-detector verification
TIFA [1674]	2023	4,081	Faithfulness (VQA over auto-generated questions)	VQA / LLM-as-judge
HRS-Bench [1672]	2023	45,000	Holistic; 13 skills $\times$ 50 scenarios	Automatic, mixed
HEIM [1673]	2023	62 scenarios	Holistic; 12 aspects (alignment, quality, knowledge, bias, toxicity, ...)	Automatic + human
GenEval [1675]	2023	553	Object-centric: single/two-obj., counting, color, position, attribute	Detector verification
T2I-CompBench [1676]	2023	6,000	Compositional: attribute binding, object relationships, complex prompts	BLIP-VQA + UniDet
DPG-Bench [1678]	2024	1,065	Dense, long compositional prompts	MLLM-based scoring
GenAI-Bench [1679]	2024	1,600	Compositional T2V/T2I prompts	VQAScore (LLM-judge)
WISE [1680]	2025	1,000	World-knowledge-informed semantic accuracy	LLM-based grading

the Inception Score rewards images that are individually confidently classified yet collectively diverse, and generative precision/recall [1682, 1683] separate quality from coverage through neighborhood tests in feature space. Alignment metrics score one image against its prompt; CLIPScore [1684], the cosine similarity between the CLIP embeddings of $\hat{\mathbf{y}}$ and $\mathbf{x}$, is cheap and reference-free but inherits CLIP's insensitivity to attribute binding, counting, and spatial relations. Learned preference models, HPSv2 [1685], ImageReward [1686] (fit to 137K human comparisons), and PickScore [1687] (fit to the $\sim$500K real-user comparisons of Pick-a-Pic), regress a scalar that correlates with average user preference, bundling realism, alignment, and aesthetics; they have become the default automatic ranking while inheriting their raters' biases and degrading out of distribution.

The newest tier uses a vision-language model as judge. TIFA [1674] has a language model write verification questions from the prompt and a VQA model answer them against the image, the score being the fraction answered correctly; VQAScore [1688] collapses this to a single yes/no probability under a fine-tuned CLIP-FlanT5 model and reports the best human correlation on compositional suites; VIEScore [1689] and LLMScore [1690] prompt a multimodal LLM with a structured rubric and return per-axis scores with brief rationales, and direct prompting of frontier VLMs with custom rubrics has become routine for ad-hoc capability probes. Judge metrics track human preference on compositional and spatial prompts far better than CLIPScore, at the cost of a frontier-model call per image, sensitivity to rubric wording, and drift across judge-model versions that makes numbers from different papers hard to compare. No single tier suffices, and current papers report a tuple (FID or IS, CLIPScore, one or two preference scores, one judge score) on several suites plus a small human study; cross-metric disagreement is read as a warning rather than averaged away.

8.1.1.4 Challenges and Trends

The weakness that matters most for this handbook is **spatial-relation faithfulness**, a sub-problem of compositionality. Models frequently misplace objects relative to one another: a prompt that puts a cup to the left of a laptop is often rendered with the two swapped or stacked, and VISOR [1677] shows that language-specified relations are frequently ignored outright. Data-centric remedies help; SPRIGHT [1691]

re-captions training images with explicit spatial descriptions and measurably improves consistency, but reliable control over where objects appear remains open. Because layout-conditioned generation (§8.1.2) and text-to-3D (§8.2) inherit whatever relational competence this prior provides, the limitation is foundational rather than cosmetic. Fine structure is a second gap: legible text, hands, and faces still fail in otherwise strong models and appear to need architectural or data-specific fixes rather than scale alone.

Evaluation reliability is the third. Automatic scores correlate imperfectly with human judgment precisely on the compositional, counting, and spatial-relation prompts that matter here; judge-based metrics close part of the gap but add rubric sensitivity and judge-version drift, so headline numbers across papers are not directly comparable. As GenEval and T2I-CompBench become standard, their prompts risk leaking into web-scraped training corpora, in which case high scores increasingly measure memorization rather than capability; periodic refresh of held-out prompts, blinded judges, and explicit train/test decontamination are the obvious countermeasures and are not yet routine.

Societal concerns follow the data. Models inherit demographic bias from their corpora and will produce unsafe content on request; prompt classifiers, safety filters, and fine-tuning on curated subsets reduce but do not remove the problem, and photorealistic synthesis raises misinformation and intellectual-property questions that watermarking and data curation address only in part. The extension of these priors along the time axis, where identity and motion must stay consistent across frames, is the subject of §8.3; the next two subsections stay in the image plane and ask how the prior can be made to obey spatial constraints and local edits.

8.1.2 2D Spatial-aware Image Generation

2D spatial-aware image generation synthesizes images that are faithful not only to a text prompt but to explicit spatial constraints in the image plane: where objects appear, how large they are, what shape they take, and how they relate to one another (*left of*, *on*, *behind*). The constraint may be a set of boxes or a scene layout, an instance or semantic mask, an edge or sketch map, a depth or normal map, 2D keypoints, a scene graph, or a phrase bound to a region, and the pipelines that honor it are almost all diffusion models augmented with a conditioning pathway or a guidance term. The tension that organizes the subfield is three-way: *adherence* to hard constraints against *perceptual realism* and *diversity*, and both against *compositionality* when many small or occluding objects carry discrete relations. Depth and normal maps enter here as structural guides for occlusion ordering and surface orientation, not as metric measurements; recovering true distance belongs to reconstruction (§8.2).

8.1.2.1 Task Definition

We inherit the latent-generator framework of §8.1.1. Let $\mathcal{X}$ denote tokenized prompts, $\mathcal{Z}$ the latent space induced by the autoencoder pair (E, D) of Eq. (8.1), and $\phi : \mathcal{X} \to \mathcal{E}$ the text encoder. Spatial-aware generation *augments* this framework with a structural condition $\mathbf{c} \in \mathcal{C}$ drawn from one or more conditioning channels (boxes, masks, edges, depth, pose, scene graphs, region phrases, reference images), encoded by a condition encoder $\kappa : \mathcal{C} \to \mathcal{C}'$. The conditional latent sampler of Eq. (8.2) is extended to a joint conditional

$$G_\theta : \mathcal{E} \times \mathcal{C}' \to \mathcal{Z}, \qquad \hat{\mathbf{z}} \sim p_\theta\big(\mathbf{z} \,\big|\, \phi(\mathbf{x}),\, \kappa(\mathbf{c})\big), \tag{8.7}$$

and the generated image $\hat{\mathbf{y}} = D(\hat{\mathbf{z}})$ is required to be (i) photorealistic, (ii) semantically aligned with $\mathbf{x}$, and (iii) faithful to the structural condition $\mathbf{c}$ in placement, scale, shape, and pairwise relation. The unconditional limit $\mathbf{c} = \varnothing$ recovers pure T2I generation as in §8.1.1; the substance of the present subsection lies in how κ is realized for each conditioning channel and how its output is injected into G_θ. Which channel populates $\mathbf{c}$, and how densely, ranges from pixel-level maps (masks, edges, depth) through coordinate-level scaffolds (boxes, keypoints) to discrete symbols (scene graphs, region phrases); the modality survey below follows

that ordering. Interactive constraints supplied by dragging handles during generation belong to the editing regime of §8.1.3.

8.1.2.2 Methodology

Two axes organize the methods: the *conditioning modality* (what kind of structural input $\mathbf{c}$ is consumed) and the *injection mechanism* (how $\kappa(\mathbf{c})$ enters the latent sampler G_θ). They are nearly independent. One mechanism, a ControlNet-style side branch, serves canny edges, pose skeletons, depth, and segmentation alike, and one modality, bounding boxes, is served by side branches, grounded cross-attention, or inference-time attention shaping, so the design space is a matrix of (modality $\times$ mechanism) cells rather than a list of monolithic methods. Figure 8.2 draws the five mechanisms on one backbone silhouette.

Conditioning Modalities. *Dense pixel-level scaffolds* supply a value at every location, which makes them the easiest channels to follow and the most restrictive on creativity. Instance and semantic masks drive region-conditioned synthesis, with MultiDiffusion [1692] offering a training-free protocol that samples jointly over overlapping mask windows; edges, sketches, and line art, popularized by the canny and soft-edge variants of ControlNet [1693] and the edge head of T2I-Adapter [1694], fix only the location of intensity discontinuities and leave texture and appearance to the prompt; depth and normal maps, served by ControlNet's depth variant, impose coarse occlusion ordering and surface orientation without requiring a metric reconstruction. *Coordinate-level scaffolds* drop to per-region or per-point supervision. Boxes with labels are the most studied channel, a set of $(\text{box}, \text{label})$ tuples fixing location and rough scale, with LayoutDiffusion [1695], LayoutLLM [1696], and the open-set GLIGEN [1697] as representatives; 2D keypoints (typically human pose) are sparser still, a handful of joint coordinates, injected through ControlNet or handled by GLIGEN as a special case of grounding. *Discrete symbolic structure* carries no pixel- or coordinate-level information at all, and the model must map it onto the image plane itself. Scene graphs reach back to the sg2im line of Johnson et al. [1698] and re-emerged in the diffusion era through LLM-emitted layouts, as in LayoutLLM, where the graph is first decoded into a coordinate-level layout; region phrases and referring expressions bind a text span to a box, mask, or free-form polygon and are served by GLIGEN's grounded cross-attention and by phrase-grounding heads attached to SD and SDXL [1699].

Injection Mechanisms. Five mechanism families cover the literature; production pipelines routinely stack a side branch for dense geometric channels, an adapter for style or a reference image, and inference-time guidance to sharpen adherence on adversarial prompts.

Side-branch controllers. The backbone weights stay frozen and an auxiliary branch mirroring the encoder half of the base U-Net is trained, connected through *zero-convolution* layers whose weights are initialized to zero so that the augmented model reproduces the base model exactly at the start of training and degrades gracefully if the branch is poorly tuned. ControlNet [1693] is the prototype and the first widely adopted retrofit pathway for non-text structural control: the condition (a canny edge map, an OpenPose skeleton, a depth or normal map, a segmentation mask, or a scribble) is encoded by the side branch in parallel with the noisy latent and injected into the base U-Net through additive residual connections at each scale. One branch is trained per modality on labeled image–condition pairs, which produced a library of plug-in controllers for SD 1.5 and SDXL [1699] that compose with prompt conditioning at inference time without retraining the backbone; most later designs either shrink this recipe or fold it into a unified adapter library.

Lightweight adapters. Where the side branch spends capacity (half a U-Net per modality), T2I-Adapter [1694], developed concurrently with the same goal of plug-in control over a frozen backbone, spends as little as possible: a few residual blocks per modality whose multi-scale features are added directly into the corresponding encoder stages. Several adapters can be combined at inference, an image conditioned simultaneously on an OpenPose skeleton, a depth map, and a color palette with a scalar weight on each, so the adapter is preferred for multi-channel fusion and fast iteration, and the side branch where maximum single-channel fidelity matters.

Grounded cross-attention. Side branches and adapters run an external pathway alongside the backbone; the next two families intervene *inside* its attention stack. GLIGEN [1697] inserts a new *gated self-attention* layer between the existing self-attention and cross-attention of each U-Net transformer block; the layer attends jointly over the visual tokens and a set of *grounding tokens* that encode each (phrase, box) pair, a learned scalar gate controls how strongly the grounding signal enters the residual stream, and only the new layer is trained, so the network falls back to ungrounded behavior when no constraint is supplied. The problem it solves is complementary to the side branch: not how to inject a dense pixel-level scaffold, but how to introduce *open-set* symbolic grounding (any category named in text, localized by a box, keypoints, or a reference image) into a frozen T2I backbone, with several grounding channels interleaved within one mechanism.

Layout-aware cross-attention. LayoutDiffusion [1695] is the dense-supervised cousin of grounded attention: rather than gating a new layer onto a frozen backbone, it co-trains a layout-aware cross-attention block end to end for layout-to-image (L2I) generation specifically. A structured layout, a set of (box, class label) tuples, becomes a sequence of region tokens fused with the noisy latent at every denoising step, and an explicit *object–scene* factorization models global scene context separately from per-object features so that object identity, scale, and pairwise relations survive in cluttered scenes. It reports stronger layout IoU and relation satisfaction than ControlNet or T2I-Adapter on multi-object COCO benchmarks, at the cost of being specialized to box-and-label input rather than the broader modality library.

Inference-time guidance. The fifth family adds no parameters and modifies only sampling. BoxDiff [1700] is the canonical training-free spatial controller: it shapes the cross-attention maps so that activations of box-localized prompt tokens concentrate inside their boxes while the backbone stays untouched. MultiDiffusion [1692] fuses overlapping mask windows at sampling time, and composing scores from several conditional models or tuning the classifier-free guidance scale per region are related strategies. Because nothing is retrained, these methods suit rapid prototyping, backbones whose weights are unavailable for fine-tuning, and per-prompt sharpening on top of the trained mechanisms above.

8.1.2.3 Benchmarks

Two things change relative to §8.1.1. The ground truth is a *structural annotation* (a layout, a mask, a keypoint set) rather than a reference image, so reconstruction-style metrics give way to adherence metrics that score how closely the generated image honors $\mathbf{c}$; and every channel admits its own adherence metric, so a meaningful report combines a channel-specific score with the realism and alignment scores inherited from the previous subsection.

Training Corpora. Spatial-aware models are typically trained on corpora that pair images with rich structural annotations: COCO Captions [440] for layouts and captions; ADE20K [492] (20K images, 150 classes) for semantic and instance masks; Visual Genome [796] for object boxes, attributes, and pairwise relations supporting scene-graph conditioning; RefCOCO / RefCOCO+ / RefCOCOg [1082] for phrase grounding; and PhraseCut [1092] for region-prompt mask supervision. Additional structure (canny edges, depth, normals, OpenPose skeletons) is typically derived on the fly from these base corpora by running pretrained predictors over each image, producing the paired image–condition data on which ControlNet-style and adapter-style pipelines are trained.

Evaluation Benchmark Suites. Most of the suites surveyed in §8.1.1 (Table 8.1) include axes that probe spatial-aware generation directly and serve as its primary out-of-the-box benchmarks. GenEval [1675] exposes a position subset that scores whether two objects appear in a specified relation (*left of*, *above*); T2I-CompBench [1676] provides explicit *spatial* and *non-spatial relationship* categories with BLIP-VQA and UniDet scoring; HRS-Bench [1672] includes a *spatial* skill among its thirteen; GenAI-Bench [1679] offers compositional prompts on which spatial relations are a frequent failure of CLIP-based scoring; and WISE [1680] measures knowledge-grounded location accuracy. Beyond these, the L2I literature reports task-specific evaluations on COCO layouts: detection-based AP for box-localized objects alongside FID and CLIPScore.

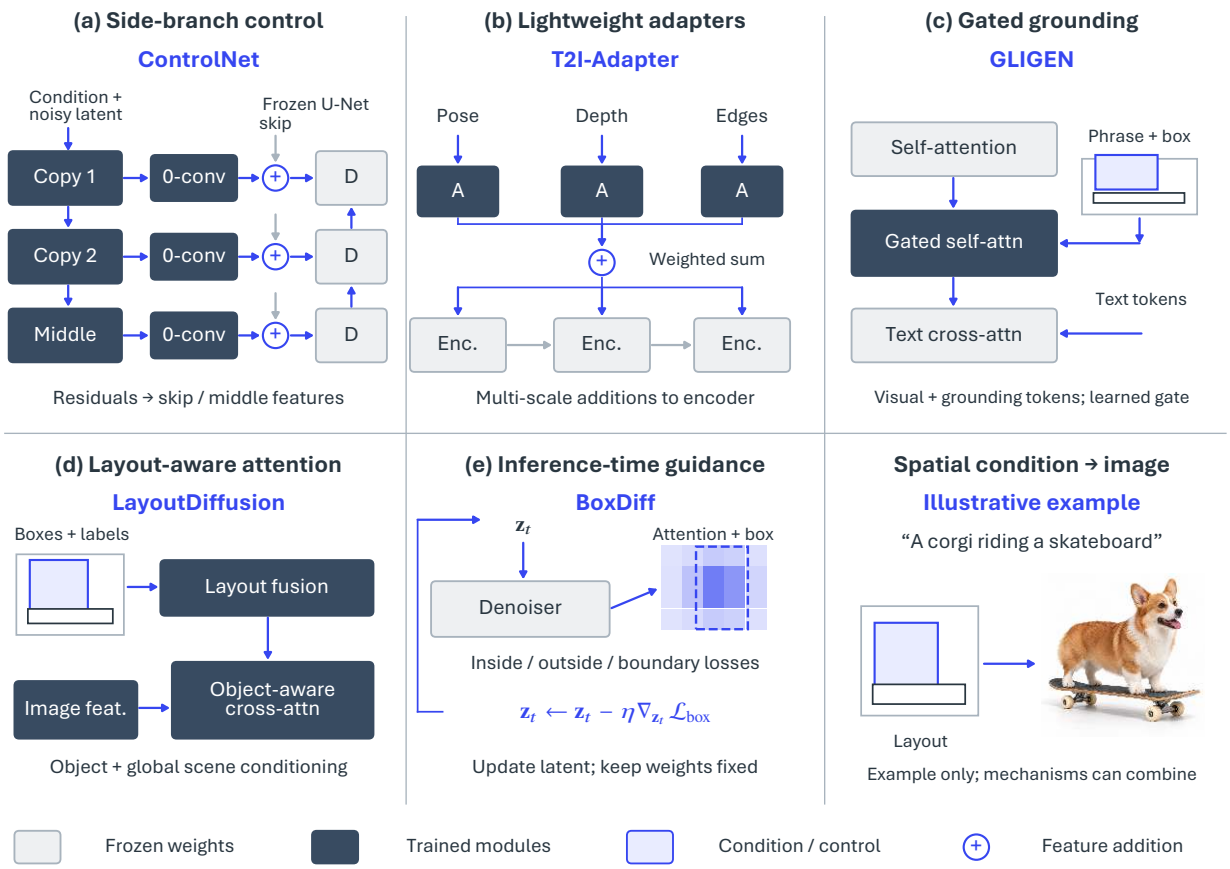


Figure 8.2: Five mechanisms for injecting a spatial condition $\kappa(\mathbf{c})$ into a text-to-image denoiser: (a) side-branch control (ControlNet [1693]), (b) lightweight adapters (T2I-Adapter [1694]), (c) gated grounding (GLIGEN [1697]), (d) layout-aware cross-attention (LayoutDiffusion [1695]), and (e) inference-time guidance (BoxDiff [1700]). The corgi example is illustrative, not a comparison of outputs.

Constraint-specific Metrics. Layout adherence is box IoU between the conditioning layout and the boxes recovered by a frozen detector on $\hat{\mathbf{y}}$, often with per-object center distance; mask adherence uses mask IoU and boundary F-score against the conditioning mask; edge and sketch adherence report recall (sometimes precision) of edges re-extracted from $\hat{\mathbf{y}}$; geometry adherence reports depth RMSE and normal angular error where ground-truth geometry exists; pose adherence reports keypoint reprojection error or PCK; and relational adherence reports relation-satisfaction accuracy and per-entity object AP for scene-graph and phrase-grounded settings. These numbers do not aggregate: averaging layout IoU with mask IoU has no operational meaning, since the two live on different scales and reward different shape regimes, and most papers report a different subset of them, which is the main obstacle to cross-paper comparison in this subfield.

8.1.2.4 Challenges and Trends

Spatial-aware T2I inherits the compositionality and evaluation problems of §8.1.1 and sharpens them.

Faithful multi-object composition. Crowded scenes with many small or partially occluded objects and hard relations (support, containment, depth ordering) fail in three recognizable ways: *attention dilution*, in which a prompt token's cross-attention spreads across the canvas and no object is well localized; *attribute leakage*, in which color, material, or count attaches to the wrong object (the canonical "red cube on a blue sphere" confusion); and *object dropping*, in which small or rare entities in a multi-object layout are simply omitted. On GenEval the position and counting subsets remain the weakest axes across SD 1.5, SDXL, and

SD 3 [1659]. Remedies work at three levels: Attend-and-Excite [1701] reweights cross-attention at inference so that every subject token is strongly activated somewhere in the canvas; StructureDiffusion [1702] uses prompt syntax to constrain attention structure without training; and LLM-emitted layouts, LayoutGPT [1703] and the multimodal-LLM-driven RPG of Yang et al. [1704], plan a structured layout from the free-form prompt and hand it to an L2I renderer. Whether the gap closes through a stronger compositional prior in the backbone or through planning as a separate stage is unresolved, and the two are not exclusive.

Constraint satisfaction against creativity and diversity. Turning guidance up tightens adherence until, past a model-dependent threshold, edges oversharpen, textures flatten, and variation across seeds collapses; the same curve appears for the ControlNet conditioning weight, GLIGEN's gating strength, and the classifier-free guidance scale, leaving practitioners to tune two or three knobs at once. Annealing the control weight along the denoising trajectory, per-region or per-token guidance scales, and down-weighting guidance where the base model is already confident are the current responses; how to expose such schedules to end users without overwhelming the interface is open. Chaining spatial-aware generation with repeated local edits adds a drift problem of its own, treated in §8.1.3.

Geometry-aware realism in 2D. Depth and normal conditioning fix coarse occlusion but introduce their own artifacts: *boundary banding* where a sharp depth discontinuity meets a smooth gradient in the generated image, *scale ambiguity* because a monocular depth map carries no absolute scale and two objects at the same depth value can be rendered at incompatible physical sizes, and *lighting incoherence* because the map fixes geometry but not the light field, so shadows disagree with the implied surface orientations. Better estimators are the first lever, with the diffusion-prior Marigold [636] producing sharper and more globally consistent depth than CNN baselines and Depth Anything [140] staying robust on out-of-distribution conditioning images; differentiable geometry proxies that translate depth into shading constraints are the second, explored at the boundary with full 3D synthesis (§8.2) but not yet integrated into general T2I backbones. Region grounding also changes the safety picture, since a harmful edit can be confined to a footprint that whole-image forensics miss; that concern belongs to editing and is taken up in §8.1.3.

8.1.3 2D Spatial Inpainting & Editing

2D spatial inpainting & editing modifies an image inside a specified region while leaving the rest intact: filling holes and occlusions, removing or inserting objects, changing attributes (color, material, texture, text), adjusting layout, or harmonizing pasted content with its new background. The intent arrives as a mask, a structural hint (edges, sketches), a geometry proxy (depth, normals), or a natural-language instruction, and the tensions are between **local editability** and **global consistency** of lighting, perspective, and shadow, between **constraint adherence** and **realism**, and between **semantic plausibility** and **identity preservation** for person- or product-centric edits. Editing holds the model to its strongest geometric account: the *unedited* region of I supplies ground-truth pixels that any plausible edit must respect, exposing failure modes that pure synthesis can hide, among them color drift, lighting reflow, identity loss, and attribute leakage across the mask boundary.

8.1.3.1 Task Definition

We inherit the latent-generator framework of §8.1.1. Let (E, D) denote the autoencoder pair of Eq. (8.1), $\phi : \mathcal{X} \to \mathcal{E}$ the text encoder, and G_θ the conditional latent sampler of Eq. (8.2). Inpainting and editing depart from de-novo generation in one crucial respect: the input is no longer the empty set but an existing image $I \in \mathcal{Y}$, encoded into $\mathbf{z}^{\text{in}} = E(I)$. A mask $\mathcal{M} \in \{0,1\}^{H \times W}$ designates the editable region, and an editing control set $\mathbf{c} \in \mathcal{C}$ (text prompts, structural cues, exemplar patches, or drag-style handles) populates the conditioning channel, encoded by $\kappa : \mathcal{C} \to \mathcal{C}'$. The conditional latent sampler is extended to

$$G_\theta : \mathcal{Z} \times \{0,1\}^{H \times W} \times \mathcal{E} \times \mathcal{C}' \to \mathcal{Z}, \qquad \hat{\mathbf{z}} \sim p_\theta\big(\mathbf{z} \,\big|\, \mathbf{z}^{\text{in}},\, \mathcal{M},\, \phi(\mathbf{x}),\, \kappa(\mathbf{c})\big), \tag{8.8}$$

and the edited image $\hat{I} = D(\hat{\mathbf{z}})$ is required to (i) realize the requested change *inside* $\mathcal{M}$, (ii) preserve I *outside* $\mathcal{M}$ up to imperceptible deviations, and (iii) remain globally coherent in color, lighting, geometry, and semantics. Setting $\mathbf{z}^{\text{in}} = \varnothing$ and $\mathcal{M} = \mathbf{1}$ recovers the spatial-aware synthesis of §8.1.2 (Eq. (8.7)); the distinctive substance of editing therefore lies in the interplay between the *preservation* constraint outside $\mathcal{M}$ and the *change* constraint inside it. The mask may be hard, soft, inferred from a referring expression, or implicit (whole-image instruction editing with $\mathcal{M} = \mathbf{1}$).

Variants by Task Type. The tasks differ in what populates $\mathbf{c}$ and $\mathcal{M}$ and in whether ground-truth pixels exist for evaluation. Reconstruction-style completion carries no semantic intent: hole filling repairs scratches and occlusions from $(I, \mathcal{M})$ alone, and object removal reconstructs the background behind a specified instance. Constraint-driven local synthesis adds intent inside $\mathcal{M}$: text-guided local editing follows an instruction scoped to a region ("make the *left cup* red"), structure-guided inpainting takes edges, sketches, or semantic maps as a scaffold for shape and boundary, and geometry-guided inpainting conditions on depth or normals to respect occlusion ordering. Object-level edits treat the region as an object whose identity or embedding in the scene is the unit of change: insertion with harmonization of color, grain, lighting, shadows, and reflections; attribute and appearance editing that changes material or style while preserving identity (garments, products, logos); and exemplar-guided transfer that drives content or style from a reference patch. Interactive workflows replace declarative constraints with actions taken at edit time: point- or handle-based reshaping, and multi-round, multi-region sessions in which stability across rounds is itself the requirement. The metrics appropriate to each are collected in §Benchmarks.

8.1.3.2 Methodology

Methods are organized by *editing mechanism*, that is, by how G_θ of Eq. (8.8) is realized so that change is confined to $\mathcal{M}$ and preservation is enforced outside it. The mechanism is largely independent of the task type above, since text-guided local editing can be served by masked denoising, by inversion and rewrite, or by an instruction-tuned editor, and one mechanism such as inversion serves both attribute edits and structure-guided ones. Figure 8.3 sets the five mechanisms against a single example edit.

Editing Mechanisms.

Masked latent denoising. The most direct realization of Eq. (8.8) runs the diffusion sampler only inside $\mathcal{M}$ while clamping the latent outside $\mathcal{M}$ to a noise-perturbed copy of $\mathbf{z}^{\text{in}}$ at every step. RePaint [1705] does this with a *pretrained, unconditional* DDPM and no retraining, replacing the known-region pixels with a forward-noised sample of the input at each reverse step and adding a *resampling* schedule that re-injects noise periodically so the in-fill can re-harmonize with its context over several rounds. Blended Diffusion [1706] made the same scheme text-driven by interleaving a CLIP-guided gradient step that steers the content inside $\mathcal{M}$ toward a target prompt with the context-clamping step outside it, the first widely cited text-guided local editor. LaMa [1707] predates the diffusion line and is instead a feed-forward inpainter whose fast Fourier convolutions give every layer an image-wide receptive field, a Fourier-domain inductive bias that stays robust on the very large holes that defeat purely local CNN backbones. Region-aware noise schedules and classifier-free guidance are standard add-ons, and ControlNet-inpaint and the inpainting variants of T2I-Adapter (§8.1.2.2) compose with masked denoising by supplying the inside-$\mathcal{M}$ condition.

Inversion + rewrite. A complementary strategy first recovers the noise trajectory that the model would have used to generate I (*inversion*) and then re-runs the reverse process under an edited prompt while reusing or swapping the attention maps of the original (*rewrite*). Prompt-to-Prompt [1708] reuses or reweights source cross-attention maps during generation under the edited prompt, which preserves spatial structure while changing selected semantic content; for a real image an inversion step first supplies a suitable noise trajectory. Imagic [1709] jointly optimizes the text embedding and lightly fine-tunes the model to reconstruct I, then interpolates the embedding toward the target prompt, preserving identity-level features (faces, garments) under non-rigid edits. Null-Text Inversion [1710] fixes the practical failure that plain DDIM inversion drifts

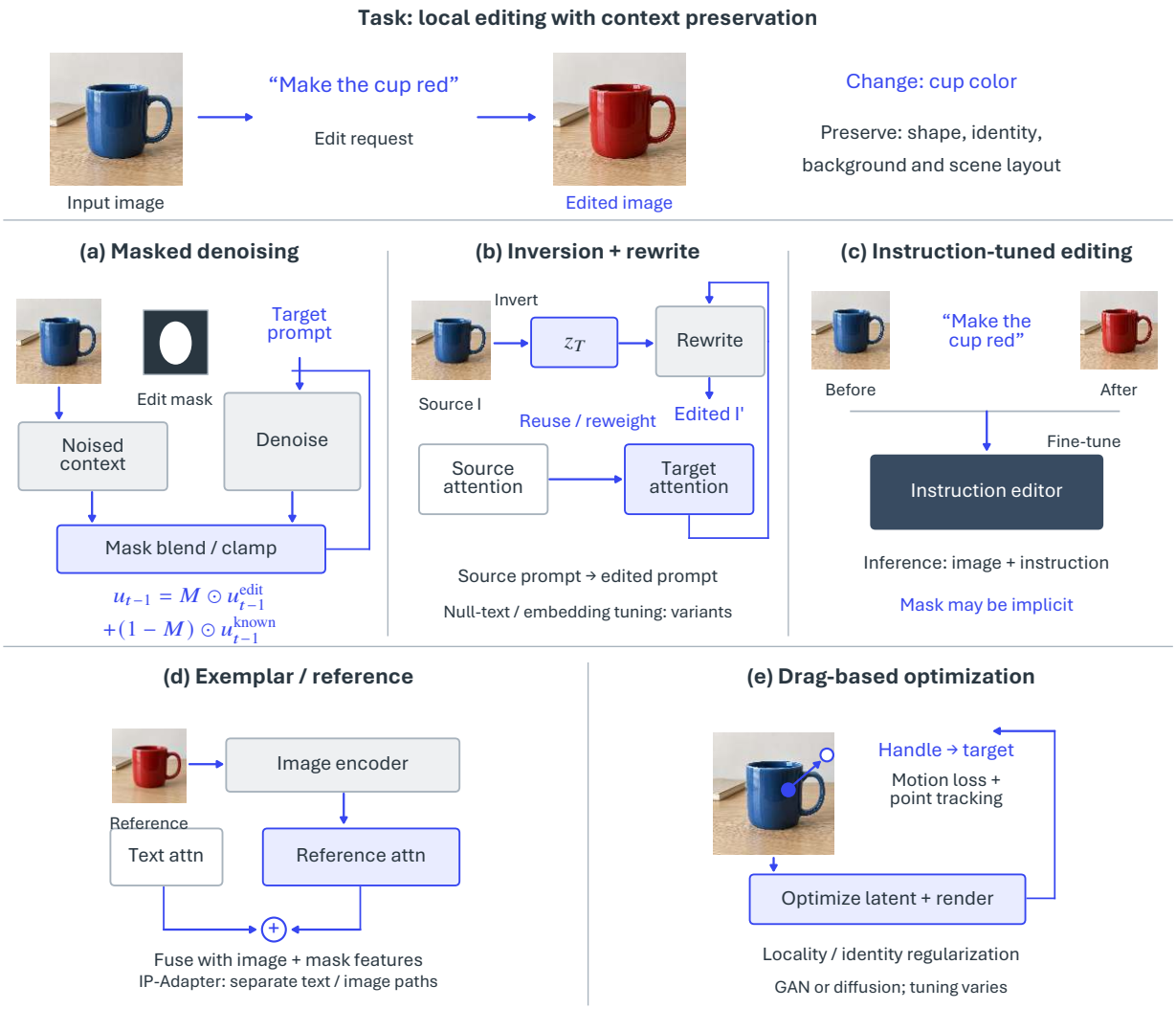


Figure 8.3: Local 2D editing with context preservation (top) and five editing mechanisms: (a) masked denoising, where $M = 1$ marks the editable region; (b) inversion and rewrite; (c) instruction-tuned editing; (d) exemplar / reference-guided editing; and (e) drag-based optimization. The cup images are illustrative, not outputs of the surveyed methods.

away from I under classifier-free guidance, by optimizing the *null-text* embeddings used in the inverse pass so that exact reconstruction holds at ordinary guidance scales. The family needs no full backbone fine-tune for Prompt-to-Prompt or Null-Text and dominates attribute, style, and structure-preserving real-image edits.

Instruction-tuned editing. A third family avoids inversion entirely by retraining a diffusion editor on paired (before, instruction, after) triplets, mapping I directly to $\hat{I}$ from a natural-language instruction with no explicit mask. InstructPix2Pix [1711] built its corpus by pairing GPT-3-generated edit instructions with Prompt-to-Prompt edits of Stable Diffusion samples and fine-tuned Stable Diffusion on the triplets, so that at inference the user supplies only $(I, \text{instruction})$ and the editor localizes the change itself. MagicBrush [1712] replaced the synthetic data with about 10K manually annotated real-image edits across multi-turn sessions, yielding both a stronger supervised editor and the standard instruction-editing test set. Emu Edit [1713] trained a single model on a heterogeneous mixture of *recognition* (segmentation, detection) and *generation* (insertion, removal, style change) tasks with learned task embeddings that route each input, and leads its own

benchmark on precision and instruction following. Instruction editors are the most user-friendly mechanism, no mask and no inversion, at the cost of large supervised retraining.

Exemplar / reference-guided editing. A visual *exemplar*, an example image or patch encoding the desired content or style, replaces the textual instruction and enters through an auxiliary cross-attention pathway beside the standard text conditioning. Paint-by-Example [1714] fine-tunes Stable Diffusion to fill $\mathcal{M}$ with the content of an exemplar whose identity is encoded by a CLIP image encoder, with strong exemplar augmentation (cropping, color jitter) at training time to prevent copy-paste behavior and force semantic rather than pixel-level transfer. IP-Adapter [1715] generalizes this into a *plug-in* reference channel: a lightweight decoupled cross-attention head consumes a CLIP image embedding of the exemplar and is inserted into a frozen SD or SDXL backbone, so that any number of reference images can be combined with text prompts and with structural adapters such as ControlNet at inference time. It is the standard route for subject personalization and style transfer when the desired content is hard to describe in words.

Drag-based latent optimization. The most interactive family takes user-supplied (source point, target point) pairs and iteratively optimizes the latent so that the features at each source point migrate to the corresponding target while a locality regularizer pins the latent outside the user-marked region. DragGAN [1716] introduced the recipe on a StyleGAN backbone: a motion-supervision loss tracks the source feature and pulls it toward the target by gradient descent in the $\mathcal{W}^+$ latent space, and a point-tracking step re-localizes the moved feature at each iteration to prevent drift. DragDiffusion [1717] ported the recipe to the LDM latent space, running a per-timestep latent-update loop together with an identity-preserving LoRA fine-tune of the U-Net so that both the dragged subject and the unmarked context survive. Both are inference-time optimizers with latency proportional to the number of drag iterations, and no other family yet offers the same directness of spatial control.

Read across the five families, the mechanisms track the generative substrate: feed-forward inpainters such as LaMa gave way to inversion-based diffusion editors (Prompt-to-Prompt, Imagic), which in turn yielded to supervised instruction-tuned editors (InstructPix2Pix, Emu Edit) as paired data became available at scale. Orthogonal to that migration, the structural conditioning interfaces of §8.1.2 compose with every mechanism here, so a production pipeline pairs a mechanism with a structural channel rather than choosing one: Blended Diffusion with ControlNet-depth for geometry-aware text-guided edits, or IP-Adapter with DragDiffusion for exemplar-driven drag edits.

8.1.3.3 Benchmarks

Editing is evaluated in two regimes: *reference-based*, where a held-out ground-truth region exists (holes synthesized over real images), and *reference-free*, where only constraint satisfaction and human preference are available (instruction editing). No single suite covers both, so the field keeps a small library of task-specific benchmarks next to the holistic T2I suites of §8.1.1.

Training Corpora. Inpainting draws on Places2 [609] and Paris StreetView for scene-level masks, DIV2K for high-resolution patches, and the COCO / ADE20K mask supervision of §8.1.2 [440, 492]. Constraint-driven local synthesis reuses the structural channels of the previous subsection, RefCOCO / RefCOCO+ / RefCOCOg [1082] and PhraseCut [1092] for referring-expression masks, SA-1B masks [138], and COCO edge and depth predictions, with the convention that the mask is supplied or derived from the expression; geometry-guided editing adds depth and normal supervision from NYUv2 [504], Replica [339], and synthetic corpora, and insertion with harmonization uses composite datasets such as iHarmony4. Instruction editing has its own data ecology: the synthetic InstructPix2Pix corpus of Prompt-to-Prompt-generated triplets, the manually annotated MagicBrush triplets [1712], and the multi-task Emu Edit training mixture [1713].

Evaluation Benchmark Suites. The holistic suites of §8.1.1 (Table 8.1) carry over in part: T2I-CompBench [1676] and GenAI-Bench [1679] score the attribute and compositional fidelity that edited

images must also satisfy, and HEIM [1673] includes alignment and quality axes that apply to edits. The dedicated suites are EditBench [1718], released with Imagen Editor, a curated set of text-guided inpainting prompts over object, attribute, and scene categories that serves as the standard evaluation for text-driven mask inpainting; the MagicBrush test set [1712], $1{,}053$ manually annotated single- and multi-turn triplets, which is the standard for instruction-tuned editors; and the Emu Edit benchmark [1713], which extends instruction editing to per-task subsets (insertion, removal, style change, color change, global change). Drag-based methods evaluate on small custom test sets curated by Pan et al. [1716] and Shi et al. [1717]; a community-wide drag-editing benchmark has not yet emerged.

Constraint-specific Metrics. Reconstruction quality (PSNR / SSIM / LPIPS against ground truth, with FID / KID for diverse-completion settings) is defined only where held-out content exists. Region adherence uses mask-restricted CLIPScore or region-level VQAScore to ask whether the edit honors the prompt *inside* $\mathcal{M}$; edit locality measures ℓ_1 or LPIPS *outside* $\mathcal{M}$ against I, lower being better; harmonization is scored by color-statistic divergence, shadow-direction consistency, and the rate at which an off-the-shelf forgery detector flags the composite; identity preservation uses feature-cosine similarity over face or landmark regions for people and CLIP-image similarity for products and characters; and instruction following is paired accuracy, whether $\hat{I}$ honors the instruction better than I, judged by human raters or by a VLM over a structured rubric, as is now standard on MagicBrush and Emu Edit. Change inside $\mathcal{M}$ and preservation outside it are reported together; a method that wins on one and loses on the other has not solved the task.

8.1.3.4 Challenges and Trends

Several problems are specific to editing an existing image.

Structure-faithful hallucination in large holes. Large irregular holes and rare structures remain the hardest case: inpainters tile a salient local texture across the hole instead of synthesizing appropriate content, straight lines, perspective cues, and silhouettes break at the mask seam, and in-fills that are locally plausible contradict the global scene (a patch of grass inside a room). Quality degrades sharply with hole size. LaMa's image-wide Fourier convolutions and RePaint's resampling schedule (§8.1.3.2) are the established responses; retrieval-augmented priors, which search a paired corpus for candidate completions and use them as soft conditioning, are an emerging one at the intersection of editing and retrieval.

Edit locality against drift across rounds. A single well-calibrated edit does not guarantee a stable session. Across rounds the unedited background shifts in white balance or saturation (*color drift*), shadows and highlights migrate as if the edit had changed the global illumination (*lighting reflow*), and salient textures from the edited region reappear in nearby unedited regions through attention leakage (*texture replication*); the errors compound even when each round individually meets its locality budget. Hard attention masking that confines non-edit tokens to $\mathcal{M}$, mask-aware noise schedules that anneal noise inside $\mathcal{M}$ only, and edit-history-aware sampling that conditions on the trajectory of prior edits are the active mitigations. Inversion-based editors (Null-Text Inversion [1710], Imagic [1709]) carry a variant of the problem: any residual reconstruction error in the inverted trajectory leaks into the edit and accumulates.

Harmonization under complex illumination. Inserted content betrays itself when the host scene has structured light: its shadow disagrees with the dominant light direction recoverable from the scene, there is no interreflection between the object and adjacent surfaces, and its albedo is tone-mapped inconsistently with the host, producing the cut-and-paste look that off-the-shelf forgery detectors flag. Differentiable rendering proxies that turn depth and normals into shading constraints and back-propagate harmonization losses into the editor, illumination maps as an additional conditioning channel, and explicit shadow-synthesis modules are being explored; integration with the 3D-aware methods of §8.2 is the natural next step and remains nascent.

Language grounding and safety. Region-level instructions must be grounded reliably: an edit aimed at one object spills onto neighbors with similar attributes, spatial referring expressions are misparsed ("the left

cup" lands on the right one), and mildly perturbed instructions produce large deviations in the edit trajectory. Borrowing GLIGEN's gated grounding [1697] (§8.1.2.2) to confine edit attention, and using LLM-emitted layouts to harden the grounding before the edit is applied, are the current responses. The same precision cuts the other way for safety. A harmful edit (an identity swap inside one bounding box, watermark removal inside one mask) can be confined to a footprint that whole-image forensic detectors miss; large generative models reproduce individual training images under targeted prompting [1719], which becomes acute when the training set contains copyrighted artwork or identifiable faces and motivates artist-side defenses such as Glaze [1720], which perturbs source artworks so that downstream fine-tunes mimic a different style; and most pipelines still lack robust per-image watermarking, so a photograph with a small synthesized region cannot be told from a wholly synthetic one. Watermarking, region-aware safety filters, and licensing frameworks for the underlying corpora are all needed, and none is settled.

Evaluation under plausibility ambiguity. Ground-truth pixels are often not the only plausible completion: a removed pedestrian leaves a hole that could contain road, curb, or shadow, and any reconstruction metric that compares against one ground truth over-penalizes good alternatives, while PSNR and SSIM rate sharp but contextually wrong in-fills highly. The field is converging on *paired evaluation*, asking whether $\hat{I}$ honors the instruction better than I under a held-out judge, now standard on MagicBrush [1712] and Emu Edit [1713]; the judge-prompt sensitivity and benchmark-contamination caveats of §8.1.1 apply here as well.

Scalability and interactive latency. Inversion-based methods store cross-attention activations at every timestep, multi-step sampling at 4K and above scales with image area for standard U-Net backbones, and drag editors run an inner optimization loop per operation, so interactive editing lags behind the user's hand by hundreds of milliseconds. Consistency-style distillation [1661, 1662] (§8.1.1) compresses the reverse trajectory to a handful of steps, memory-efficient adaptive patching runs the denoiser only in a window around $\mathcal{M}$, and progressive-refinement schedules return a coarse preview within a perceptual-latency budget and refine in the background. Closing the gap to the responsiveness of a photo editor is increasingly a systems problem shared with production T2I inference stacks.

8.2 3D Spatial Reconstruction and Generation



The 3D spatial reconstruction and generation encompasses the tasks that turn images, video, sensor readings, or high-level prompts into an explicit 3D representation (a textured mesh, a signed-distance or occupancy field, a neural radiance field, or a 3D Gaussian field) of an object, a scene, or a complete navigable environment. Whereas §8.1 taught a 2D semantic-to-pixel before honor spatial structure on the image plane, this section lifts that prior off the plane and into the volumetric substrate on which embodied perception, simulation, and interaction ultimately operate. This is where appearance and geometry are forced into a single, view-consistent representation: a model must not only produce plausible pixels but place matter correctly in $\mathbb{R}^3$ so that the result can be rendered from arbitrary viewpoints, physically simulated, navigated, and edited. It is the bridge between the flat generative priors of §8.1 and the embodied world models, simulators, and digital twins that consume 3D structure downstream.

The five subsections are organized by the direction and source of the map into a shared 3D-representation space $\mathcal{R}$. §8.2.1 starts with a single asset: 3D object synthesis turns the 2D prior of §8.1 into a volumetric generator through 3D-native training, score distillation, multi-view lifting, or feed-forward reconstruction. §8.2.2 scales that generator to a structured scene of placed objects and a room envelope. §8.2.3 reverses the direction: reconstruction recovers a representation faithful to sensor observations, and meets synthesis exactly where the observations run out. §8.2.4 routes through a 360-degree panorama as a structured 2D intermediate that is then lifted to 3D, and §8.2.5 closes the loop by editing an existing representation under user control.

8.2.1 3D Object Synthesis

3D object synthesis generates a complete, usable 3D asset (geometry and, optionally, appearance) from *text*, *a single image*, or *a sparse set of views*. Where reconstruction recovers an object consistent with given observations, synthesis additionally asks for plausible completion of what was never observed, for generative diversity, and for usability in downstream graphics and embodied pipelines (*e.g.*, game engines, simulators, manipulation); the output is a textured mesh, a radiance field, or a set of 3D Gaussians depending on which of fidelity, editability, rendering speed, and toolchain compatibility matters most. It is also the point at which the *semantic-to-pixel* prior of §8.1 (the conditional latent sampler G_θ of Eq. (8.2)) becomes a *semantic-to-3D* prior G_θ^{3D}: 3D-native generators train on 3D corpora directly, score-distillation pipelines invert the frozen 2D prior under rendering supervision, multi-view pipelines fine-tune it to emit consistent views and lift them, and feed-forward reconstructors and 3D-native latent diffusion exploit the recent object-scale 3D corpora to learn one-shot text- or image-to-3D maps. Figure 8.4 illustrates the text- or single-image-conditioned setting.

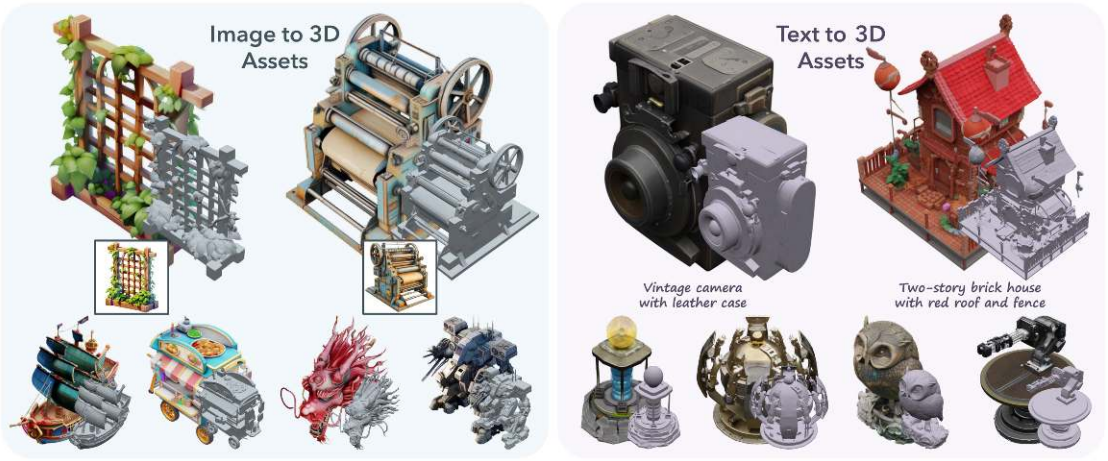


Figure 8.4: Illustration of 3D object generation conditioned on text or a single image (adapted from [1721]).

8.2.1.1 Task Definition

We cast 3D object synthesis as a paradigm-agnostic extension of the latent-generator framework of §8.1.1. Let $\mathcal{R}$ denote a 3D-representation space taken from $\{$ *mesh*, *SDF*, *voxel*, *triplane*, *NeRF*, *3DGS*, *structured latent* $\}$, and let $\mathcal{C}$ denote the conditioning-input space comprising a text prompt $\mathbf{x} \in \mathcal{X}$ embedded via ϕ (as in Eq. (8.2)), a single image $I \in \mathcal{Y}$, or a multi-view set $\{(I_v, \pi_v)\}_{v=1}^{V}$ pairing images with camera poses π_v. A condition encoder $\kappa : \mathcal{C} \to \mathcal{C}'$ maps each input into a shared conditioning embedding. The core learning problem is to train a conditional 3D generator

$$G_\theta^{\text{3D}} : \mathcal{C}' \to \mathcal{R}, \qquad \hat{\mathbf{r}} \sim p_\theta^{\text{3D}}\big(\mathbf{r} \,\big|\, \kappa(\mathbf{c})\big), \tag{8.9}$$

so that the sampled asset $\hat{\mathbf{r}} \in \mathcal{R}$ admits high-fidelity geometry, plausibly completes unobserved regions, and renders consistently across viewpoints. Two limiting cases locate Eq. (8.9) in the literature. When $\kappa(\mathbf{c}) = \varnothing$ the sampler degenerates to *unconditional* 3D generation, the regime of early GAN- and diffusion-based 3D generators trained on category-specific corpora. At the other extreme, when $\mathbf{c}$ is text-only and G_θ^{3D} is realized by distilling the 2D sampler G_θ of Eq. (8.2) (via score-distillation sampling, multi-view consistency objectives, or volume-rendering supervision), one recovers the DreamFusion-style text-to-3D pipeline that has dominated open-vocabulary 3D synthesis since 2022.

8.2.1.2 Methodology

The four paradigms below are four ways of parameterizing and optimizing the sampler G_θ^{3D} of Eq. (8.9), taken in the order in which each became prominent. Figure 8.5 contrasts them, making explicit which components are trained versus frozen and where the 3D representation $\mathcal{R}$ is instantiated.

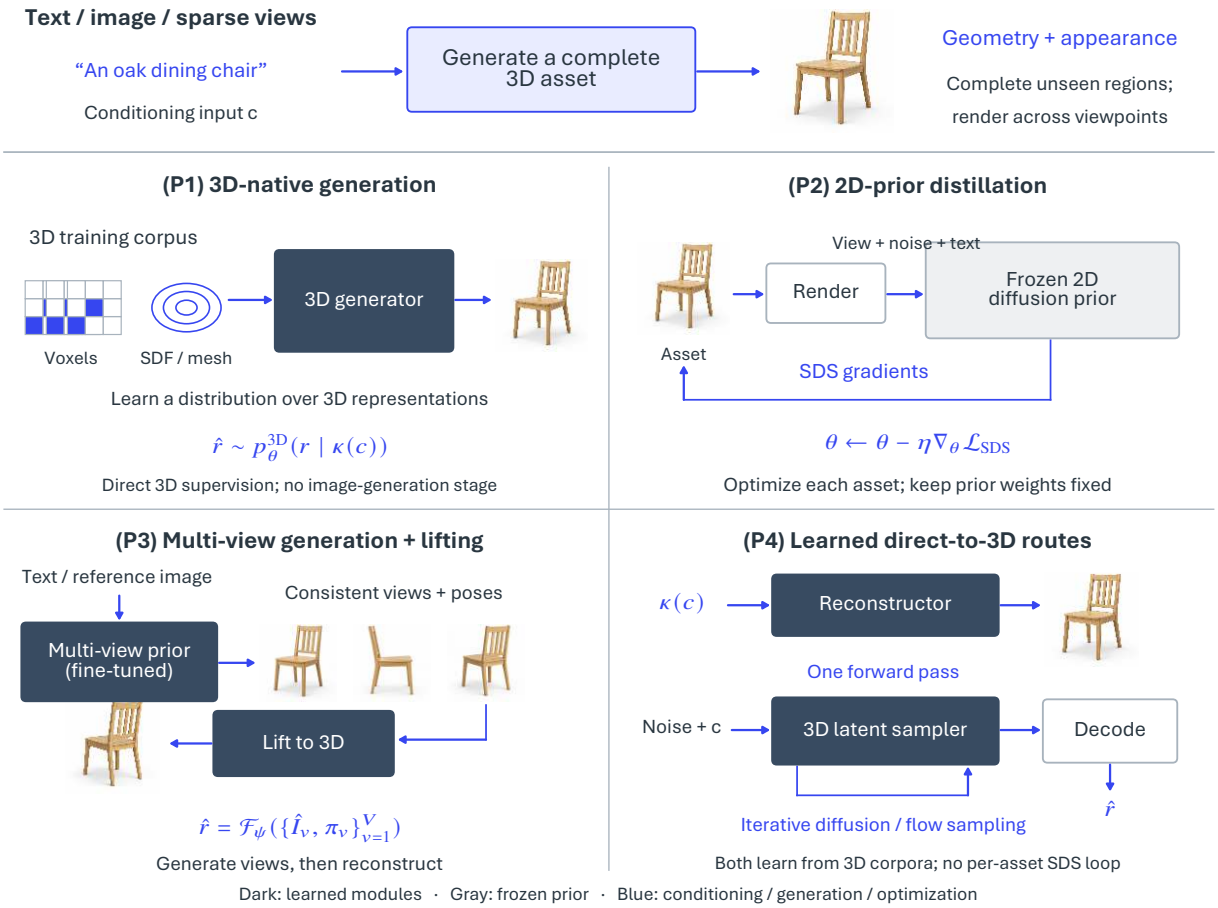


Figure 8.5: Four paradigm families for 3D object synthesis: (P1) 3D-native generation; (P2) distillation from a frozen 2D diffusion prior (SDS); (P3) multi-view generation with a fine-tuned prior, then lifting to 3D; and (P4) learned direct-to-3D routes, by one-pass reconstruction or iterative 3D latent sampling. The families can overlap, notably P1 and the latent route of P4; chair images are illustrative.

3D-Native Generators. The earliest 3D generators instantiate G_θ^{3D} as a sampler trained *directly* on a 3D representation, by denoising score matching on voxels, SDFs, or triplanes or, in earlier work, by an adversarial fit to a category-specific 3D corpus $p_{\text{data}}^{3D}(\mathbf{r})$,

$$\mathcal{L}_{\text{native}}(\theta) = \mathbb{E}_{\mathbf{r}_0 \sim p_{\text{data}}^{3D},\, t,\, \epsilon}\left[\left\|\epsilon - \epsilon_\theta^{3D}(\mathbf{r}_t,\, t,\, \kappa(\mathbf{c}))\right\|^2\right], \quad \mathbf{r}_t = \alpha_t \mathbf{r}_0 + \sigma_t\, \epsilon, \tag{8.10}$$

where the noise predictor ϵ_θ^{3D} operates directly on $\mathbf{r} \in \mathcal{R}$ with no 2D intermediary, so the supervision signal scales with the size of the 3D corpus rather than with 2D image–text pairs. Because 3D data is heterogeneous, these generators are themselves diverse, having been instantiated on signed distance fields [1722], voxel grids [1723, 1724, 1725], point clouds [1726, 1727], view sets [1728, 1729], neural radiance fields [278, 1730, 1731, 1732, 1733], triplanes [1734, 1735, 1736], and 3D Gaussian Splatting [1737, 1738, 1739]. The scarcity and monotony of 3D corpora remain the bottleneck: even with sophisticated architectures, generated geometry and texture lack fidelity and stay confined to a handful of canonical categories, which

is what pushed the field, around 2022–2023, toward paradigms that use the much larger *2D* corpora as an indirect supervision signal for 3D structure.

2D-Prior Distillation (SDS Family). The second paradigm sidesteps the 3D-data bottleneck by realizing G_θ^{3D} as a *per-shape optimization* that distills gradients from a pretrained 2D generator (a CLIP model [143] or a 2D diffusion model [185]) trained on web-scale image data [1670]. Given the *frozen* 2D noise predictor ϵ_ϕ of Eq. (8.2) and a differentiable renderer $\text{Render}(\mathbf{r}_\theta, \pi)$ that images a 3D representation $\mathbf{r}_\theta \in \mathcal{R}$ under camera pose π, score distillation updates only the *per-shape* 3D parameters θ via the gradient

$$\nabla_\theta \mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\,\epsilon,\,\pi}\Big[\, w(t)\,\big(\epsilon_\phi(\mathbf{y}_t^\pi,\, t,\, \phi(\mathbf{x})) - \epsilon\big)\,\tfrac{\partial \mathbf{y}^\pi}{\partial \theta}\,\Big], \quad \mathbf{y}^\pi = \text{Render}(\mathbf{r}_\theta, \pi), \tag{8.11}$$

where $w(t)$ is a step-dependent weighting and $\mathbf{y}_t^\pi$ is the rendered view perturbed by the diffusion forward process. The contrast with Eq. (8.10) is sharp: the 2D prior of §8.1 is held *fixed* and no 3D corpus is consumed, while the trainable parameters live in the representation of a *single* asset, hence the per-shape optimization cost paid at inference. A line of work [1740, 1741, 1742, 1743] applies score-distillation sampling (SDS) or its variants to a NeRF or mesh; DreamFusion [1741] obtains zero-shot text-to-3D by optimizing a NeRF whose multi-view renderings the 2D diffusion prior scores, and GaussianDreamer [1744] moves the same recipe onto a 3DGS representation. These methods need no 3D training data, but because the optimization is rerun for every prompt they are slow, prone to over-saturated color, and subject to the "multi-face" (Janus) artifact in which the 2D prior fails to disambiguate the front and back of an object; variational score distillation [1745] mitigates the over-saturation but not the per-prompt cost.

Multi-View Generation and Lifting. The third paradigm realizes G_θ^{3D} as a two-stage pipeline: a fine-tuned 2D diffusion model generates a small set of geometrically consistent views, and a separate reconstructor lifts them to a 3D asset, so that Eq. (8.9) factorizes as

$$\hat{\mathbf{r}} = \mathcal{F}_\psi\Big(\big\{\hat{\mathbf{y}}_v,\, \pi_v\big\}_{v=1}^{V}\Big), \qquad \hat{\mathbf{y}}_v \sim p_\theta\big(\mathbf{y}_v \,\big|\, \mathbf{y}_{\text{ref}},\, \pi_v,\, \phi(\mathbf{x})\big), \tag{8.12}$$

where the view-conditional 2D sampler $p_\theta(\cdot \mid \cdot)$ is a fine-tuned variant of Eq. (8.2) that additionally consumes a reference image or text $\mathbf{y}_{\text{ref}}$ and a target camera pose π_v, and $\mathcal{F}_\psi$ is a (typically feed-forward) multi-view-to-3D reconstructor. Two differences from Eq. (8.11) follow: the 2D prior is *fine-tuned* rather than frozen, and the 3D representation appears only inside the deterministic lift $\mathcal{F}_\psi$, with no per-shape inner loop at inference, which is what lets this paradigm run one to two orders of magnitude faster than SDS at the price of depending on the view-consistency of the synthesized $\hat{\mathbf{y}}_v$. The first step was to make a pretrained diffusion model pose-aware. Following fine-tuning recipes such as ControlNet [1693] and LoRA [1746], several works [1747, 1748] fine-tune for cross-view consistency; Zero-1-to-3 [1749] opened the open-world single-image setting by training a diffusion model to synthesize a novel view conditioned on an input view and a relative camera pose; MVDream [1750] and ImageDream [1751] fix a world-camera coordinate convention from which geometry can be recovered; Magic123 [1752] combines the Zero-1-to-3 prior with the 2D prior of Stable Diffusion; One-2-3-45 [1753] lifts Zero-1-to-3 views to a mesh; and Consistent123 [1754] and MVD-Fusion [1755] add boundary and depth priors. Attention then turned to consistency *across* the generated views: SyncDreamer [1756] correlates features across views with a 3D-aware attention, CAT3D and Free3D [1757, 1758] model the multi-view distribution jointly, and Zero123++ [1759] tiles six surrounding views into a single grid image, a trick inherited by One-2-3-45++ [1760], Instant3D [1761], InstantMesh [1762], and AR-1-to-3 [1763].

Feed-Forward Reconstruction and 3D-Native Latent Diffusion. The fourth group amortizes generation across object-scale 3D corpora rather than optimizing each asset against a frozen 2D prior, in two instantiations. A *feed-forward reconstructor* trains G_θ^{3D} end-to-end to map the conditioning embedding $\kappa(\mathbf{c})$ directly to a 3D representation $\hat{\mathbf{r}} \in \mathcal{R}$, supervised by differentiable rendering against V ground-truth views $\{\mathbf{y}_v, \pi_v\}$ drawn

from the corpus,

$$\mathcal{L}_{\text{LRM}}(\theta) = \mathbb{E}_{(\mathbf{c},\,\{\mathbf{y}_v,\pi_v\}_v)\sim p^{\text{3D}}_{\text{data}}} \sum_{v=1}^{V} \left\| \text{Render}\big(G^{\text{3D}}_\theta(\kappa(\mathbf{c})),\,\pi_v\big) - \mathbf{y}_v \right\|^2. \tag{8.13}$$

A *3D-native latent diffusion* model instead learns a 3D autoencoder pair $(E^{\text{3D}}, D^{\text{3D}})$ with $\mathbf{z}^{\text{3D}} = E^{\text{3D}}(\mathbf{r})$, trains a denoising model on the compact latent, and decodes via $\hat{\mathbf{r}} = D^{\text{3D}}(\hat{\mathbf{z}}^{\text{3D}})$,

$$\mathcal{L}_{\text{3DL}}(\theta) = \mathbb{E}_{\mathbf{z}^{\text{3D}}_0,\,t,\,\epsilon}\Big[\big\| \epsilon - \epsilon_\theta\big(\mathbf{z}^{\text{3D}}_t,\,t,\,\kappa(\mathbf{c})\big) \big\|^2 \Big], \tag{8.14}$$

which is the 2D LDM objective Eq. (8.5) of §8.1.1 with the autoencoder moved from pixels to $\mathcal{R}$. Both variants avoid the per-shape optimization of Eq. (8.11), but only the reconstructor uses a single prediction pass; latent diffusion still samples iteratively. The unlocking factor was the arrival of Objaverse and Objaverse-XL [1764, 1765]. LRM [1766] and its follow-ups [1767, 1768, 1769] show that large transformers trained purely with rendering losses capture object geometry and texture in detail and generalize to unseen images, and a parallel line introduces compact *3D-native latent codes* [1770, 1771, 1772, 1773] whose distribution a latent diffusion model can learn; TRELLIS [1721], CLAY [1774], HiGS-RGBN [1775], and Hunyuan3D [1776] build on this to bring generated assets close to production grade while keeping the open-vocabulary reach that the SDS family pioneered.

Two design choices cut across the four paradigms. The representation $\mathcal{R}$ (*mesh*, *SDF*, *voxel*, *triplane*, *NeRF*, *3DGS*, or *structured latent*) is largely independent of the paradigm: SDS has been run on NeRFs and on 3DGS, feed-forward reconstructors output triplanes, meshes, or Gaussians, and latent-diffusion stacks decode to almost any explicit representation through a learned VAE. So is the conditioning modality: the same backbone takes text, a single image, or sparse views by changing only the encoder κ and the camera-pose embedding, and production systems chain them (image-to-3D conditioned on a text-to-image preview). The succession of paradigms itself tracks the accumulation of supervision at each level, from 3D-native corpora to 2D image–text pairs to Zero-1-to-3-style multi-view priors to Objaverse-XL-scale 3D, with the most recent industrial systems sitting in the fourth paradigm while inheriting the multi-view-consistency tricks of the third.

8.2.1.3 Benchmarks

Evaluation draws on object-scale training corpora, on dedicated benchmark suites that curate prompts and human ratings, and on metrics that probe alignment, geometry, and appearance separately.



Figure 8.6: Demonstration of the Objaverse [1765] dataset.

Table 8.2: Representative training corpora for 3D object synthesis. "Scale" gives order-of-magnitude object counts as reported in each release.

Dataset	Year	Scale (#objects)	Acquisition	Primary annotation / modality
ShapeNet [1777]	2015	∼51K	Curated CAD	Semantic category, part labels
Google Scanned Objects [1778]	2022	∼1K	Real-world scans	High-fidelity geometry + PBR textures
OmniObject3D [1779]	2023	∼6K	Real-world scans	Dense multi-view + reconstructions
Objaverse [1764]	2023	∼800K	Web-scraped	Open-vocabulary text tags
Objaverse-XL [1765]	2023	∼10M	Web-scraped at scale	Open-vocabulary text tags
G-Objaverse [1780]	2024	∼280K	Synthesized + curated	Text-driven assets, rendered views

Training Corpora. Object-scale 3D corpora drive the feed-forward and latent-diffusion paradigms; Table 8.2 lists the ones most commonly used, and Figure 8.6 shows representative Objaverse-XL assets. They span the classical CAD repository ShapeNet [1777], category-bounded real-world scans with accurate geometry and PBR textures (Google Scanned Objects [1778], OmniObject3D [1779] with dense multi-view imagery), the open-vocabulary web-scraped Objaverse and Objaverse-XL [1764, 1765] that unlocked the fourth paradigm, and G-Objaverse [1780], which adds text-driven synthesized assets between the two.

Evaluation Benchmark Suites. Recent benchmarks have moved to *multi-dimensional* assessment beyond plain text–image alignment. GT23D-Bench [1781] curates a high-fidelity dataset and a suite of 3D-aware metrics spanning alignment and visual quality; MATE-3D [1782] provides designed prompt categories with large-scale human annotations across several perceptual dimensions and introduces HyperScore as a learned evaluator; T23D-QA [1783] adopts a question-answer protocol that scores geometry, texture, and alignment independently; and Gen3DEval [1784] fine-tunes a vision-language model for 3D quality assessment, scoring text fidelity, appearance, and surface quality from rendered surface normals. Together with the 3D-relevant subsets of the 2D suites collected in §8.1.1 (Table 8.1), they form the de facto evaluation protocol for 3D object synthesis.

Constraint-specific Metrics. Because an asset is judged from many viewpoints and under different downstream uses, evaluation reports complementary metrics along three axes. **Text-3D alignment** renders multi-view images from the asset and measures text–image similarity with CLIP-like encoders, aggregated across views (mean, worst-case, or view-consistency variants); pure CLIP alignment can be misled by texture "shortcuts" and does not guarantee geometric correctness, which motivates the 3D-aware alignment scores of GT23D-Bench [1781] and learned evaluators such as HyperScore [1782]. **Geometry quality** is measured against ground-truth shapes, where they exist, by Chamfer Distance (CD) [1785], F-score at varying thresholds, normal consistency, and volumetric IoU for watertight meshes; for text-to-3D without ground truth it falls back on multi-view depth and normal consistency, silhouette consistency, and learned 3D-quality predictors. **Appearance and multi-view consistency** are measured by perceptual distances (LPIPS) between rendered views and references where available, by view-consistency measures (variance across views under fixed lighting and camera trajectories), and by human studies, which recent benchmarks collect as multi-dimensional ratings to calibrate the automatic metrics [1782, 1783].

The paradigm succession of §8.2.1.2 shows up in two measurable quantities. Generation time has fallen by about three orders of magnitude: DreamFusion [1741] needs roughly 1.5 hours per asset and Magic3D [1742] about 40 minutes, whereas LRM [1766] reconstructs a shape in about 5 seconds and InstantMesh [1762] completes image-to-3D in around 10 seconds. Image-to-3D geometry is now routinely scored on Google Scanned Objects [1778] with Chamfer Distance (lower is better) and F-score at τ=0.2 (higher is better), where InstantMesh [1762] reaches CD ≈ 0.18 and FS ≈ 0.88, ahead of TripoSR [1786] (0.217 / 0.843), CRM [1768] (0.252 / 0.787), and LGM [1767] (0.345 / 0.671). Coarse text-to-3D alignment, by contrast, is saturated: CLIP-style scores no longer separate the strongest systems, so Hunyuan3D 2.0 [1787] reports ULIP- and Uni3D-based image- and text-to-shape similarity backed by a study in which 50 participants rated 300

unselected results, and learned evaluators (HyperScore on MATE-3D [1782], T23D-QA [1783]) score geometry and texture on separate axes. The reconstruction metrics carry their own caveat: CD and F-score saturate on coarse geometry and can reward visually plausible but physically thin assets, so reported gains are credible mainly where geometric, perceptual, and human ratings agree. Commercial text- and image-to-3D services now ship these capabilities with PBR export to game engines, alongside research systems such as Meta 3D Gen [1788] and the open-vocabulary 3D-native-latent systems CLAY [1774], TRELLIS [1721], and Hunyuan3D [1776, 1787].

8.2.1.4 Challenges and Trends

Four problems remain open, and they depend on one another: the multi-view consistency of the asset, the representation it lands in, whether progress on either can be measured reliably, and the data that ultimately drives all three.

Multi-View and 3D Consistency under Open-Vocabulary Prompts. Enforcing consistent geometry and texture across viewpoints while preserving fine detail remains difficult in all four paradigms, and the failures have names. The *Janus artifact* renders both the front and the back of an object as a frontal face, most acutely in SDS pipelines, where the 2D prior cannot disambiguate camera direction; *view drift* makes adjacent views of a multi-view stack disagree on geometry, a residual concern in Zero-1-to-3-style cascades that SyncDreamer's cross-view attention and Zero123++'s tiled six-view trick only partly remove; and *texture seams* appear where neighboring views' color and lighting cannot be merged into a globally coherent UV map, a persistent weakness of feed-forward reconstructors. These failures motivate 3D-native backbones with *structured latents* [1721] that share parameters across views by construction, and benchmarks that score multi-view consistency rather than a single rendered view [1782].

Representation Trade-offs between Fidelity, Editability, and Deployability. Even a consistent asset must *land somewhere* a downstream tool can consume, and the choice of representation is not neutral. Meshes with UV/PBR textures are what game engines and simulators want but lose detail to discretization; radiance fields and Gaussians give high visual fidelity and fast rendering but require conversion, which introduces its own failures: *topology breakage* during marching-cubes-style meshing, *baked illumination* that fuses lighting into albedo when converting from a NeRF or 3DGS, and *texture bleeding* across UV seams. The response is *unified latent representations* that decode to several formats [1721], so that a single generator serves diverse downstream requirements and the choice of $\mathcal{R}$ is deferred to inference.

Scalable, Reliable Evaluation and Human-Aligned Metrics. These representation choices complicate evaluation, because the metric panel a paper reports tends to depend on which format its generator targets, and the automatic metrics remain imperfect for geometry plausibility and asset usability. CLIP-style alignment is fooled by *texture shortcuts* that ignore the underlying shape; Chamfer Distance saturates on coarse geometry while penalizing semantically irrelevant fine detail; and LPIPS on rendered views cannot distinguish a physically plausible asset from a visually plausible cardboard cut-out. Multi-dimensional evaluators trained on large human-annotation sets (*e.g.*, HyperScore on MATE-3D [1782]) and diagnostic benchmarks (*e.g.*, GT23D-Bench [1781], T23D-QA [1783]) that score alignment, geometry, and texture on independent axes are the current answer, so that progress on one axis is no longer masked by averaging.

Data Scaling and Asset Readiness. Underlying consistency, representation, and evaluation alike is the supply of training data and the industrial scale at which it can be exploited. Large-scale corpora such as Objaverse-XL [1765] and industrial-scale systems such as Hunyuan3D [1787, 1789] and Meta 3D Gen [1788] point toward *high-resolution, engine-ready assets* with controllable materials, relightable textures, and post-processing for watertightness and topology. The question has shifted from *whether* a paradigm can be trained at scale, which the fourth paradigm has settled, to *which slice* of the 3D-asset distribution the available corpora still under-represent: articulated objects, semi-transparent or refractive materials, and topologically

non-trivial assemblies are visibly thinner in current open corpora than in production pipelines, and closing this gap is the explicit target of recent industrial releases.

8.2.2 3D Scene Synthesis

Scene-level synthesis extends 3D generation from a single asset $\mathbf{r} \in \mathcal{R}$ to an entire environment: a clutter of objects whose positions, orientations, and pairwise relations together compose a navigable room or open-world scene. Where §8.2.1 models the conditional distribution of one object given a prompt, scene synthesis must jointly model (i) a variable number of objects, (ii) their global $\mathrm{SE}(3)$ placements, (iii) their pairwise spatial and functional relations, and (iv) the envelope geometry (floor, walls, ceiling) that closes the scene; the output is a structured set rather than a vector in $\mathcal{R}$, and the conditioning broadens from text to layouts, scene graphs, sparse calibrated views, and full camera trajectories. The result must be navigable, not only renderable, which makes the problem harder than object synthesis along three axes at once: geometric coherence across the clutter, semantic consistency between related objects, and view generalization along an $\mathrm{SE}(3)$ trajectory rather than a fixed look-at orbit. Four families answer it: *layout- and scene-graph-conditioned* generators that produce the structured set from a symbolic prior; *view-to-3D* synthesizers that regress or sample a continuous radiance or Gaussian field from sparse calibrated views; *view-to-view* synthesizers that skip the explicit 3D representation and learn a conditional map from observed to novel views; and *video-model-based* synthesizers that treat a pretrained video diffusion model as an implicit 3D prior with the camera trajectory as an additional conditioning channel.

8.2.2.1 Task Definition

We cast 3D scene synthesis as a scene-level extension of the object-synthesis framework of §8.2.1. Let $\mathcal{S}$ denote the scene-output space whose elements $\mathbf{s} = \big(\{\mathbf{r}_i\}_{i=1}^{N}, \{\mathbf{T}_i\}_{i=1}^{N}, \mathbf{G}, \mathcal{E}_{\text{room}}\big)$ package N object representations $\mathbf{r}_i \in \mathcal{R}$ (with $\mathcal{R}$ the per-object space of §8.2.1), their global poses $\mathbf{T}_i \in \mathrm{SE}(3)$, a relational graph $\mathbf{G}$ over the object set, and an envelope geometry $\mathcal{E}_{\text{room}}$ encoding the room shell (walls, floor, ceiling). The conditioning space broadens to $\mathcal{C}^{\text{scene}} \supseteq \{$ text $\mathbf{x}$, layout $\mathbf{L}$, scene graph $\mathbf{G}_{\text{in}}$, sparse views $\{(I_v, \pi_v)\}_v$, camera trajectory $\{\pi_v\}_v \}$, with a condition encoder $\kappa^{\text{scene}} : \mathcal{C}^{\text{scene}} \to \mathcal{C}^{\text{scene}\prime}$ mapping each input mode into a shared embedding. The core learning problem is to train a conditional scene generator

$$G_\theta^{\text{scene}} : \mathcal{C}^{\text{scene}\prime} \to \mathcal{S}, \qquad \hat{\mathbf{s}} \sim p_\theta^{\text{scene}}\big(\mathbf{s} \,\big|\, \kappa^{\text{scene}}(\mathbf{c})\big), \tag{8.15}$$

so that the sampled scene $\hat{\mathbf{s}} \in \mathcal{S}$ admits coherent inter-object geometry, semantically plausible relations, and view-consistent renderings under arbitrary $\mathrm{SE}(3)$ trajectories. Two limiting cases fix the scope of Eq. (8.15). When $N = 1$, $\mathbf{G} = \varnothing$, and $\mathcal{E}_{\text{room}} = \varnothing$, the scene degenerates to a single asset and the sampler recovers Eq. (8.9): scene synthesis is a strict generalization of object synthesis. At the opposite extreme, when $\mathcal{S}$ is instantiated as an ordered sequence of views $\{I_v\}_{v=1}^{V}$ *without an explicit 3D substrate*, the sampler reduces to a camera-trajectory-conditioned video generator (the regime of §8.2.2.2 below), so that "scene synthesis" and "consistent video generation" are two boundary regimes of the same problem.

8.2.2.2 Methodology

Each of the four paradigms is a different factorization of G_θ^{scene} in Eq. (8.15), from sparse structural controls (layouts and scene graphs) through continuous 3D representations to video-based synthesis; Figure 8.7 previews them against a common scene.

Layout and Scene-graph Conditioned Methods. A major line of research generates 3D environments from high-level structural priors such as layouts $\mathbf{L}$ or scene graphs $\mathbf{G}_{\text{in}}$, instantiating Eq. (8.15) by materializing the structured tuple as an explicit object set,

$$\hat{\mathbf{s}} = \big\{(\hat{\mathbf{r}}_i,\, \hat{\mathbf{T}}_i)\big\}_{i=1}^{N}, \qquad \hat{\mathbf{s}} \sim p_\theta\Big(\{(\mathbf{r}_i, \mathbf{T}_i)\}_i \,\Big|\, \kappa(\mathbf{L} \text{ or } \mathbf{G}_{\text{in}}),\, \phi(\mathbf{x})\Big), \tag{8.16}$$

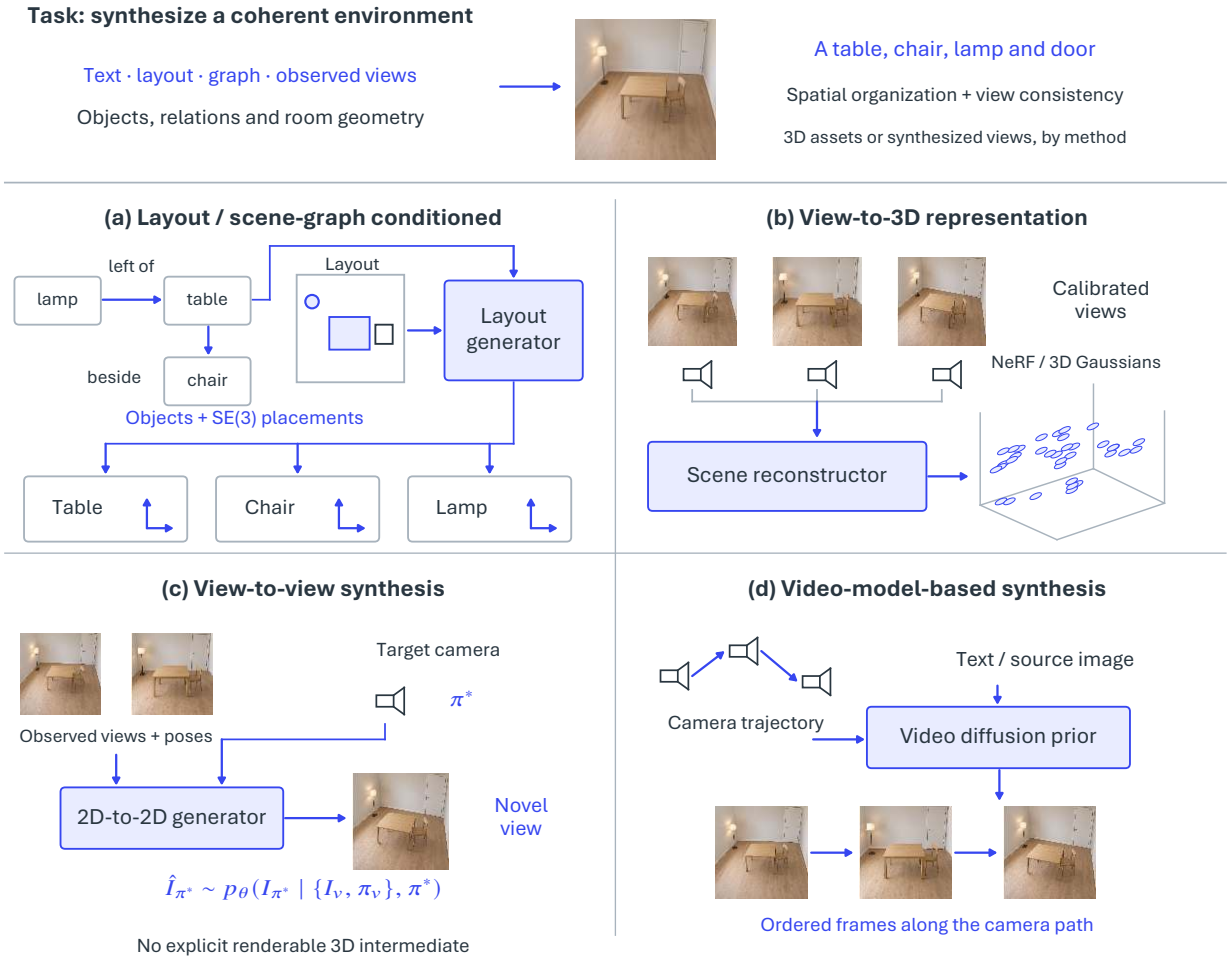


Figure 8.7: Four paradigm families for 3D scene synthesis (Eq. (8.15)): (a) layout- or scene-graph-conditioned placement of objects, (b) view-to-3D reconstruction into a radiance field or 3D Gaussians, (c) view-to-view synthesis with no explicit 3D, and (d) video-model-based synthesis along a camera path. The families are not mutually exclusive, and view consistency does not guarantee a valid 3D asset; room images are illustrative.

where the joint over the N slots and their poses is factorized autoregressively (a transformer over object slots) or as a denoising diffusion over the slot tensor. SceneFormer [1790] explored autoregressive modeling of room layouts and furniture placement, and DiffuScene [1791] introduced diffusion over object arrangements, improving realism and diversity. Recent work adds language and visual reasoning for compositional control: LayoutGPT [1703] plans the layout with an LLM before neural scene construction, and LayoutVLM [1792] optimizes layout consistency under multimodal alignment. Scene-graph approaches (GraphDreamer [1793], CommonScenes [1794], EchoScene [1795]) use relational reasoning to keep placements plausible, and InstructScene [1796] adds natural-language conditions for controllable scene creation. Beyond indoor settings, CityCraft [1797] scales layout synthesis to the city level, and GALA3D [1798] couples layout-guided diffusion with Gaussian-based rendering for photorealistic multi-object environments.

View-to-3D Representation Synthesis. The second paradigm trades the explicit structured set for a *continuous* 3D representation, recovering the scene as a radiance field or 3D Gaussian collection from sparse calibrated views; the scene tuple collapses into a single field,

$$\hat{\mathcal{R}}^{\text{scene}} = \mathcal{F}_\psi\Big(\{(I_v, \pi_v)\}_{v=1}^{V}\Big), \qquad \hat{\mathbf{s}} \equiv \hat{\mathcal{R}}^{\text{scene}}, \tag{8.17}$$

where $\mathcal{F}_\psi$ is either a deterministic feed-forward regressor or a generative sampler over the field. This scales the multi-view-and-lift recipe of §8.2.1.2 from one asset to a full scene by replacing the per-object reconstructor with a scene-scale one. Deterministic regressors such as PixelNeRF [1799], PixelSplat [1800], DepthSplat [675], and MVSplat [1801] / MVSplat360 [1802] stay strictly faithful to the input views and are reconstruction methods, treated in §8.2.3.2. The generative variant samples the field instead: DreamScene [1803] introduces a diffusion prior over Gaussian fields for text-conditioned scene creation, and Generative 3DGS [1804] generates 3D Gaussians for scenes with video priors.

View-to-View Synthesis (Regression and Generative). The third paradigm skips the explicit 3D representation altogether and formulates novel view synthesis as a direct conditional 2D-to-2D map from observed to target views,

$$\hat{I}_{\pi^*} \sim p_\theta\Big(I_{\pi^*} \,\Big|\, \{(I_v, \pi_v)\}_{v=1}^{V}, \pi^*\Big), \tag{8.18}$$

where π^* is the target camera pose and no intermediate 3D representation is materialized. Regression-based approaches learn a deterministic map: SRT [1805] encodes the scene into latent tokens that support interpolation and extrapolation across viewpoints, and LVSM [1806] predicts novel views with a large feed-forward transformer and minimal 3D inductive bias. Generative approaches model the conditional distribution over unseen views, which permits stochastic synthesis under ambiguity and large viewpoint changes: Photoconsistent-NVS [1807] and SEVA [1808] use diffusion to improve diversity and robustness along extended camera trajectories, and CausNVS [1809] generates views autoregressively and causally, modeling inter-view dependencies explicitly to stay coherent across complex viewpoint transitions.

Video-model Based Scene Synthesis. The fourth paradigm adds *temporal* coherence to the viewpoint variety of the view-to-view family by exploiting large video generation models as spatiotemporal priors; the scene tuple is materialized as a frame sequence,

$$\{\hat{I}_v\}_{v=1}^{V} \sim p_\theta^{\text{video}}\Big(\{I_v\}_{v=1}^{V} \,\Big|\, \{\pi_v\}_{v=1}^{V}, \phi(\mathbf{x})\Big), \tag{8.19}$$

treating the scene as the implicit 3D structure of a coherent video and the camera trajectory $\{\pi_v\}$ as an explicit conditioning channel. Where the feed-forward reconstructors of §8.2.1.2 output a static asset in one pass, the video paradigm trades static 3D for temporally coherent multi-frame output that mirrors the navigability requirement of a scene more directly. CameraCtrl [1810] and MotionCtrl [1811] condition video generation on camera trajectories directly; ViewCrafter [1812] and Gen3C [1813] add geometric guidance by unprojecting observed views into an intermediate 3D representation and reprojecting it to the target viewpoints, which improves pose fidelity; and CAT4D [1814] and 4DiM [1815] train jointly on posed static multi-view data and on temporal video data, so that viewpoint variation and temporal dynamics are modeled together for 4D scene synthesis.

8.2.2.3 Benchmarks

The four paradigms draw on different training corpora and report on different metric panels, each inherited from a precursor task.

Training Corpora. Table 8.3 lists the corpora in use. Layout-conditioned methods rely overwhelmingly on 3D-FRONT [1816], usually combined with its companion 3D-FUTURE [1817] furniture library for instance-level geometry. View-based methods (P2 / P3) draw on real-world scans of residential interiors (ScanNet [448], Matterport3D [340]), on photorealistic synthetic corpora with ground-truth depth and normals (Hypersim [664]), and on Replica [339] for compact high-fidelity indoor benchmarks. Video-model-based methods (P4) inherit the web-scale supervision of their underlying video prior, far larger than any 3D corpus but without explicit 3D supervision, supplemented by camera-pose-annotated subsets distilled from ScanNet, RealEstate10K [734], or hand-labeled trajectories.

Table 8.3: Representative training corpora for 3D scene synthesis.

Dataset	Year	Scale	Acquisition	Primary use / paradigm
ScanNet [448]	2017	∼1.5K scenes	Real-world RGB-D scans	Sparse-view supervision (P2 / P3)
Matterport3D [340]	2017	∼90 buildings	Real-world RGB-D scans	Whole-house multi-view (P2 / P3)
Replica [339]	2019	∼18 scenes	Real-world high-fidelity scans	Photorealistic indoor benchmarks (P2 / P3)
Hypersim [664]	2021	∼460 scenes	Photorealistic synthetic	Ground-truth depth / normals (P2 / P3)
3D-FRONT [1816]	2021	∼19K rooms	Synthetic indoor layouts	Layout / object-arrangement supervision (P1)
3D-FUTURE [1817]	2021	∼10K assets	Synthetic furniture library	Companion to 3D-FRONT (P1)

Evaluation Benchmark Suites. Layout-conditioned methods report the metrics introduced with ATISS on 3D-FRONT (object-category frequency, pairwise-relation accuracy, and FID on rendered top-down or perspective views). View-to-3D and view-to-view methods adopt the NeRF-era panel of PSNR / SSIM / LPIPS on held-out views of ScanNet, RealEstate10K [734], or DL3DV-10K [735]. Video-model-based methods add FVD and a camera-trajectory-adherence metric, the mean translation and rotation error between requested and recovered camera poses.

Constraint-specific Metrics. Four axes mirror the scene tuple of Eq. (8.15): **layout adherence** (object-count agreement, pairwise-relation accuracy on $\mathbf{G}$, and bounding-box IoU between requested and synthesized placements); **geometric fidelity** where ground truth exists (Chamfer Distance, F-score, and depth and normal consistency against held-out views); **photometric quality** (PSNR / SSIM / LPIPS against ground-truth views, FID / KID against held-out scene distributions, and FVD for video outputs); and **camera-pose adherence**, most relevant for P4 (translation and rotation error of poses recovered by structure-from-motion or a learned pose regressor). A paper reports the subset of axes its paradigm can be scored on, which is why comparison across paradigms remains difficult.

8.2.2.4 Challenges and Trends

Scene synthesis meets the problems of object synthesis on a structured set: more conditioning variables, more output components, and longer trajectories enter the cost at once.

Cross-Paradigm Inconsistency. The four paradigms solve subtly different problems, and each leaves a gap when used alone: a layout-conditioned generator can satisfy the symbolic prior $\mathbf{G}_{\text{in}}$ yet produce a non-traversable static scene, and a view-to-3D generator can produce a navigable Gaussian field that ignores the requested object semantics. The gaps surface as *layout-adherence drift*, where the generated NeRF or 3DGS no longer satisfies the input layout once the user roams outside the trained view envelope; *object drop*, where small or thin objects from $\mathbf{L}$ are silently omitted during continuous-field synthesis; and *geometry flattening*, where a multi-object composition collapses into a single textured slab when the 2D or video prior dominates. Hybrids such as GALA3D [1798] and ViewCrafter [1812] chain a layout or geometry prior with a Gaussian or video renderer to keep both halves honest, and future systems will more likely compose the four paradigms than choose among them.

Scaling Beyond Indoor Rooms. Almost all paradigms remain anchored to furnished indoor rooms because that is where the training corpora live: 3D-FRONT [1816] alone supplies the layout supervision for the dominant P1 methods, and ScanNet, Matterport3D, and Hypersim dominate P2 / P3. The result is an outdoor and urban scale gap that CityCraft [1797] is among the few systems to address, a category imbalance with many furniture classes but few terrain, vegetation, or industrial ones, and an open-world domain shift under which methods trained on residential interiors fail on industrial sites, excavations, or invented environments. Web-scraped open-vocabulary scene corpora and hybrid synthetic-real pipelines that blend

procedurally generated environments with captured scans are the active responses, both for breadth and to control the long tail of the scene-class distribution.

Camera-Trajectory Coherence under Long-Horizon Synthesis. The video paradigm (§8.2.2.2) excels at short camera moves but degrades on long, unbounded trajectories that require persistent world memory. Long trajectories expose *temporal blur*, where textures shimmer or drift between frames; *scene amnesia*, where objects reappear with different geometry when the camera revisits a location; and *pose-conditioning collapse*, where the model learns to ignore the camera signal in favor of the marginal video distribution. ViewCrafter [1812]'s unprojection and reprojection scaffold, Gen3C [1813]'s 3D cache, and CAT4D [1814]'s joint multi-view and video training all inject an explicit 3D inductive bias back into the video prior, but the trade-off between the video prior's perceptual quality and the geometric prior's rigidity remains an open design tension.

Evaluation. A P1 layout-adherence score and a P3 PSNR cannot be ranked against each other, no single dataset evaluates layout faithfulness, geometric fidelity, photometric quality, and pose adherence jointly, and no human-aligned evaluator analogous to HyperScore [1782] has been validated on scene outputs.

Trends: Toward 4D and World Models. Scene synthesis targets static geometry conditioned on camera poses, but the same formulation is being extended toward 4D synthesis and action-conditioned world modeling: Dreamer [147], GameNGen [1818], Diamond [1819], RTFM [1820], and Genie3 [157] move from static view or video synthesis toward open-ended environment generation in which spatial structure, temporal evolution, and interaction are modeled together. From the standpoint of this subsection, scalable, temporally coherent video generation with explicit viewpoint control is the foundation those systems build on; they are treated as world models in §8.6.

8.2.3 3D Scene Reconstruction

3D reconstruction recovers a digital 3D representation of a scene or object from sensor observations such as RGB images, video, depth maps, or point clouds, and has long been a foundational problem in computer vision, graphics, and immersive systems such as augmented reality (AR), virtual reality (VR), and digital twins. Its difference from the synthesis tasks of §8.2.1 and §8.2.2 is one of *intent*: synthesis is open-ended generation in which plausibility and diversity dominate, whereas reconstruction must explain the input views faithfully before any inference about unobserved regions is attempted. Reconstruction is where perception becomes geometry: it turns raw sensor streams into an explicit, metric 3D substrate that simulators, AR / VR systems, digital twins, and embodied agents can reason over, manipulate, and render, and it supplies the ground-truth geometry against which the generative tasks of §8.2 are ultimately measured. Three families follow: *iterative reconstruction*, which solves a per-scene optimization against the input views; *feed-forward reconstruction*, which amortizes that cost into a learned map from observations to a 3D representation; and *complete reconstruction*, which generatively fills the occluded regions and so makes the boundary with synthesis explicit. Figure 8.8 previews representative reconstruction tasks.

8.2.3.1 Task Definition

We cast 3D scene reconstruction as the observation-conditioned counterpart of the generative frameworks of §8.2.1 and §8.2.2. Let $\mathcal{O}$ denote the space of multi-sensor observations. Each observation $\mathbf{O} \in \mathcal{O}$ is represented as a finite collection of sensor measurements. Depending on the sensing setup, $\mathbf{O}$ may consist of multi-view RGB images $\{I_v\}_{v=1}^{V}$, posed images $\{(I_v, \pi_v)\}_{v=1}^{V}$, RGB-D or RGB-normal observations $\{(I_v, d_v)\}_{v=1}^{V}$, point clouds, LiDAR scans, or their combinations. Here, I_v denotes the RGB image from view v, π_v denotes the corresponding calibrated camera pose, and d_v denotes an associated depth or normal map. Let $\mathcal{R}^{\text{rec}}$ denote the output 3D-representation space, taken from the same $\mathcal{R}$ as in §8.2.1, including mesh, SDF, voxel, triplane, NeRF, 3DGS, point map, structured latent. The core learning problem is to obtain a reconstruction operator

$$G_\theta^{\text{rec}} : \mathcal{O} \to \mathcal{R}^{\text{rec}}, \qquad \hat{\mathbf{r}}^{\text{rec}} = G_\theta^{\text{rec}}(\mathbf{O}) \quad \text{(deterministic)} \quad \text{or} \quad \hat{\mathbf{r}}^{\text{rec}} \sim p_\theta^{\text{rec}}\big(\mathbf{r}^{\text{rec}} \,\big|\, \mathbf{O}\big) \quad \text{(generative)}, \tag{8.20}$$

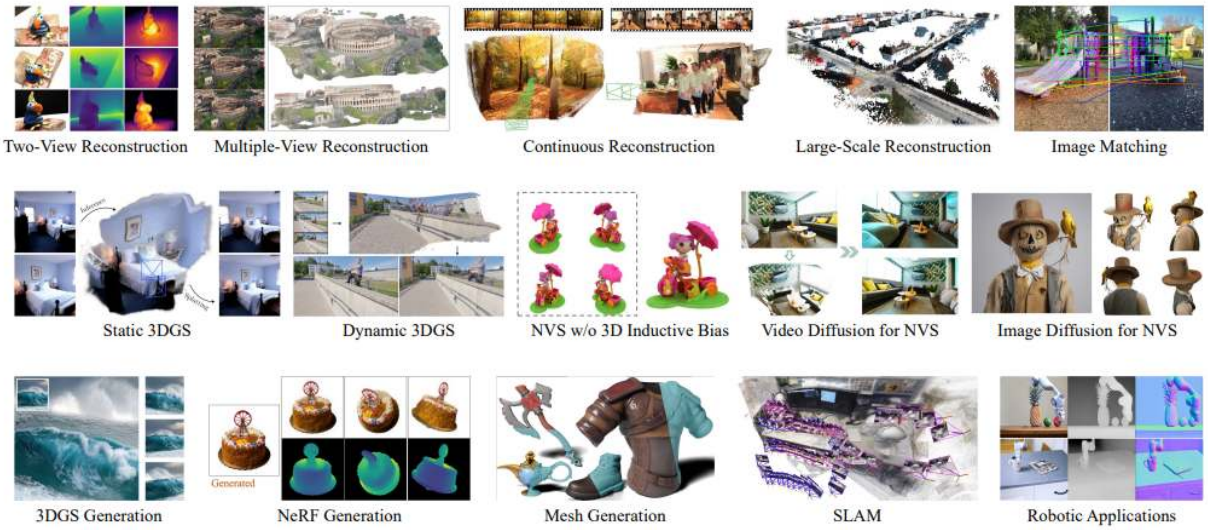


Figure 8.8: Illustrations of 3D reconstruction, adapted from [1821].

that minimizes a reconstruction loss against the input views,

$$\mathcal{L}_{\text{rec}}\big(\hat{\mathbf{r}}^{\text{rec}}, \mathbf{O}\big) \;=\; \sum_{(I_v,\pi_v)\in\mathbf{O}} \big\|\text{Render}(\hat{\mathbf{r}}^{\text{rec}}, \pi_v) - I_v\big\|^2 \;+\; \lambda\,\Omega(\hat{\mathbf{r}}^{\text{rec}}), \tag{8.21}$$

where Ω is an optional geometric or smoothness regularizer and Render is a differentiable renderer of $\mathcal{R}^{\text{rec}}$. The structural contrast with §8.2.1 is sharp: there, the conditioning $\kappa(\mathbf{c})$ is a high-level prior (text, single image, multi-view) and the objective is a *generative* likelihood; here, the conditioning $\mathbf{O}$ is sensor data and the objective is an *observation-fidelity* loss.

8.2.3.2 Methodology

The three paradigms are three parameterizations of the reconstruction operator G^{rec}_θ of Eq. (8.20), taken in the order in which each became dominant. Figure 8.9 contrasts them, making explicit whether the 3D representation is optimized per scene, regressed in a single forward pass, or sampled from a generative posterior over the unobserved regions.

Iterative 3D Reconstruction. The classical paradigm casts reconstruction as a *per-scene* optimization: there is no learned θ shared across scenes; the 3D representation $\mathbf{r}$ itself is the optimization variable, and Eq. (8.21) is minimized from scratch for each new input,

$$\hat{\mathbf{r}} \;=\; \arg\min_{\mathbf{r}\in\mathcal{R}^{\text{rec}}} \sum_{(I_v,\pi_v)\in\mathbf{O}} \big\|\text{Render}(\mathbf{r}, \pi_v) - I_v\big\|^2 \;+\; \lambda\,\Omega(\mathbf{r}). \tag{8.22}$$

Classical image-based reconstruction is rooted in Structure-from-Motion (SfM) [1822, 107, 1823, 115], which estimates camera poses from unposed images, followed by Multi-View Stereo (MVS) [1824, 112, 1179, 113, 1825, 1826], which recovers dense geometry. Decades of refinement have made SfM + MVS a mature and widely used pipeline (see [112] for a comprehensive treatment), yet it still struggles with textureless surfaces, reflective objects, and large illumination changes.

A more recent line represents the scene as a continuous function parameterized by a neural network: signed distance functions [117, 1827, 288, 1828], occupancy fields [118, 1829], and neural radiance fields [119, 271, 1830, 1831]. The original formulations were expensive to train and render and limited in quality; PlenOctrees [1832], Plenoxels [289], Instant-NGP [120], RefNeRF [1833], TensoRF [1834], and ZipNeRF [272] are among the many follow-ups that improved efficiency and fidelity, but all of them still require a large number

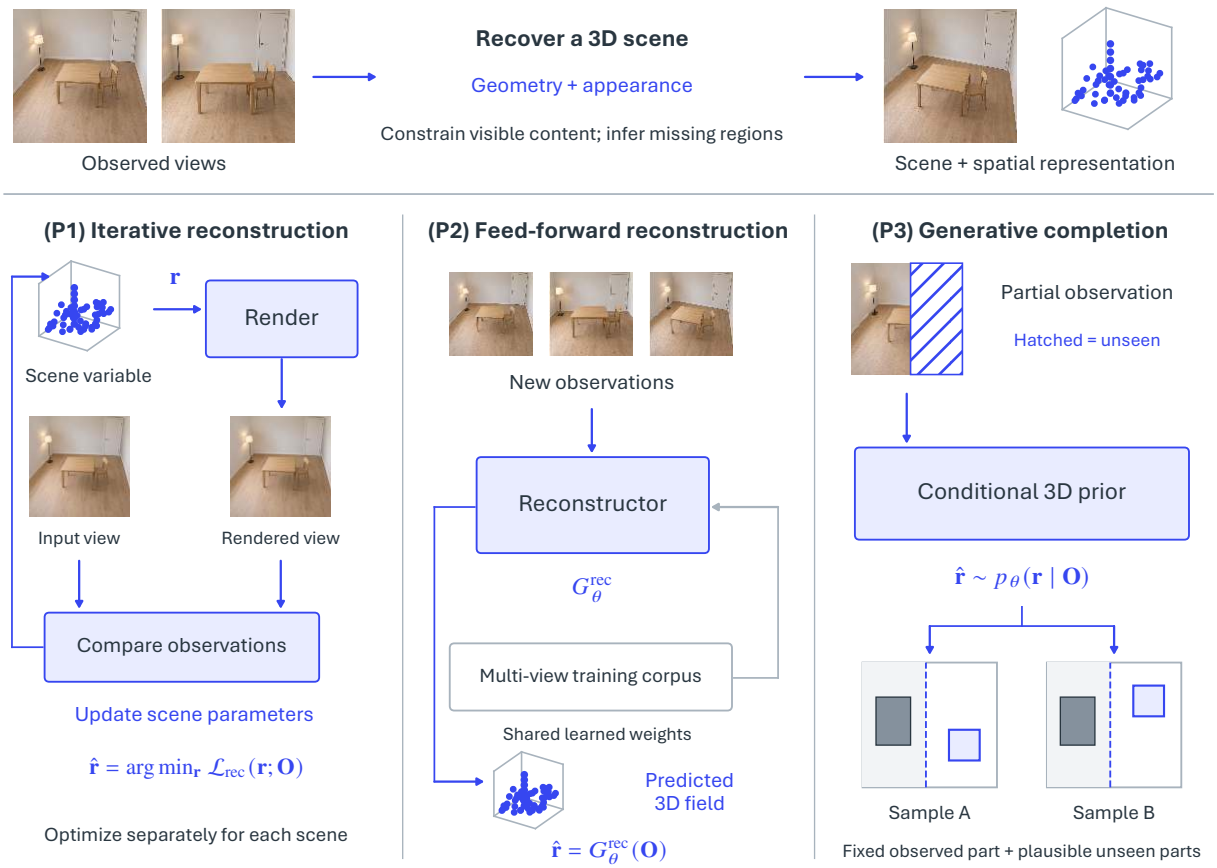


Figure 8.9: Three paradigm families for 3D scene reconstruction (Eq. (8.20)): (P1) iterative per-scene optimization, (P2) feed-forward reconstruction with weights shared across scenes, and (P3) generative completion of unseen (hatched) regions from a learned prior.

of input images per scene. Sparse-view variants add auxiliary information such as regularization [1835], depth priors [1836, 1837], point clouds [1838], and surface priors [1839], or distill a 2D diffusion prior into the per-scene representation, the SDS recipe of §8.2.1.2 applied to reconstruction from one or a few images [1840, 1841, 1842]. 3D Gaussian Splatting [121] then made per-scene optimization fast, training within minutes and rendering in real time, with subsequent work adding depth regularization, anti-aliasing, mesh extraction, and surface alignment [1843, 290, 1844, 1845]. All of these systems, however, work well only over nearby views or a small scene and must be re-optimized from scratch for every new input, which motivates the amortized paradigm.

Feed-forward 3D Reconstruction. The feed-forward paradigm amortizes the per-scene optimization of Eq. (8.22) across a learned prior: a network G_θ^{rec} is trained once on a large multi-view corpus, then deployed in a *single forward pass* on unseen inputs, minimizing

$$\mathcal{L}_{\text{FF}}(\theta) \;=\; \mathbb{E}_{(\mathbf{O},\,\mathbf{r}_{\text{gt}})\sim p_{\text{data}}}\Big[\mathcal{D}\big(G_\theta^{\text{rec}}(\mathbf{O}),\,\mathbf{r}_{\text{gt}}\big)\Big], \tag{8.23}$$

where $\mathcal{D}$ is a representation-specific distance (Chamfer Distance for point maps, a rendering loss for radiance fields) and $(\mathbf{O}, \mathbf{r}_{\text{gt}})$ are observation–asset pairs drawn from a 3D corpus. This is the reconstruction-side analogue of the view-to-3D paradigm of §8.2.2.2: sparse calibrated views are mapped feed-forward to a continuous scene representation, but the supervision is a reconstruction loss against ground-truth 3D rather than a rendering loss against held-out views alone.

Early feed-forward work applied 2D networks to individual 3D representations, for depth estimation [55, 1846], voxel reconstruction [1847, 1377], point-cloud reconstruction [1848, 132], and mesh deformation [285, 1849], but remained limited to single objects or small datasets. Learned components then replaced individual stages of the SfM + MVS pipeline, keypoint detection and description [539, 1850, 540], image matching [541, 1851], pose estimation [723, 625, 1852], and depth prediction [133, 625], proving more robust and accurate than their hand-designed counterparts and being integrated into the classical pipeline [1853, 116, 622]. DUSt3R [122] and MASt3R [728] then reconstructed large scenes from unposed images end to end: similar in spirit to MVS but without explicit keypoint detection, matching, and pose estimation, the model predicts point maps in the reference view directly and recovers camera poses in post-processing, and follow-ups [1854, 731, 732, 1855, 123] have improved quality and efficiency at scale.

Beyond geometry alone, a parallel line recovers geometry and appearance together through differentiable rendering. Scene Representation Networks [183] and Light Field Networks [1856] introduced ray-based querying for novel-view synthesis and were followed by epipolar attention [1857, 1858], large transformers [1806, 1859], and integration with pretrained generative models [1758, 1757]; PixelNeRF [1799] pioneered pixel-aligned features for NeRF-based reconstruction and inspired 3D volume representations [1860, 1861], feature-matching fusion [1862], and large transformers [1863, 1766]. 3D Gaussian Splatting [121] has since become the representation of choice for feed-forward reconstruction, in PixelSplat [1800], Splatter Image [1864], MVSplat [1801], Flash3D [1865], LatentSplat [1737], Splatt3R [1866], GS-LRM [1867], NoPoSplat [1868], and DepthSplat [675]. These methods synthesize novel views with high appearance fidelity, but the views remain confined to small camera changes within the observed envelope, which motivates the generative paradigm below.

Complete 3D Reconstruction. A harder regime is to reconstruct a *complete* scene from an observation $\mathbf{O}$ that covers only part of it, which requires inferring both *visible* and *occluded* regions; the operator shifts from a deterministic map to a generative posterior over the unobserved regions,

$$\hat{\mathbf{r}}^{\text{rec}} \sim p_\theta^{\text{rec}}\big(\mathbf{r}^{\text{rec}} \,\big|\, \mathbf{O}\big), \qquad \mathbf{r}^{\text{rec}}\big|_{\text{visible}} \approx \text{observation-consistent}, \quad \mathbf{r}^{\text{rec}}\big|_{\text{occluded}} \sim p_\theta(\,\cdot\,\big|\,\mathbf{O}), \tag{8.24}$$

which shares the formal structure of the synthesis sampler of Eq. (8.9) with the conditioning replaced by a strong observation constraint $\mathbf{O}$.

Complete reconstruction has been studied extensively for single objects [1869, 1771, 1870, 1774, 1871, 1721, 1786, 1776, 1787, 1872] but remains under-explored for large scenes. Classical scene-scale methods fuse depth maps from multiple views into a volumetric representation [109, 114, 392, 334, 170], but they are sensitive to depth errors and produce noisy, incomplete reconstructions. A newer line fine-tunes large pretrained diffusion models [185, 1873] for the task, conditioning on controlled camera trajectories [1811, 1874, 1875, 1757, 1876, 1808] or on auxiliary signals [1877, 1812, 1802, 1447]; these methods synthesize observation-consistent views but leave the underlying geometry implicit. A small but growing body of work addresses geometry directly: WVD [1878] and Bolt3D [1879] generate a hybrid RGB + point-map output that recovers geometry and appearance rather than novel views alone, and LaRI [1880] reconstructs a complete scene from a single image, still limited to small indoor scenes.

Two design choices cut across the three paradigms. The density of the observation decides which paradigm applies: a dense calibrated multi-view set calls for per-scene optimization, a sparse uncalibrated capture for a feed-forward map, and a single image for a generative posterior, since progressively more of the scene must be inferred rather than observed. The output representation $\mathcal{R}^{\text{rec}}$, by contrast, is largely independent of the paradigm and driven by the downstream consumer: a point map (DUSt3R), a mesh (classical MVS post-processing), a NeRF (per-scene optimization, PixelNeRF), a 3DGS field (3DGS, PixelSplat), or a hybrid RGB + point map (Bolt3D, WVD).

Table 8.4: Representative training corpora for 3D scene reconstruction. "Type" indicates synthetic / real-world acquisition; "Dynamics" indicates whether the corpus contains temporal variation.

Dataset	Year	Scale	Type	Dynamics	Primary use
Objaverse [1764]	2023	∼800K objects	Synthetic (CAD)	Static	Object-centric multi-view supervision
Zeroverse [1882]	2024	Procedural	Synthetic	Static	LRM-zero scalable supervision
MegaSynth [1881]	2025	Procedural	Synthetic	Static	Scalable multi-view scene supervision
ARKitScenes [669]	2021	∼5K scans	Real-world	Static	Indoor RGB-D / mesh supervision
ScanNet++ [939]	2023	∼460 scenes	Real-world	Static	High-fidelity indoor multi-modal scans
KITTI-360 [662]	2022	∼300K frames	Real-world	Dynamic	Outdoor driving scenes
TartanAir [666]	2020	∼1M frames	Synthetic	Mixed	Static + dynamic mixed-domain evaluation

8.2.3.3 Benchmarks

Datasets span object-centric collections, indoor scenes, and outdoor environments, and partition further by acquisition (synthetic or real-world) and dynamics (static or dynamic); Table 8.4 summarizes those most commonly used.

Training Corpora. Synthetic corpora such as Objaverse [1764] supply large and diverse 3D assets, and MegaSynth [1881] and Zeroverse [1882] explore procedurally generated data for scalable supervision. Real-world static corpora include ARKitScenes [669] and ScanNet++ [939] for indoor capture, dynamic corpora such as KITTI-360 [662] capture outdoor scenes with temporal variation, and TartanAir [666] contains both static and dynamic scenes, which permits evaluation across both settings.

Evaluation Benchmark Suites. Iterative methods (P1) are evaluated on DTU [1174] and Tanks-and-Temples [1173] for geometric fidelity and on the NeRF-era benchmarks LLFF [1883], MipNeRF360 [1831], and RealEstate10K [734] for novel-view quality. Feed-forward methods (P2) adopt the same panel together with ScanNet [448] subsets that supply ground-truth point maps for direct geometry supervision. Complete-reconstruction methods (P3) have no standardized suite and typically report on the precursor benchmark of their underlying 2D or video prior.

Constraint-specific Metrics. Four axes mirror the rendering-versus-geometry decomposition of Eq. (8.21). **Photometric quality** is measured by PSNR, SSIM [1884], and LPIPS [1885], covering pixel-level fidelity, structural similarity, and perceptual consistency of novel views. **Camera-pose adherence** uses Relative Translation Accuracy (RTA), Relative Rotation Accuracy (RRA), and the Area Under the Curve (AUC) that summarizes pose accuracy across angular thresholds. **Geometric fidelity** of point maps and point clouds is measured by Accuracy (precision), Completeness (recall), and Chamfer Distance, where Accuracy reflects geometric precision, Completeness measures surface coverage, and Chamfer Distance aggregates both. **Depth quality** for monocular estimation uses the absolute relative error $|y - \hat{y}|/y$ and the inlier ratio $\delta < T$ (the fraction of predictions within a multiplicative threshold of the ground truth), with median scaling or least-squares fitting applied beforehand because monocular depth is scale-ambiguous; multi-view depth reports Accuracy, Completeness, and Overall, the mean of the first two and equivalent to Chamfer Distance.

The paradigm succession of §8.2.3.2 is visible in two quantities. Per-scene optimization has moved from NeRF [119], which trains for hours and renders at a few frames per second, to 3D Gaussian Splatting [121], which trains a scene in minutes and renders in real time ($\geq$30 fps at 1080p); the feed-forward paradigm removes per-scene optimization altogether, with DUSt3R / MASt3R [122, 728] and VGGT [123] recovering point maps, depth, and camera poses in a single sub-second forward pass and Fast3R [732] processing over a thousand images at once. On pose, learned reconstructors now rival classical pipelines: VGGT [123] reports a camera-pose AUC@30 of 85.3 on RealEstate10K in pure feed-forward mode and 93.5 with bundle-adjustment refinement. Two caveats apply. There is no cross-paradigm leaderboard, since P1 is scored on novel-view PSNR / SSIM / LPIPS, P2 adds geometric Accuracy / Completeness / Chamfer Distance and pose AUC, and P3 inherits whichever protocol its 2D or video prior used; and on raw depth-map accuracy feed-forward

models still trail tuned MVS pipelines, with quality degrading under free-viewpoint extrapolation (§8.2.3.4), so the strongest numbers hold inside the observed view envelope. Phone-based NeRF and Gaussian-Splatting capture apps and the open-source Nerfstudio stack have brought per-scene reconstruction to commodity devices, while classical photogrammetry suites remain the production baseline for metric accuracy.

8.2.3.4 Challenges and Trends

Four problems span the pipeline end to end: the data the model trains on, the accuracy it can deliver, the quality of the views it renders, and the compute envelope that constrains all three at long input lengths; reconstruction shares the first and third with its synthesis counterparts but sharpens the other two because it commits to recovering ground truth from raw sensor data.

Limited Modality in Datasets. Most reconstruction and view-synthesis datasets cover a narrow band of modalities. Widely used benchmarks such as RealEstate10K [734] and MipNeRF360 [1831] are RGB-only, without depth, LiDAR, or semantic annotation, and even Objaverse-XL [1765], with 10.2M objects, is primarily synthetic mesh data without real-world sensor modalities. Merging datasets of different modalities is the usual workaround, at the cost of domain shift and annotation inconsistency, and the gap is widest for dynamic scenes: datasets with synchronized RGB, depth, optical flow, and 3D motion remain far fewer than static ones, and most dynamic datasets record either camera motion or object motion rather than both with a full sensor suite. The open problem is a scalable, modality-rich corpus that combines the diversity of Objaverse [1764] with the multi-sensor completeness of ScanNet++ [939], collected under one protocol so that aligned RGB, depth, and semantic signals supervise the same model.

Reconstruction Accuracy. Feed-forward models still trail traditional MVS pipelines [1886, 1821] on raw depth-map accuracy, because MVS exploits known camera parameters and a hypothesized depth set to build cost volumes from which depth is predicted. Whether feed-forward reconstructors [732, 123] learn an approximation of such cost volumes implicitly in their attention layers is an open question; injecting explicit camera parameters or other priors into the backbone, for instance through Diffusion Transformers [1645], is one route to closing the gap.

Free-viewpoint Rendering. The accuracy gap is most visible at *free-viewpoint* rendering, where the model must produce novel views far from the input camera distribution. Disocclusions, geometric uncertainty, and limited generalization conspire: regions never observed come out blurred, ghosted, or geometrically wrong, because current methods rely on local consistency and cannot infer plausible content for occluded areas, and view-dependent effects and complex light transport demand reasoning beyond interpolation. Hallucinating missing detail while keeping novel views consistent is exactly the generative regime of §8.2.3.2.

Long Context Input. Reconstruction from many views meets a hard computational ceiling: methods for geometry reasoning and novel-view synthesis rely on full attention, whose compute grows quadratically with the number of tokens; exact-attention kernels such as FlashAttention-2 [1887] remove the quadratic memory of the attention matrix but not the quadratic compute. VGGT [123], for example, needs about 21 GB of GPU memory for 100 input frames and about 41 GB for 200, so inference over long sequences and training on many views quickly exceed a single GPU. Recurrent alternatives such as CUT3R [731] integrate new views incrementally while maintaining a memory state, which keeps inference memory near constant, but they forget earlier views and degrade as the number of inputs grows. Reasoning efficiently over hundreds or thousands of views while keeping memory and compute manageable remains open, with recurrent and sparse-attention architectures the main candidates.

8.2.4 3D Panorama Synthesis

3D panorama synthesis occupies a distinctive position in spatial generation: rather than producing a 3D representation directly, it generates a complete 360-degree omnidirectional image as a structured *intermediate*

from which a full 3D environment can then be lifted. The panorama gives embodied agents, AR / VR pipelines, and world models a holistic spatial context with minimal occlusion anchored at one observation point, and it bridges the mature 2D generative prior of §8.1 with the 3D substrate on which the rest of §8.2 operates. Four method families trace the path from a single omnidirectional snapshot to an explorable world: panoramic image synthesis, panoramic video generation, panoramic 3D reconstruction, and panorama-based 3D scene generation.

8.2.4.1 Task Definition

Panorama synthesis generates a complete 360-degree environment conditioned on text, a reference panorama or NFoV image, a spatial layout, or a specified viewpoint; unlike object-centric or narrow-field-of-view (NFoV) generation, it covers the whole environment around the observer at once rather than a fragment of it. Formally, we cast it as the omnidirectional-intermediate counterpart of the generative frameworks of §8.2.1 and §8.2.2. Let $\mathcal{P}$ denote the space of panoramic images (functions defined on the sphere $\mathbb{S}^2$ and encoded through a projection operator $\Pi \in \{\text{ERP}, \text{cubemap}, \text{tangent}, \text{fisheye}\}$ onto a processable 2D array) and let $\mathbf{c} \in \mathcal{C} \supseteq \{\text{text}, \text{reference panorama / NFoV image}, \text{layout}, \text{viewpoint / camera trajectory}\}$ be the conditioning. Panorama synthesis first learns a panoramic generator and then optionally lifts its output into 3D,

$$G_\theta^{\text{pano}} : \mathcal{C}' \to \mathcal{P}, \qquad \hat{P} \sim p_\theta^{\text{pano}}\big(P \,\big|\, \kappa(\mathbf{c})\big), \qquad \hat{\mathbf{r}}^{\text{rec}} = \Lambda\big(\hat{P}\big), \tag{8.25}$$

where κ is the condition encoder of §8.2.1, $\hat{P} \in \mathcal{P}$ is the synthesized omnidirectional image, and $\Lambda : \mathcal{P} \to \mathcal{R}^{\text{rec}}$ is an optional *lifting operator* mapping the panorama into the 3D-representation space $\mathcal{R}^{\text{rec}}$ of §8.2.3 (drawn from $\mathcal{R}$ of §8.2.1, *e.g.* NeRF, 3DGS, mesh, point map). Equation (8.25) makes the role of the panorama as a structured *intermediate* explicit: the hard generative work is performed on the sphere by G_θ^{pano} (inheriting the 2D generator G_θ of Eq. (8.2) with its outputs defined on $\mathbb{S}^2$ rather than the plane), while Λ reuses the reconstruction machinery of §8.2.3. This is also what separates panorama synthesis from the scene synthesis of §8.2.2, which produces 3D representations directly from arbitrary viewpoints distributed over $\text{SE}(3)$: here every direction around one observation point is represented coherently first, and 3D structure is recovered from that complete view. Two limiting cases locate Eq. (8.25) within §8.2. When $\Lambda = \text{id}$ (no lifting), the task collapses to omnidirectional 2D image generation, a spatially aware variant of the text-to-image sampler of §8.1 defined on the sphere rather than the plane. When $\mathbf{c}$ is a set of sparse posed panoramas and Λ is a per-scene optimization, Eq. (8.25) reduces to panoramic reconstruction, recovering the observation-conditioned operator of Eq. (8.20).

Panoramic Image Formats. The projection Π fixes how the sphere is laid out for a 2D network and which distortions the generator must absorb. Equirectangular projection (ERP) maps the sphere onto a rectangle in latitude–longitude coordinates; it stretches content near the poles but is continuous and directly compatible with standard 2D pipelines and convolutional networks, which has made it the dominant format for learning-based synthesis. Cubemap projection represents the sphere as six perspective faces (front, back, left, right, top, bottom), removing polar distortion at the cost of discontinuities along face boundaries, and is the native format of real-time rendering and VR / AR systems, where each face is processed as an ordinary perspective image. Tangent images and perspective patches decompose the panorama into overlapping tangent-plane projections so that models trained on ordinary perspective images can be reused. Fisheye and dual-fisheye projections capture 180–220 degrees per lens with radial distortion; most consumer 360-degree cameras capture two hemispheres this way and stitch them, so dual-fisheye is the native acquisition format even when methods operate on ERP.

8.2.4.2 Methodology

The four families are four instantiations of the panoramic generator G_θ^{pano} and lifting operator Λ of Eq. (8.25): image synthesis stops at a static panorama, video generation extends it over time, panoramic reconstruction

applies Λ to observed panoramas, and panorama-based scene generation composes the two. Figure 8.10 schematizes them and the panorama-as-intermediate pipeline that connects them.

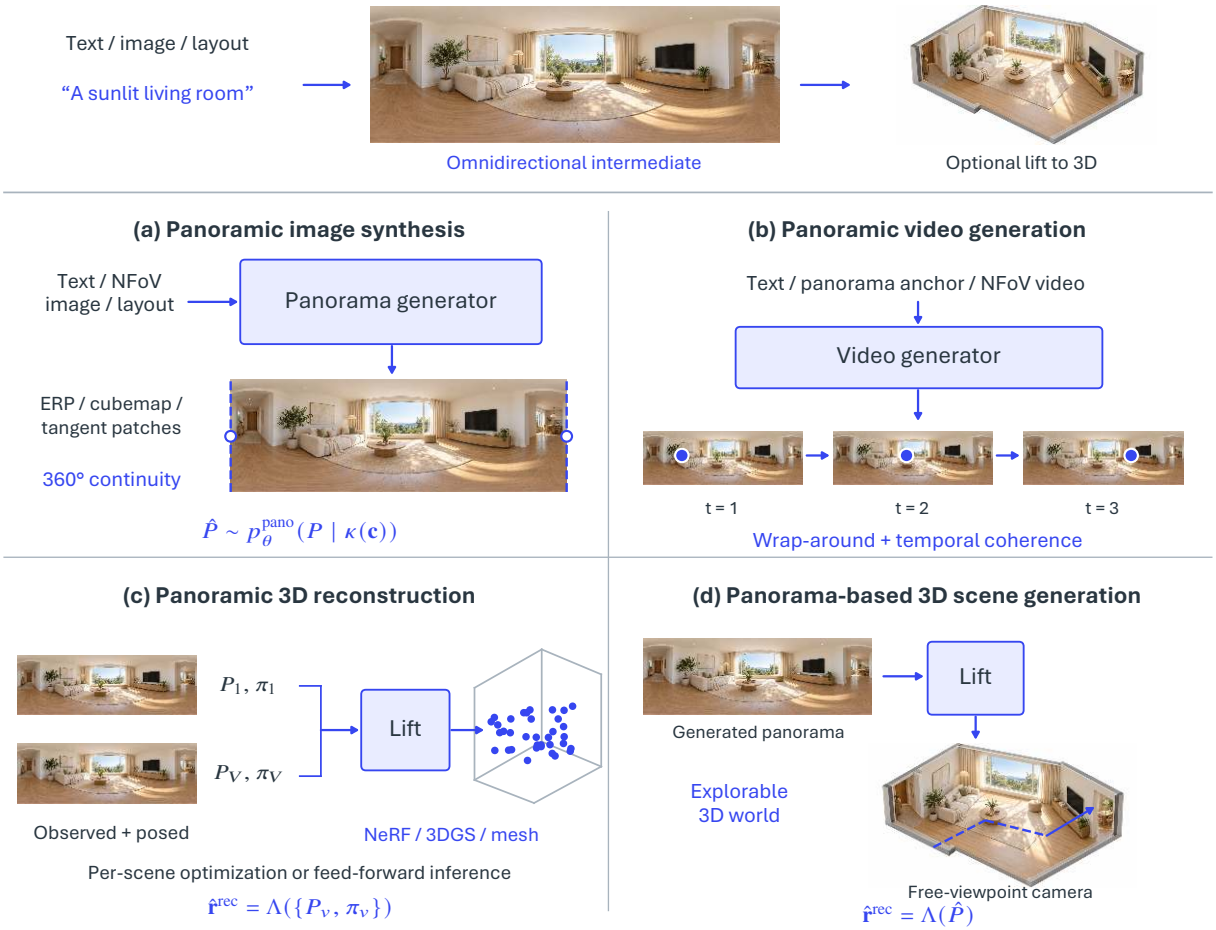


Figure 8.10: The four method families of 3D panorama synthesis (Eq. (8.25)), linked by a panoramic intermediate (top): (a) panoramic image synthesis, (b) panoramic video generation, (c) panoramic 3D reconstruction, and (d) panorama-based 3D scene generation. Room imagery is illustrative.

Panoramic Image Synthesis. Two task formulations differ in their conditioning: *text-to-panorama generation*, which leans on the prior of a pretrained generator and is judged on semantic alignment and diversity, and *image-conditioned panorama generation*, which expands an NFoV image or an incomplete panorama by inpainting or outpainting and must preserve the content, style, and panoramic distortion of its input. Formally, this family is the base case of Eq. (8.25) with the lifting operator disabled: it samples a single static omnidirectional image and stops,

$$\hat{P} \sim p_\theta^{\text{pano}}\big(P \mid \kappa(\mathbf{c})\big), \qquad \Lambda = \mathrm{id}, \tag{8.26}$$

where the generative work is carried out under a chosen projection $\Pi \in \{\text{ERP}, \text{cubemap}, \text{tangent}\}$ that dictates how spherical distortion is handled; Figure 8.10(a) depicts this static-panorama pathway.

Early approaches [1888, 1889] relied on GANs [1646] and were constrained by limited training data and mode collapse; diffusion models [184, 185] brought diversity and fidelity, and current methods divide by projection format and architecture. The three strategies below are three realizations of the sampler of

Eq. (8.26), differing in how the spherical output is factorized before being mapped onto the sphere,

$$\hat{P} = \begin{cases} \Pi_{\text{ERP}}^{-1}(\hat{Z}), \;\; \hat{Z} \sim p_\theta\big(Z \mid \kappa(\mathbf{c})\big) & \text{(direct ERP)}, \\ \Pi_{\text{cube}}^{-1}\big(\{\hat{F}_i\}_{i=1}^{6}\big), \;\; \{\hat{F}_i\} \sim p_\theta\big(\{F_i\} \mid \kappa(\mathbf{c})\big) & \text{(cubemap)}, \\ \mathcal{S}\big(\{\hat{I}_k\}_{k=1}^{K}\big), \;\; \hat{I}_k \sim p_\theta^{\text{2D}}\big(I \mid \kappa(\mathbf{c})\big) & \text{(perspective + stitch)}, \end{cases} \tag{8.27}$$

where Π^{-1} lifts the generated representation onto the sphere and $\mathcal{S}$ is a spherical stitch-and-blend operator. The first two cases synthesize the panorama *directly* (and so require panoramic training data), whereas the third reuses a frozen perspective generator p_θ^{2D} and pushes the cross-view-consistency burden into $\mathcal{S}$.

▶ *Equirectangular projection (ERP) methods.* The direct-ERP case generates on the latitude–longitude grid and must handle polar distortion and the wrap-around seam architecturally. For text-to-panorama generation, DiffPano [1890] fine-tunes a diffusion model with LoRA on ERP, SMGD [1891] guides the diffusion process with spherical manifold convolutions that respect geometry across the sphere, and PAR [1892] replaces diffusion with autoregressive modeling to avoid the mismatch between ERP and the noise assumptions of standard diffusion. The seam is treated at the denoising or decoding stage: PanoDiff [1893] uses geometric augmentation when completing a panorama from unregistered NFoV images, PanoDiffusion [1894] generates depth jointly and rotates the latent circularly to enforce left–right continuity, StitchDiffusion [1895] re-denoises the wrap-around region at every step together with global cropping, and PanFusion [1896] pairs an ERP branch with a multi-view perspective branch to draw on both representations. For image-conditioned generation, OmniDreamer [1897] introduced mask-based outpainting from NFoV images with GANs, AOG [1898] and ODIS [1899] explore alternative conditioning for inpainting and outpainting, and PanoDecouple [1900] separates distortion guidance from content completion to counter the "visual cheating" in which a model buys image quality at the cost of geometric accuracy.

▶ *Cubemap-based methods.* The second case of Eq. (8.27) generates six perspective faces, which removes polar distortion but moves the problem to the face boundaries. CubeDiff [1901] introduced the cubemap representation for text-to-panorama generation, and DreamCube [1902] synchronizes the six planes to suppress discontinuous seams and inconsistent color tone; overlapping the fields of view of adjacent faces reduces seams further but duplicates generation in the overlap and lowers the effective resolution. JoPano [1903] models the faces jointly with a DiT and a Joint-Face Adapter, applies Poisson blending along the boundaries, and introduces the seam metrics (Seam-SSIM, Seam-Sobel) used in §8.2.4.3.

▶ *Perspective image-based methods.* The third case leaves a perspective text-to-image model untouched and pushes the consistency burden into the stitch operator $\mathcal{S}$, with or without panorama-specific training. *Training-free methods* apply a pretrained perspective model directly: MultiDiffusion [1692] denoises overlapping patches independently and blends them, PanoGen [1904] outpaints recursively from an initial view until the sphere is covered, DiffCollage [1905] casts the panorama as a factor graph whose factor nodes are portions and whose variable nodes are their overlaps so that portions are generated in parallel rather than autoregressively, PanoFree [1906] iterates warping and inpainting over multi-view images, and SphereDiff [1907] moves MultiDiffusion onto a spherical latent with dynamic spherical sampling and distortion-aware weighted averaging so that quality holds up to the poles. These methods ride on advances in perspective generation but struggle with global consistency across the full sphere. *Training-based methods* fine-tune on panoramic data: MVDiffusion [1908] generates eight perspective views with consistent geometry and reprojects them into a closed-loop panorama, and TanDiT [1909] decomposes the panorama into a structured grid of tangent-plane images that a Diffusion Transformer generates jointly in one denoising pass under spatial attention.

Panoramic Video Generation. Extending synthesis over time adds temporal coherence to omnidirectional consistency. Formally, this family extends the static generator of Eq. (8.26) over time, sampling a sequence

under a joint spatial-and-temporal coherence constraint,

$$\{\hat{P}_t\}_{t=1}^{T} \sim p_\theta^{\text{pano}}\big(\{P_t\}_{t=1}^{T} \mid \kappa(\mathbf{c})\big), \qquad \text{s.t. } \hat{P}_t \approx \mathcal{W}^{\text{t}}(\hat{P}_{t-1}), \tag{8.28}$$

where $\mathcal{W}^{\text{t}}$ enforces frame-to-frame motion consistency on top of the per-frame 360-degree wrap-around continuity; Figure 8.10(b) depicts this temporally-extended pathway. The methods share this generator but differ in the conditioning argument $\mathbf{c}$ supplied to Eq. (8.28), which separates them into three input modalities,

$$\mathbf{c} = \begin{cases} \mathbf{x} & \text{(text-to-video)}, \\ P_0 & \text{(image-to-video)}, \\ \{(I_t, \mathcal{M}_t)\}_{t=1}^{T} & \text{(perspective-to-video)}, \end{cases} \tag{8.29}$$

where $\mathbf{x}$ is a text prompt, P_0 a static panorama anchor reused as the first frame, and $(I_t, \mathcal{M}_t)$ an NFoV frame paired with its visibility mask. The difficulty grows down the list: as the conditioning becomes spatially sparser, the model must hallucinate progressively more of the sphere while preserving the temporal coherence of Eq. (8.28).

▶ *Text-to-panoramic video generation.* 360DVD [1910] adapts a pretrained text-to-video model to the ERP domain through adapter modules, keeping wrap-around consistency across frames but struggling to balance text alignment against spherical geometry and to vary style under several control conditions. TiP4GEN [1911] adopts a dual-branch panoramic-and-perspective architecture in the spirit of PanFusion and lifts the generated video into a geometry-aligned 4D Gaussian scene. PanoWan [1912] and PanoDiT [1913] fine-tune large video diffusion transformers on panoramic video data and gain motion quality from scale, and PanoLora [1914] treats the task as a projection adaptation from perspective views that LoRA can learn with little data. Training-free alternatives carry the static-panorama machinery into time: SphereDiff [1907] extends its spherical latent to video, and DynamicScaler [1915] denoises through a seamless rotating window with an offset-shifting scheme and Global Motion Guidance, which yields arbitrary output resolution at constant memory. Text alone, however, offers little control over layout and motion, which motivates the richer conditioning below.

▶ *Image-to-panoramic video generation.* Animating a static panorama anchors the video spatially and asks for plausible omnidirectional motion that respects the input geometry. 4K4DGen [1916] animates user-selected regions with an image-to-video model under overlapping windows, at the cost of fixed-window artifacts and limited motion. ViewPoint [1917] introduces a panorama representation that keeps global spatial continuity while Pano-Perspective attention transfers a perspective video prior, and HoloTime [1918] reconstructs the result as 4D Gaussian Splatting so that the animated scene can be explored under viewpoint changes. GenEx [1919] generates a 3D-consistent imaginative environment from a single image as a panoramic video stream with loop consistency over long trajectories, which makes it usable for embodied exploration. For explicit control, PanFlow [1920] conditions on synthetic optical flow as a scene-agnostic motion signal and warps noise spherically so that camera and object motion can be controlled separately while the seam stays consistent; OmniDrag [1921] adds drag-style interaction at scene and object level through a spherical motion estimator; and PanoWorld-X [1922] reprojects ERP features onto the sphere inside a Sphere-Aware Diffusion Transformer to follow diverse camera trajectories.

▶ *Perspective-to-panoramic video generation.* Expanding an NFoV video to a full 360-degree video requires coherent hallucination in space and time at once, under visibility masks that change with camera motion. VidPanos [1923] frames the task as space-time outpainting and uses a generative video model to fill the regions a panning camera never observed; Imagine360 [1924] combines a global panoramic branch with local perspective priors and an elevation-aware design for the moving masks; and Beyond the Frame [1925] adds geometry-aware modules and motion-aware conditioning to improve realism and motion consistency in the outpainted regions.

Panoramic 3D Reconstruction and Generation. Reconstruction bypasses the generator of Eq. (8.25): rather than generating $\hat{P}$, it takes *observed* posed panoramas and applies the lifting operator directly,

$$\hat{\mathbf{r}}^{\text{rec}} = \Lambda\big(\{P_v, \pi_v\}_{v=1}^{V}\big), \tag{8.30}$$

where Λ is realized either as a per-scene optimization (iterative) or as an amortized network (feed-forward), mirroring the split of §8.2.3; Figure 8.10(c) depicts this reconstruction pathway. Generation then composes the two directions, feeding a synthesized panorama into the same Λ.

▶ *Iterative optimization-based panoramic reconstruction.* Realizing Λ of Eq. (8.30) as a *per-scene* optimization,

$$\hat{\mathbf{r}}^{\text{rec}} = \arg\min_{\mathbf{r}} \sum_{v} \big\|\text{Render}(\mathbf{r}, \pi_v) - P_v\big\|^2 + \lambda\, \Omega(\mathbf{r}), \tag{8.31}$$

re-solved from scratch for each scene, these methods fit a radiance field or an explicit representation to multi-view panoramas, the panoramic analogue of §8.2.3.2. NeRF-based work includes 360Roam [1926] for geometry-aware indoor reconstruction, 360FusionNeRF [1927] for novel views from a single panorama under geometric and semantic supervision, EgoNeRF [1928] with spherical grids for egocentric capture, and OmniNeRF [1929], which processes raw dual-fisheye input and its lens distortion. Gaussian-splatting alternatives are faster: 360GS [1930] projects Gaussians onto spherical tangent planes, ODGS [1931] rasterizes on tangent planes with elevation-aware densification, OP43DGS [1932] corrects the projection bias of the local affine approximation across camera models, OmniGS [1933] exploits omnidirectional optical properties, and Seam360GS [1934] calibrates dual-fisheye input by jointly optimizing translation vectors and learnable angular-distortion embeddings. Scene4U [1935] reconstructs from a single panorama in depth layers: an open-vocabulary segmenter and an LLM decompose it semantically, diffusion inpainting restores occluded regions, and 3D Gaussians are fitted per layer.

▶ *Feed-forward generalization-based panoramic reconstruction.* Amortizing Λ of Eq. (8.30) into a trained network applied in a single forward pass,

$$\hat{\mathbf{r}}^{\text{rec}} = \Lambda_{\psi}\big(\{P_v, \pi_v\}_{v=1}^{V}\big), \tag{8.32}$$

where Λ_{ψ} is trained once over a panoramic corpus (the analogue of §8.2.3.2), trades the per-scene cost of Eq. (8.31) for a learned prior that reconstructs from sparse panoramas without scene-specific training. PanoGRF [1936] pioneered feed-forward reconstruction of wide-baseline panoramas with depth guidance; PanoSplatt3R [1937] extends DUSt3R [122] to unposed wide-baseline panoramas through RoPE rolling, a rotary positional embedding rolled across attention heads so that ERP periodicity and correspondences across the seam are modeled; and PanSplat [1938] builds spherical 3D Gaussian pyramids on a Fibonacci lattice to reach 4K novel views with feed-forward inference and cubemap rendering for memory efficiency.

▶ *Panorama-based 3D generation and world modeling.* This family realizes the full composition of Eq. (8.25): it first *generates* the panorama and then lifts it,

$$\hat{\mathbf{r}}^{\text{rec}} = \Lambda\big(\hat{P}\big), \qquad \hat{P} \sim p_{\theta}^{\text{pano}}\big(P \mid \kappa(\mathbf{c})\big), \tag{8.33}$$

so that the synthesized omnidirectional image of Eq. (8.26) becomes the structured intermediate fed to the reconstruction operator of Eq. (8.30); Figure 8.10(d) depicts this generate-then-lift pathway. Text $\rightarrow$ panorama $\rightarrow$ 3D pipelines generate the omnidirectional image with a diffusion model [1890, 1901] before reconstructing a NeRF, Gaussian, or mesh scene. LayerPano3D [1939] parses the panorama into hierarchical depth layers and reconstructs each with 3D Gaussian Splatting, DreamScene360 [1940] fits panoramic Gaussians for free-roaming text-conditioned scenes, Matrix-3D [1941] couples trajectory-guided panoramic video generation with a dual-pathway reconstruction that conditions on scene-mesh renders, and HunyuanWorld [1942] treats the panorama as a 360-degree world proxy for text- and image-conditioned explorable scenes with mesh export and disentangled object representations. The panoramic video generators GenEx [1919] and

Table 8.5: Representative training corpora for 3D panorama synthesis, grouped by primary modality. "Modality" indicates the data type and synthetic / real acquisition. Structured3D and SUN360 scales are those of the splits used by recent panorama generators (SUN360 in full has 67,583 panoramas).

Dataset	Year	Scale	Modality	Primary use
Structured3D [1943]	2020	∼21K panoramas	Synthetic image	Indoor synthesis; distortion / seam benchmark
SUN360 [1944]	2012	∼29K panoramas	Real image	Real-scene generation benchmark
Matterport3D [340]	2017	10,800 RGB-D panoramas	Real image + depth	Generation and reconstruction supervision
Upright360 [1939]	2024	∼9.4K panoramas	Mixed image	Geometrically upright indoor synthesis
360+x [1946]	2024	2,152 videos	Real multimodal video	Panoptic scene understanding
WEB360 [1910]	2024	∼2K videos	Real captioned video	Text-to-panoramic video
360-1M [1876]	2024	> 1M videos	Real video	Large-scale world-model pretraining
PANOVID [1912]	2025	Aggregated	Real video	Text-to-panoramic video training
Matrix-Pano [1941]	2025	116,759 sequences	Synthetic video + depth + trajectory	Trajectory-conditioned generation
Replica [339]	2019	18 scenes	Synthetic (rendered scans)	Indoor reconstruction benchmark
Residential [1947]	2022	Indoor scenes	Synthetic image	Indoor reconstruction benchmark
360Loc [1948]	2024	18 posed sequences	Real reconstruction	4K reconstruction fine-tuning

PanoWorld-X [1922] (above) turn the same intermediate into an action- or trajectory-conditioned observation space, which is where panorama synthesis meets the world models of §8.6.

8.2.4.3 Benchmarks

Panoramic datasets must supply geometric consistency over the full sphere, a declared projection format, and, for reconstruction, camera poses and depth; Table 8.5 lists the corpora in use, grouped by modality.

Training Corpora. For static synthesis, Structured3D [1943] (synthetic indoor scenes with controlled geometry) and SUN360 [1944] (real-world indoor and outdoor panoramas) are the standard benchmarks for distortion handling and seam artifacts, in the splits adopted by JoPano, PanoDecouple, and DreamCube [1903, 1900, 1902]; Matterport3D [340], with RGB-D panoramas, camera poses, and mesh geometry across 90 buildings, is the most widely used corpus for both generation and reconstruction [1938, 1939, 1942, 1908, 1896], and Upright360 [1939] filters Matterport3D, web, and Blockade Labs panoramas with GeoCalib [1945] to near-zero pitch and roll for realistic indoor synthesis. Panoramic video with geometric annotation has been the bottleneck. 360+x [1946] and WEB360 [1910] are small real-world collections (the latter captioned for text conditioning), 360-1M [1876] scales to over a million videos for world-model pretraining, PanFlow [1920] adds optical flow and camera poses on motion-rich sequences, PANOVID [1912] aggregates these with Imagine360 [1924] and public VR videos but lacks poses and depth, and Matrix-Pano [1941] fills that gap with Unreal Engine 5 renders carrying depth maps and camera trajectories; embodied work renders action-conditioned observations in Unreal Engine directly [1919, 1922]. For reconstruction, Replica [339] and Residential [1947] serve as synthetic indoor benchmarks at baselines of about 1.0 m and 0.3 m respectively [1938], 360Loc [1948] supplies posed panorama sequences used for 4K fine-tuning [1938], and the scarcity of posed real panoramic video led PanSplat [1938] to capture its own dual-fisheye footage with poses from visual SLAM.

Evaluation Metrics. Panorama evaluation combines general image and video quality metrics with panorama-specific measures of distortion, seam consistency, and spherical topology.

► *Image fidelity and semantic alignment.* FID measures distributional similarity between generated and real panoramas [1903, 1939, 1900, 1909]; PanFusion [1896] shows that FAED (Fréchet Auto-Encoder Distance), computed in the latent space of an auto-encoder trained on the target data, tracks panorama quality more closely than Inception features do. Text-to-panorama generation reports CLIP-Score between prompt and output [1942, 1903, 1939, 1909], image-conditioned tasks report CLIP-I between input and output [1942],

and no-reference quality is scored with BRISQUE and NIQE [1942, 1939, 1941], complemented by aesthetic scorers and human preference studies, for instance the Average User Ranking protocol of LayerPano3D [1939]. Layout-conditioned synthesis measures adherence by 3D and 2D IoU against layouts recovered with HorizonNet [864], following PanFusion [1896].

▶ *Panorama-specific consistency.* Projection-induced distortion and boundary discontinuities call for metrics of their own. JoPano [1903] scores cubemap seams with Seam-SSIM (SSIM between adjacent edge pixels of neighboring faces) and Seam-Sobel (gradient discontinuity along the seam); PanoDecouple [1900] introduces Distort-CLIP, a CLIP variant trained to detect panoramic distortion, so that the "visual cheating" of models that improve image quality while degrading geometry becomes measurable; and loop consistency across the 360-degree wrap-around is assessed by PSNR in overlapping regions [1908] or across the left–right boundary [1920, 1912].

▶ *Reconstruction and novel-view synthesis.* Because pixel density on the sphere is non-uniform, WS-PSNR (Weighted Spherical PSNR) weights pixels by latitude and replaces plain PSNR for equirectangular outputs [1938, 1936], complemented by SSIM and LPIPS [1939, 1938, 1941].

▶ *Video quality and spatiotemporal dynamics.* Panoramic videos are often projected to several perspective views (*e.g.*, four views at azimuths $\{0°, 90°, 180°, 270°\}$ with a $90°$ field of view) so that human raters and perspective-domain tools can be used [1942, 1939, 1924]. Q-Align [1949], a large multimodal model trained to score visual quality on discrete text-defined levels, is increasingly used for images and videos alike [1942, 1924]; FVD [1950], computed in the feature space of an inflated 3D ConvNet, is the standard distributional metric for video [1920, 1912]; VBench [1951] adds imaging quality, aesthetic quality, motion smoothness, and subject consistency over projected views; and trajectory-conditioned generation is scored by frame-to-frame consistency and by the range of camera motion it sustains [1922]. Two limits remain. Ground truth is scarce for dynamic panoramic video with depth and camera annotation, so evaluation falls back on projected perspective views or user studies; and world-level properties such as loop consistency over long trajectories and action-conditioned spatial coherence have no dedicated benchmark yet.

8.2.4.4 Challenges and Trends

Unique Challenges in Panoramic Synthesis. Modeling a complete omnidirectional environment imposes constraints that NFoV synthesis never meets. The first is **distortion-aware learning**: ERP stretches content near the poles, so visual information is non-uniformly distributed over the image and convolutional kernels designed for translation invariance on the plane struggle with the varying distortion, which motivates spherical manifold convolutions and latitude-aware architectures [1891, 1900]. The second is **seamless boundary consistency**: wrap-around coherence across the longitudinal boundary must be preserved to avoid visible seams, typically through circular padding or latent rotation during denoising [1920, 1912]. The third is **scaling structured generation** to 4K and beyond while keeping global geometric and semantic consistency over the whole sphere, which pushes toward memory-efficient attention and hierarchical spherical representations [1938, 1915].

From Panorama to World Models. The clearest trend is the move from static spherical snapshots toward **interactive 3D world modeling**, in which the panorama is the interface through which an embodied agent reasons about a complete environment rather than a fragment of one. The systems that lift a generated panorama into an explorable world (HunyuanWorld [1942], Matrix-3D [1941]) and World Labs' Marble [158] show what panoramic priors make possible, and **temporal panoramic modeling** supplies the 4D half: generating temporally coherent panoramic video requires motion dynamics that respect spherical topology and occlusion boundaries [1920, 1922]. Integrated with interactive systems, such panoramic world simulations become a basis for embodied interaction, long-range spatial reasoning, and environment simulation in robotics and immersive media.

Representation Innovations. Efficiency and fidelity both hinge on the omnidirectional representation. **Efficient omnidirectional representations** such as the Fibonacci-lattice Gaussian pyramids of PanSplat [1938] render at high fidelity without the redundancy of uniform grids, and **multi-modal panoramic fusion**, which binds 360-degree RGB to depth, semantic labels, and geometric proxies such as surface normals [1939, 1935], makes the synthesized environment both more realistic and more editable, closing part of the gap between 2D generative priors and full 3D scene intelligence.

8.2.5 3D Scene Editing

The preceding subsections build 3D content; 3D scene editing takes an existing asset and modifies it under user control while preserving everything the edit should leave untouched. Such incremental control is exactly what spatial intelligence asks of a 3D world model: an agent that must correct a mistake, personalize an environment, or keep a scene consistent as its task unfolds cannot afford to regenerate the world from scratch every time it changes. Technically, 3D editing lifts the 2D inpainting-and-editing machinery of §8.1.3 from the image plane to a multi-view-consistent 3D representation, adding a cross-view coherence requirement that has no analogue in the 2D case; the five method families below are distinguished by how the edit signal is injected into that representation.

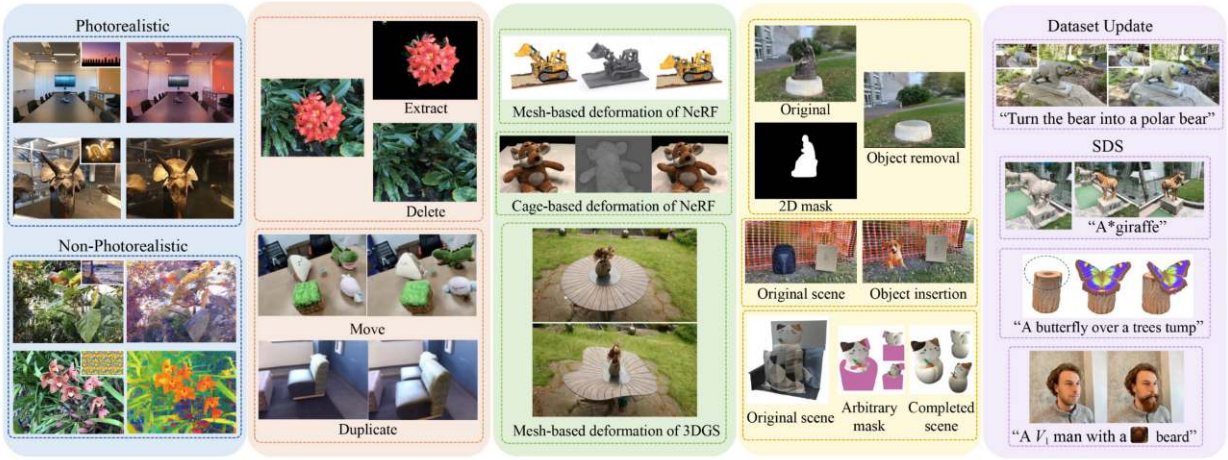


Figure 8.11: Demonstrations of 3D editing tasks. (a) appearance editing, (b) object transformation, (c) shape deformation, (d) scene inpainting, and (e) creative editing. Figure adapted from [1952].

8.2.5.1 Task Definition

3D scene (and object) editing modifies an existing 3D asset or reconstructed scene according to a user intent (typically language or reference images, as in Figure 8.11) while preserving 3D consistency for novel-view rendering. Unlike text-to-3D generation, editing operates on a given representation (*e.g.*, NeRF, 3D Gaussian Splatting, or meshes) and requires (i) locality and identity preservation for unedited regions, (ii) multi-view (and potentially temporal) coherence, and (iii) controllability over the edited attributes, whether geometry, appearance, material, or semantics.

Formally, 3D editing is the 3D-lifted counterpart of the 2D inpainting-and-editing operator of §8.1.3 (Eq. (8.8)). Given an input 3D representation $\mathbf{r}^{\text{in}} \in \mathcal{R}$ drawn from the same representation space $\mathcal{R}$ as §8.2.1, an edit intent $\mathbf{c} \in \mathcal{C} \supseteq \{\text{text instruction}, \text{reference image}, \text{mask / region}, \text{scribble}\}$, and an optional 3D region of interest $\mathcal{M} \subseteq \mathbb{R}^3$, the goal is an editing operator

$$E_\theta : \mathcal{R} \times \mathcal{C}' \to \mathcal{R}, \qquad \hat{\mathbf{r}} = E_\theta\big(\mathbf{r}^{\text{in}}, \kappa(\mathbf{c})\big), \tag{8.34}$$

trained or optimized to minimize an objective that couples three competing terms,

$$\mathcal{L}_{\text{edit}} = \underbrace{\mathcal{L}_{\text{intent}}\big(\hat{\mathbf{r}}, \mathbf{c}; \mathcal{M}\big)}_{\text{edit-region adherence}} + \lambda_{\text{p}} \underbrace{\mathcal{L}_{\text{preserve}}\big(\hat{\mathbf{r}}, \mathbf{r}^{\text{in}}; \mathbb{R}^3 \setminus \mathcal{M}\big)}_{\text{identity preservation}} + \lambda_{\text{c}} \underbrace{\mathcal{L}_{\text{consist}}\big(\{\text{Render}(\hat{\mathbf{r}}, \pi_v)\}_v\big)}_{\text{multi-view coherence}}, \tag{8.35}$$

where κ is the condition encoder of §8.2.1, Render is the differentiable renderer of §8.2.3, and $\lambda_{\text{p}}, \lambda_{\text{c}}$ trade off preservation and consistency against edit strength. The limiting case that anchors Eq. (8.34) to its 2D origin is instructive: when $\mathcal{R}$ collapses to a single image ($V = 1$, no rendering), E_θ reduces exactly to the 2D editing operator of Eq. (8.8) and the multi-view-consistency term vanishes, so 3D editing is 2D editing plus a cross-view coherence constraint.

8.2.5.2 Methodology

Recent 3D object and scene editing methods can be categorized by *how* the edit signal of Eq. (8.34) is injected into the 3D representation, from lifting 2D edits view by view, through enforcing cross-view consistency, to operating directly on 3D primitives, reasoning with language models, and amortizing the edit into a single feed-forward pass. Figure 8.12 schematizes the five families and how each realizes the operator E_θ.

2D-lifted Iterative Editing. A dominant paradigm inherits from instruction-based NeRF editing [1953, 1954, 1955, 1956]: the method iterates between rendering views from the current scene, applying a 2D diffusion-based editor to them, and updating the 3D representation to match the edited guidance under geometric regularization. Formally, this family instantiates E_θ of Eq. (8.34) as a *per-scene* optimization that drives the rendered views toward the output of a *frozen* 2D editor $\mathcal{E}_{\text{2D}}$ (the InstructPix2Pix-style operator of §8.1.3),

$$\hat{\mathbf{r}} = \arg\min_{\mathbf{r} \in \mathcal{R}} \sum_v \big\| \text{Render}(\mathbf{r}, \pi_v) - \mathcal{E}_{\text{2D}}\big(\text{Render}(\mathbf{r}^{\text{in}}, \pi_v),\ \mathbf{c}\big) \big\|^2 + \lambda\, \Omega(\mathbf{r}), \tag{8.36}$$

so that the 2D edits are *lifted* into the 3D representation and the $\arg\min$ is re-solved from scratch for every scene; Figure 8.12(a) depicts this render-edit-update loop. Instruct-NeRF2NeRF [1957], EditNeRF [1958], Free-Editor [1959], and CTRL-D [1960] lift instruction-driven 2D edits into the radiance field this way, and with the rise of 3D Gaussian Splatting (3DGS) [121] the same principle moved to Gaussian primitives: Instruct-GS2GS [1961] iteratively edits the training images with InstructPix2Pix [1711] and re-optimizes the 3DGS to reflect the instruction while preserving 3D structure [1962, 1963]. The recipe is simple and borrows the full strength of 2D editors, but it is slow, prone to view-dependent artifacts, and only as good as the stability of the 2D editor and the coverage of the rendered guidance views.

Multi-view-consistent Guidance. Because independently edited views can disagree (inconsistent textures, hallucinated geometry), recent work [1964] enforces multi-view consistency at the *guidance* level rather than relying on the 3D update to reconcile conflicting edits. Formally, rather than editing each view independently as in Eq. (8.36), this family makes the guidance images mutually consistent before the lift,

$$\{\hat{I}_v\} = \arg\min_{\{I_v\}} \sum_v \mathcal{L}_{\text{intent}}(I_v, \mathbf{c}) \ + \ \beta \sum_{(u,v)} \big\| I_u - \mathcal{W}_{u \to v}(I_v) \big\|^2, \tag{8.37}$$

where $\mathcal{W}_{u \to v}$ warps view v into view u using geometry or depth derived from the current 3D estimate, penalizing cross-view disagreement; the consolidated $\{\hat{I}_v\}$ then drive the 3D update of Eq. (8.36), as in Figure 8.12(b). GaussCtrl [1965] edits several rendered views in a coupled fashion rather than sequentially, improving multi-view consistency for text-driven 3DGS editing [1962, 1966], and VcEdit [1962] integrates the 3DGS into the image-editing process itself with consistency modules that counter cross-view disagreement and mode collapse. A complementary line transfers a *single-view* edit to other views via geometry: Gomel and Wolf [1967] extracts diffusion attention features from a reference edit, warps them across views using depth from the Gaussian splats, and injects the warped features to propagate a coherent edit. These methods

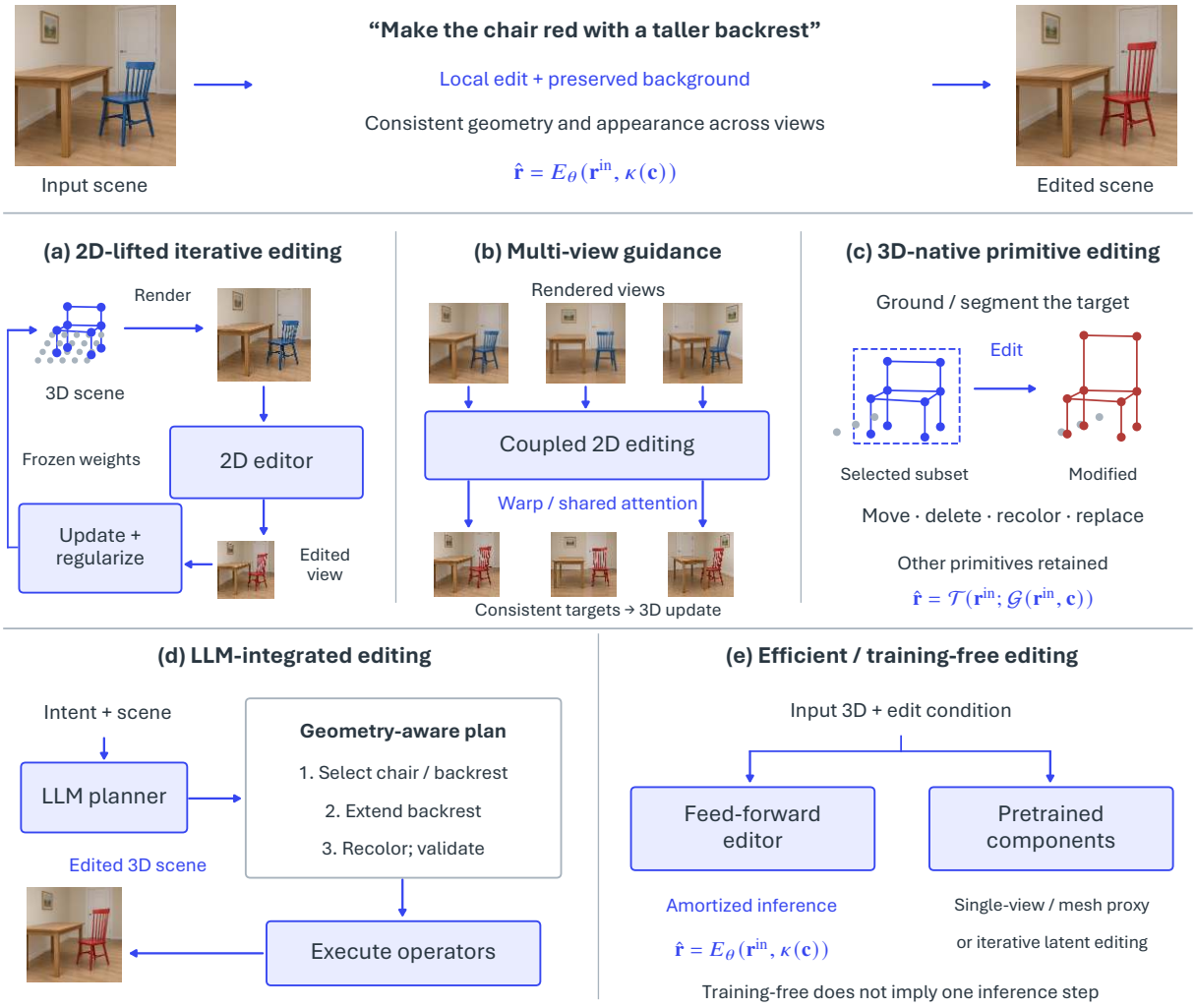


Figure 8.12: Five method families for 3D scene editing: (a) 2D-lifted iterative editing, (b) multi-view-consistent guidance, (c) 3D-native primitive editing, (d) LLM-integrated editing, and (e) efficient and training-free editing. The chair edit (top) is illustrative, not a measured result.

preserve 3D coherence more robustly but still struggle with large non-rigid shape changes or severe occlusion, where reliable correspondence and geometry are hard to come by.

3D-native Primitive Editing. To reduce reliance on unstable 2D guidance and enable interactive use, 3D-native approaches operate directly on explicit scene primitives. Formally, this family bypasses the rendering loop of Eq. (8.36) entirely: a grounding or segmentation operator $\mathcal{G}$ selects the target subset of primitives and a direct geometric transform $\mathcal{T}$ applies the edit,

$$\hat{\mathbf{r}} = \mathcal{T}\big(\mathbf{r}^{\text{in}};\, \mathcal{G}(\mathbf{r}^{\text{in}}, \mathbf{c})\big), \qquad \mathcal{T} \in \{\text{add, delete, move, recolor, replace}\}, \tag{8.38}$$

acting on explicit Gaussians or mesh facets so that the unedited primitives are preserved *by construction*: the identity-preservation term of Eq. (8.35) is satisfied structurally rather than by a penalty; Figure 8.12(c) shows this pathway. 3DSceneEditor [1968] runs a fully 3D pipeline, combining instance segmentation and zero-shot grounding to localize the target and then performing Gaussian-level addition, deletion, relocation, recoloring, and replacement with strong controllability in real time, and InterGSEdit [1969] adds user-in-the-loop key-view selection and a geometry-consistent attention prior for iterative, preference-guided 3DGS editing

with control over the edit degree. More broadly, GaussianEditor and 3DSceneEditor [1970, 1968] impose higher-level structure on the Gaussians, grouping them by object identity or semantic coherence (typically through zero-shot grounding or segmentation [1971, 1972] followed by anchoring) so that an edit can be formulated at the level of an instance rather than of individual primitives, which localizes the target more precisely and protects the background.

LLM-integrated Editing. Large language models have been integrated into 3D editing pipelines not merely as text encoders but as high-level reasoning and control modules. Formally, the LLM decomposes the instruction into an ordered program of geometry-aware operators applied to the asset,

$$\hat{\mathbf{r}} = (o_K \circ \cdots \circ o_1)(\mathbf{r}^{\text{in}}), \qquad (o_1, \ldots, o_K) = \text{LLM}\big(\mathbf{c},\, \mathbf{r}^{\text{in}}\big), \tag{8.39}$$

so that high-level reasoning (attribute inference, edit planning, dependency resolution) is separated from the low-level operators o_k, often the primitive transforms of Eq. (8.38), that it dispatches; Figure 8.12(d) depicts this plan-then-execute pathway. ShapeLLM-Omni [1973] casts 3D editing as language-grounded reasoning in which the LLM performs shape understanding, attribute inference, and edit planning before invoking geometry-aware operators, and LL3M [1974] extends this to multimodal 3D manipulation requiring compositional understanding of geometry, appearance, and spatial relations. A complementary line [1975] uses the LLM to guide or parameterize image-to-3D or text-to-3D backbones in a feed-forward setting rather than a per-scene optimization, and LLMs also serve as coordination agents for global consistency [1976, 1614]: Xia et al. [1966] shows that reasoning over object identities, relations, and edit dependencies reduces the conflicts that arise in purely diffusion- or optimization-based pipelines when several objects and attributes must be edited at once.

Efficient and Training-free Editing. A final trend reduces test-time cost through feed-forward reconstruction or reuse of pretrained models [1977, 1978]. These strategies are distinct: a training-free editing pipeline can still involve iterative diffusion inference. The amortized feed-forward case evaluates the editing operator of Eq. (8.34) directly,

$$\hat{\mathbf{r}} = E_\theta\big(\mathbf{r}^{\text{in}},\, \kappa(\mathbf{c})\big) \quad \text{(amortized, no per-scene optimization)}, \tag{8.40}$$

eliminating the per-scene $\arg\min$ of Eq. (8.36) by amortizing inference into a trained network. Training-free alternatives instead reuse pretrained components, potentially through single-view guidance, mesh proxies, or iterative native-3D latent editing; Figure 8.12(e) distinguishes these routes from the per-scene optimization loop in panel (a). Free-Editor [1959] edits a NeRF zero-shot from text without scene-specific retraining, countering multi-view style inconsistency with a "single-view editing" scheme and an Edit Transformer; on the scene-operation side (insert, replace, delete), FreeEdit [1979] uses a mesh proxy to perform training-free edits guided by foundation models at near real-time rates in complex indoor scenes. These methods improve usability but typically trade away edit fidelity and coverage, especially for fine-grained geometry, and depend strongly on robust grounding and localization in 3D.

8.2.5.3 Benchmarks

Several benchmarks have recently been introduced for systematic evaluation. 3D-Alpaca-Editing [1973] focuses on instruction-following 3D editing, assessing whether a model interprets natural-language commands correctly while preserving unedited content. CMD [1980] emphasizes compositional and multi-step instructions, where reasoning and planning matter. Edit3D-Bench [1978] is a general-purpose suite covering geometry, appearance, and object-level operations, with 3D edit regions annotated on 100 source models (300 edits), which allows fair comparison across editing paradigms. 3DEditVerse [1966] is a large-scale paired benchmark built for robustness and generalization across varied objects, scenes, and intents. Together they standardize evaluation in terms of instruction compliance, edit locality, and scalability; Table 8.6 summarizes their scale and the evaluation axes each one annotates.

Table 8.6: Representative 3D editing benchmarks. "Edit region" indicates whether a localized edit mask / region is annotated; "Consistency" and "Harmony" indicate whether multi-view consistency and edit harmonization are evaluated. The criteria follow the four requirements set out with 3DEditVerse [1966].

Dataset	Training size	Testing size	Edit region	Train / test split	Consistency	Harmony
3D-Alpaca-Editing [1973]	∼70,000	–	✓	✗	✗	✓
CMD [1980]	∼40,000	50	✗	✓	✓	✗
Edit3D-Bench [1978]	–	300	✓	✗	✓	✓
3DEditVerse [1966]	116,309	1,500	✓	✓	✓	✓

Evaluation Metrics. Evaluation considers geometric fidelity, appearance quality, and semantic alignment with the instruction. Geometry is measured by Chamfer Distance (CD) [1975], lower being better, and by F-score [1966], which captures surface overlap at a distance threshold. Texture and appearance are assessed by comparing renderings of the edited representation against target or inpainted images with SSIM [1884], LPIPS [1885], and DreamSim [1981] [1982]. Visual–text alignment is scored with CLIP- or DINO-based embedding similarity [1983], and GPTEval3D [212] uses GPT-4V to compare multi-view renderings of edited shapes on text alignment, 3D plausibility, and texture detail. To measure the edit relative to the input rather than in isolation, directional CLIP (CLIP_{dir}) [1963] checks that the change in image embedding, computed over multi-view renderings of input and output, follows the change in text embedding.

Edit cost is the clearest axis separating the five families. Instruct-NeRF2NeRF [1957] takes on the order of 30–90 minutes per edit; GaussianEditor [1970] brings this under 20 minutes by operating on explicit Gaussians; PrEditor3D [1977] edits a shape in a few minutes, and VoxHammer [1978] performs precise local edits training-free in a native 3D latent space. On quality the field reports DINO-I and CLIP-T on Edit3D-Bench [1978], directional CLIP [1963], and GPT-4V judgment [212]; the 3D-native and feed-forward families lead on locality and speed, while 2D-lifted methods keep an edge on open-vocabulary appearance edits where a strong 2D editor does the work. The speed of feed-forward editing comes at a cost in fine-grained geometry control and a dependence on robust 3D grounding, taken up below.

8.2.5.4 Challenges and Trends

Several problems remain genuinely open and define the current frontier of 3D editing.

Geometry-level, not merely appearance, editing. Most current methods edit color and texture well but treat geometry as fixed: precise, instruction-driven changes to shape, including thin, transparent, or high-frequency structures, remain unreliable, and edits that should alter the underlying surface often only repaint it. Closing this gap requires editors that operate on geometry and appearance jointly rather than deferring geometry to a frozen reconstruction.

Large non-rigid and structural edits. Adding, removing, or substantially deforming structure while preserving the identity of the rest of the asset is still difficult: drag-driven, as-rigid-as-possible Gaussian editors [1984] handle moderate deformation, but large non-rigid motion and topology changes frequently lose original detail. Robust structural and compositional editing, including relation-aware multi-object edits, remains an active frontier.

Material–lighting disentanglement and relightable editing. Because most representations bake illumination into appearance, an edit that changes an object's material or color is not guaranteed to stay consistent when the scene is relit. Relightable representations that decompose geometry, BRDF, and incident light [1985] are a prerequisite for edits that survive relighting and for inserting edited content into new lighting, yet stable material–light decomposition under editing is not solved.

The consistency bottleneck has moved to 3D grounding. Now that native-3D editing handles appearance coherence, the limiting factor is localization: methods such as VoxHammer [1978] are highly sensitive to 3D

mask accuracy and degrade rapidly when the target region is imprecisely grounded. Robust open-vocabulary 3D grounding and segmentation [1971, 1972], ideally without user-provided masks [1966], is therefore the new gate on edit quality, and it is most acute in dynamic scenes, where the edited region must be tracked with temporal identity preserved across frames [1960].

Two pressures cut across all four. Evaluation remains immature: the benchmarks of §8.2.5.3 are small (Edit3D-Bench annotates 100 source models and 300 edits) and no metric panel is yet standard, so progress on the harder edits above is difficult to measure reliably. And the strongest forward trend is unification, bringing generation, segmentation, grounding, and editing into a single structured pipeline [1986] with interfaces that go beyond text to reference images [1983], masks [1987], sketches [1955], and object-level operations, so that an agent can specify and verify a geometry-, material-, and relighting-consistent edit rather than only a repaint.

8.3 Video Spatial Generation



IDEO spatial generation is the task of synthesizing a temporally coherent image sequence (a video clip) from a high-level prompt together with optional spatial, temporal, and other conditioning signals. It has advanced substantially in recent years, evolving from early generative adversarial network (GAN)-based approaches [1646] through autoregressive token models to the diffusion- and diffusion-transformer-based systems that now dominate the field [1988, 1989, 1990, 1991]. This rapid progress has stimulated a wide range of downstream tasks, extending beyond basic text-to-video generation to encompass fine-grained control mechanisms (spatial, *e.g.*, image, pose, depth, landmark, sketch, and bounding box; temporal, *e.g.*, optical flow, trajectory, camera, and motion; and other conditional controls, *e.g.*, style, audio, bird's-eye view) as well as sophisticated video editing capabilities.

Within the arc of §8, video spatial generation is the *temporal* extension of the priors built in the preceding two sections. Whereas §8.1 learned a 2D semantic-to-pixel prior on a single image plane and §8.2 lifted that prior into a static volumetric representation, this section adds the time axis: the model must place matter and appearance correctly *and* make them evolve plausibly as the camera or the scene itself moves through time. Appearance, geometry, and dynamics must now be *jointly* consistent: a generated clip is judged not only frame by frame but on object permanence, motion plausibility, and multi-view coherence under camera motion, which makes video the natural substrate for the dynamic world models and simulators that consume spatial structure downstream.

Formally, we cast every system in this section as a conditional spatio-temporal sampler that extends the latent-generator framework of §8.1.1. Let $\mathcal{V} = [0,1]^{T\times H\times W\times 3}$ be the space of T-frame clips and $\mathcal{Z}^{\text{vid}}$ a compressed spatio-temporal latent space supplied by a video autoencoder that extends the image autoencoder of Eq. (8.1) along the time axis. Writing ϕ for the text encoder of Eq. (8.2) and $\kappa : \mathcal{C} \to \mathcal{C}'$ for a generic encoder of an auxiliary control signal $\mathbf{c} \in \mathcal{C}$, the core learning problem is to train

$$G_\theta^{\text{vid}} : \mathcal{E} \times \mathcal{C}' \to \mathcal{Z}^{\text{vid}}, \qquad \hat{\mathbf{z}}^{1:T} \sim p_\theta\big(\mathbf{z}^{1:T} \,\big|\, \phi(\mathbf{x}),\, \kappa(\mathbf{c})\big), \tag{8.41}$$

so that the decoded clip $\hat{\mathbf{v}} = D(\hat{\mathbf{z}}^{1:T})$ is photorealistic, temporally coherent, and faithful to both the prompt $\mathbf{x}$ and the control $\mathbf{c}$. Two limiting cases anchor the framework: with no auxiliary control ($\kappa = \varnothing$) Eq. (8.41) reduces to text-to-video generation, and with a single frame ($T = 1$) it recovers the image generator of Eq. (8.2) exactly, so video spatial generation is precisely the image prior of §8.1 unrolled in time and equipped with a richer conditioning interface.

The subsections are organized by the axis of control each introduces into Eq. (8.41). Text-to-video generation (§8.3.1) is the base case without auxiliary control; spatial control (§8.3.2) adds a per-frame scaffold such as a pose or depth sequence; temporal control (§8.3.3) adds a motion or camera signal that governs dynamics rather than appearance; §8.3.4 covers the remaining modalities and the frameworks that fuse several at once;

and video editing (§8.3.5) modifies an existing clip, lifting the 2D editing of §8.1.3 into time. §8.3.6 compares the five and discusses the challenges they share.

8.3.1 Text-to-Video Generation

Text-to-video (T2V) generation [1991, 1988, 1989, 1990] synthesizes photorealistic, temporally coherent video from a natural-language description. It extends text-to-image (T2I) synthesis along time, so the model must produce plausible motion and state transitions as well as high-fidelity frames, and it does so with no explicit spatial scaffold: object permanence, motion plausibility, and view coherence under camera motion cannot be supplied from outside but must *emerge* from the generator itself. That is why the unconditional T2V model is the substrate on which the explicit control axes of §8.3.2–§8.3.5 are built. Its architecture has followed the image case, from generative adversarial networks (GANs) through autoregressive models to the diffusion models that dominate today, with progress driven as much by data as by architecture [1992, 1993].

8.3.1.1 Task Definition

Text-to-video generation is the base case of Eq. (8.41) with no auxiliary control ($\kappa = \varnothing$): the generator samples a latent from the prompt embedding alone,

$$\hat{\mathbf{z}}^{1:T} \sim p_\theta\big(\mathbf{z}^{1:T} \,\big|\, \phi(\mathbf{x})\big), \tag{8.42}$$

which is exactly the latent sampler G_θ of Eq. (8.2) unrolled over T frames in the spatio-temporal latent space $\mathcal{Z}^{\text{vid}}$.

8.3.1.2 Methodology

The three paradigms factorize the clip distribution $p_\theta(\mathbf{z}^{1:T} \mid \phi(\mathbf{x}))$ adversarially, causally over discrete tokens, or by iterative spatio-temporal denoising, and each changes its image counterpart of §8.1.1 in one structural respect; Figure 8.13 contrasts how each handles the temporal dimension.

Generative Adversarial Networks (GANs). The first T2V systems extended the image GANs of §8.1.1.2 to clips. The video-specific change is in the discriminator: rather than scoring a single image, it must judge both per-frame realism *and* the plausibility of motion across the sequence, which is naturally realized as a two-critic objective

$$\min_\theta \max_{\psi_s,\psi_t} \mathbb{E}_{\mathbf{v}\sim p_{\text{data}}}\big[\log D_{\psi_s}(\mathbf{v}) + \log D_{\psi_t}(\mathbf{v})\big] + \mathbb{E}_{\mathbf{n}}\big[\log\big(1 - D_{\psi_s}(\hat{\mathbf{v}})\big) + \log\big(1 - D_{\psi_t}(\hat{\mathbf{v}})\big)\big], \tag{8.43}$$

where $\hat{\mathbf{v}} = D\big(G_\theta(\mathbf{n}, \phi(\mathbf{x}))\big)$ is the decoded clip, the spatial critic D_{ψ_s} scores individual frames, and the temporal critic D_{ψ_t} scores motion coherence over the clip, the term absent from the single-discriminator image objective of Eq. (8.3). TGAN [1994] and TGAN-v2 [1995] used a temporal generator to produce latent motion vectors from which an image generator rendered frames; MoCoGAN [1996] and MoCoGAN-HD [1997] disentangled content from motion, generating the motion code recurrently; DVD-GAN [1998] instantiated Eq. (8.43) directly with separate spatial and temporal discriminators; and DIGAN [1999] used implicit neural representations for longer sequences. Even with architectures such as StyleGAN-V [2000], GANs suffered from flicker, training instability, and mode collapse, which limited the diversity of their motion and pushed the field toward other paradigms.

Autoregressive Models. Autoregressive (AR) models recast T2V as the sequential prediction of §8.1.1.2, tokenizing a clip with a 3D Vector Quantized Variational Autoencoder (3D VQ-VAE) and predicting each token from its predecessors with a large transformer. The video-specific change is the *order* of the factorization: where the image model of Eq. (8.4) runs over the spatial tokens of a single frame, the video model factorizes



Figure 8.13: Three mechanisms for text-to-video generation: (a) a GAN with spatial and temporal critics, a representative rather than universal design; (b) autoregressive prediction of discrete space–time tokens, with pale cells as generated context and the blue cell as the next token; and (c) diffusion with factorized spatial / temporal attention, where s is the sampling step and t is video time. The ball sequence is illustrative.

over space *and* time,

$$p_\theta\big(\mathbf{z}^{1:T} \,\big|\, \phi(\mathbf{x})\big) \;=\; \prod_{t=1}^{T} \prod_{n=1}^{N} p_\theta\big(z_{t,n} \,\big|\, \mathbf{z}^{<t},\, z_{t,<n},\, \phi(\mathbf{x})\big), \tag{8.44}$$

so that each frame is conditioned on all previously generated frames $\mathbf{z}^{<t}$. This inter-frame causal dependency is the source of both the paradigm's strength (global temporal coherence) and its characteristic failure, the accumulation of per-step error over long roll-outs. VideoGPT [27] demonstrated the approach for unconditional generation, NÜWA [2001] and CogVideo [1989] adapted it to text-to-video synthesis, and TATS [2002] addressed quality degradation in long sequences. The paradigm has since resurged in large multimodal backbones: Emu3 [2003] generates video inside a language-model backbone by treating frames as discrete visual tokens, and CausVid [2004] and MAGI-1 [2005] scale to long-form generation with autoregressive diffusion transformers. Sequential decoding nevertheless remains expensive and prone to the error accumulation that Eq. (8.44) makes explicit.

Diffusion Models. Diffusion models, which now dominate T2V, lift the latent-denoising objective of §8.1.1.2 from a single image to the clip latent $\mathbf{z}^{1:T}$. The forward and reverse processes are those of Eq. (8.5); the video-specific change is that the denoiser ϵ_θ operates on a spatio-temporal tensor and is trained to predict the

noise added to the whole clip at sampling step s,

$$\min_{\theta} \mathbb{E}_{s,\, \mathbf{z}_0^{1:T},\, \epsilon}\left[\left\|\epsilon - \epsilon_\theta\big(\mathbf{z}_s^{1:T},\, s,\, \phi(\mathbf{x})\big)\right\|_2^2\right], \tag{8.45}$$

where ϵ_θ models dependencies across both space and time, and s is distinct from the frame index t. One widely used design *factorizes* attention into spatial attention within each frame and temporal attention across frames, as illustrated in Figure 8.13(c); other architectures use joint spacetime attention. VDM [2006] applied the idea in pixel space with a 3D U-Net; the breakthrough was moving diffusion into the compressed latent space of a video-specific autoencoder, as in VideoLDM [2007], LVDM [2008], and the open-source Stable Video Diffusion (SVD) [1873], which cut training and inference cost enough to make high-resolution generation practical.

A second recipe adapts pretrained T2I models. Imagen Video [2009] and Make-A-Video [2010] freeze a T2I model and insert trainable temporal modules (*e.g.*, temporal attention layers or 1D convolutions) between its spatial layers, which are then trained on video to learn motion while the image prior is inherited intact. VideoCrafter [2011, 2012] and LaVie [2013] built high-quality generators on this recipe, Tune-A-Video [2014] adapted it to one-shot tuning on a single clip, and AnimateDiff [2015] turned the temporal modules into plug-and-play motion modules that animate personalized T2I models without further fine-tuning.

Diffusion Transformers (DiTs) have since replaced the U-Net backbone, operating on "spacetime patches" (unified tokens representing blocks of video) and scaling well. Sora [1991] demonstrated long-duration (up to one minute), high-resolution clips and presented the approach as a step toward world simulation, and a wave of DiT models followed, including Open-Sora [2016], LATTE [2017], CogVideoX [2018], StepVideo [2019], Movie Gen [1988], Wan [1990], and Veo [2020]. Large open DiT foundation models have substantially narrowed the gap to proprietary systems. HunyuanVideo [2021], at 13B parameters the largest open T2V model at release, couples a "dual-stream-to-single-stream" full-attention DiT (text and video tokens processed in separate streams before fusion) with a multimodal-LLM text encoder and a 3D causal VAE, and reports human-evaluated quality competitive with leading closed systems. Mochi 1 [2022] introduces an asymmetric DiT (AsymmDiT) that allocates most capacity to the visual stream, and LTX-Video [2023] uses a high-compression video VAE to run latent video diffusion in near real time. NVIDIA's Cosmos [156] reframes the video generator as a world-foundation model for "Physical AI", trained so that embodied and robotics systems can use it as a learned simulator, which ties video generation to the world models of §8.6. Across these shifts the T2V trajectory has trailed the T2I trajectory of §8.1.1 by roughly a year, each image paradigm being inflated along the time axis once it matured.

8.3.1.3 Benchmarks

Video adds a requirement at every level of evaluation: corpora must pair clips with captions at scale, suites must stratify by temporal phenomena (motion, object permanence, dynamic degree), and metrics must measure cross-frame coherence as well as per-frame quality.

Training Corpora. Caption–clip scale has been the principal driver of T2V quality, and high-quality paired video–text data is far scarcer than image–text data. Early work repurposed action-recognition corpora such as UCF-101 [902] and Kinetics [239], which supply short, class-labeled clips but no captions. WebVid-10M [2024], the first web-scale captioned corpus, underpinned most 2022–2023 systems despite noisy alt-text captions and watermarks; later corpora raised caption density and clip length, with Panda-70M [1992] captioned automatically by cross-modality teachers, MiraData [1993] targeting long clips with dense structured captions, and Koala-36M [2025] explicitly optimizing the consistency between caption and content, the property that most directly limits a generator's controllability. Table 8.7 gives their scale.

Evaluation Benchmark Suites. Classical T2V evaluation reported zero-shot distributional scores on the held-out splits of UCF-101 and MSR-VTT [2026], which probe frame quality and coarse text relevance but

Table 8.7: Representative training corpora for text-to-video generation. Action-recognition sets (top) supply class labels rather than captions; web-scale captioned corpora (bottom) supply the clip–text pairs on which modern generators are fit. MiraData scale is that of its largest release.

Corpus	Year	Scale	Annotation / Notes
UCF-101 [902]	2012	13,320 clips	101 action classes; labels, no captions
Kinetics [239]	2017	∼650K clips	Action classes; labels, no captions
WebVid-10M [2024]	2021	10M clip–text pairs	Web alt-text captions; noisy, watermarked
Panda-70M [1992]	2024	70M clip–text pairs	Auto-captioned by cross-modality teachers
MiraData [1993]	2024	330K clips	Long-duration; dense structured captions
Koala-36M [2025]	2025	36M clip–text pairs	Fine-grained structured captions; condition–content consistency

say nothing about temporal or compositional faithfulness. Recent suites are purpose-built: VBench [1951] decomposes quality into fine-grained dimensions (subject and background consistency, motion smoothness, dynamic degree, temporal flicker, per-frame aesthetic and imaging quality), each with an automatic protocol validated against human preference; EvalCrafter [2027] aggregates objective metrics over prompts derived from real-world usage; and T2V-CompBench [2028] targets compositional faithfulness over time (attribute binding, spatial relations, motion binding, interactions), the temporal analogue of the T2I-CompBench probe of §8.1.1. Table 8.8 lists them.

Table 8.8: Representative evaluation resources for text-to-video generation. Distributional resources (top) reuse captioned-clip splits for zero-shot FVD/IS/CLIPSIM; multi-dimensional suites (bottom) score temporal coherence, dynamics, and compositional faithfulness directly.

Suite	Year	Evaluation Axis	Scoring
UCF-101 / MSR-VTT [902, 2026]	2012–16	Frame quality + coarse text relevance (MSR-VTT: 10K captioned clips)	Zero-shot FVD / IS / CLIPSIM
VBench [1951]	2024	16 dimensions (consistency, motion, dynamics, flicker, quality)	Per-dimension automatic, human-validated
EvalCrafter [2027]	2024	17 objective metrics over 700 structured prompts	Aggregated automatic, fitted to human opinion
T2V-CompBench [2028]	2024	Compositional: attribute/motion binding, relations, interactions	MLLM- / detector-based

Evaluation Metrics. The metric that defines video evaluation is the Fréchet Video Distance (FVD) [1950], the spatio-temporal analogue of FID: it compares distributions of features extracted from generated and real clips by a pretrained *video* backbone (I3D), and is therefore sensitive to motion realism in a way that per-frame FID [1681] and Inception Score are not. Text–clip alignment is measured by CLIPSIM, the per-frame CLIPScore [1684] averaged over the clip, which captures topical relevance but is blind to whether the *described* motion actually occurs. Because none of these scalars captures coherence or dynamics, papers report several VBench dimensions alongside them, which exposes a trade-off a single number hides: *dynamic degree* (how much genuine motion a clip contains) against *temporal consistency* (how stable it stays), since a near-static clip trivially maximizes consistency while failing the prompt. Headline claims are corroborated with human preference.

8.3.1.4 Challenges and Trends

Three problems belong to the unconditional base case. **Long-horizon coherence from text alone** is the first: with no scaffold, object permanence and motion plausibility must emerge implicitly, and all three paradigms degrade as clips lengthen (GANs flicker, autoregressive roll-outs accumulate the error of Eq. (8.44), diffusion models lose consistency beyond their trained window), so coherence beyond a few seconds remains the binding constraint. **Motion realism without a control signal** is the second: prompts under-specify

dynamics, so the generator must sample from a highly multi-modal motion posterior, and the easy optimum is near-static or physically implausible motion that still scores well on per-frame metrics. The **video–text data bottleneck** is the third: paired data is scarce relative to images, web captions are short and weakly aligned, and this caps both controllability and the reliability of the automatic captioning used to scale corpora, which is what the dense-caption efforts (Panda-70M, MiraData, Koala-36M) address. The explicit control interfaces of the following subsections relieve the first two by moving structure and dynamics from the prompt into the input.

8.3.2 Spatial Controlled Video Generation

While textual descriptions provide high-level semantic guidance, they often lack the granularity required for precise control over object placement, scene geometry, or character appearance. Spatial-controlled video generation addresses this limitation by augmenting the generation process with explicit spatial conditioning signals, affording users more direct and deterministic influence over the synthesized visual output. The geometry, articulation, and layout that the text-to-video base case of §8.3.1 had to *hallucinate* from the prompt alone are here *externalized* into a per-frame scaffold: the spatial structure becomes an input the model must honor rather than an emergent property it must invent, which is precisely what makes the output controllable and verifiable.

8.3.2.1 Task Definition

Spatial control instantiates Eq. (8.41) with a control encoder κ_{s} acting on a spatial condition $\mathbf{c}_{\mathrm{s}}^{1:T}$ (a static or sparse reference image, a pose sequence, a depth-map sequence, facial landmarks, a sketch sequence, or a bounding-box track),

$$\hat{\mathbf{z}}^{1:T} \sim p_\theta\big(\mathbf{z}^{1:T} \,\big|\, \phi(\mathbf{x}),\, \kappa_{\mathrm{s}}(\mathbf{c}_{\mathrm{s}}^{1:T})\big), \tag{8.46}$$

so that the scaffold pins down the per-frame spatial layout while the prompt $\mathbf{x}$ and the temporal modules supply appearance and motion. The conditioning channels surveyed below differ only in what $\mathbf{c}_{\mathrm{s}}$ encodes (appearance for an image, articulation for pose or landmarks, 3D structure for depth, free-form layout for a sketch, or object location for a bounding box), and recover the text-to-video base case of Eq. (8.42) exactly when $\mathbf{c}_{\mathrm{s}} = \varnothing$.

8.3.2.2 Methodology

The six channels carry different spatial information, while the injection mechanism depends on the model. A common design is a ControlNet-style adapter [1693]: a trainable control branch processes the encoded condition and injects a residual into intermediate features of a video backbone, schematically at layer ℓ

$$h'_\ell = h_\ell + \Delta h_\ell, \qquad \Delta h_\ell = \mathcal{A}_{\psi,\ell}\big(\kappa_{\mathrm{s}}(\mathbf{c}_{\mathrm{s}}^{1:T})\big), \tag{8.47}$$

where h_ℓ denotes backbone features and $\mathcal{A}_{\psi,\ell}$ a trainable control branch (dependence on the noisy latent, diffusion step, and other conditioning is omitted, and the backbone may be frozen while the adapter trains); latent concatenation and attention-based conditioning are the other common implementations. Figure 8.14 shows the six channels alongside this adapter pathway: a reference image acts as a static appearance anchor, the other five are usually frame-indexed sequences, and compatible channels can be combined.

Image Control. A reference image (Figure 8.14a) specifies the style, content, or initial state of a video, which is the basis of animating a still and of keeping a subject's appearance consistent. I2VGen-XL [2029], Stable Video Diffusion (SVD) [1873], VideoCrafter1 [2011], and Lumiere [2030] use the reference image as an initial condition, often by concatenating its latent with the initial noise, and ConsistI2V [2031] refines this for consistency. The model then generates a plausible motion sequence that evolves from the image, typically under a text prompt, as in DynamiCrafter [2032].

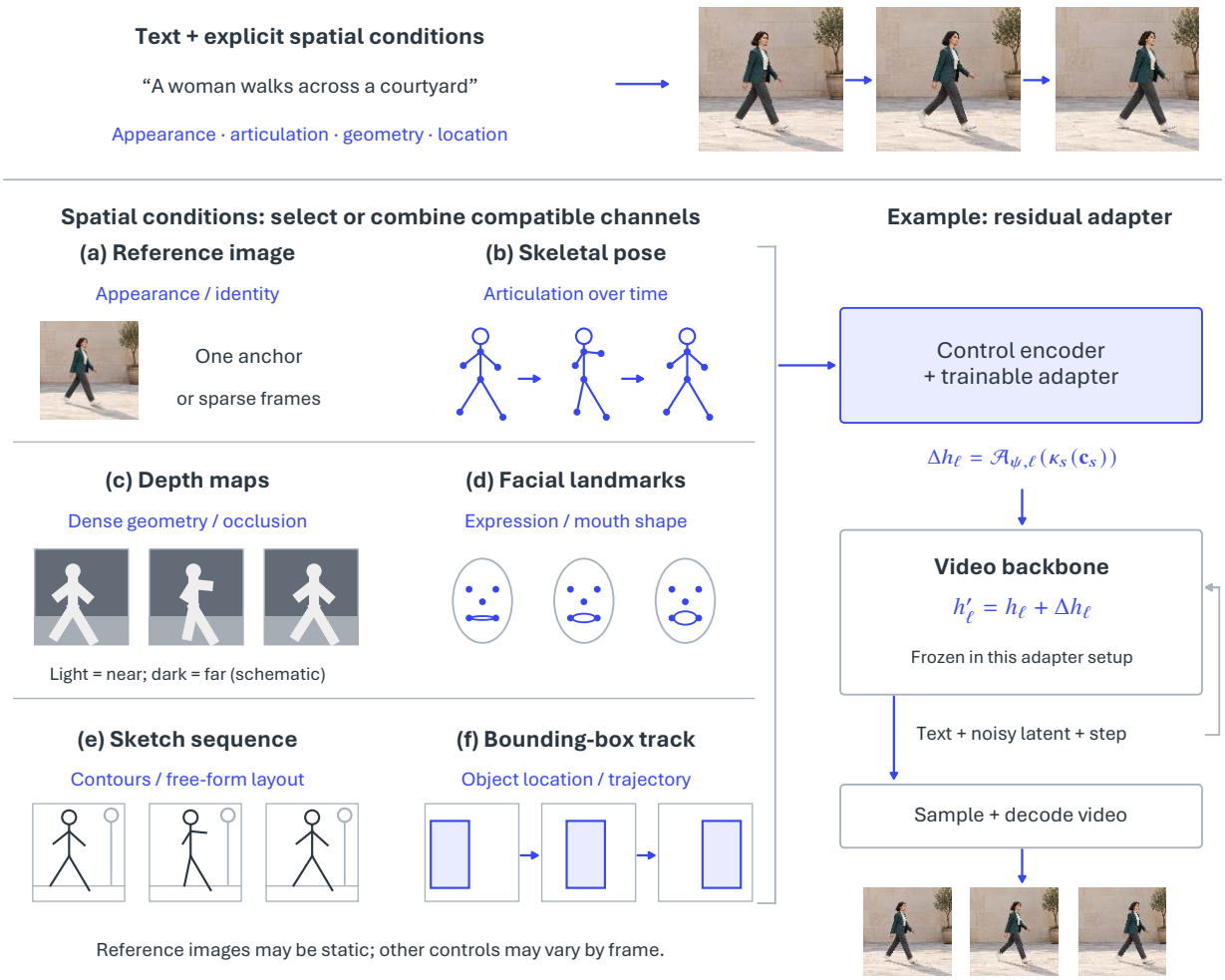


Figure 8.14: Six spatial-control channels for video generation, (a)–(f), which can be selected or combined, and an example injection route (right) in which a trainable adapter adds residual features to a frozen video backbone (Eq. (8.47)). Frames and control maps are illustrative.

Pose Control. Pose-guided generation (Figure 8.14b) animates articulated subjects, predominantly humans, along a predefined sequence of skeletal poses, the video counterpart of pose-guided image generation [2033, 1693, 2034] and the core of avatar animation and character generation. The pose keypoints (*e.g.*, from MMPose [2035]) are encoded and injected at each denoising step, frequently through a ControlNet-style adapter [1693]. MagicAnimate [2036], Animate Anyone [2037], Follow-Your-Pose [2038], and DreamPose [2039] pair a reference image for appearance with a pose sequence for motion, DisPose [2040] disentangles the pose guidance, and Champ [2041] and MagicPose [2042] target high-fidelity motion, so that the output keeps the subject's identity while following the target poses.

Depth Control. A sequence of depth maps (Figure 8.14c), estimated from an existing video or rendered from a 3D model, specifies dense 3D structure and so enforces consistent object layout and realistic occlusion. ControlVideo [2043], Make-Your-Video [2044], Control-A-Video [2045], and ControlNeXt [2046] adapt the ControlNet framework to video, processing each frame's depth map with a trainable adapter that guides the denoiser; DreamDance [2047] and SparseCtrl [2048] also use depth for dynamic scene generation.

Landmark Control. Facial landmarks (Figure 8.14d) give fine-grained control over expression and are the standard interface for talking-head animation. EchoMimic [2049], HunyuanPortrait [2050], and Live-

Portrait [2051] animate a static portrait from landmarks, often themselves driven by audio, so that lip movements, blinks, and subtle expressions stay synchronized; TASTE-Rob [2052], Takin-ADA [2053], and Follow-Your-Emoji [2054] extend landmarks to more expressive control.

Sketch Control. Sketches (Figure 8.14e) let users define compositional layout and motion with simple line drawings, an intuitive interface for animation. ToonCrafter [2055], VidSketch [2056], AniDoc [2057], and SketchBetween [2058] take a sequence of sketches to guide interpolation or generation, encoding the line art and injecting it as a condition; LayerAnimate [2059] animates hand-drawn characters and scenes layer by layer, and MagicStick [2060] uses sketch-like structural cues for editing.

Bounding Box (BBox) Control. Bounding-box tracks (Figure 8.14f) specify where objects are and how they move over time, and dominate autonomous-driving simulation: MagicDrive [2061], DriveDreamer [2062], DriveScape [2063], and DreamForge [2064] use 3D boxes to control the position and motion of vehicles and pedestrians. The box information enters either through cross-attention, as in DiVE [2065], or as a spatial feature map, and Boximator [2066] and Ctrl-V [2067] generate and control object motion from box sequences.

Nearly every structural channel (pose, depth, sketch, box) is paired with an appearance reference image, since structure alone cannot fix identity or texture, and the channels cluster by application: image and landmark control in portrait and talking-head synthesis, pose and sketch control in character animation, depth and box control in geometry-faithful and driving-simulation settings.

8.3.2.3 Benchmarks

Unlike the text-to-video base case of §8.3.1, spatial control has no single canonical corpus or suite: each channel is trained and evaluated on data specialized to its application. Models are typically built on the general web-scale corpora of §8.3.1 (WebVid-10M [2024] and successors) for appearance priors, then specialized on task data: human-animation channels on dance- and fashion-video collections paired with extracted pose keypoints (*e.g.*, via OpenPose [526] or MMPose [2035]), talking-head landmark control on portrait-video sets, and bounding-box driving simulation on annotated driving datasets such as nuScenes [445], whose 3D box and trajectory labels supply the control signal directly.

Control-fidelity Metrics. Spatial control is judged on two axes simultaneously: the generic video-quality metrics of §8.3.1 (FVD [1950], CLIPSIM, and VBench [1951] dimensions) and channel-specific *adherence* metrics that measure how faithfully the output respects $\mathbf{c}_s$. The latter are necessarily per-channel: pose and landmark control report keypoint accuracy (re-detected-keypoint distance to the target skeleton or face mesh); depth control reports depth consistency between the re-estimated and the conditioning maps; image control reports identity and appearance preservation (face-/subject-similarity to the reference); and bounding-box control reports detection mAP and trajectory IoU of the generated objects against the conditioning boxes. Neither axis suffices alone, so methods report a quality-plus-adherence tuple: a model can maximize adherence by ignoring the prompt (rigid, texture-collapsed output) or maximize quality by ignoring the control. The same holds for every controlled axis that follows.

8.3.2.4 Challenges and Trends

Single-channel spatial control has three problems of its own; those that arise once several conditions are fused are discussed in §8.3.6. Because the structural scaffold and the appearance reference pass through the same denoiser, strong control leaks structure into texture: pose and depth conditioning induce identity drift or bake the scaffold's silhouette into the surface, and disentangling what moves from what it looks like (as DisPose [2040] attempts for pose) remains open. Dense scaffolds (image, depth) over-constrain and suppress plausible motion, whereas sparse ones (boxes, coarse sketches) leave appearance and fine geometry ambiguous, so no channel is uniformly preferable and the right granularity depends on the task. And adapters trained on, say, frontal human dance transfer poorly to unusual viewpoints or non-human subjects,

which is the data cost of per-channel specialization and one motivation for the multi-condition frameworks of §8.3.4.

8.3.3 Temporal Controlled Video Generation

Temporal control regulates how motion and timing evolve across frames, the lever behind cinematic camera work, narrative pacing, and believable physical interaction. It is the complement of the spatial scaffold of §8.3.2: where spatial control fixes *where things are* in each frame, temporal control fixes *how they move between* frames, externalizing the dynamics (object motion, camera path, timing) that the text-to-video base case of §8.3.1 otherwise had to sample blindly from a highly multi-modal motion posterior.

8.3.3.1 Task Definition

Temporal control instantiates Eq. (8.41) with an encoder κ_{t} of a motion or timing signal $\mathbf{m}^{1:T}$ (an optical-flow field, an object trajectory, a camera-pose path, or a learned motion embedding),

$$\hat{\mathbf{z}}^{1:T} \sim p_\theta\big(\mathbf{z}^{1:T} \,\big|\, \phi(\mathbf{x}),\, \kappa_{\mathrm{t}}(\mathbf{m}^{1:T})\big). \tag{8.48}$$

The contrast with spatial control (Eq. (8.46)) is that $\mathbf{m}^{1:T}$ constrains the *inter-frame dynamics* (how content moves between frames) rather than the per-frame appearance, so κ_{t} typically steers the temporal-attention or latent-warping pathway of the denoiser. The channels below differ in how the motion is specified: densely (optical flow), as object paths (trajectory), as virtual-camera parameters (camera), or as a transferable behavioral style (motion).

8.3.3.2 Methodology

The four channels specify inter-frame dynamics through different signals, and no single injection architecture serves them all. One implementation uses a trainable temporal-control branch $\mathcal{T}_\psi$ to modulate features of a pretrained video backbone,

$$h'_\ell = h_\ell + \Delta h_\ell, \qquad \Delta h_\ell = \mathcal{T}_{\psi,\ell}\big(\kappa_{\mathrm{t}}(\mathbf{m}^{1:T})\big), \tag{8.49}$$

with dependence on noisy latents, diffusion step, and text omitted. More broadly, two routes recur: explicit transformations that use the motion to warp noise, latents, or attention features, and encoded conditioning that steers generation through adapters, attention, or learned motion features. They are not tied to particular channels (flow and trajectories can drive learned conditioning, and camera control can use geometry-aware warping), and Figure 8.15 keeps the four input channels separate from the two routes.

Flow Control. The most explicit motion signal (Figure 8.15a) is an optical-flow field, which records the per-pixel displacement between consecutive frames and so specifies dynamics at the finest granularity; conditioning on a sequence of such fields yields specific, coherent motion at the cost of authoring or estimating a dense signal. Motion-I2V [2068] integrates a motion-field predictor into its diffusion framework so that generation follows the supplied flow, I2VControl [2069] disentangles flow for unified control, and MOFA-Video [2070] animates a still image from user-painted motion fields. Go-with-the-Flow [2071] dispenses with the adapter altogether, warping the diffusion *noise prior* along the target flow so that the motion constraint lives in the sampling process rather than in an added branch.

Trajectory Control. Sparsifying the dense field into a few object paths (Figure 8.15b) trades pixel-level precision for an interface a user can actually author: a sequence of 2D or 3D coordinates, or a path drawn interactively, directs character movement or object kinematics, entering the denoiser either by modifying the latents or by steering cross-attention maps. MotionCanvas [2072], MagicMotion [2073], and Tora [2074] consume a predefined curve, FreeTraj [2075] relaxes this to free-form trajectories, Peekaboo [2076] also schedules when objects appear and disappear along the path, and DragNUWA [2077] and DragAnything [2078] let the user drag regions through the frame interactively. MotionCtrl [1811] handles object trajectory and camera

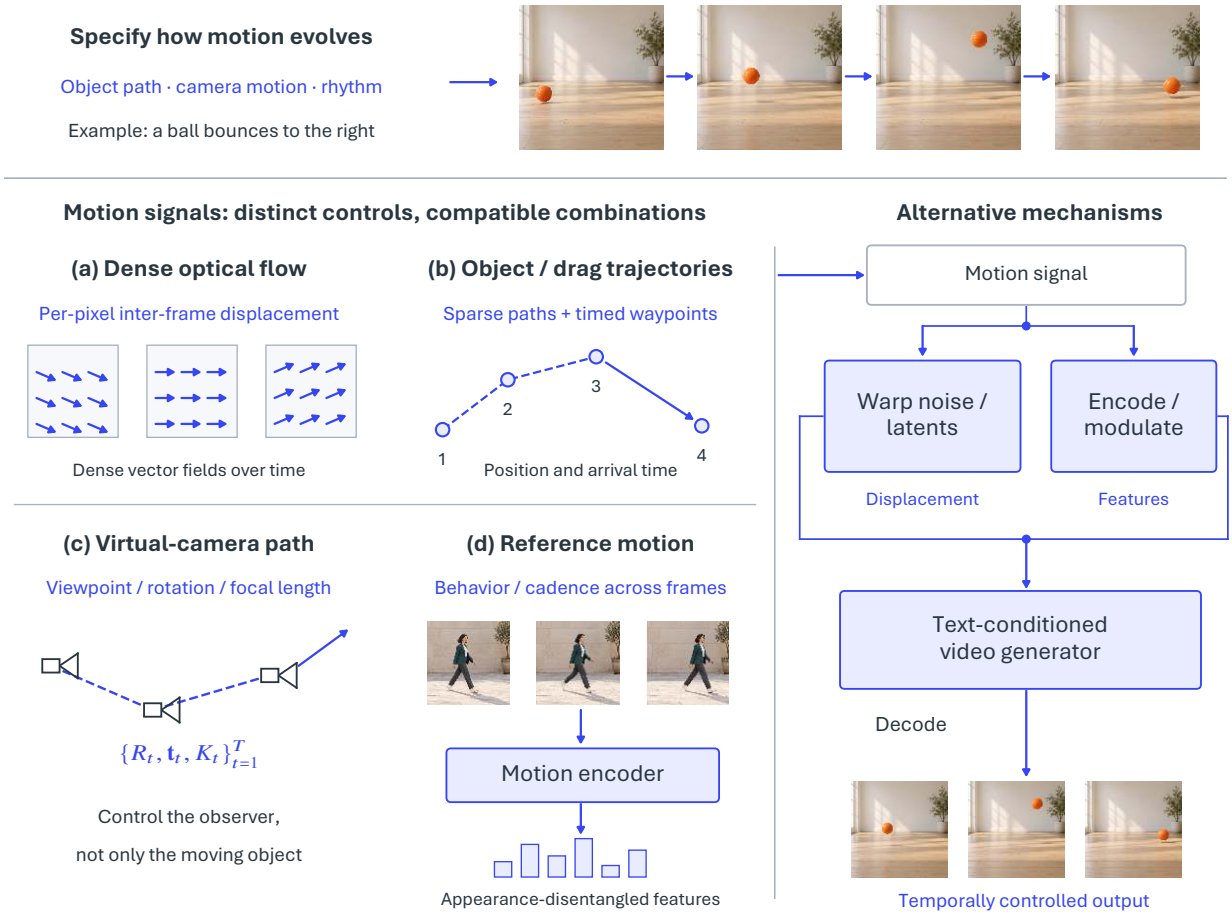


Figure 8.15: Four temporal-control channels for video generation, (a)–(d), and two alternative ways of applying them (right): warping noise or latents, and encoding the signal as conditioning features. Frames are illustrative.

motion in one controller, treating an object path and a camera path as two facets of the same inter-frame displacement.

Camera Control. Moving that displacement from objects to the observer (Figure 8.15c), camera-guided generation manipulates the virtual camera's extrinsics (translation, rotation) and intrinsics (*e.g.*, focal length) to produce pans, tilts, zooms, and dolly moves. CameraCtrl [1810] and CameraCtrl II [2079] encode the camera trajectory as a conditioning feature; Direct-a-Video [2080], ViewCrafter [1812], and Cavia [2081] couple it with novel-view synthesis; and 3DTrajMaster [2082], MotionFlow [2083], and CineMaster [2084] control camera and objects jointly in 3D. Two directions now lead. ReCamMaster [2085] and TrajectoryCrafter [2086] re-render an *existing* monocular clip under a new camera path rather than generating from scratch, and Diffusion-as-Shader [2087] and GS-DiT [189] route the camera signal through a 3D proxy (a rendered shader pass or a pseudo-4D Gaussian field), which sharpens control by making it geometry-aware.

Motion Control. At the abstract end (Figure 8.15d), the signal is no longer a geometric displacement but a *behavioral concept* (walking, dancing, jumping) transferred from a reference video as a pose sequence or a learned motion embedding. Unlike the pose control of §8.3.2, it captures the dynamic style and cadence of a behavior rather than a frame-exact skeleton, so the central technical problem is disentanglement: extracting the motion while discarding the reference's appearance. MotionDirector [2088], MotionClone [2089], and VMC [2090] learn this motion–appearance separation; Motion Prompting [2091], MotionAgent [2092], and MotionMatcher [2093] refine its granularity through trajectory- or feature-level motion descriptors; and

DreamMotion [2094] and Customize-A-Video [2095] apply a reference clip's dynamics to a new subject or scene fixed by the text prompt.

The channels trade precision against ease of specification: dense flow gives pixel-accurate control but is laborious to author, a motion embedding is effortless to supply but steers the result only loosely, and the most usable systems therefore pair an abstract signal (a drawn path, a reference clip) with a learned predictor that densifies it into the flow- or latent-level displacement the backbone actually consumes.

8.3.3.3 Benchmarks

As with spatial control (§8.3.2), temporal control has no single canonical suite: each channel is evaluated against data specialized to its signal. Models build on the general corpora of §8.3.1 for appearance and motion priors, then evaluate on task-specific resources: camera control on the posed-video collection RealEstate10K [734], whose per-frame camera poses supply both the conditioning signal and the evaluation ground truth, increasingly supplemented by the larger and more scene-diverse DL3DV-10K [735] to test generalization beyond real-estate walkthroughs; flow control against the MPI-Sintel optical-flow benchmark [956] or flow re-estimated from generated clips with RAFT [255]; and motion- and trajectory-control quality through the motion-smoothness and dynamic-degree dimensions of VBench [1951].

Control-fidelity Metrics. Alongside the generic video-quality metrics of §8.3.1, each channel measures adherence by re-extracting the control signal from the generated clip and comparing it with the conditioning input; the estimator and the error differ per channel.

Flow adherence. The optical flow of the generated clip is re-estimated with RAFT [255] and compared to the conditioning flow by the *end-point error* (EPE), the per-pixel L_2 displacement difference averaged over pixels and frames. A complementary *flow-warping error* warps each generated frame onto the next with the *target* flow and reports the photometric residual, which tests whether the rendered motion realizes the commanded displacement rather than merely correlating with it.

Trajectory adherence. The controlled object is re-localized in the output with a dense point tracker such as CoTracker [2096] or a detector, and its tracked center is compared to the target path by the mean L_2 distance, normalized by frame size and averaged over time and instances; this is the object motion-control error reported by Tora [2074] and DragNUWA [2077], and it penalizes both spatial deviation and mistimed arrival.

Camera adherence. The camera trajectory of the generated clip is recovered by structure-from-motion (COLMAP [115]) and aligned to the target up to a similarity transform, since pose is recoverable only up to scale. Following CameraCtrl [1810], a *rotation error* (the geodesic angle between estimated and target rotations, summed over frames) and a *translation error* (the L_2 distance between scale-normalized camera centers) are reported; failure to reconstruct a stable trajectory at all is itself diagnostic of an incoherent output.

Motion-transfer fidelity. With no ground-truth video, fidelity is measured from two sides: a *motion-similarity* term compares the dynamics of output and reference (the correlation of tracked point trajectories or temporal feature statistics), and an *appearance-divergence* term (CLIP similarity to the new prompt together with low feature similarity to the reference's appearance) confirms that the subject was re-synthesized rather than copied.

As for spatial control, these adherence errors are reported together with generic quality, since a near-static clip trivially minimizes EPE and pose error while ignoring the prompt.

8.3.3.4 Challenges and Trends

Three problems are specific to temporal control; fusing it with other conditions is discussed in §8.3.6. Because the motion signal passes through the same backbone that renders appearance, a transferred motion often drags the source clip's identity or texture with it, the leakage that MotionDirector [2088] and VMC [2090] try

to disentangle and that remains unsolved for large appearance gaps between reference and target. Dense flow over-constrains and suppresses generative freedom, whereas a single trajectory or a short reference clip under-specifies the global dynamics, which is why methods increasingly interpose a learned predictor that densifies an abstract signal into a flow- or latent-level field. And a commanded camera path or object trajectory can demand views or dynamics inconsistent with the scene geometry, producing parallax errors and object distortion; 3D-aware camera control (ViewCrafter [1812], CineMaster [2084]) routes the control through an explicit geometric proxy for exactly this reason, which links temporal control back to the volumetric representations of §8.2.

8.3.4 Other Controlled Video Generation

Beyond the spatial scaffolds of §8.3.2 and the motion signals of §8.3.3, a video can be steered by further, application-specific conditions: an artistic style reference, a handful of draggable points, a bird's-eye-view road layout, a target text string to render, or a synchronizing audio track. Each externalizes an attribute that text specifies only loosely, extending controllable generation to creative, interactive, and domain-specific settings; together they raise the question the single-axis interfaces leave open, how to honor *several* conditions at once, which the universal frameworks at the end of this subsection try to answer.

8.3.4.1 Task Definition

Taken one at a time, each "other" modality instantiates the generic conditional sampler of Eq. (8.41) with an encoder κ_{o} of a heterogeneous control signal $\mathbf{c}$ (a style reference, a set of draggable points, a BEV layout, a target text string, or an audio track). The distinctive task of this subsection, however, is to honor *several* such conditions simultaneously: universal and multi-task frameworks fuse K encoded signals into a single sequence,

$$\hat{\mathbf{z}}^{1:T} \sim p_\theta\big(\mathbf{z}^{1:T} \,\big|\, \phi(\mathbf{x}),\, \kappa_1(\mathbf{c}_1), \ldots, \kappa_K(\mathbf{c}_K)\big), \tag{8.50}$$

so that the K encoded conditions are processed jointly rather than one axis at a time, with $K = 1$ recovering the single-modality case and $\kappa_k = \varnothing$ for all k recovering the text-to-video base case of §8.3.1.

8.3.4.2 Methodology

The methods fall into single-modality channels (style, points, BEV, text rendering, audio) and multi-task frameworks that combine compatible conditions. Writing $u_k = \kappa_k(\mathbf{c}_k)$, the common interface is

$$\hat{\mathbf{z}}^{1:T} = G_\theta\big(\mathbf{n}, \phi(\mathbf{x}); \mathcal{F}_\psi(u_1, \ldots, u_K)\big), \tag{8.51}$$

where $\mathcal{F}_\psi$ is a model-specific conditioning interface that may act at several layers rather than produce one fixed-size embedding. Three implementations are representative and can be combined: summed adapter residuals $\sum_k \Delta h_k$, concatenation of the encoded channels with the model input or features, and unified attention over condition tokens. Figure 8.16 shows the five input modalities and these three fusion designs; $K = 1$ denotes one selected modality and $K > 1$ a supported combination.

Style Control. Style-guided generation transfers the artistic style of a reference image or video to a target video. Per-frame style transfer flickers; diffusion-based approaches such as Rerender A Video [2097], StyleCrafter [2098], FRESCO [2099], and StyleMaster [2100] instead integrate the style into generation, often by guiding it with the CLIP embedding of the style reference so that the style holds across frames, and UniVST [2101] generalizes this into a universal framework. FrameGuidance [2102] and OmniVDiff [2103] add training-free frame-level guidance and omni-controllable diffusion, respectively.

Point Control. Point control fixes *motion* through a sparse, interactive interface: the user drags a few points on an object from an initial frame to a new position in a later frame, and the model infers a plausible motion for the whole object. Drag-A-Video [2104] introduced the approach by optimizing latent codes to satisfy the

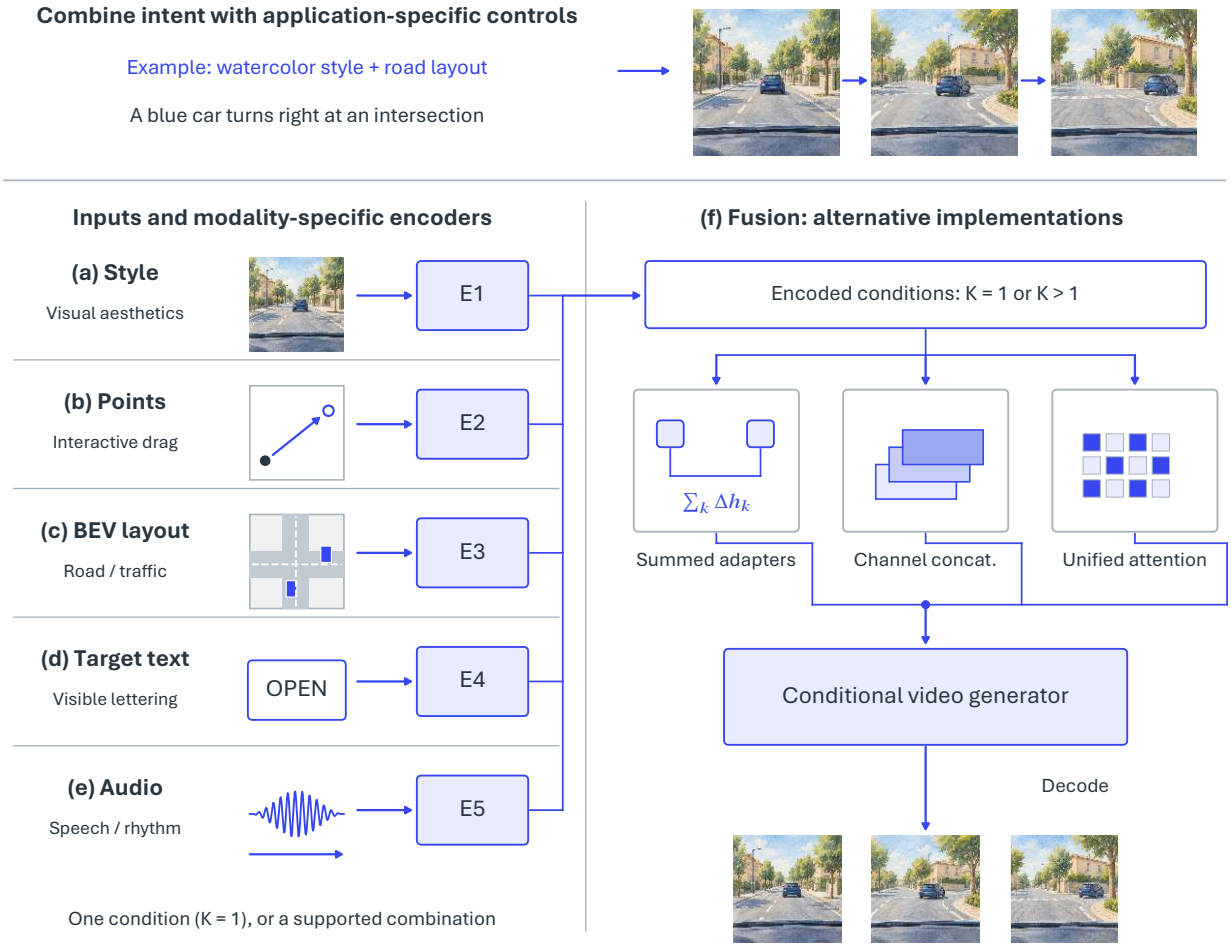


Figure 8.16: Further control modalities for video generation, (a)–(e), each with its own encoder E_k implementing κ_k, and (f) alternative ways of fusing $K \geq 1$ encoded conditions. The watercolor driving sequence is illustrative.

point constraints, Track4Gen [2105] focuses on tracking the user-specified points robustly, and the 3D-aware camera controllers of §8.3.3 [2087, 189] also accept sparse point input for non-rigid manipulation.

Bird's-Eye View (BEV) Control. BEV-guided generation serves autonomous-driving simulation: a sequence of top-down semantic maps dictates the road layout, the positions of other vehicles, and the movements of pedestrians. DriveDreamer-2 [2106], MagicDrive-V2 [2107], DiVE [2065], DreamForge [2064], and Panacea [2108] take these maps as input and generate first-person driving video consistent with the specified layout and dynamics.

Text Rendering. Many generators cannot render coherent, legible text inside a scene, so a dedicated channel treats the target string itself as the control signal. Text-Animator [2109] adds text-embedding injection and refinement modules that improve the structure and temporal stability of rendered text, and Wan [1990] reports being the first video foundation model able to render legible Chinese and English text.

Audio Control. Audio-controlled generation synthesizes video synchronized with an input audio signal, in two sub-domains. *Voice-guided generation* produces talking heads: EMO [2110], VASA-1 [2111], DreamTalk [2112], and AniPortrait [2113] take a reference image and a speech clip, extract audio features with a pretrained encoder, and drive facial motion with accurate lip-sync and expression; Say Anything [2114], MuseTalk [2115], and CAFE-TALK [2116] focus on high-fidelity lip synchronization, and V-Express [2117] and

MegActor [2118] balance audio against other control signals. *Sound-guided generation* follows more general audio such as music or ambient sound, using its rhythm, pitch, and emotional tone to drive visual elements, motion, and scene transitions: TA2V [2119], MotionCraft [2120], AV-Link [2121], and ASVA [2122] generate music visualizations or scenes that follow the energy and structure of a track, and MagicInfinite [2123] and Sound2Sight [2124] explore more open-ended audio-driven generation.

Universal and Multi-Task Control. The single-modality channels each add one encoder; universal frameworks realize the fusion of Eq. (8.50) in a single model, along two lines. Autoregressive models built on multimodal large language models generate video from discrete visual tokens, as in VideoPoet [2125] and Emu3 [2003]. Diffusion models are being adapted to many conditions at once: VideoComposer [2126] integrates depth, pose, motion, and other conditions by channel concatenation, OmniHuman [2127] generates human video under multiple controls by mixing cross-attention and concatenation, and FullDiT [2128] processes all conditions with unified attention, the third realization of $\mathcal{F}_\psi$.

8.3.4.3 Benchmarks

Being heterogeneous, the "other" modalities share no single suite: each is evaluated against data and metrics specialized to its signal, with the general corpora of §8.3.1 supplying the appearance and motion priors. BEV control is evaluated on driving collections such as nuScenes [445], whose paired BEV layouts and front-camera footage furnish both the conditioning signal and the ground truth. Voice-guided talking-head generation is evaluated on portrait-video collections (HDTF [2129] and VFHQ [2130] are standard) and sound-guided generation on audio–video corpora with aligned music or ambient tracks. Style and point control, having no canonical benchmark, are assessed on the general corpora plus qualitative and user studies.

Control-fidelity Metrics. Beyond the generic video-quality metrics of §8.3.1 (FVD [1950], CLIPSIM, and VBench [1951] dimensions), each modality adds an *adherence* metric tailored to its signal, reported together with generic quality as in §8.3.2. *Style adherence* combines a CLIP style-similarity to the reference with a temporal-consistency term that penalizes flicker. *Point adherence* re-localizes the dragged keypoints in the output and reports their L_2 deviation from the target path, as in the trajectory metric of §8.3.3. *BEV adherence* renders or detects objects in the generated clip and reports detection-mAP and layout-IoU against the input map. *Text adherence* runs OCR on the rendered frames and reports recognition accuracy and legibility. *Audio adherence* measures lip-sync quality with SyncNet-style confidence and landmark error for voice, and audio–visual synchrony for sound.

8.3.4.4 Challenges and Trends

The central difficulty here, fusing conditions of incommensurate dimensionality and strength without one washing out or dominating the others, is shared with the spatial and temporal axes and discussed in §8.3.6. Two problems are specific to these modalities. Text rendering must keep glyphs legible and stable at once, and per-frame legibility and temporal coherence pull against each other as the camera or the text moves. BEV control covers a narrow domain: models trained on a handful of driving datasets inherit their sensor rigs, cities, and traffic statistics, and generalize poorly beyond them.

8.3.5 Video Editing

Video editing takes an *existing* clip and modifies it under user instruction, rather than generating one from scratch (§8.3.1) or steering that generation with controls (§8.3.2–§8.3.4). It is the temporal lift of the image editing of §8.1.3, and its defining demand is selective preservation: the operator must change only the targeted content while holding the source's geometry, layout, and motion fixed across every frame, so the central technical problem is to balance editability against spatio-temporal consistency without artifacts or unintended drift.

8.3.5.1 Task Definition

Given a source clip encoded as $\mathbf{z}^{\text{in},1:T} = E(\mathbf{v}^{\text{in}})$ and an edit instruction $\mathbf{c}$, an editing operator

$$\hat{\mathbf{z}}^{1:T} = E_\theta^{\text{vid}}\big(\mathbf{z}^{\text{in},1:T},\, \phi(\mathbf{x}),\, \kappa(\mathbf{c})\big) \tag{8.52}$$

returns an edited clip that applies the requested change while preserving unedited content and the source motion, recovering the image editing operator of Eq. (8.8) exactly at $T = 1$. Eq. (8.52) encodes two competing objectives: *editability* (the output must realize the instruction $\mathbf{c}$) and *preservation* (the unedited regions and the source dynamics must survive untouched).

8.3.5.2 Methodology

The five paradigms are overlapping choices of training strategy and preservation mechanism: training-free editing keeps the weights fixed, one-shot tuning updates selected parameters on the source clip, trained instruction editors learn across editing data, attention-based editing reuses or modifies source attention, and structural guidance conditions the edit on source-derived depth, edges, or poses. Many of them share a diffusion recipe that inverts the source clip and re-denoises it under the instruction; writing Inv for DDIM inversion of the pretrained or adapted backbone ϵ_θ^0 of §8.3.1,

$$\mathbf{z}_S^{1:T} = \mathrm{Inv}\big(\mathbf{z}^{\text{in},1:T};\, \phi(\mathbf{x}^{\text{src}})\big), \qquad \hat{\mathbf{z}}^{1:T} = \mathrm{Denoise}\big(\mathbf{z}_S^{1:T};\, \phi(\mathbf{x}),\, \kappa(\mathbf{c})\big), \tag{8.53}$$

where S denotes the terminal diffusion step, distinct from the clip length T. Inversion supplies a source-correlated initialization; how much of the source survives then depends on the sampling, feature reuse, and guidance applied during re-denoising. Figure 8.17 schematizes the five paradigms, with panels (a)–(e) keyed to the paragraphs below.

Training-Free (Zero-Shot) Methods. The lightest realization of Eq. (8.53) (Figure 8.17a) edits a video without any fine-tuning, analogous to zero-shot image editing [2131]: DDIM inversion encodes the source into latent codes and noise maps, and the denoising process is rerun on them under the edited prompt. Text2Video-Zero [2132] and Pix2Video [2133] established the category, and vid2vid-zero [2134], I2VEdit [2135], RAVE [2136], and FastVideoEdit [2137] add latent-code warping and feature injection to improve temporal consistency without fine-tuning.

One-Shot Tuning Methods. Trading a little training for sharper fidelity (Figure 8.17b), Tune-A-Video [2014] fine-tunes a limited subset of a pretrained text-to-image diffusion model's parameters on the frames of a single source video, so that the model learns its content and motion and can regenerate it under a new prompt. Dreamix [2138] builds on one-shot fine-tuning to improve both editability and fidelity to the original motion.

Training-Based Methods. The limited control and quality of inference-time editing have motivated editors that learn a reusable operator from editing data (Figure 8.17c), particularly for instruction-based editing. InsV2V [2139] extends the InstructPix2Pix [1711] framework directly to video; GenProp [2140] propagates an edit from a user-modified first frame to the rest of the sequence; and VideoPainter [2141] uses a video inpainting model as an intermediary. UNIC [2142] handles many editing tasks in one model by concatenating the conditions into a sequence with task-aware positional embeddings, and EditVerse [2143] unifies video generation and editing by representing text, image, and video as one token sequence processed with self-attention, which permits in-context learning and flexible inputs.

Attention-Based Editing. To protect preservation within the inference-time recipe (Figure 8.17d), many techniques manipulate the cross-attention maps directly: following Prompt-to-Prompt in images, they blend the attention maps of the original and edited prompts so that the inverted source structure of Eq. (8.53) survives the re-denoising. FateZero [2144] and Video-P2P [2145] were early examples, and Edit-A-Video [2146],

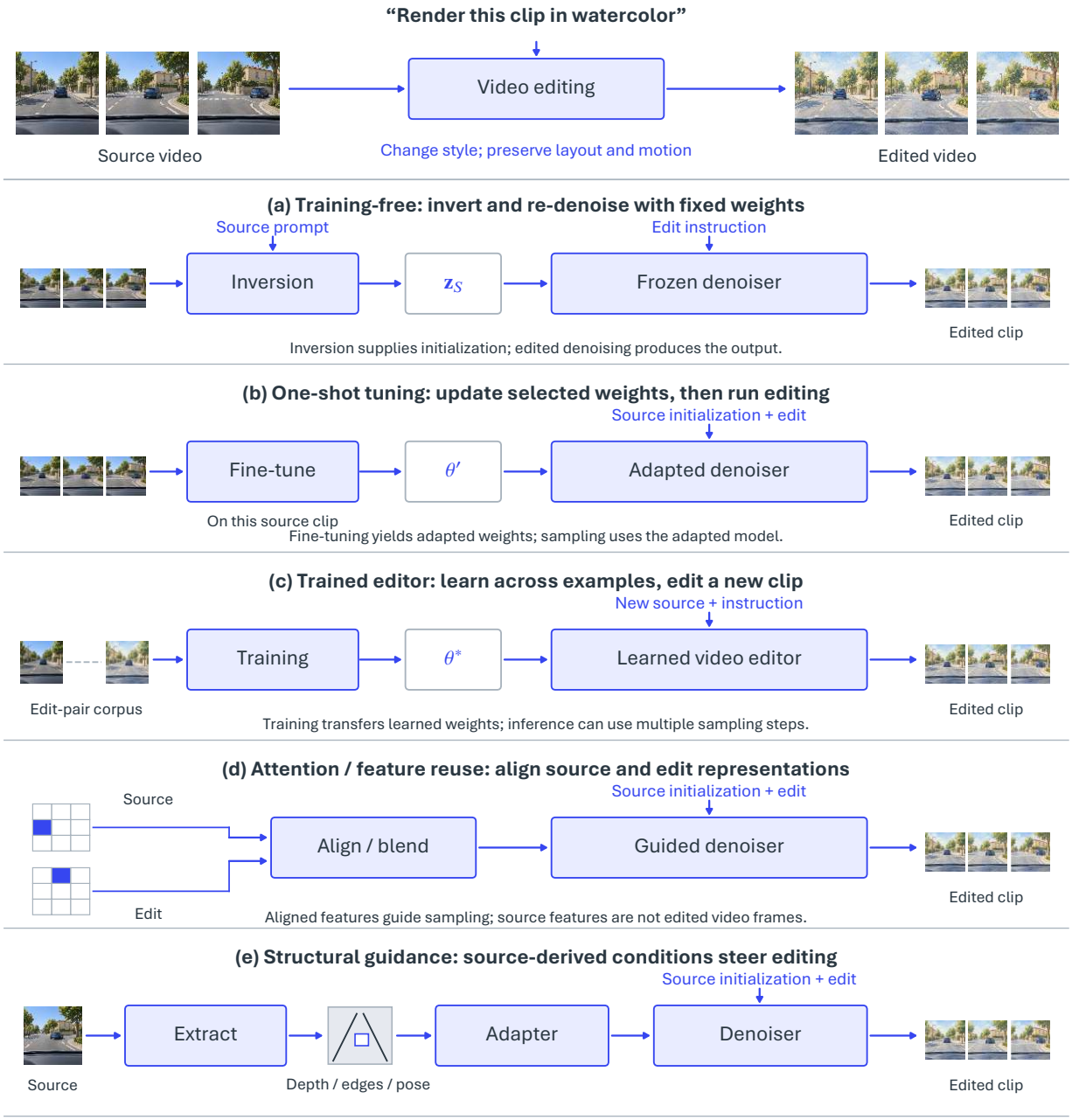


Figure 8.17: Five overlapping video-editing paradigms, (a)–(e), applied to one task (top): restyling a driving clip while preserving its layout and motion. The dashed link in (c) marks a paired training example; scene sequences are illustrative.

TokenFlow [2147], UniEdit [2148], and AnyV2V [2149] improve temporal consistency at the token or feature level, enabling localized edits such as changing an object's material or color while the scene structure is kept.

Editing with Structural Guidance. At the explicitly guided end (Figure 8.17e), editing is conditioned on spatial signals extracted from the source video, the conditioning machinery of §8.3.2 repurposed for preservation rather than generation. Rerender A Video [2097] feeds the source's depth or edge maps to a ControlNet-like module during editing so that geometry and layout stay intact, and MagicEdit [2150] accepts a variety of spatial inputs (*e.g.*, pose, sketches, segmentation maps) for high structural fidelity, which suits

style transfer and object replacement, where preserving structure is what prevents distortion and keeps motion coherent.

The paradigms trade cost against control: training-free editing avoids per-clip weight updates but may still need iterative inference, whereas a trained instruction editor is expensive to build but generalizes across edits.

8.3.5.3 Benchmarks

Video editing has no single canonical suite, and evaluation has moved from shared source-video collections toward purpose-built, fine-grained editing benchmarks. Early evaluations report on DAVIS [987], the de facto source set for object-centric edits, and on the LOVEU text-guided video-editing (TGVE) benchmark [2151], which supplies prompt–clip pairs with human ratings; VPBench, released with VideoPainter [2141], adds masked clips with dense captions for inpainting-style edits. The most recent suites target instruction-guided editing and pair richer data with model-based scoring. FiVE [2152] contributes 74 real clips (drawn from DAVIS) and 26 generated clips across six fine-grained editing types, with 420 object-level prompt pairs and masks, evaluated under a 15-metric protocol spanning background preservation, text–video similarity, temporal consistency, video quality, and runtime. IVEBench [2153] scales this to 600 source videos spanning seven semantic dimensions and clip lengths from 32 to $1{,}024$ frames, across eight task categories (35 subcategories) whose prompts are generated by LLMs and refined by experts, and adopts a three-dimensional protocol (video quality, instruction compliance, and video fidelity) that fuses traditional metrics with multimodal-LLM-based assessment.

Editing-fidelity Metrics. Editing is judged on three axes that mirror the editability–preservation tension of Eq. (8.52). The classic metrics are, for *edit adherence*, the CLIP similarity between each edited frame and the target prompt; for *temporal consistency*, frame-to-frame CLIP similarity together with a warping error that flow-aligns adjacent frames and reports the photometric residual; and for *structure and background preservation*, PSNR / SSIM / LPIPS on the unedited regions against the source, atop the generic FVD [1950] quality score. These scalars correlate only loosely with human judgment on fine-grained edits (an object recolored but mislocalized can still score well), so the latest benchmarks add vision-language judges: FiVE-Acc [2152] uses a VLM to test whether the intended fine-grained edit succeeded, and IVEBench [2153] scores instruction compliance and fidelity with multimodal LLMs alongside the traditional metrics. Results are reported as an edit-versus-preservation pair, since ignoring the instruction trivially maximizes preservation.

8.3.5.4 Challenges and Trends

Three problems are specific to editing. Pushing the edit harder erodes the unedited regions and the source motion, while protecting them too strongly suppresses the requested change; this editability–preservation trade-off of Eq. (8.52) is what every paradigm above negotiates differently. The invert-and-re-denoise recipe of Eq. (8.53) is imperfect, and its reconstruction error compounds over long clips, so edits that are clean over a few seconds degrade over minute-scale video. And training-based editors follow only the edit types seen in their training pairs, so a flexible instruction editor that handles arbitrary edits remains open; the unified generate-and-edit models UNIC [2142] and EditVerse [2143], which treat the source clip as one stream among many in shared attention, are the current attempt.

8.3.6 Summary and Critical Challenges and Future Directions

Table 8.9 places the five control axes side by side: the control signal each introduces, the operator it instantiates over the shared sampler of Eq. (8.41), the paradigms it surveys, the methods anchoring each, and the benchmarks on which it reports.

Table 8.9: The five control axes of video spatial generation; each operator specializes the spatio-temporal sampler of Eq. (8.41).

Control axis	Control signal / operator	Dominant paradigms	Canonical methods	Benchmarks / metrics
§8.3.1 Text-to-video	Text only, $\kappa = \varnothing$; $G_\theta^{\text{vid}} : \mathcal{E} \to \mathcal{Z}^{\text{vid}}$ (Eq. (8.42))	GAN; autoregressive token; diffusion / diffusion transformer	CogVideo, Sora, Wan, HunyuanVideo, Cosmos	UCF-101, MSR-VTT, VBench, EvalCrafter; FVD, CLIPSIM
§8.3.2 Spatial control	Per-frame scaffold $\kappa_{\text{s}}(\mathbf{c}_{\text{s}}^{1:T})$ (Eq. (8.46)); residual injection (Eq. (8.47))	Image, pose, depth, landmark, sketch, bounding box	Animate Anyone, MagicAnimate, ControlVideo, EchoMimic, MagicDrive	nuScenes, pose-annotated video; keypoint / depth / identity / detection adherence
§8.3.3 Temporal control	Motion signal $\kappa_{\text{t}}(\mathbf{m}^{1:T})$ (Eq. (8.48)); temporal-branch injection (Eq. (8.49))	Flow, trajectory, camera, motion; latent warping or encoded conditioning	Motion-I2V, Tora, DragNUWA, CameraCtrl, MotionDirector	RealEstate10K, DL3DV-10K, MPI-Sintel; EPE, path / pose error, transfer fidelity
§8.3.4 Other control	Heterogeneous κ_{o} fused by $\mathcal{F}_\psi$ (Eqs. (8.50), (8.51))	Style, point, BEV, text, audio; summed adapters, concatenation, unified attention	StyleCrafter, DriveDreamer-2, EMO, VideoComposer, FullDiT	nuScenes, HDTF, VFHQ; style / point / BEV / OCR / lip-sync adherence
§8.3.5 Video editing	Existing clip + edit intent → edited clip; E_θ^{vid} (Eq. (8.52))	Training-free, one-shot tuning, trained instruction editors, attention reuse, structural guidance	Tune-A-Video, TokenFlow, InsV2V, UNIC, EditVerse	DAVIS, LOVEU-TGVE, FiVE, IVEBench; edit-vs-preservation tuple, VLM / MLLM judges

The axes partition not by output domain, since all five decode a clip from the same $\mathcal{Z}^{\text{vid}}$, but by the control fed to one shared sampler: text-to-video is the $\kappa = \varnothing$ limit, spatial, temporal, and other control progressively enrich κ, and editing alone reverses the direction of the map, taking an existing clip in place of pure noise. The injection mechanism converges across axes: the spatial residual branch (Eq. (8.47)), the temporal branch (Eq. (8.49)), and the multi-condition interface (Eq. (8.51)) are one idea, a trainable control pathway added to a pretrained backbone, generalized from one condition to many, and the most recent systems (VideoComposer, FullDiT, UNIC, EditVerse) go further by folding every axis, editing included, into a single token sequence processed by shared attention. Evaluation has not converged: no suite spans the five axes, each reports generic quality together with a channel-specific adherence score, and the newest editing benchmarks (FiVE, IVEBench) add vision-language and multimodal-LLM judges to improve human alignment.

Video generation inherits every difficulty of the 2D and 3D priors of §8.1 and §8.2 and adds the time axis, which turns several already-hard problems into open ones; the four below cut across all five axes.

Temporal coherence and long-horizon drift. The defining difficulty of video over image generation is keeping content stable across frames. The signature artifact is *flicker*, the high-frequency appearance instability between adjacent frames that plagues per-frame and early GAN-based methods (§8.3.1); over longer spans it gives way to *identity drift*, the gradual mutation of a subject's appearance, color, or fine texture across a clip; and in autoregressive roll-outs to *error accumulation*, the compounding of small per-step mistakes that caps usable clip length. Latent video autoencoders with temporal compression [1873, 2007] and dedicated spatio-temporal attention suppress flicker, while autoregressive–diffusion hybrids such as CausVid [2004] and MAGI-1 [2005] combine sequential consistency with diffusion fidelity to push toward minute-scale generation; coherence over arbitrarily long horizons remains unsolved and is the binding constraint on the world-model applications that motivate the section.

Controllability versus fidelity under multi-condition fusion. As §8.3.2–§8.3.4 stack heterogeneous conditions (spatial scaffolds, camera paths, audio, style) onto Eq. (8.41), the conditions begin to conflict. Over-strong conditioning yields rigidity and texture collapse; weak conditioning ignores the control; and naive fusion produces *control-signal dilution* (summed adapters wash one another out), *condition leakage* across frames, and *collapse to the dominant condition* when one modality (*e.g.*, audio in talking-head generation) overwhelms the rest. Compositional and universal frameworks (VideoComposer [2126], OmniHuman [2127], and the

unified-attention design of FullDiT [2128]) replace per-axis injection with joint token fusion and task-aware positional embeddings, but a principled account of how to weight and arbitrate competing conditions is still missing.

Physical and 3D plausibility. Even a temporally coherent, well-controlled clip may depict an impossible world, which is the bridge back to §8.2. Generated motion frequently violates object permanence (objects teleport, morph, or vanish), contact and rigid-body dynamics (rigid objects warp non-rigidly; feet slide; occlusions resolve implausibly), and *multi-view consistency* under camera motion, where the same scene is rendered inconsistently from successive viewpoints. Systems presented as steps toward "world simulators", Sora [1991] among them, still exhibit these failures on closer inspection. Camera-control methods [1810, 1812] and explicitly 3D-aware video diffusion (Diffusion-as-Shader [2087] and GS-DiT [189]) narrow the gap by conditioning generation on geometry, which suggests that the more promising route to physical plausibility runs through the volumetric representations of §8.2 rather than through scale alone.

Compute, latency, and evaluation. Underlying all three is cost: a spatio-temporal latent is an order of magnitude larger than an image latent, so long, high-resolution clips strain memory and make iterative sampling slow, which in turn limits the context length available for coherence and the conditioning budget available for control. Consistency distillation and few-step samplers, together with sparse and causal attention, are the active responses. Evaluation compounds the problem: distributional scores such as FVD and per-frame FID are blind to temporal and control adherence, and automatic metrics correlate weakly with human judgment, so reported progress is hard to compare across methods; benchmarks and human-aligned protocols that jointly score fidelity, temporal coherence, and per-axis control adherence are a central near-term need.

Coherence and physical plausibility both demand longer temporal context, which inflates compute, and every added condition enlarges the evaluation surface, so progress on any one of the four is bounded by the others. Together they mark the last step of the progression traced across §8, from the static 2D prior of §8.1 through the static volumetric prior of §8.2 to dynamic, navigable models that consume spatial structure over time.

8.4 Motion Generation



Motion generation is the temporal, embodied output channel of spatial generation: where §8.1–§8.3 synthesize static images, volumetric assets, and controllable video, motion generation produces the *articulated dynamics* that an avatar, character, or robot can actually enact. A generated motion is a time series of body poses that must be believable in its dynamics, physically plausible (balance, ground contact, non-penetration), and faithful to its driving signal, whether a text description, audio, a scene, or a key frame. The conditional-sampler template that turns a prompt into pixels (Eq. (8.2)) is here instantiated over a sequence of poses in a structured kinematic space, so geometric priors meet physical constraints: the output must respect the biomechanics of an articulated body as well as the semantics of the condition.

The subsections are indexed by the moving agent and then by one cross-cutting operation: human motion (§8.4.1), animal motion with its cross-species morphology (§8.4.2), vehicle motion planning, where the body is rigid and the difficulty moves into the multi-agent traffic scene (§8.4.3), and motion editing (§8.4.4).

8.4.1 Human Motion Generation

Human motion generation produces natural, varied sequences of human body poses from intentions, environmental stimuli, or explicit conditions, and supports animation, virtual characters in film and games, AR/VR, and human–robot interaction.

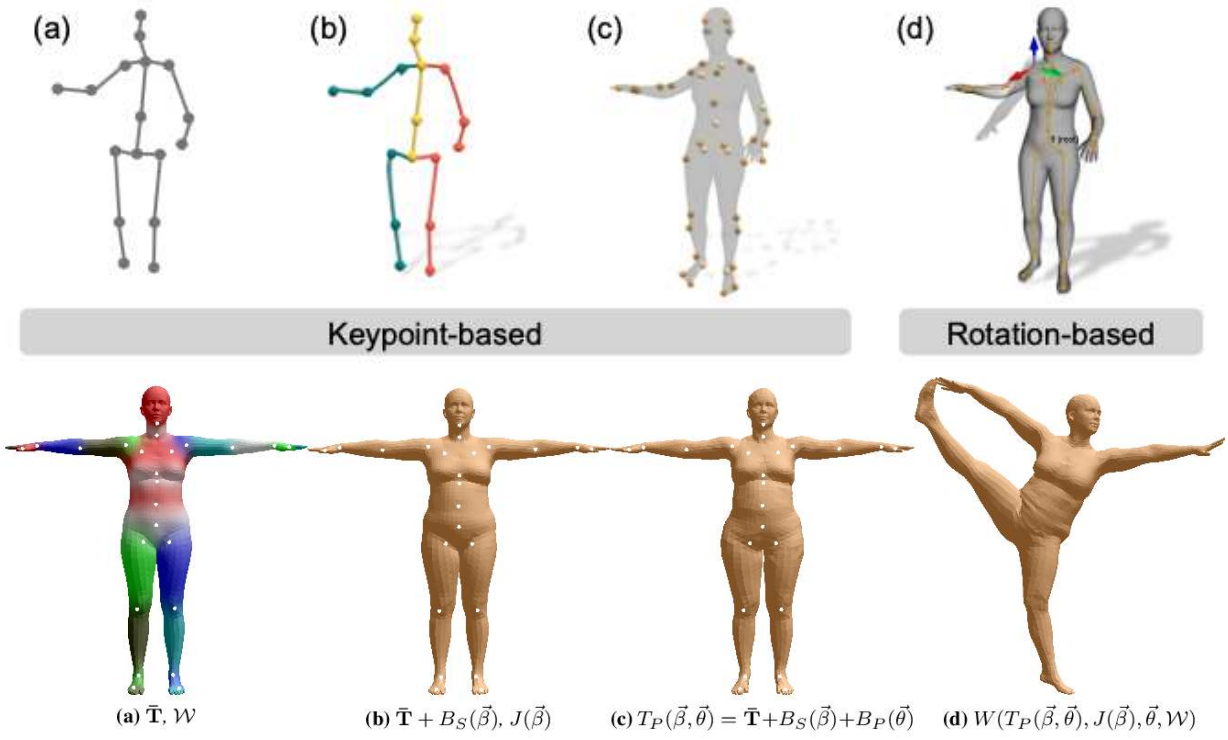


Figure 8.18: Human motion representations. Top: keypoint-based, as (a) joint positions, (b) joints grouped by kinematic chain, and (c) dense surface markers; and rotation-based, as (d) per-joint rotations in $\mathrm{SO}(3)$ on a kinematic tree. Bottom: the SMPL body model, from (a) the template mesh and skinning weights through (b) shape and (c) pose blend shapes to (d) the skinned, posed mesh.

8.4.1.1 Task Definition

Let a motion be a temporal sequence of body poses $\mathbf{m}^{1:T} = (\mathbf{m}^1, \ldots, \mathbf{m}^T)$, where each frame $\mathbf{m}^t \in \mathcal{P}$ is a pose in a chosen representation space $\mathcal{P}$ and $\mathcal{M} = \mathcal{P}^T$ is the resulting motion space. Writing ϕ for the text encoder of Eq. (8.2) and $\kappa : \mathcal{C} \to \mathcal{C}'$ for an encoder of a driving signal $\mathbf{c} \in \mathcal{C}$ (an action label, a text description, a music or speech track, or a scene), human motion generation learns a conditional sampler

$$G_\theta^{\text{mot}} : \mathcal{C}' \to \mathcal{M}, \qquad \hat{\mathbf{m}}^{1:T} \sim p_\theta\big(\mathbf{m}^{1:T} \mid \kappa(\mathbf{c})\big)\,. \tag{8.54}$$

This is the conditional-sampler template of Eq. (8.2) instantiated over a temporal pose sequence rather than an image latent: with no driving signal ($\kappa = \varnothing$) it reduces to unconditional motion synthesis, and the choice of $\mathbf{c}$ indexes the five conditioning settings surveyed in §8.4.1.2.

Figure 8.18 illustrates the pose representations that instantiate $\mathcal{P}$. Broadly, representations fall into keypoint-based and rotation-based categories, and the two are convertible: joint rotations map to spatial keypoints via forward kinematics (FK), and keypoints recover joint rotations via inverse kinematics (IK).

Keypoint-based Representation. The body is modeled as a collection of keypoints, each a distinct anatomical landmark (joints or salient surface locations) given by its 2D or 3D coordinates in a chosen frame (pixel space or world space), and motion is encoded as a temporal series of such configurations. Keypoint representations are directly available from many motion-capture systems and are highly interpretable, but many downstream uses (character animation, robotic control) require converting keypoints to joint rotations through the IK problem. Some works instead augment or replace sparse skeletal keypoints with dense body markers

distributed across the body surface [2154, 2155], which carry richer information about body shape, limb twisting, and fine geometric detail.

Rotation-based Representation. A pose is represented by joint angles, the rotations of body segments relative to their parent in a hierarchical kinematic tree. Most studies model these in three dimensions, treating each non-root joint as an element of $\mathrm{SO}(3)$, parameterized as Euler angles, axis–angle vectors, or unit quaternions.

Skinned Multi-Person Linear Model. Beyond pure joint rotation, statistical mesh models jointly capture pose and body shape, including non-rigid deformations associated with articulation. The Skinned Multi-Person Linear (SMPL) model [338] defines a template mesh with fixed topology (6,890 vertices, 23 joint rotations plus a root) and uses linear blend skinning augmented by learned blend shapes: identity (shape) blend shapes for inter-subject variation and pose blend shapes for rotation-dependent deformations such as muscle bulging and twisting.

8.4.1.2 Methodology

Methods for human motion generation are organized along two axes. The first is the *learning paradigm*: regression-based techniques learn a deterministic mapping from conditions to a single target motion, whereas generative approaches model the full conditional distribution of motion using GANs, VAEs, normalizing flows, or diffusion models (the generative paradigms themselves are detailed in §8.1.1). Formally, a regression model realizes Eq. (8.54) as a point estimate,

$$\hat{\mathbf{m}}^{1:T} = f_\theta(\kappa(\mathbf{c}))\,, \qquad \min_\theta \; \big\|\hat{\mathbf{m}}^{1:T} - \mathbf{m}^{1:T}\big\|, \tag{8.55}$$

which is simple and stable but collapses the inherently one-to-many condition–motion relationship onto an averaged trajectory. Generative models instead sample $\hat{\mathbf{m}}^{1:T} \sim p_\theta(\mathbf{m}^{1:T} \mid \kappa(\mathbf{c}))$; the now-dominant diffusion instantiation trains a denoiser G_θ to recover the clean motion at every noise level n,

$$\mathcal{L} = \mathbb{E}_{\mathbf{m}^{1:T},\, n} \left\| \mathbf{m}^{1:T} - G_\theta(\mathbf{m}_n^{1:T},\, n,\, \kappa(\mathbf{c})) \right\|^2, \tag{8.56}$$

which preserves diversity and supports flexible conditioning, at higher sampling cost. The second axis is the *conditioning modality* $\mathbf{c}$, which indexes the five settings below; each setting instantiates the same Eq. (8.54) and has migrated, broadly, from regression and adversarial models toward diffusion. Figure 8.19 previews this two-axis organization.

Action-to-Motion. Given a discrete action label (*e.g.*, "walk", "throw"), these methods generate a corresponding sequence. Early work used GANs with graph convolution and self-attention (SA-GAN [2156], Kinetic-GAN [2157]) for realism and diversity, while VAE/CVAE frameworks (Action2Motion [2158], ACTOR [2159]) produced probabilistic, controllable motions. More recent models adopt contrastive latent learning (ODMO [2160]), discrete autoregressive sequences (PoseGPT [2161]), and diffusion (MDM [2162], the latent-diffusion hybrid MLD [2163]). As most methods excel only at single actions, MultiAct [2164] recurrently composes multiple action motions into long sequences via a canonicalization scheme.

Text-to-Motion. Conditioning on free-form language exploits richer semantic cues than fixed labels allow. Early methods used GANs (Text2Action [2165]) or joint text–motion embeddings (JL2P [2166], two-stream encoder–decoders [2167]); later ones added variational modeling and attention (TEMOS [2168], TEACH [2169]). For zero-shot generalization, MotionCLIP [2170] and AvatarCLIP [2171] align the motion latent space with CLIP. The current frontier combines VQ-VAE tokenization [2172], autoregressive transformers (T2M-GPT [2173]), and diffusion (FLAME [2174]), with multimodal conditioning extensions such as MoFusion [2175] fusing text and audio. More recent masked-generative models (MoMask [2176]) push quality

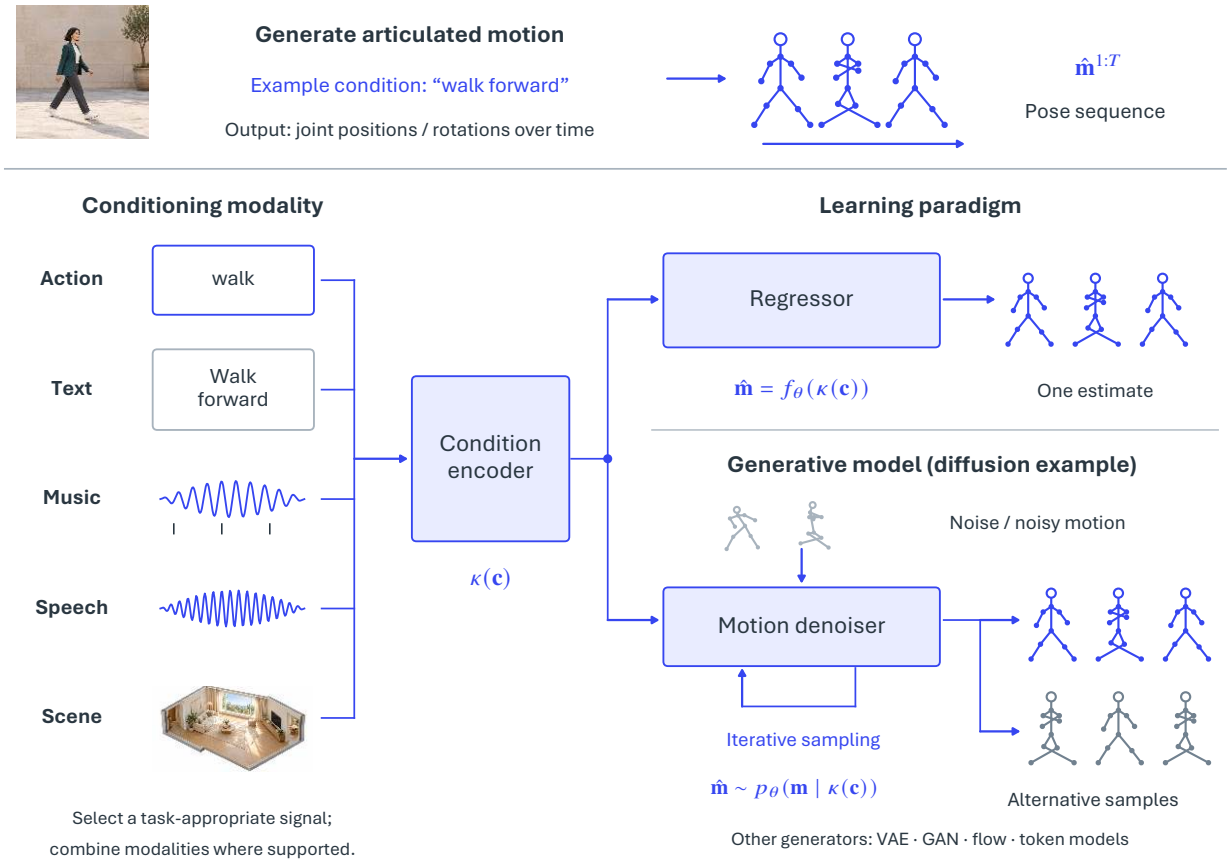


Figure 8.19: Two-axis organization of human motion generation: the conditioning modality (left), encoded by κ, and the learning paradigm (right), either a regressor that returns one estimate (Eq. (8.55)) or a generative model that returns alternative samples, shown with diffusion (Eq. (8.56)). Branches are alternatives, not consecutive stages.

and inference speed further by predicting quantized motion tokens with a bidirectional, iteratively-refined masking schedule rather than left-to-right.

Music-to-Dance. The goal is coherent, expressive dance aligned to a music track. Early approaches framed this as supervised sequence-to-sequence translation via LSTM autoencoders [2177], transformers (AI Choreographer [2178]), and multi-person frameworks (GroupDancer [2179], GDancer [2180]). Because music–dance is inherently one-to-many, generative models model the diverse distribution: MNET [2181] with style conditioning, Transflower [2182], Bailando [2183] (VQ-VAE + GPT), and EDGE [2184] (diffusion). Graph-based methods (ChoreoMaster [2185], PC-Dance [2186], ChoreoGraph [2187]) use motion graphs and shared music–motion embeddings for rhythmic coherence, while curriculum learning, dynamics-prior denoising, and hierarchical pose/motif/choreography modeling target long-term, genre-consistent full-song dances [2188, 2189, 2190].

Speech-to-Gesture. Co-speech gesture synthesis generates upper-body or full gesture sequences from speech (audio, sometimes text). Early adversarial methods modeled person-specific gesture patterns [2191]; subsequent work disentangled shared versus motion-specific variation [2192] and decomposed body, hands, and face [2193]. Recent methods employ diffusion [2194, 2195, 2196], VQ-VAE quantization [2197], and style or speaker embeddings [2198] to handle diversity, identity, rhythm, and expression more robustly.

Scene-to-Motion. Here the condition is a 3D scene, and most methods adopt a multi-stage pipeline that predicts a goal or interaction anchor, plans a trajectory, then generates motion along it. Early works predict a goal from 2D/RGB scene inputs and synthesize motion through trajectory-guided VAEs [2199, 2200], sometimes with scene-aware discriminators. SAMP [2201] models 3D paths and human–object interactions; later methods refine interaction with per-joint contact maps [2202] or path-search (A^*) for trajectory diversity [2203], and GOAL [2155] and SAGA [2154] target grasping via two-step goal-to-motion VAEs. Some assume given goals and stitch sub-goal motions hierarchically for long-term synthesis [2204], while refinement-based frameworks such as CIRCLE [2205] initialize by interpolation and refine with scene-geometry-aware modules.

8.4.1.3 Benchmarks

Human motion datasets group naturally by conditioning modality; Table 8.10 gives their representation, scale, and condition. Text- and action-conditioned corpora dominate, led by the language-annotated KIT Motion-Language [2206] and HumanML3D [2207], BABEL [2208]'s sequence- and frame-level labels over the AMASS [2209] aggregate, and the action taxonomies of UESTC [2210], NTU-RGB+D 120 [2211], and HumanAct12 [2158]. Audio-conditioned resources split into music-to-dance sets (AIST++ [2178], AIOZ-GDANCE [2180], and several genre-specific collections) and co-speech gesture sets (Trinity [2212], BEAT [2213], PATS [2198]), many built by fitting parametric body models to in-the-wild video. Scene-conditioned datasets (PROX [2214], GRAB [2215], HUMANISE [2216], CIRCLE [2205]) capture human–scene and human–object interaction with synchronized geometry, from Vicon and IMU rigs to synthetic engines such as GTA-IM [2199]. Across all three groups, the cost of high-quality capture forces a trade-off between scale and annotation fidelity.

Evaluation Metrics. Evaluating synthesized motion is hard because the condition–motion map is one-to-many, judgments are partly subjective, and quality depends on high-level conditional cues, so the literature reports an ensemble of metrics along three axes. **Fidelity** captures naturalness, smoothness, and plausibility: distance-based measures such as NPSS [2229] and NDMS [2230], keypoint accuracy (PCK [2231]), motion-space statistics (AVE, Hellinger distance over speed histograms [2232], Fréchet Distance over pose rotations [2201, 2203], PSKL [2233]), feature-space distributional distances led by FID (with IS [2198], MMD [2156], and related scores), and physical-plausibility proxies for foot sliding and foot–ground contact (FCS [2184], non-collision ratios [2154], contact accuracy [2234]), the last still lacking a standardized definition. **Diversity** guards against the long-sequence "freezing" failure via intra-motion variation and freezing rate [2189], inter-motion manifold coverage, and multimodality metrics (n-APD, ADE/FDE and their multimodal variants [2235]) computed by sampling many motions per condition. **Condition consistency** measures alignment to the driving signal: recognition accuracy and R-Precision for text/action, feature-level text–motion distance (Multimodal Distance [2207], mCLIP and MID [2174]), audio–motion beat metrics (beat coverage, hit rate [2218], beat alignment [2178], SRGR [2213], Semantic Score [2196]), and, for scene conditioning, non-collision and human–scene contact scores [2154, 2228]. No single metric is decisive, so methods are compared on a task-appropriate composition of the three.

8.4.1.4 Challenges and Trends

Data scarcity is the most basic constraint: high-quality motion capture is expensive and slow, so dataset scale trades against annotation fidelity, and heterogeneous pose representations and conditioning signals fragment the data that exist; fusing heterogeneous sources through weak supervision and multimodal foundation models is the main response. **Semantic alignment** is the second. Motion encodes culture, communication, and context, not only mechanics, and a model that fits joint trajectories while missing their relation to language, music, or environment produces motion that is kinematically clean but perceptually empty; semantic priors from language and foundation models help, but how to measure the result is unresolved. That is the **evaluation** problem: every objective metric above has blind spots, none replaces a

Table 8.10: Benchmarks for human motion generation, grouped by condition (none, text / action, music, speech, scene/object).

Dataset	Representation	Subjects	Sequences	Frames	Length	Condition	Remarks
Human3.6M [550]	Keypoint	11	-	3.6M	5h	-	15 actions
AMASS [2209]	Rotation	344	11,265	-	40h	-	Unifies 15 marker-based MoCap datasets
HuMMan [2217]	Rotation	1000	400K	60M	-	-	500 actions
KIT Motion Language [2206]	Keypoint	111	3,911	-	10.3h	Text	6.3K text descriptions
UESTC [2210]	Keypoint	118	25.6K	-	83h	Action	40 action classes
NTU-RGB+D 120 [2211]	Keypoint	106	114.5K	-	74h	Action	120 action classes
HumanAct12 [2158]	Keypoint	12	1,191	90K	6h	Action	12 action classes
BABEL [2208]	Rotation	344	-	-	43.3h	Text	260 action classes
HumanML3D [2207]	Keypoint + Rotation	344	14.6K	-	28.5h	Text	44.9K text descriptions
DanceWithMelody [2177]	Keypoint	-	61	907K	1.6h	Music	4 genres
Lee [2218]	Keypoint	-	361K	-	71h	Music	3 genres
Huang [2188]	Keypoint	-	790	-	12h	Music	3 genres
AIST++ [2178]	Rotation	30	1,408	10.1M	5.2h	Music	10 genres
PMSD [2182]	Keypoint	8	-	-	3.8h	Music	4 genres
ShaderMotion [2182]	Keypoint	8	-	-	10.2h	Music	2 genres
Chen [2185]	Rotation	-	-	160K	9.9h	Music	9 genres
PhantomDance [2219]	Rotation	-	260	795K	3.7h	Music	13 genres
MMD-ARC [2186]	Rotation	-	213	-	11.3h	Music	-
MDC [2179]	Rotation	-	798	-	3.5h	Music	2 genres
Aristidou [2190]	Rotation	32	-	-	2.4h	Music	3 genres
AIOZ-GDANCE [2180]	Rotation	>4000	-	-	16.7h	Music	7 styles, 16 genres
Trinity [2212]	Keypoint	1	23	-	4.1h	Speech	Casual talks
TED-Gesture [2220]	Keypoint	-	1,295	-	52.7h	Text	TED talks
Speech2Gesture [2191]	Keypoint	10	-	-	144h	Speech	TV shows, lectures
TED-Gesture++ [2221]	Keypoint	-	1,766	-	97.0h	Speech, Text	Extension of [2220]
PATS [2198]	Keypoint	25	-	-	251h	Speech, Text	Extension of [2191]
Speech2Gesture-3D [2193]	Keypoint	6	-	-	33h	Speech	Videos from [2191]
BEAT [2213]	Rotation	30	2,580	30M	76h	Speech, Text, Emotion	8 emotions, 4 languages
Chinese Gesture [2222]	Rotation	5	-	-	4h	Speech, Text	Chinese
ZEGGS [2223]	Rotation	1	67	-	2.3h	Speech, Style	19 styles
SHOW [2197]	Rotation	-	-	-	27h	Speech	Videos from [2191]
WBHM [2224]	Rotation	43	3,704	691K	7.68h	Object	41 objects
PiGraph [2225]	Keypoint	5	63	0.1M	2h	Scene, Object	30 scenes, 19 objects
PROX [2214]	Rotation	20	60	0.1M	1h	Scene, Object	12 indoor scenes
i3DB [2226]	Keypoint	1	-	-	-	Scene, Object	15 scenes
GTA-IM [2199]	Keypoint	50	119	1M	-	Scene	Synthetic, 10 indoor scenes
GRAB [2215]	Rotation	10	1,334	1.6M	-	Object	51 objects
HPS [2227]	Rotation	7	-	300K	-	Scene	8 large scenes
SAMP [2201]	Rotation	1	-	185K	0.83h	Scene, Object	7 objects
COUCH [2228]	Rotation	6	>500	-	3h	Scene, Chairs	3 chairs
HUMANISE [2216]	Rotation	-	19.6K	1.2M	-	Scene, Object, Text	643 scenes
CIRCLE [2205]	Rotation	5	>7K	4.3M	10h	Scene	9 scenes

user study, and the field still needs metrics that are both interpretable and better correlated with human perception. **Controllability** matters for practical use as much as in image generation, and joint masking, attribute prompts, and style conditioning are moving toward editing tools with textual, spatial, and temporal handles (§8.4.4). **Interactivity** remains underexplored: most methods animate one person in a static scene, and socially coupled groups (conversation, group dance) and motion grounded in dynamic, actionable scenes need both new corpora and new models.

8.4.2 Animal Motion Generation

Most motion-generation work targets the human body and ignores the diversity of animal behavior, although synthesizing animal motion matters for wildlife conservation, ecology, and biomechanics. Relative to §8.4.1, animal motion is harder on two fronts that the human setting never confronts: species vary widely in skeletal topology and morphology, so a single fixed-skeleton representation no longer suffices, and the same textual description ("walk stealthily") maps to radically different gaits across species. These two pressures, *i.e.*, data scarcity and morphological heterogeneity, shape every method below, which is why animal motion generation borrows the architectures, datasets, and training recipes of human motion generation while adding species-conditioning machinery on top.

8.4.2.1 Task Definition

Animal motion generation extends the human sampler of Eq. (8.54) with an explicit species (morphology) descriptor. Let $s \in \mathcal{S}$ index a species and let $\mathcal{P}_s$ be its species-dependent pose space (its skeleton topology and joint count depend on s), so that $\mathcal{M}_s = \mathcal{P}_s^T$ is the corresponding motion space. With κ encoding a driving signal $\mathbf{c} \in \mathcal{C}$ (an action class, a text description, an anchor pose, or past motion), the task is to learn

$$G_\theta^{\text{amot}} : \mathcal{C}' \times \mathcal{S} \to \mathcal{M}_s, \qquad \hat{\mathbf{m}}^{1:T} \sim p_\theta\big(\mathbf{m}^{1:T} \mid \kappa(\mathbf{c}),\, s\big)\,. \tag{8.57}$$

Restricting $\mathcal{S}$ to the single human morphology recovers Eq. (8.54) exactly; the open difficulty is that s ranges over a heterogeneous, long-tailed set of skeletons for which paired text–motion data are scarce, so the species condition must do double duty as both a generation control and a regularizer that shares statistical strength across morphologies.

8.4.2.2 Methodology

Almost all animal motion generation is text-driven, and the methods share a common response to the data-scarcity and morphology problems of Eq. (8.57): transfer representations learned from abundant human motion, then specialize them to a species through *feature modulation*. Concretely, given a backbone feature $\mathbf{h}$ at some layer, species-aware modulation applies a FiLM-style affine transform conditioned on s,

$$\tilde{\mathbf{h}} = \gamma(s) \odot \mathbf{h} + \beta(s), \tag{8.58}$$

where $\gamma(s), \beta(s)$ are predicted from the species descriptor, allowing one shared network to adapt its internal representation to diverse anatomical forms. The methods differ in how they obtain the shared representation and how strictly they enforce cross-species plausibility, as Figure 8.20 summarizes.

OmniMotionGPT [2236] tackles limited animal data by jointly training motion autoencoders on human and animal corpora and aligning their latent encodings to text via CLIP embeddings; it relies most directly on transfer, and at inference it still depends on human-motion priors, which prevents truly text-only animal generation. **Motion Avatar** [2237] uses SinMDM [2238] for data augmentation in its ZooGen pipeline (the route by which the Zoo-300K dataset is built) and extends the masked-motion modeling of MoMask [2176] to animals, adding an LLM planner that orchestrates joint motion and 3D-avatar mesh generation; it makes little use of species-specific structure, which is what Eq. (8.58) addresses. **AniMo** [2239] makes that remedy central, injecting species-aware modulation into a joint-aware spatiotemporal tokenizer and learning, through

masked modeling, a mapping from text plus species attributes to motion tokens; its AniMo4D dataset (78,149 sequences over 114 species) yields markedly better morphological and behavioral fidelity than transfer-only baselines. **X-MoGen** [2240] pushes toward a single model spanning humans and animals: it embeds motion into a shared space with graph-based VAEs and latent autoencoders, generates via masked modeling, and adds a morphological-consistency module that enforces skeletal plausibility across species, supported by the UniMo4D corpus ($\sim$115 species, $\sim$119K sequences). OmniMotionGPT and Motion Avatar still lean on human priors, whereas AniMo and X-MoGen promote species morphology to a first-class, explicitly regularized conditioning variable, so the four together trace a progression from transfer toward native cross-species modeling.

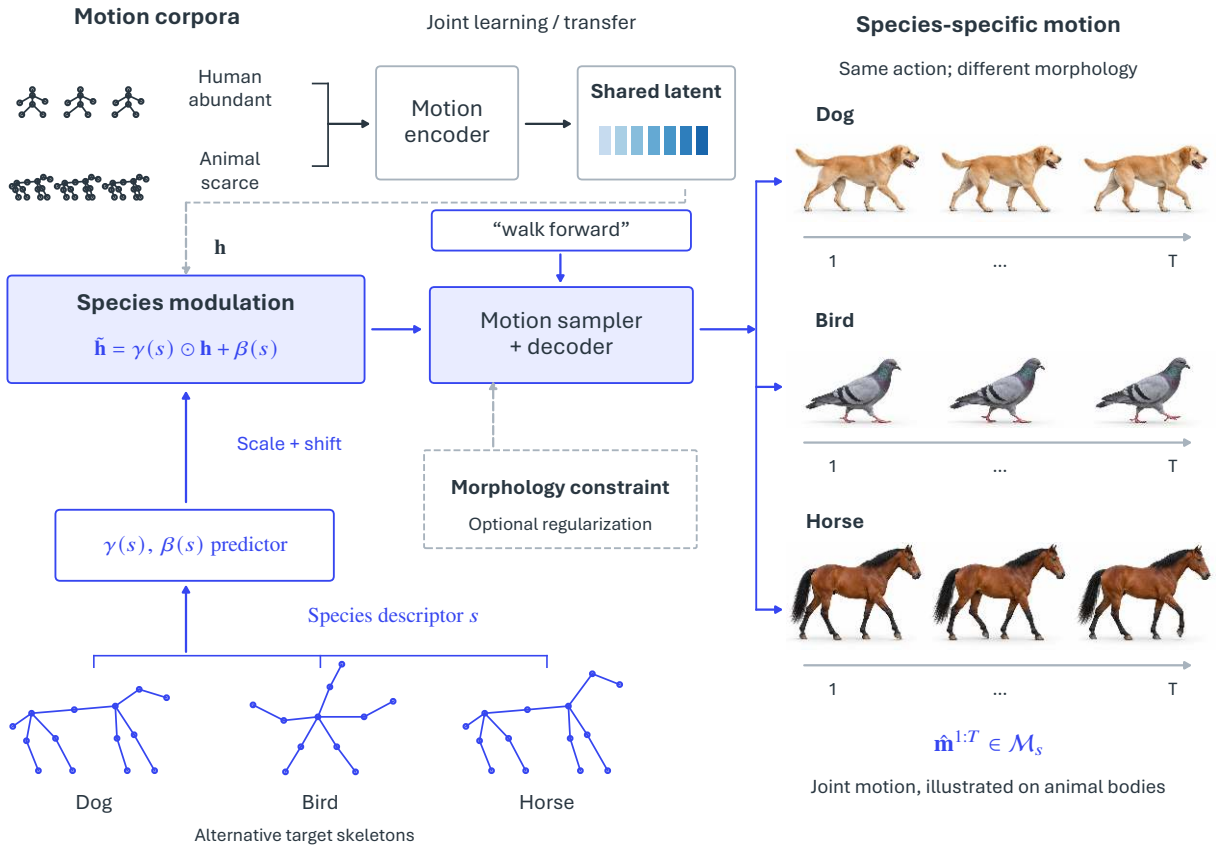


Figure 8.20: Abstract cross-species motion generation: human and animal corpora support a shared representation, species-dependent modulation (Eq. (8.58)) adapts features, and an action-conditioned sampler produces motion for the selected skeleton. Dog, bird, and horse sequences illustrate alternative target morphologies, not simultaneous outputs. The dashed constraint denotes optional morphology regularization; implementations vary across methods.

8.4.2.3 Benchmarks

Animal motion resources fall into two groups (Table 8.11). **Capture and pose-estimation corpora** supply ground-truth articulated motion but no language: RGBD-Dog [2241] and 3DDogs-Wild [2242] pair RGB-D or RGB video with marker-based mocap for canine pose, and DeformingThings4D [2243] provides synthetic 4D sequences with dense mesh-deformation annotations across humanoid and animal categories. **Text-to-motion corpora** drive recent progress: AnimalML3D [2236] is the first text–animal-motion dataset, and the field then scales sharply through Zoo-300K [2237] (built by the ZooGen pipeline with SinMDM and LLM annotators), AniMo4D [2239] (normalized to a 30-joint cross-species skeleton), and UniMo4D [2240], which

merges AniMo4D with HumanML3D and KIT-ML. Even the largest text-conditioned sets remain smaller and more morphologically fragmented than their human counterparts of §8.4.1.3, which is what forces the transfer-and-modulate methodology above.

Table 8.11: Benchmarks for animal motion generation. "Subjects / species" counts individual animals for the canine capture sets and species or categories elsewhere; "Texts / frames" gives text descriptions for text-conditioned corpora and frames for capture corpora.

Dataset	Type	Modality	Subjects / species	Sequences	Texts / frames
RGBD-Dog [2241]	Canine	RGB-D, mocap	5 dogs	5 motion types	8,346 frames
3DDogs-Wild [2242]	Canine	RGB, mocap	37 dogs	143 recordings	-
DeformingThings4D [2243]	Animal / humanoid	4D mesh	31 categories	1,972	122,365 frames
AnimalML3D [2236]	Animal / humanoid	4D	36 identities	1,240	-
Zoo-300K [2237]	Animal	3D mesh	65 categories	∼300K text–motion pairs	-
AniMo4D [2239]	Animal	4D	114 species	78,149	185,435 texts
UniMo4D [2240]	Animal + human	4D	∼115 species	∼119K	-

Evaluation Metrics. Animal motion evaluation reuses the three-axis human protocol of §8.4.1.3, adapted for species-specific biomechanics. *Fidelity* is measured by FID against real movement distributions and by MPJPE against ground-truth poses, augmented with physical-plausibility checks for gait-cycle regularity, ground contact, and the absence of foot sliding or limb penetration. *Condition alignment* carries over R-Precision and Multimodal Distance to score text-/action-to-motion correspondence, and *diversity / multimodality* metrics quantify behavioral variety within and across species. The animal-specific addition is perceptual and biomechanical judgment of naturalness, balance, and anatomical consistency, which matters more for quadrupeds and multi-limbed morphologies than for the single human skeleton.

8.4.2.4 Challenges and Trends

Physical realism is the most basic gap: most methods do not simulate ground-contact forces, quadruped balance, or the fluid dynamics of swimming and flight, so artifacts the human setting has largely learned to suppress reappear, and differently for every body plan. **Behavioral diversity** is harder than for humans because one instruction such as "walk stealthily" realizes into different gaits, postures, and speeds across species, so the conditional distribution to be captured is far wider while the data to capture it are scarcer. **Unseen species and radical morphologies** remain out of reach: species-aware modulation (Eq. (8.58)) and morphological-consistency constraints generalize across the skeletons seen in training, but insects and winged or aquatic creatures violate the shared-skeleton, fixed-topology assumptions on which current methods rest. AniMo and X-MoGen already move toward treating morphology as a variable rather than a template, and topology-flexible representations are the direction this points.

8.4.3 Vehicle Motion Planning

Where §8.4.1 and §8.4.2 generate the motion of an articulated body, vehicle motion planning generates the trajectory of a (near-)rigid agent moving through a traffic scene. The body is far simpler, since a vehicle has essentially no internal articulation, but the difficulty moves outward into the environment: a planned trajectory must obey vehicle dynamics, stay on the drivable surface dictated by an HD map, and remain socially compatible with the future motion of many surrounding agents whose intentions are unobserved. This is the setting where motion generation becomes inseparable from *multi-agent* spatial reasoning and physical feasibility: trajectory forecasting (predicting how others will move) and planning (deciding how the ego vehicle should move) are two readings of the same generative object.

8.4.3.1 Task Definition

Let a trajectory be a sequence of agent states $\boldsymbol{\tau}^{1:H} = (\mathbf{s}^1, \ldots, \mathbf{s}^H)$ over a horizon H, where each state $\mathbf{s}^t$ collects position, heading, and velocity. The driving signal is now a structured scene context $\mathbf{c} = (\mathbf{s}^{-P:0}, \mathcal{G}, \mathcal{A})$ comprising the agent's own past states, an HD-map / lane graph $\mathcal{G}$, and the observed histories of neighboring agents $\mathcal{A}$. With κ encoding this context, vehicle motion planning learns

$$G_\theta^{\text{veh}} : \mathcal{C}' \to \mathcal{T}, \qquad \hat{\boldsymbol{\tau}}^{1:H} \sim p_\theta\big(\boldsymbol{\tau}^{1:H} \mid \kappa(\mathbf{c})\big) \quad \text{s.t.} \quad \mathbf{s}^{t+1} = f_{\text{dyn}}(\mathbf{s}^t, \mathbf{u}^t), \tag{8.59}$$

the conditional-sampler template of Eq. (8.54) with the pose sequence replaced by a state trajectory and an added kinematic-feasibility constraint f_{dyn} (*e.g.*, a bicycle model) that distinguishes this setting from articulated-body generation. Dropping $\mathcal{A}$ recovers single-agent forecasting; supplying a goal or reward in place of $\mathbf{c}$ turns prediction into planning.

8.4.3.2 Methodology

Prediction methods vary along two axes: how the scene context is *encoded* and how the multimodal future is *decoded*. Early encoders rasterize the scene into a bird's-eye-view image, but the field has converged on *vectorized* encoders that represent lanes and agent tracks as polylines: VectorNet [2244] and LaneGCN [2245] build graph representations of the HD map, and transformer encoders such as Wayformer [2246] and the Motion Transformer (MTR) [2247] attend jointly over map and agents; graph-attention predictors [2248] model spatial and temporal interaction in the same spirit. On the decoding side, the defining property is multimodality, since a single context admits many plausible futures, so models predict a set of K modes with probabilities,

$$p_\theta(\boldsymbol{\tau} \mid \kappa(\mathbf{c})) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(\boldsymbol{\tau};\, \mu_k(\kappa(\mathbf{c})),\, \Sigma_k)\,, \tag{8.60}$$

rather than the single point estimate of Eq. (8.55), which here would average distinct maneuvers (turning *vs.* going straight) into an infeasible mean. The earliest learned predictors modeled social interaction directly in agent-coordinate space, Social-LSTM [2249] through pooled recurrent states and Social-GAN [2250] through adversarial sampling, and Trajectron++ [2251] added dynamic feasibility to a graph-structured latent-variable model. Goal-based decoders make the modes explicit: DenseTNT [2252] samples dense candidate endpoints on the map and scores trajectories conditioned on them, exposing the maneuver structure that Eq. (8.60) captures only implicitly. Generative decoders replace the fixed mixture with a learned sampler: MotionDiffuser [2253] denoises whole multi-agent trajectory sets jointly.

Planning turns the same machinery on the ego vehicle. ChauffeurNet [2254] learns the ego policy by imitating expert driving and augments the demonstrations with synthesized perturbations and worst cases, so that the policy stays robust when it drifts off the data manifold. Figure 8.21 situates these families along the encode/decode axes.

8.4.3.3 Benchmarks

Progress is measured on large real-world driving logs that pair agent tracks with HD maps: nuScenes [445], Argoverse and Argoverse 2 [2255, 661], and the Waymo Open Motion Dataset [218], the last two offering the scale and interaction density that current transformer models require. Evaluation centers on the multimodal-aware displacement metrics that match Eq. (8.60): minimum average and final displacement error over the K predicted modes (minADE_K, minFDE_K), miss rate (the fraction of scenes where no mode lands within a tolerance of the ground truth), and the soft mean-average-precision of the Waymo protocol, which jointly scores accuracy and the calibration of mode probabilities π_k. Off-road and collision rates further test the feasibility constraint f_{dyn} that purely displacement-based scores ignore.

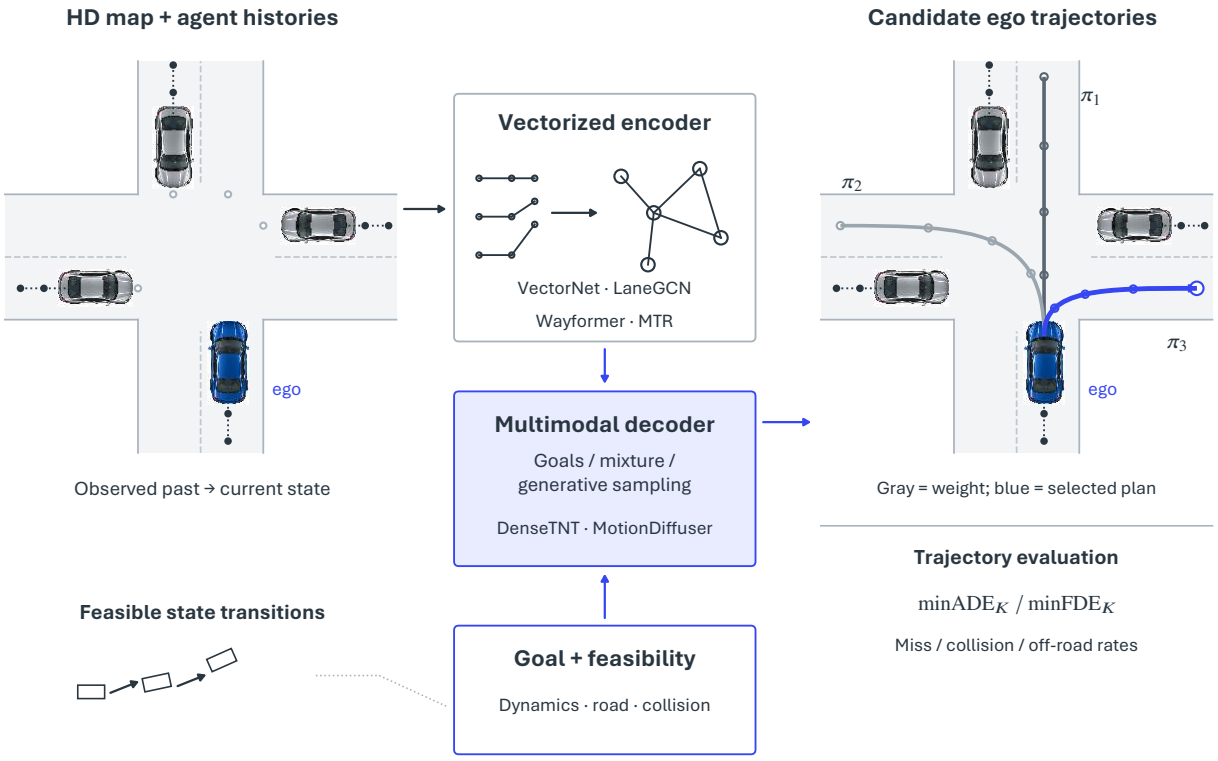


Figure 8.21: Vehicle motion planning as conditional trajectory generation: a vectorized encoder represents map polylines and agent interactions, and a multimodal decoder generates candidate futures (Eq. (8.60)). Gray intensity indicates mode weight; blue marks a plan selected using a goal or cost and feasibility constraints, not necessarily the highest-probability mode. Road geometry and trajectories are illustrative.

8.4.3.4 Challenges and Trends

A planner must assign trustworthy probabilities to the futures it enumerates, not merely list them: an over-confident missed mode of Eq. (8.60) is a safety failure rather than a metric error, so **calibrated multimodality** is the first open problem. **Feasibility and safety** are the second: displacement metrics reward trajectories a vehicle cannot physically execute, so feasibility has to be enforced inside the model (f_{dyn}) rather than checked afterward, and the long tail of near-collision scenarios is exactly where data are scarcest. The third is **closed-loop distribution shift**: models trained to imitate logged, open-loop futures drift once their own predictions feed back into the scene, the difficulty ChauffeurNet's worst-case synthesis was built to expose. Diffusion- and transformer-based scene-level samplers that forecast all agents jointly and run closed-loop are the current direction, and they connect directly to the controllable-video and world-model lines of §8.3 and §8.6, in which a driving world model both predicts and is conditioned on the ego plan.

8.4.4 Motion Editing

The subsections above *synthesize* motion from scratch; motion editing instead modifies an *existing* motion so that it satisfies a new constraint while leaving the rest intact. It is the temporal, articulated analogue of the image editing of §8.1.3 and the video editing of §8.3.5, and its demand is again selective preservation: change a targeted aspect (a trajectory, a joint, a style, a semantic action) while holding the body's identity, timing, and untouched portions fixed, which makes editing the controllability layer that closes the generate-and-refine loop for motion.

8.4.4.1 Task Definition

Given a source motion $\mathbf{m}^{\text{in}} \in \mathcal{M}$, an edit region $\mathbf{M} \in \{0,1\}^{T \times J}$ over time T and joints J, and an edit control $\mathbf{c}$ (a text instruction, target keyframes, a desired root trajectory, or a target joint), motion editing learns an operator

$$E_\theta : \mathcal{M} \times \{0,1\}^{T \times J} \times \mathcal{C}' \to \mathcal{M}, \qquad \hat{\mathbf{m}} = E_\theta\big(\mathbf{m}^{\text{in}}, \mathbf{M}, \kappa(\mathbf{c})\big)\,, \tag{8.61}$$

which lifts the image-editing operator of Eq. (8.8) into the motion domain. As in image and video editing, Eq. (8.61) encodes two competing objectives, *editability* (the output must realize the control $\mathbf{c}$ inside $\mathbf{M}$) and *preservation* (it must match $\mathbf{m}^{\text{in}}$ outside $\mathbf{M}$ and remain a plausible, smooth motion), and methods differ chiefly in how they trade these off.

8.4.4.2 Methodology

Because a pretrained motion diffusion model already samples plausible motion, most editors realize Eq. (8.61) without retraining, by *conditioning the sampler* of Eq. (8.56) rather than learning a dedicated E_θ. The simplest mechanism is **masked infilling**: at every denoising step the observed entries are overwritten with the (noised) source while only the masked entries are generated,

$$\hat{\mathbf{m}} = (1 - \mathbf{M}) \odot \mathbf{m}^{\text{in}} + \mathbf{M} \odot G_\theta(\mathbf{m}_n,\, n,\, \kappa(\mathbf{c}))\,, \tag{8.62}$$

which directly enforces preservation by construction. This is the basis of motion in-betweening, where $1 - \mathbf{M}$ marks given keyframes: robust in-betweening [2256] predates the diffusion era, and CondMDI [2257] casts flexible keyframe-conditioned in-betweening as masked diffusion, while PriorMDM [2258] reuses a frozen motion-diffusion prior for editing and composition tasks such as joining and in-filling. A second family handles *spatial* controls that masking alone cannot express (a target root path, or a specific joint at specific times) through classifier-style guidance: GMD [2259] and OmniControl [2260] steer the denoiser toward arbitrary spatial constraints at any joint and time. A third, optimization-based family treats the initial diffusion noise itself as the free variable, solving $\min_{\mathbf{z}} \mathcal{L}_{\text{edit}}(\mathcal{D}(\mathbf{z}))$ so that the decoded motion satisfies an arbitrary differentiable criterion: DNO [2261] shows this single mechanism serves editing, in-betweening, and refinement as one universal motion prior. Finally, *semantic* editors take a natural-language instruction in place of a geometric mask: MotionFix [2262] learns text-driven edits from paired source–target motions, and LLM-driven approaches [2263] decompose an instruction into a program of primitive edits applied iteratively. Figure 8.22 groups these four mechanisms by how they obtain E_θ, paralleling the editing taxonomies of §8.1.3 and §8.3.5.

8.4.4.3 Benchmarks

Motion editing inherits the corpora of §8.4.1.3 (HumanML3D [2207] and BABEL [2208] over AMASS), used in editing protocols where keyframes, joints, or trajectories are held out as the constraint, together with the dedicated MotionFix [2262] benchmark of paired source–edit–instruction triples that supplies the ground truth most editing tasks lack. Evaluation must report the editability–preservation pair jointly: editability through control adherence (instruction-conditioned R-Precision, target-joint or trajectory error) and overall realism (FID, foot-skating), and preservation through the deviation from the source outside the edit region. As in video editing, single scalars correlate weakly with perceived edit quality, so reporting a tuple, and increasingly a learned or VLM-style judge of instruction compliance, is becoming standard.

8.4.4.4 Challenges and Trends

The central tension is the **editability–preservation trade-off** of Eq. (8.61): pushing harder on a control propagates unwanted change through the kinematic chain (editing one arm disturbs balance and foot contact), so a good editor must respect the body's global coordination, not just the masked entries. This is

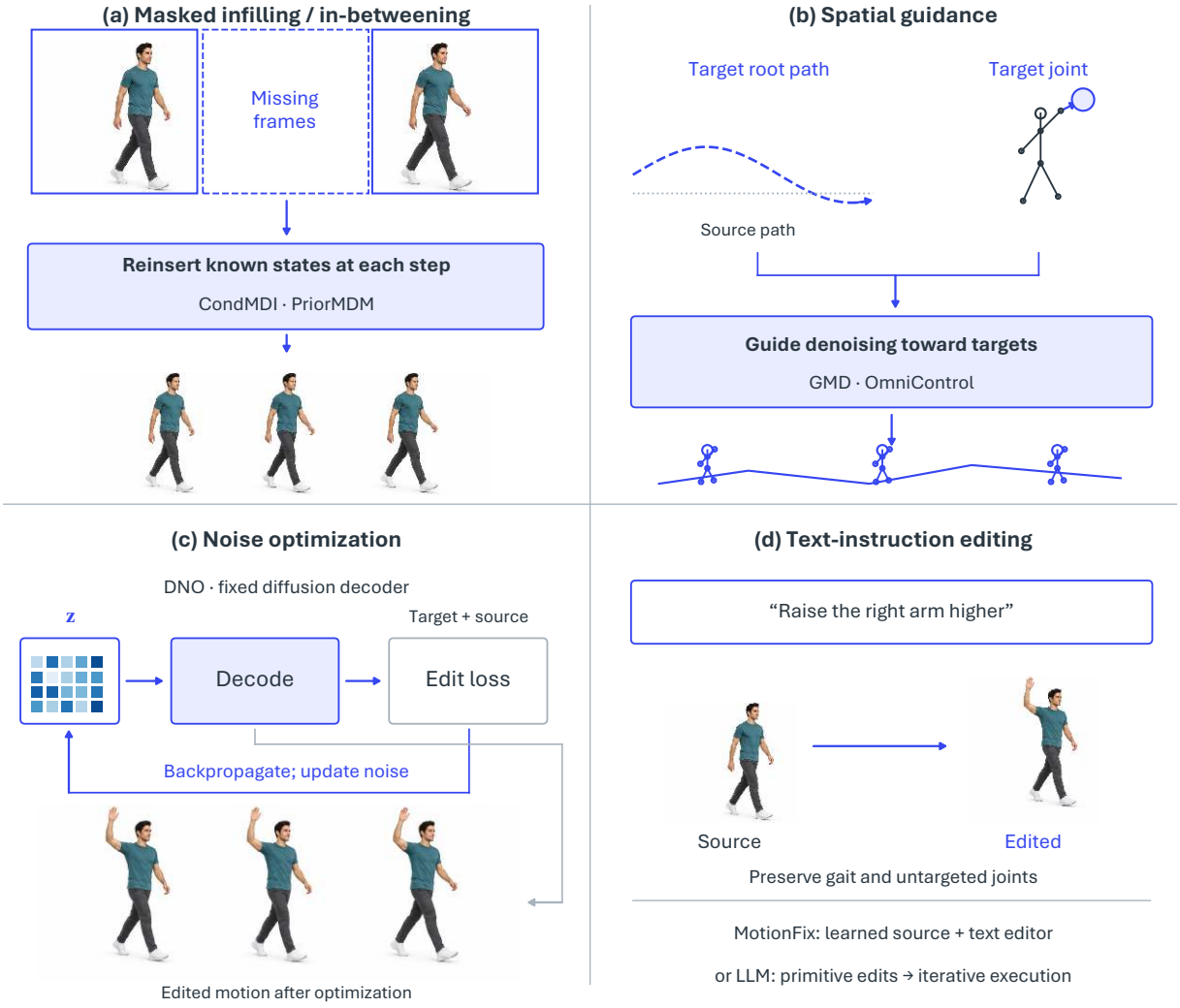


Figure 8.22: Four motion-editing mechanisms: (a) reinsert known states while infilling missing frames; (b) guide denoising with root-path or joint targets; (c) optimize initial noise through a fixed decoder using an edit loss; and (d) edit source motion from text, using a learned editor or iterative primitive edits. Rendered people illustrate edits rather than measured model outputs.

compounded by **sparse-control ambiguity**: a single keyframe or a root path under-determines the full-body motion, and the model must fill the gap plausibly, which the guidance and noise-optimization families address with differing robustness. Underlying both is an **evaluation** gap sharper than in synthesis: because the "correct" edit is rarely unique, ground-truth-based metrics under-reward valid alternatives, motivating the paired-triplet and judge-based protocols above. The trend mirrors the rest of §8.4 and §8.3: from per-mechanism, training-free editors toward unified models that generate, control, and edit motion within a single conditional sampler, increasingly driven by natural-language instructions rather than geometric masks.

8.5 4D Spatial Generation



ENERATIVE modeling has moved outward from static images [185, 1671] to video [2012, 2011, 2006, 2264], multi-view imagery [1750, 1749, 1759, 2265, 2266], and static 3D assets [1743, 1747], yet each

stops short of a scene that is at once three-dimensional and alive. A 3D asset holds geometry but not motion; a video holds motion, but no view-consistent geometry one can navigate or simulate. 4D generation is where the two meet: the synthesis of dynamic 3D content whose geometry stays view-consistent as it moves and deforms over time [2267, 2268, 2269, 2270, 1769]. This is the representation an embodied system needs to rehearse how a scene will unfold, and it underwrites photorealistic digital humans [2271, 2265, 2266], interactive AR/VR [2272, 2273], and high-fidelity simulation for autonomous driving and embodied AI [2274, 2275].

Two advances make this tractable. Temporal 3D representations, deformation-aware fields over NeRFs [119], point clouds, and Gaussian splatting [121, 2275], supply a geometry that can bend and move without losing view consistency, while the extension of generative diffusion into this domain supplies the realism and controllability to drive it. Together they open dynamic scene synthesis, 4D editing, and interactive agent simulation, at the cost of the difficulty this section returns to throughout: holding long-range temporal coherence while still granting flexible, user-driven control.

Formally, we cast 4D generation as the latent-generator framework of §8.1.1 unrolled along time over a shared 3D representation. Let $\mathcal{R}$ be the 3D-representation space of Eq. (8.9) (mesh, NeRF, 3DGS, point cloud, and so on), and let $\kappa : \mathcal{C} \to \mathcal{C}'$ encode a conditioning signal $\mathbf{c} \in \mathcal{C}$ (text, an image, a video, a static 3D asset, or a combination thereof). 4D generation learns a conditional sampler

$$G_\theta^{\text{4D}} : \mathcal{C}' \to \mathcal{R}^{[0,T]}, \qquad \hat{\mathbf{r}}^{1:T} \sim p_\theta\big(\mathbf{r}^{1:T} \mid \kappa(\mathbf{c})\big)\,, \tag{8.63}$$

that emits a time-indexed sequence of 3D representations $\hat{\mathbf{r}}^{1:T}$ with view-consistent geometry and coherent motion. Two limiting cases anchor the framework: with a single time step ($T = 1$) Eq. (8.63) recovers the static 3D generator of Eq. (8.9), and rendering each frame $\mathbf{r}^t$ to a 2D view recovers the video generator of Eq. (8.41), so 4D generation is precisely the 3D generator of §8.2 unrolled in time, or equivalently the video generator of §8.3 lifted from per-frame pixels to a shared, renderable geometry. The subsections are organized by the conditioning input (text, image, video, a static 3D asset, or several at once). Figure 8.23 previews this organization together with two methodological strategies that recur across all five, per-scene score distillation and explicit "generate-then-reconstruct" supervision; feed-forward generators and rig-based animation of meshes appear alongside them in the subsections where they apply.

8.5.1 Text-to-4D Generation

Text-to-4D generation synthesizes dynamic content directly from natural language, supporting text-driven asset creation and character animation [2171, 2276]. Its progress has been propelled by large language models for semantic understanding and by video foundation models that supply motion priors.

8.5.1.1 Task Definition

Text-to-4D generation specializes the 4D sampler of Eq. (8.63) to a textual condition $\mathbf{c} = \mathbf{x}$ encoded by the text encoder ϕ of Eq. (8.2),

$$\hat{\mathbf{r}}^{1:T} \sim p_\theta\big(\mathbf{r}^{1:T} \mid \phi(\mathbf{x})\big)\,. \tag{8.64}$$

Because language fixes only semantics and leaves geometry and motion grossly under-determined, the sampler must hallucinate both a plausible 3D shape and its temporal evolution from the prompt alone, the source of the semantic-to-motion ambiguity examined in §8.5.1.4.

8.5.1.2 Methodology

Two strategies recur throughout this section, implicit distillation and explicit supervision, implemented across various 4D representations (Figure 8.23). The more common implicit strategy leverages Score Distillation Sampling (SDS) to indirectly optimize the geometry and motion of a 4D representation, using pre-trained

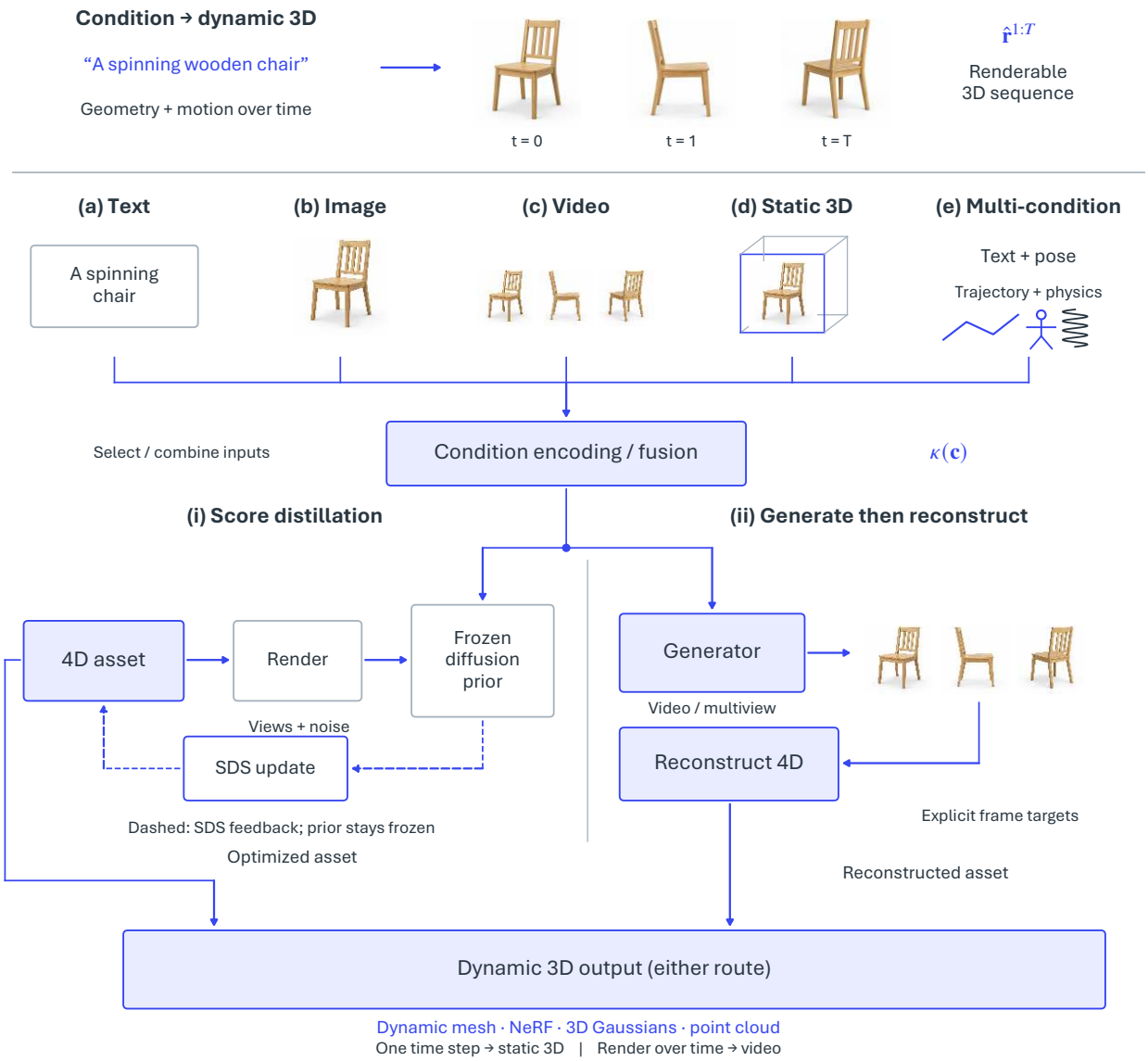


Figure 8.23: 4D spatial generation by conditioning input, (a)–(e), with two representative strategies: (i) score distillation from a frozen diffusion prior (dashed: SDS feedback), and (ii) generating video or multiview targets, then reconstructing a dynamic 3D asset. The strategies are alternatives that may be combined; the chair sequence is illustrative.

diffusion models as frozen priors [2267, 2268]. Formally, it applies a score-distillation update to per-scene parameters θ [1741],

$$\nabla_\theta \mathcal{L}_{\text{SDS}} = \mathbb{E}_{s,\epsilon,\pi}\left[w(s)\left(\epsilon_\phi(\mathbf{z}_s; s, \kappa(\mathbf{c})) - \epsilon\right) \frac{\partial \mathbf{x}_\theta}{\partial \theta} \right], \quad \mathbf{z}_s = \alpha_s \mathbf{x}_\theta + \sigma_s \epsilon, \tag{8.65}$$

where $\mathbf{x}_\theta = \text{Render}(\hat{\mathbf{r}}_\theta^{1:T}, \pi)$ is a differentiable rendering at camera π, s indexes diffusion noise levels, and ϵ_ϕ is a frozen 2D/video diffusion prior. The update differentiates through the renderer while treating the prior noise residual as the supplied gradient signal; this lifts the static score-distillation objective of §8.2.1.2 to a temporal sequence. In contrast, explicitly supervised methods initialize the 4D generation from text-to-3D or text-to-video outputs and use these generated assets as direct supervisory signals [2275, 191, 2277, 2270, 2007, 2278].

This "generate-then-reconstruct" strategy first synthesizes target frames $\hat{\mathbf{y}}$ with a fast generator $\mathcal{G}_{2D}$ and then fits the 4D asset to them,

$$\hat{\mathbf{y}} = \mathcal{G}_{2D}(\mathbf{c}), \qquad \hat{\mathbf{r}}^{1:T} = \arg\min_{\mathbf{r}^{1:T}} \sum_{\pi,t} \left\| \mathrm{Render}(\mathbf{r}^t, \pi) - \hat{\mathbf{y}}_\pi^t \right\|^2, \tag{8.66}$$

replacing score-distillation guidance with explicit pixel targets; reconstruction may itself require per-scene optimization, so a speed advantage depends on the implementation. The two strategies are realized across mesh, NeRF, and 3DGS representations.

Mesh-based Methods. Meshes offer explicit geometric control, which suits precise text-driven shape manipulation and animation. CT4D [2276] implements a three-stage "Generate-Refine-Animate" pipeline in which a mesh is generated from text, its texture refined, and its vertices animated; TextMesh4D [2279] deforms a generated mesh through a per-face Jacobian deformation field under video-diffusion priors; and AvatarCLIP [2171] generates a static avatar zero-shot and animates it by registering motion to an SMPL skeleton.

NeRF-based Methods. NeRFs represent continuous geometry and complex temporal variation well. MAV3D [2267] introduced multi-stage SDS optimization that refines texture, geometry, and motion with priors from both T2I and T2V diffusion models; 4D-fy [2280] blends 3D-aware, image, and video score signals in a hybrid SDS; and Dream-in-4D [2268] first generates a static asset with a multiview diffusion model and then adds dynamics with a video diffusion model.

Gaussian Splatting-based Methods. 4D Gaussian Splatting (4DGS) [2272, 191] renders efficiently and differentiably, which speeds up text-driven dynamic scene synthesis. Early methods followed the two-stage implicit pipeline of Eq. (8.65): AYG [2270] generates a static 3D Gaussian model from text via SDS and adds dynamics under a T2V prior, and Comp4D [2278] uses an LLM to parse a complex prompt into several dynamic objects and their trajectories for a compositional scene. Later works moved to explicit supervision (Eq. (8.66)): PLA4D [2273] aligns the 4D Gaussians pixel by pixel with generated video targets, and 4Real [2281] uses generated multiview video frames as the reference for photorealistic scene-level generation. For digital humans, AvatarGO [2282] generates 4D human–object interactions zero-shot by animating a text-to-3D human with the SMPL-X skinning function.

8.5.1.3 Benchmarks

Text-to-4D, like the other subfields of this section, has no standardized benchmark, because large-scale dynamic 3D data with text descriptions is hard to capture and annotate. Training and evaluation instead adapt static 3D repositories (Table 8.12): Diffusion4D [2283] filtered Objaverse and Objaverse-XL [1764, 1765] down to 54K high-quality animated assets, and SV4D [2284] built ObjaverseDy by adaptive sampling and viewpoint adjustment. These are synthetic renderings and lack the multimodal annotations, text in particular, needed to train and evaluate text-conditioned models robustly.

Table 8.12: Datasets used for 4D generation across the subfields of §8.5.

Dataset	Content	Scale	Used for
Objaverse-XL [1765]	Static and animated 3D objects	10M+ objects	Source pool for dynamic subsets
Diffusion4D [2283]	Animated assets curated from Objaverse / Objaverse-XL	54K assets	Training image- / video-to-4D
ObjaverseDy [2284]	Animated Objaverse objects, re-rendered	-	Training and evaluating video-to-4D
Consistent4D [2269]	Synthetic and real monocular videos	-	Evaluating video-to-4D

Evaluation Metrics. Evaluation across 4D generation combines user studies with metrics borrowed from image and video generation, and the subsections that follow use this same set. Semantic alignment with a prompt is measured by CLIP Score [143] between rendered views and the text; fidelity by PSNR [2285], SSIM [1884], and LPIPS [1885] where a reference view exists; temporal quality and motion by the Fréchet Video Distance (FVD) [1950]; and overall plausibility and aesthetics by user studies. None of these scores the two properties that make content 4D, cross-view consistency and long-range spatio-temporal coherence.

8.5.1.4 Challenges and Trends

The challenge specific to text-to-4D is semantic-to-motion ambiguity: a prompt such as "a dancing dog" admits endlessly many movements, styles, and trajectories, and methods that lean on generic T2V priors struggle to produce diverse, fine-grained motion that matches a nuanced description. One response uses LLMs for motion planning, as Comp4D [2278] does when it decomposes a prompt into a structured spatio-temporal plan for several interacting objects; another targets human–object interaction [2282], moving toward story-driven dynamic scenes.

8.5.2 Image-to-4D Generation

Image-to-4D generation synthesizes dynamic content from a single static image, combining the spatial information in the image with temporal priors learned by video models [2286, 2287]. No large image–4D paired dataset exists, which has ruled out direct end-to-end training and pushed the field toward optimization and toward generated supervision.

8.5.2.1 Task Definition

Image-to-4D generation specializes the 4D sampler of Eq. (8.63) to a single-image condition $\mathbf{c} = I$,

$$\hat{\mathbf{r}}^{1:T} \sim p_\theta\big(\mathbf{r}^{1:T} \mid \kappa(I)\big)\,, \tag{8.67}$$

where the source view at $t = 0$ is fixed by I while every unseen viewpoint and time step must be inferred. Both the unseen 3D geometry and the entirely absent temporal dynamics have to be inferred from one 2D view, which requires combining the spatial cues of the input with temporal priors from large video models [2288, 2289, 2287].

8.5.2.2 Methodology

Both strategies of §8.5.1.2 apply, and most methods use 4D Gaussian Splatting as the representation [2288, 2286].

Implicit Optimization with Diffusion Priors. These methods instantiate Eq. (8.65), typically combining a multiview diffusion model for geometric consistency with a video diffusion model for temporal coherence. Animate124 [2289] first generates a static NeRF from the image and then refines its dynamics with a video-diffusion SDS loss, and Diffusion^2 [2286] reconciles the scores of heterogeneous priors (multiview and video) through Variance-Reducing Sampling, improving both multiview and temporal consistency of the resulting 4D Gaussians.

Explicit Supervision with Generated Data. The slow, iterative nature of SDS motivates the two-stage pipeline of Eq. (8.66), in which an image-to-video model first generates an intermediate video that then supervises the 4D reconstruction directly. DreamGaussian4D (DG4D) [2288] generates a static 3D Gaussian model and learns a deformation field to animate it, using the generated video for texture refinement and dynamic supervision, and EG4D [2287] avoids SDS entirely by chaining image-to-video and image-to-multiview models into coherent image sequences that guide a fast reconstruction.

Unified and Tuning-free Generation. GenXD [2290] trains one model on both 3D and 4D data, using multiview-temporal modules that separate camera motion from object motion, and Free4D [190] builds a 4D scene from a single image without fine-tuning any model, distilling pretrained image-to-video and multiview priors into a consistent 4D representation.

Specialized Methods for Human Generation. Dynamic human avatars form a sub-domain in which strong body priors raise fidelity. Human4DiT [2291] uses a hierarchical 4D transformer that integrates human-specific features, camera parameters, and temporal signals to generate 360-degree human animations from a single image, and Disco4D [2292] models the body with SMPL-X while representing clothing separately with Gaussian splats, allowing detailed animation of clothed humans.

8.5.2.3 Benchmarks

With no image–4D pairs, methods are evaluated on images from Objaverse [1764] or on in-the-wild photographs chosen by the authors, which makes comparison across papers unreliable. The metric set is that of §8.5.1.3, applied to the source view and to pseudo-ground-truth novel views where available.

8.5.2.4 Challenges and Trends

Inferring a complete shape and a plausible motion from one viewpoint is extremely ill-posed, which shows up as geometric inconsistency in unseen views (*e.g.*, the "Janus problem") and as unnatural or repetitive motion, since every dynamic detail must be hallucinated. Domain-specific priors are the main remedy: Human4DiT [2291] and Disco4D [2292] regularize geometry and motion with parametric body models and reach much higher fidelity, at the cost of generality.

8.5.3 Video-to-4D Generation

Video-to-4D generation lifts a single-view video into a dynamic 4D representation that can be viewed from new angles with consistent geometry and motion, turning ordinary video into interactive 4D content. Annotated video-to-4D datasets are hard to curate, so current methods build on pretrained video diffusion priors, either as multi-view supervision or through score distillation.

8.5.3.1 Task Definition

Video-to-4D generation specializes the 4D sampler of Eq. (8.63) to a monocular video condition $\mathbf{c} = \mathbf{v}^{1:T}$,

$$\hat{\mathbf{r}}^{1:T} \sim p_\theta\big(\mathbf{r}^{1:T} \mid \kappa(\mathbf{v}^{1:T})\big)\,, \tag{8.68}$$

so that re-rendering $\hat{\mathbf{r}}^{1:T}$ from the input viewpoint reproduces $\mathbf{v}^{1:T}$ while novel views remain geometrically consistent. Unlike image-to-4D (Eq. (8.67)), the temporal dynamics are now *observed* and the residual ambiguity is purely geometric: the unseen off-axis structure.

8.5.3.2 Methodology

Most methods densify the monocular input into multi-view supervision and reconstruct against it (Eq. (8.66)), some add score distillation (Eq. (8.65)), and the newest predict the 4D asset feed-forward.

Mesh-based Methods. Meshes are interpretable and efficient to animate. DreamMesh4D [2293] obtains a coarse mesh and drives it with a sparse deformation graph and linear blend skinning (LBS), reaching strong spatio-temporal consistency more cheaply than implicit volumetric methods, and V2M4 [2294] reconstructs an animated, textured mesh asset directly from a monocular video.

NeRF-based Methods. NeRF-based methods model the continuous volumetric content of a video well, and they overcome the single view by generating data. Consistent4D [2269] densifies the monocular input

through temporal and multiview interpolation before optimizing a NeRF with SDS, 4Diffusion [2295] fine-tunes a video diffusion model with a learnable motion module on synthetic multiview data so that it produces consistent multiview video to guide generation, and SV4D [2284] completes missing views and frames with a diffusion model and uses the resulting multiview video to supervise the reconstruction.

Gaussian Splatting-based Methods. 4DGen [2296] grounds 4D Gaussian generation in the input video with spatial-temporal pseudo labels; Efficient4D [2297] converts the video into a multiview image matrix with a multiview-consistent generator (SyncDreamer) for rapid reconstruction; SC4D [2298] first fits sparse control Gaussians for coarse shape and motion and then drives dense Gaussians through LBS; and L4GM [2299] predicts the 4D Gaussians of a whole clip in a single feed-forward pass, trained on rendered animated objects.

8.5.3.3 Benchmarks

Video-to-4D has the one dedicated evaluation set in this section, released with Consistent4D [2269]: synthetic and real monocular videos with training and test splits. The metrics are those of §8.5.1.3, with reconstruction fidelity measured at the input view and novel-view quality measured where synthetic ground-truth multiview video exists.

8.5.3.4 Challenges and Trends

The video constrains motion but not the unseen 3D structure, which leads to geometric distortion, depth ambiguity, and a "painted-on" look in novel views. Explicit geometric structure is the main remedy: DreamMesh4D [2293] and SC4D [2298] constrain the reconstruction with meshes or sparse control points under LBS, which improves both coherence and efficiency, and combining such structure with the appearance modeling of neural representations is the obvious next step.

8.5.4 3D-to-4D Generation

3D-to-4D generation animates a static 3D model, so that the geometry is given and only motion and deformation must be synthesized, which makes the large stock of existing 3D assets usable for simulation, virtual reality, and character animation. Unconditional generators of dynamic neural fields, such as HyperDiffusion [2300], which runs diffusion directly on the weights of neural fields, sample 4D content without a given asset; 3D-to-4D instead starts from one, either through explicit mesh deformation and rigging or through neural and physics-based deformation models.

8.5.4.1 Task Definition

Here the condition is itself a 3D representation $\mathbf{c} = \mathbf{r}_0 \in \mathcal{R}$, so only its temporal evolution must be generated. We therefore specialize Eq. (8.63) to a *deformation operator* that takes the static asset and an optional auxiliary driving signal $\mathbf{c}_{\text{aux}}$ (text, audio, or a motion sketch) to a sequence,

$$\hat{\mathbf{r}}^{1:T} = \mathcal{D}_\theta(\mathbf{r}_0,\, \kappa(\mathbf{c}_{\text{aux}}))\,, \qquad \hat{\mathbf{r}}^0 = \mathbf{r}_0, \tag{8.69}$$

with the first frame pinned to the input asset. This places 3D-to-4D in the same family as the editing operators of §8.2.5 and §8.4.4: it transforms an existing representation rather than synthesizing one from scratch.

8.5.4.2 Methodology

Methods realize the deformation operator $\mathcal{D}_\theta$ of Eq. (8.69) either with explicit mesh structure, increasingly through learned rigging, or with neural and physics-informed deformation fields.

Mesh-based Methods. Meshes give explicit structure and precise control over deformation. Motion3DGAN [2301] generates 3D facial animation from a static face mesh by first predicting the displacement

of facial landmarks for a target expression and then deforming the full mesh, with a GAN keeping the sequence temporally coherent. A recent line makes arbitrary meshes animatable by predicting the rig itself: MagicArticulate [2302] and RigAnything [2303] predict skeletons and skinning weights feed-forward, Puppeteer [2304] rigs a mesh and then animates it by fitting the rig to videos generated from renderings of the mesh, and AnimateAnyMesh [2305] generates text-driven mesh animation in a single forward pass with a rectified-flow model over a dynamic-mesh latent.

NeRF-based Methods. ElastoGen [2306] generates physically grounded elastodynamic behavior: it rasterizes a static 3D model and then predicts and refines deformation fields iteratively with recurrent and 3D convolutional modules, so that the motion obeys physical principles.

8.5.4.3 Benchmarks

No benchmark with ground-truth dynamic sequences exists; methods are evaluated on a few individual assets, often faces or generic objects, mostly by user study and with the metrics of §8.5.1.3 where a reference animation is available. Physical plausibility, specific to this subfield, can be measured for physics-informed methods such as ElastoGen [2306] as deviation from a high-fidelity simulator or as violation of conserved quantities such as energy and momentum.

8.5.4.4 Challenges and Trends

Even with the geometry given, the space of plausible animations of a static model is unbounded, so without guidance the task is unconstrained, and current methods specialize to narrow categories such as faces. Physical priors constrain it (ElastoGen [2306] grounds motion in elastodynamics), learned rigging makes arbitrary assets animatable, and pairing the asset with text, audio, or motion sketches turns open-ended generation into controllable animation.

8.5.5 Multi-conditional 4D Generation

Multi-conditional 4D generation steers dynamic scene synthesis with several signals at once: semantic conditions (text), appearance conditions (images, video), structural conditions (skeletal poses, explicit trajectories), and rule-based priors (differentiable physics). The central design problem is to fuse them so that each is respected without one overpowering the others.

8.5.5.1 Task Definition

We generalize the single-condition cases of Eqs. (8.64)–(8.69) to a set of K conditions, each mapped by its own encoder κ_k and fused into the shared embedding:

$$\hat{\mathbf{r}}^{1:T} \sim p_\theta\big(\mathbf{r}^{1:T} \mid \kappa_1(\mathbf{c}_1), \ldots, \kappa_K(\mathbf{c}_K)\big)\,, \tag{8.70}$$

which recovers a single-modality sampler when $K = 1$. Users can then specify not only what is generated (semantics) but how it behaves (motion path, physical behavior, artistic style), which complex applications such as character animation and driving simulation require.

8.5.5.2 Methodology

Methods fuse the encoders κ_k of Eq. (8.70), usually combining score distillation (Eq. (8.65)) with explicit supervision (Eq. (8.66)) tailored to each condition.

Motion and Trajectory Control. TC4D [2307] lets users define a spline trajectory that guides both the global rigid transformation and the local deformation of an object, with text supplying the semantics. For human animation, skeletons are the natural condition: Beyond Skeletons [2308] integrates text and image conditions

with the skeleton to refine motion, and STAR [2309] uses skeleton-aware SDS and motion retargeting so that avatar movement is both visually coherent and anatomically accurate.

Physics-Informed Control. Sync4D [2310] and Phy124 [2311] integrate differentiable Material Point Method (MPM) simulation, so that the generated content follows realistic dynamics such as elasticity and collisions under video or text guidance.

User-Centric and Artistic Control. Generative Rendering [2312] uses a pre-existing dynamic mesh to guide a pretrained text-to-image model, giving precise control over the style and detail of the rendered output, and sketch-based interfaces [2313] let users draw the dynamics they want.

Complex, Application-Specific Control. MagicDrive3D [2274] generates dynamic driving scenes from 2D bounding boxes, Bird's-Eye-View (BEV) maps, agent trajectories, and text together, producing tightly controlled traffic scenarios.

8.5.5.3 Benchmarks

No benchmark exists; building one would require 4D assets annotated with several aligned conditions (*e.g.*, a clip with a description, a skeletal motion sequence, and a trajectory map). Text adherence is scored with the CLIP Score of §8.5.1.3, while adherence to sketches or trajectories relies on task-specific errors (*e.g.*, distance from the specified trajectory) or on user studies, which remain the only holistic judge of how well a combination of inputs was followed.

8.5.5.4 Challenges and Trends

Semantic guidance from text, geometric constraints from skeletons, and rule-based priors from physics can contradict one another, and keeping each respected without one overpowering the rest while preserving fidelity is the open problem. Domain-specific stacks such as MagicDrive3D [2274] sidestep it by fixing the set of inputs for one application, and interactive interfaces such as sketches [2313] point toward iterative creation in which several control channels are adjusted in turn.

8.5.6 Summary, Overarching Challenges and Future Directions

Table 8.13 compares the five subfields as specializations of the 4D sampler of Eq. (8.63): image- and video-to-4D progressively replace hallucinated structure with observed pixels, 3D-to-4D is given the geometry and generates only motion, and multi-conditional generation fuses several signals at once.

Data Scarcity and Generative Priors. Large, annotated 4D datasets are the field's main missing ingredient: real dynamic 3D data with multi-view consistency and multimodal annotation (text, motion capture) is expensive and technically hard to collect, which has largely ruled out end-to-end training. The field has therefore converged on distilling pretrained text-to-image, text-to-video, and text-to-3D models, the de facto way to guide 4D synthesis.

From Score Distillation to Explicit Supervision. Score distillation [1741] converges slowly, is costly and unstable, and produces multi-faced geometry and flickering textures. The generate-then-reconstruct pipeline of DreamGaussian4D [2288] and EG4D [2287], which first generates target videos and then reconstructs against them, is replacing per-prompt SDS, and feed-forward predictors such as L4GM [2299] and AnimateAnyMesh [2305] take the next step by removing per-scene optimization altogether.

4D Gaussian Splatting as the Default Representation. NeRF-based works rendered well but trained and rendered slowly. 3D Gaussian Splatting [121] and its temporal extension offer real-time, high-fidelity rendering with efficient gradient-based optimization, and their explicit, point-based form is easier to edit

Table 8.13: The five subfields of 4D spatial generation; each sampler or operator specializes Eq. (8.63).

Subfield	Conditioning input	Sampler / operator	Dominant method families	Datasets / metrics
§8.5.1 Text-to-4D	Text prompt $\mathbf{x}$	G_θ^{4D}, $\kappa = \phi$ (Eq. (8.64))	SDS distillation or generate-then-reconstruct; mesh / NeRF / 3DGS; LLM motion planning	Objaverse-derived; CLIP, FVD, PSNR/SSIM/LPIPS, user study
§8.5.2 Image-to-4D	Single image I	G_θ^{4D}, $\kappa(I)$ (Eq. (8.67))	Diffusion-prior SDS; explicit video supervision; unified / tuning-free; human-specific priors	Objaverse / in-the-wild; PSNR/SSIM/LPIPS, FVD, user study
§8.5.3 Video-to-4D	Monocular video $\mathbf{v}^{1:T}$	G_θ^{4D}, $\kappa(\mathbf{v}^{1:T})$ (Eq. (8.68))	Video-diffusion densification; explicit geometric constraints (mesh / LBS); feed-forward	Consistent4D; PSNR/SSIM/LPIPS, FVD, user study
§8.5.4 3D-to-4D	Static asset $\mathbf{r}_0$	$\mathcal{D}_\theta$ deformation (Eq. (8.69))	Mesh deformation and learned rigging; physics-informed deformation	Per-asset (faces / objects); FVD, physical plausibility, user study
§8.5.5 Multi-conditional	K fused signals	G_θ^{4D}, $\{\kappa_k\}$ (Eq. (8.70))	Trajectory / skeleton / physics / sketch control; application-specific stacks	No standard suite; CLIP, PSNR/SSIM/LPIPS, FVD, user study

and to couple with physical simulation, so 4DGS has become the representation of choice for current methods [191, 2288, 2270].

Consistency, Control, and Physical Plausibility. Keeping geometry, appearance, and motion coherent across all viewpoints and over long horizons remains the central difficulty, and control is moving beyond single prompts toward fine-grained manipulation of motion, appearance, and interaction. Physical plausibility is the newest goal: integrating differentiable simulators, as PhysSplat [2275] and Phy124 [2311] do, makes generated dynamics more realistic and predictable. Progress on all three is hard to measure while evaluation relies on small, author-curated test sets and on image and video metrics that ignore cross-view and long-range consistency; 4D-aware benchmarks and metrics are a prerequisite for the rest.

8.6 World Modeling

Despite its growing popularity across AI research, the notion of a "world model" does not yet enjoy a unified or universally agreed-upon definition. Its conceptual roots trace back to the psychologist Kenneth Craik, who argued in his 1943 monograph *The Nature of Explanation* that the mind "carries a 'small-scale model' of external reality and of its own possible actions within its head". In its original form, a world model referred to an individual's internal representation of external reality; contemporary AI has generalized the concept far beyond mental representations to structured models of physical and spatial environments.

Fei-Fei Li has framed the gap between current AI systems and physical understanding in these terms [5]: modern LLMs resemble "wordsmiths in the dark", fluent in symbols but divorced from the physical world that gives them meaning, and capable AI must instead understand, act within, and reason about three-dimensional space. On this view world models are new-generation *generative* models that simulate, predict, and imagine the world's dynamics rather than only describe it, which ties world modeling directly to spatial intelligence.

Complementing Li's generative perspective, another influential formulation comes from Yann LeCun [17]. He describes a world model as a system capable of predicting how the world will evolve given its current state and an action. Formally, it consumes the current state $S(t)$ and an action $A(t)$, and produces a prediction of the next state $S'(t+1)$; in many cases, it also integrates external sensory observations $O(t)$ to

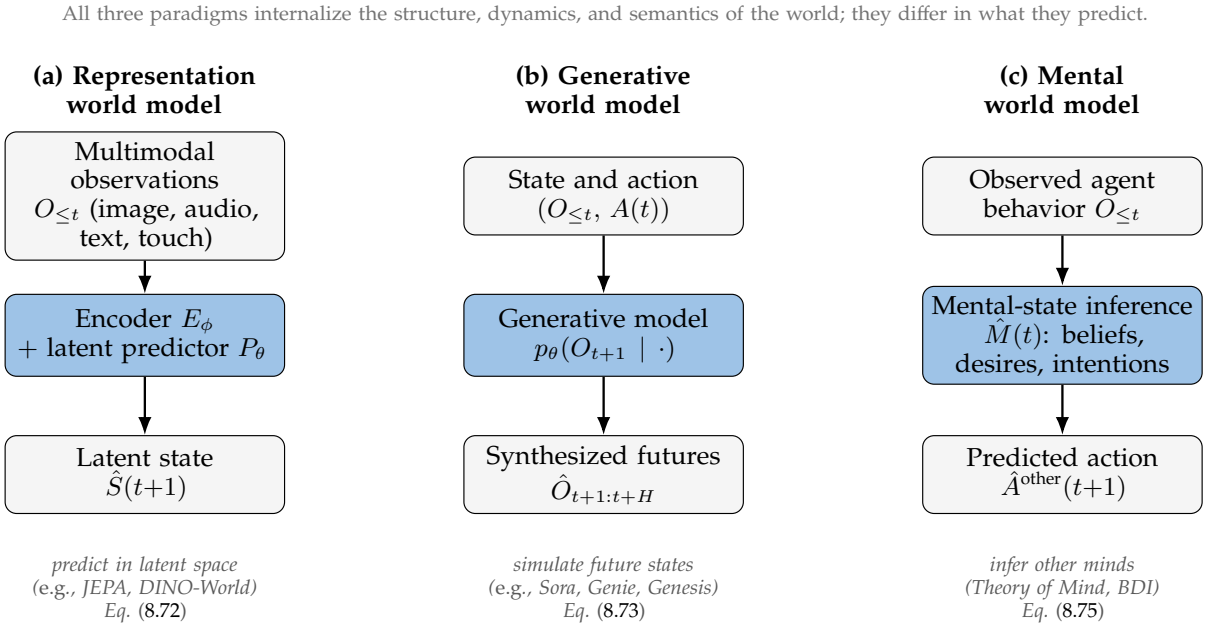


Figure 8.24: Three paradigms of world modeling, all instances of the transition $\hat{S}(t{+}1) = W_\theta(S(t), A(t), O(t))$ of Eq. (8.71), differing in what they predict: (a) a latent state, (b) future observations, and (c) another agent's mental state and next action.

update its internal state. This predictive relation can be written compactly as

$$\hat{S}(t+1) = W_\theta\big(S(t),\, A(t),\, O(t)\big), \tag{8.71}$$

where W_θ denotes the learned world model parameterized by θ. Read against the conditional-generation backbone of this chapter (Eq. (8.2)), W_θ occupies the position of the generator, but its decoder is a state-transition predictor rather than a pixel or geometry renderer; this is the sense in which world modeling extends Chapter 8 from reconstruction toward prediction. What exactly constitutes a "state" remains open: some work defines it as a compact latent representation in which the world model operates at an abstract semantic level, much as human cognition forecasts events without simulating every physical detail.

Although no single definition of world modeling has been universally accepted, a survey of the literature [2314, 2315, 2316, 2317, 186] reveals three recurring paradigms, three realizations of Eq. (8.71) that differ in what they predict. **Representation world models** learn abstract latent spaces that capture the state and dynamics of the environment. **Generative world models** explicitly simulate or generate future states of the world through data-driven or hybrid mechanisms. **Mental world models** aim to infer and represent the mental states of humans or agents, enabling social reasoning and Theory-of-Mind (ToM) capabilities. Figure 8.24 contrasts them side by side.

8.6.1 Representation World Model

8.6.1.1 Definition

A representation world model learns an internal latent description of the environment by compressing multimodal observations into structured, predictive embeddings, prioritizing the extraction of physical and semantic regularities over the generation of photorealistic pixels. Much as LLMs learn abstract linguistic representations through self-supervised pretraining, representation world models are trained on "world data" to infer the principles that govern physical and semantic dynamics; the objective is to predict how the world evolves at an abstract level, not to reconstruct pixels. The latent state is meant to encode objects and

their properties (shape, size, appearance, material), their spatial relations (topology, distance, configuration), the dynamics that govern how the scene evolves, and the causal dependence of outcomes on actions. With such a state, an agent can predict over long horizons, reason, and plan without operating in high-dimensional sensory space.

8.6.1.2 Multimodal Perception

A world model draws on several sensory streams, whose encoding is the subject of Chapter 6. Vision dominates: images and especially video carry objects, geometry, interaction, and temporal dynamics, and modern systems rely on high-capacity encoders trained with contrastive vision–language objectives for semantic and geometric priors. Audio adds direct evidence about events and the physical state of the surroundings, which language conveys only symbolically. Touch is indispensable where vision is unreliable, occluded, or unavailable (grasping inside a bag, fine manipulation, tool use), yet it remains underexplored because tactile sensors are limited and tactile data are hard to collect at scale.

8.6.1.3 World Representation Modeling

Vision-oriented Modeling. A major line of representation world modeling builds latent spaces from visual observations: video frames are encoded into a compact latent space by self-supervised learning, and a transition model predicts future latent states from past latents and actions, which captures the dynamics of the visual world more efficiently than pixel-space prediction. Concretely, an encoder E_ϕ maps observations to latent states and a predictor P_θ is trained to minimize a prediction loss defined entirely in latent space,

$$\mathcal{L}_{\text{rep}}(\theta, \phi) = \mathbb{E}\Big[\big\| P_\theta\big(E_\phi(O_{\le t}),\, A(t)\big) - \text{sg}\big(E_{\bar{\phi}}(O_{t+1})\big) \big\|^2 \Big], \tag{8.72}$$

where $\text{sg}(\cdot)$ denotes the stop-gradient operator and $E_{\bar{\phi}}$ a slowly updated target encoder. This objective instantiates the transition of Eq. (8.71) with the predicted state $\hat{S}(t+1)$ taken to be an abstract embedding rather than raw pixels (Figure 8.24(a)), which is precisely what allows representation world models to capture dynamics efficiently while discarding perceptually salient yet predictively irrelevant detail.

The *World Models* architecture [4] decomposes the problem into a VAE-based vision model, an RNN-based memory model, and a lightweight controller; the vision and memory components learn latent dynamics from raw pixels without supervision, and the controller then learns its behavior inside the "dreamed" environment, which greatly simplifies policy learning. The Dreamer agents carry this idea into reinforcement learning at scale: Dreamer [147] learns a latent dynamics model from pixels and trains an actor and critic entirely on imagined latent trajectories, and DreamerV3 [152] applies a single configuration across a wide range of domains. MuZero [148] learns a model that predicts only the quantities planning needs (reward, value, and policy) and plans with it by tree search, mastering Atari, Go, chess, and shogi without being given the rules.

Yann LeCun's autonomous intelligence framework proposes the Joint Embedding Predictive Architecture (JEPA) [17] as the core mechanism for constructing world models. JEPA learns latent representations and predictions through self-supervised objectives rather than pixel-level reconstruction, and Eq. (8.72) is its generic form. I-JEPA [151] first instantiated it on images, predicting the embeddings of masked target regions from a visible context block, and V-JEPA [2318] carried the same feature-prediction objective to video, predicting masked spatio-temporal embeddings *in latent space* with no pixel reconstruction, text, or negative examples, and so avoiding the cost and ambiguity of pixel prediction. V-JEPA 2 [155] turns this encoder into a world model. Pretrained on more than a million hours of internet video, it is extended with an action-conditioned predictor, V-JEPA 2-AC, post-trained on less than 62 hours of unlabeled robot video; deployed zero-shot on Franka arms, it picks and places objects by planning actions whose predicted embeddings reach a goal image.

The stop-gradient and slowly updated target encoder in Eq. (8.72) exist only to keep the representation from collapsing, and the most recent JEPA work removes them. LeJEPA [2319] shows that an isotropic Gaussian is the optimal embedding distribution for downstream prediction and enforces it with a single sketched regularizer (SIGReg), leaving one trade-off hyperparameter and no teacher–student machinery. LeWorldModel [2320] uses the same regularizer to train an action-conditioned JEPA world model end to end from raw pixels with only two loss terms; at about 15M parameters it trains on a single GPU, plans up to 48 times faster than world models built on foundation-model encoders, and its latents encode physical quantities well enough to flag physically implausible events. The principle has also reached language: VL-JEPA [200] predicts continuous embeddings of the target text instead of generating tokens, and LLM-JEPA [2321] adds an embedding-prediction objective to the next-token training of language models.

DINO-World [2322] instead freezes DINOv2 as a universal visual encoder, whose features are already semantic, geometric, and object-centric, learns a latent-space predictor from large-scale unlabeled video, and adds action-conditioned prediction to model how interventions change future visual states.

Hybrid World Modeling. Another important direction expands representation learning beyond vision by incorporating additional modalities such as LiDAR, camera trajectories, or multi-view geometric signals. These hybrid models aim to overcome the limitations of purely visual latent spaces by injecting explicit spatial structure, depth cues, and geometric consistency constraints. For instance, MUVO [1446] is a multimodal generative world model that learns jointly from synchronized camera and LiDAR streams. By integrating LiDAR-derived depth priors into the latent world representation, MUVO achieves more reliable 3D scene prediction, improved object motion modeling, and long-horizon forecasting in autonomous driving environments. UniWorld [2323] introduces a unified pretrained architecture that merges 3D object detection, motion prediction, and scene representation learning. Through joint training of BEV features, temporal dynamics, and multisensor context, UniWorld produces a structured latent state that simultaneously supports perception and trajectory forecasting. Aether [2324] proposes a geometry-aware unified world model where camera trajectories are treated as explicit action signals. This allows the model to learn action-conditioned scene transitions and perform predictive visual planning. By binding egomotion with geometric consistency constraints, Aether generates spatially coherent future predictions and supports counterfactual simulation within a 3D-aware latent space.

Language-oriented World Modeling. A final category examines language as a central medium for latent world representation. This raises an important question: *can LLMs themselves be viewed as world models?* From a formal perspective, LLMs indeed share structural similarities with classical world models. They maintain internal states, treat textual context as observations, accept prompts as actions that alter these states, and generate predictions that correspond to inferred future states. Models such as Dynalang [2325] demonstrate that natural language can serve as a universal intermediate representation for encoding environment states, predicting transitions, and expressing causal relations. LLMs also exhibit emergent knowledge of commonsense physics, intuitive causality, and human behavior derived from massive text corpora. However, their limitations remain clear: they are trained on symbolic descriptions rather than direct interaction with the physical world; their internal dynamics reflect linguistic statistics more than physical laws; and their world representations tend to be sparse, fragmented, and disconnected from geometric or temporal continuity. Thus, while LLMs can be interpreted as language-driven world models, their predictive capacity is primarily symbolic and semantic, rather than grounded in spatial or physical understanding.

8.6.2 Generative World Model

In contrast to representation-based foundation models, generative world models focus on explicitly predicting future states or synthesizing the evolution of the environment over time. Rather than only learning compact latent encodings, these models aim to simulate how the world might change under different conditions, capturing temporal dynamics and causal structure. Whereas the representation paradigm of Eq. (8.72)

predicts in an abstract latent space, generative world models learn an explicit conditional distribution over future observations and can be unrolled autoregressively over a horizon H,

$$p_\theta\big(O_{t+1:t+H} \mid O_{\le t},\, A_{t:t+H-1}\big) = \prod_{k=0}^{H-1} p_\theta\big(O_{t+k+1} \mid O_{\le t+k},\, A(t+k)\big), \tag{8.73}$$

so that sampling from Eq. (8.73) realizes the transition of Eq. (8.71) directly in observation space (Figure 8.24(b)). This explicit synthesis is what allows generative world models to be rendered, inspected, and acted within, at the cost of having to model a high-dimensional output distribution. Broadly, generative world models can be grouped into two main families: *physics-driven simulation*, which relies on hand-crafted rules and equations, and *learning-driven world generation*, which learns world dynamics directly from data. The former can be viewed as the deterministic limit of the latter: it fixes the transition of Eq. (8.73) to a hand-specified rule set instead of a learned distribution, trading sample diversity and coverage for exact reproducibility and verifiability. A third route, growing quickly since 2024, sits between the two: a learned model writes the rules as code and a simulator executes them (Section 8.6.2.3).

8.6.2.1 Physics-driven Simulation

Physics-driven simulation builds interactive virtual worlds from human-designed physical, geometric, and logical rules rather than from learned latent spaces. The environment is governed by predefined physical laws, rendering rules, collision logic, and behavioral scripts, specified top-down as explicit mathematical formulations, so the same initial conditions always produce the same outcome; this determinism, together with interpretability and verifiability, is what makes such systems suitable for safety-critical engineering.

Game engines are the canonical example. Physics engines such as Bullet, Havok, PhysX, and Box2D implement Newtonian, rigid-body, soft-body, cloth, particle, and volumetric dynamics, and the engines around them add acceleration structures (*e.g.*, BVH trees), real-time ray tracing, and navigation meshes for path planning, which gives them a natural foothold in spatial-intelligence tasks. Film production relies on the same principle at higher fidelity: pipelines built on Pixar USD, Maya, or Houdini simulate cloth, hair, smoke, fire, and fluids with differential-equation and fluid solvers (*e.g.*, FLIP/MantaFlow) and physically based path tracing. Engineering, architecture, and scientific visualization extend rule-based simulation to structural mechanics, electromagnetics, illumination, and material response through finite element methods, computational fluid dynamics, and photonic simulation.

For embodied AI, physics-driven simulators such as CARLA [2326], AirSim, and Isaac Sim [2327] remain the standard platforms for testing and training in autonomous driving, navigation, and manipulation, because they model lighting, collisions, dynamics, and sensor noise closely enough to yield data distributions near those of the real world. The same engines underpin digital twins built on platforms such as NVIDIA Omniverse, Unity, and Unreal Engine, which are discussed in Part IV, Section 11.1.

8.6.2.2 Learning-driven World Generation

In contrast to simulation frameworks based on explicit physical rules, learning-driven world generation seeks to acquire world structure, dynamics, and causal regularities directly from large-scale data, in a bottom-up fashion. These models do not encode Newtonian equations, lighting models, or geometric constraints by hand; instead, world behavior emerges as statistical patterns during training. For example, a model may never be given the analytic form of the gravity equation, yet by observing countless videos of falling objects, it can still infer the regularity that objects tend to accelerate downward. The resulting world models are probabilistic and can exhibit complex, emergent behaviors without explicit programming.

Current learning-driven world generation follows two technical directions: (1) video-centric generative models that learn dynamic world evolution in pixel or latent space, and (2) 3D/4D asset generation methods that directly model spatially structured worlds in three or four dimensions.

Video-centric Generative World Models. Recent advances in video generation have renewed interest in the idea of a "world simulator". Video models are intrinsically suited to encoding spatiotemporal structure, and when they produce sequences that are not only visually coherent but also consistent with basic physical intuition, they begin to resemble implicit world models. This became especially prominent with the release of OpenAI Sora [1991], whose generated videos demonstrate strong temporal consistency, long-horizon stability, and apparent physical plausibility, prompting the community to reconsider whether deep video generation might be converging towards a form of physical world modeling. Subsequently, Google DeepMind's Genie 3 [157] extended video generation from passive "watchable" content to interactive experiences: the model updates the generated world in response to user actions, enabling agent-in-the-loop, closed-loop world simulation, complementing the action-conditioned scene generators surveyed in §8.2.2. World Labs' RTFM [1820] further highlights real-time performance, achieving high frame-rate and low-latency prediction. Systems such as Wan [1990], Emu3.5 [206], OpenAI Sora 2 [2328], and Veo 3 [2329] continue this trajectory, gradually pushing video generation from mere visual fidelity towards world coherence, physical consistency, and interactivity. Action-conditioned diffusion has also turned games into testbeds: GameNGen [1818] simulates the game DOOM interactively in real time, and DIAMOND [1819] trains reinforcement-learning agents entirely inside a diffusion world model of Atari games, while NVIDIA's Cosmos [156] offers pretrained video world foundation models intended to be post-trained as simulators for robotics and driving.

Video world models are particularly promising in autonomous driving. GAIA-1 [2330] proposes a generative world model tailored for driving scenarios, built around a spatiotemporal Transformer that generates future multi-view scenes based on past observations. By implicitly capturing traffic patterns, pedestrian behavior, scene geometry, and multi-agent motion, GAIA-1 can support closed-loop evaluation, behavior prediction, and synthetic data generation, thereby reducing the cost and risk of traditional real-world testing.

3D/4D Asset Generation as World Modeling. A complementary avenue is to build world models that operate directly on structured three- or four-dimensional representations, such as 3D geometry, occupancy fields, or 4D spatiotemporal volumes. Unlike video-based approaches that primarily model dynamics in pixel space, these methods explicitly parameterize geometry and temporal evolution, making it easier to represent interactive, spatially consistent worlds.

Genesis [2331] exemplifies this direction. It is not only a generative 3D/4D physics simulation platform, but also integrates a vision-language model-driven "world-building agent" that uses a tool API to create dynamic 4D environments (geometry plus dynamics). These generated worlds can be rendered into multiple modalities (RGB, depth, semantics, contact events, etc.), providing rich supervision for training embodied AI systems. Genesis thus combines a unified physics engine with generative agents for large-scale world construction.

Autonomous driving again serves as a natural application domain for 3D world models. DriveArena [2332] builds a closed-loop generative simulation platform, allowing planning modules to iteratively probe and learn within synthesized 3D driving environments. Its diffusion-model-based multi-agent scene generation can rapidly produce long, interactive driving episodes. OccWorld [2333] and OccSora [2334] adopt 4D occupancy grids as the core representation, predicting future 3D occupancy over time from continuous video and sensor streams, thereby providing a unified motion-and-structure world model. Copilot4D [2335] tokenizes LiDAR scans and forecasts future LiDAR point clouds with discrete diffusion, learning a world model from unlabeled sensor data alone.

Generating *interactive* 3D worlds brings generative world models closest to their conceptual ideal. Marble [158] from World Labs combines action-conditioned generation, geometric consistency constraints, and multimodal inputs, so that users can act within the generated world and observe the consequences. Its next-generation model, Atlas [2336], is an omni world model pretrained from scratch on text, images, video, and 3D: an autoregressive diffusion transformer that conditions every generated image and depth map on

an explicit camera pose and grounds them all in a shared 3D context, so that the same model generates camera-controlled video of up to a minute, reconstructs scenes from a handful of images, and exports its output as point clouds or 3D Gaussian splats. World Labs reports that it outperforms specialized open-source reconstruction models, and that it will power future versions of Marble.

8.6.2.3 Code-and-Simulator World Models

Between hand-written simulation and learned generation sits a route that has grown quickly since 2024: a language model *writes* the world model as executable code, and a simulator or engine runs it. The learned component is the author of the rules, not the rules themselves,

$$\mathcal{P} \sim p_\theta\big(\mathcal{P} \mid c\big), \qquad \hat{S}(t+1) = \mathrm{Sim}\big(S(t),\, A(t);\, \mathcal{P}\big), \tag{8.74}$$

where c is the conditioning context (a description, observations, or a history of interactions), $\mathcal{P}$ the generated program, and Sim the interpreter or engine that executes it: plain Python, a physics engine such as MuJoCo or PyChrono, or Blender. Compared with Eq. (8.71), the learned W_θ is replaced by a program that can be read, tested, and repaired, so the resulting world keeps the determinism and verifiability of physics-driven simulation while gaining the open-endedness of a generative model. The scene programs of Section 8.7 rest on the same idea; the difference here is that $\mathcal{P}$ encodes how the world changes under actions, not only how it is laid out.

In one line of work, code is an agent's transition model. Code World Models [2337] synthesize Python implementations of reinforcement-learning environments with a generate, improve, and fix search over programs (GIF-MCTS), evaluated on a benchmark of 18 environments, and plan with them faster and more sample-efficiently than with learned dynamics; WorldCoder [2338] builds its world model as a program from its own interactions and transfers knowledge to a new environment by editing that program.

In the other, code builds a simulated 3D world. SceneCraft [2339] turns a description into Blender Python, planning with a scene graph, converting spatial relations into numerical layout constraints, and checking renders with a vision-language model. GenSim [2340] and RoboGen [199] have language models write robotic simulation tasks, scenes, and rewards, making the simulator an open-ended source of training data, and the world-building agent of Genesis [2331] follows the same pattern. Recent systems close the loop with the simulator itself. Code2Worlds [2341] generates 4D worlds as Blender code refined by a vision-language motion critic; Code2Worlds-2 [2342] compiles such worlds into physically valid MuJoCo scenes and revises them from penetration, contact, and stability signals; and ChronoAgentic [2343] has cooperating agents write and repair PyChrono simulations, meeting the joint semantic and physical correctness criterion on 82.5% of PhyWorldBench demos against 52.5% for the best of ten text-to-video models.

The price is coverage. A program can express only what its simulator supports, so photorealistic appearance, deformable and fluid materials, and open-ended visual variety still favor learned generators, while code-and-simulator models win where physical correctness has to be checked.

8.6.3 Mental World Modeling

Physical world models enable agents to understand and predict how the external environment evolves. However, intelligent behavior in real-world settings also requires an internal model of *other minds*. Humans naturally construct such models, *i.e.*, often referred to as *mental models*, to reason about others' beliefs, desires, intentions, emotions, and social context. This ability, commonly associated with *Theory of Mind (ToM)* [2344], underlies human social interaction, collaboration, communication, and decision-making.

Craik's "small-scale models" of reality are the origin of this notion too [2345, 2346], and Johnson-Laird later argued that human reasoning, problem solving, and causal inference rely on abstract internal representations of situations rather than on replaying detailed physical dynamics. These representations selectively capture

the most relevant structure, *i.e.*, objects, relations, causal dependencies, and possible outcomes, while abstracting away low-level perceptual details. For example, when cycling, a human does not mentally simulate every joint torque or physical trajectory; instead, one imagines "whether I would fall or not" when encountering an obstacle. This high-level, conceptual simulation is a defining property of mental world modeling.

8.6.3.1 From Physical to Cognitive Mental World Models

Human mental simulation operates by combining current observations with stored knowledge to predict plausible futures. Perception provides the immediate sensory input, which is then transformed into an internal symbolic and probabilistic representation of the world. Within this representation, the brain runs rapid, approximate simulations that evaluate different hypothetical actions. These simulations are not physically precise but are cognitively meaningful: they help determine risks, anticipate others' responses, or evaluate alternative courses of action. In this sense, mental models serve as a compact, abstract, and counterfactual "world engine" for decision making.

In artificial systems, the distinction between physical and mental world models mirrors that in human cognition. Physical world models predict state transitions in the environment (*e.g.*, dynamics, geometry, object motion), whereas *mental world models* encode the latent mental states of human users or other agents. These models allow an AI system to reason about what humans know, believe, want, or intend to do, capabilities that embodied agents, collaborative robots, intelligent tutors, and multi-agent planners all need.

Inspired by the Belief-Desire-Intention (BDI) [2347, 2348] framework, a mental world model for AI typically includes a small set of structured variables. **Beliefs.** These are representations of what the human knows, perceives, or believes about the world. **Desires.** These capture user goals, preferences, motivations, and values. **Intentions.** These are planned or ongoing actions that the agent expects the human (or another agent) to follow. Such decomposition enables an AI system to explicitly reason about human rationality, goal pursuit, and action planning. Formally, a mental world model can be cast as the inference of a latent mental state $M(t)$, comprising the beliefs, desires, and intentions above, from observed behavior, followed by a prediction of the other agent's next action,

$$\hat{M}(t) = \arg\max_{M} p\big(M \mid O_{\leq t}\big), \qquad \hat{A}^{\text{other}}(t+1) = \pi\big(\hat{M}(t)\big). \tag{8.75}$$

This formulation mirrors the transition of Eq. (8.71) but replaces the physical state with the inferred mental state of another agent (Figure 8.24(c)), so that the predicted quantity is no longer how the environment will evolve but how a human or peer agent is likely to act.

Machine Theory of Mind is now measured directly. ToMi [2349] tests false-belief reasoning in short stories with controls against shortcut solutions, BigToM [2350] generates controlled social scenarios with language models to probe whether models infer beliefs from percepts and actions, and MMToM-QA [2351] poses belief and goal questions about a person acting in a household from video and text together. Large language and multimodal models remain unreliable on these tests, whereas MMToM-QA's BIP-ALM, which performs Bayesian inverse planning over a symbolic state with a language model scoring the likelihoods, does better; it is an explicit instance of the inference in Eq. (8.75).

Dimensions of Mental State Representation. Beyond the belief-desire-intention core, a mental world model for embodied agents must span several further layers of internal state and social context. It has to read *affect*, recognizing emotional cues and how they shift a person's risk tolerance and willingness to cooperate. It has to represent *social dynamics*: the relationships, roles, group structures, and cultural norms that shape behavior. And it has to interpret *communication*, both the verbal channel and the non-verbal one carried by tone, gesture, posture, and facial expression. These layers are what turn a bare goal estimate into a socially situated account of why a person acts as they do.

Human spatial reasoning is social as well as geometric, which ties mental modeling to spatial intelligence. To assist or collaborate with humans in homes, factories, classrooms, or vehicles, an agent must infer the intention behind a spatial action (*e.g.*, where a person intends to move next), anticipate trajectories from goals or habits, adapt navigation or manipulation to user preferences, and interpret spatial configurations that reflect what a person believes or misunderstands.

8.6.3.2 A Joint Physical-Mental Formulation

The benchmarks above still treat the mental state as something inferred next to a world model, not as part of it: Eq. (8.75) estimates another agent's mind, but the environment it acts in is predicted separately. Mental World Modeling (MWM) [2352] makes the mental state part of the state that the world model of Eq. (8.71) rolls forward. We follow its notation, in which subscripts index time, so that s_t and a_t^ϵ play the roles of $S(t)$ and $A(t)$ above. The case for a joint state is that neither half suffices alone. For a target agent ϵ whose true action distribution is $p^\star$, a physical-only representation fails whenever

$$p^\star\big(a_t^\epsilon \mid s_t^{\text{phy}}, s_t^{\text{ment}}\big) \neq p^\star\big(a_t^\epsilon \mid s_t^{\text{phy}}, \tilde{s}_t^{\text{ment}}\big), \tag{8.76}$$

that is, whenever two scenes that are physically identical but differ in what people believe or want (s_t^{ment} versus $\tilde{s}_t^{\text{ment}}$) lead to different actions; the symmetric condition, with the mental state held fixed and the physical state changed, rules out a mental-only representation.

Joint State. The world state is the pair

$$s_t = \big(s_t^{\text{phy}}, s_t^{\text{ment}}\big) \in \mathcal{S}^{\text{phy}} \times \mathcal{S}^{\text{ment}}, \qquad s_t^{\text{ment}} = \big(\{m_t^i\}_{i\in\mathcal{I}},\, \{m_t^G\}_{G\in\mathcal{G}},\, R_t^{\text{ment}},\, \alpha_t\big). \tag{8.77}$$

The physical part collects the objects, the embodied characters, their spatial and contact relations, and the environmental conditions of the scene. The mental part holds an individual state m_t^i for every agent i in the set $\mathcal{I}$, optional states m_t^G for groups $G \in \mathcal{G}$, the social attitudes and role relations R_t^{ment} between agents, and the atmosphere α_t of the setting. Each individual state extends the belief-desire-intention core introduced above,

$$m_t^i = \big(\text{id}_t^i,\, b_t^i,\, q_t^i,\, g_t^i,\, \iota_t^i,\, e_t^i,\, d_t^i,\, n_t^i,\, c_t^i\big), \tag{8.78}$$

with identity attributes, beliefs, attention focus, goals, intentions, emotions, dispositions, norms, and behavioral constraints. Beliefs, goals, and intentions are the BDI variables; emotions, norms, and the relations R_t^{ment} turn the affect and social-dynamics layers discussed above into explicit state variables.

Target-Specific Observation. The model reasons from the perspective of one target agent $\epsilon \in \mathcal{I}$, which sees only a rendering of the joint state:

$$o_t^\epsilon = \Omega_\epsilon(s_t) = \Big(\Omega_\epsilon^{\text{phy}}\big(s_t^{\text{phy}};\, \kappa_t^\epsilon\big),\; \Omega_\epsilon^{\text{ment}}\big(s_t^{\text{phy}}, s_t^{\text{ment}};\, \rho_t^\epsilon\big)\Big), \tag{8.79}$$

where κ_t^ϵ is the target's perceptual access (position, line of sight, hearing range, available modalities) and ρ_t^ϵ its social and cognitive perspective (role, prior beliefs, relationships to others, awareness of norms, Theory-of-Mind capacity). The mental part of the observation contains the target's view of its own state and its estimates $\widehat{m}_{t,\epsilon}^{j,(\ell)}$ of every other agent j, nested up to a depth L; at the first level they are sampled as

$$\widehat{m}_{t,\epsilon}^{j,(1)} \sim Q_\epsilon^{(1)}\big(m_t^j \mid o_t^{\epsilon,\text{phy}},\, m_t^\epsilon,\, R_t^{\text{ment}},\, \alpha_t\big), \tag{8.80}$$

which is the inference of Eq. (8.75), made stochastic and conditioned on the target's own mind and social context. Spatial access enters the mental model directly through κ_t^ϵ: what the target can see from where it stands bounds what it can believe, so a poor decision can follow from an object outside its view as much as from a mistaken belief.

Action and Coupled Transition. The target proposes K candidate actions $\mathcal{A}_t^\epsilon = \Pi_\epsilon^K(o_t^\epsilon) = \{a_{t,1}^\epsilon, \ldots, a_{t,K}^\epsilon\}$. Each action has a physical carrier and a mental content, $a_t^\epsilon = (a_t^{\epsilon,\text{phy}}, a_t^{\epsilon,\text{ment}})$ (approaching someone, and meaning to comfort them), and the policy factorizes as $\Pi_\epsilon(a_t^{\epsilon,\text{phy}}, a_t^{\epsilon,\text{ment}} \mid o_t^\epsilon) = \Pi_\epsilon^{\text{phy}}(a_t^{\epsilon,\text{phy}} \mid o_t^\epsilon)\, \Pi_\epsilon^{\text{ment}}(a_t^{\epsilon,\text{ment}} \mid a_t^{\epsilon,\text{phy}}, o_t^\epsilon)$. The world model then predicts the next joint state with a staged transition,

$$T_\theta\big(s_{t+1} \mid s_t, a_t^\epsilon\big) = T_\theta^{\text{phy}}\big(s_{t+1}^{\text{phy}} \mid s_t^{\text{phy}}, s_t^{\text{ment}}, a_t^{\epsilon,\text{phy}}\big) \cdot T_\theta^{\text{ment}}\big(s_{t+1}^{\text{ment}} \mid s_t^{\text{phy}}, s_t^{\text{ment}}, a_t^{\epsilon,\text{phy}}, a_t^{\epsilon,\text{ment}}\big). \tag{8.81}$$

The asymmetry is deliberate. An intention moves nothing until it is carried out physically, so the physical factor ignores $a_t^{\epsilon,\text{ment}}$; the mental factor depends on both components, because the same reassuring words land differently when the speaker is approaching than when walking away. The physical factor still conditions on the mental state, since what objects, space, and visibility allow is joined by what other agents intend to do. Those other agents are not controlled: their actions $a_t^{-\epsilon}$ are marginalized under the simulator's predictive distribution,

$$T_\theta\big(s_{t+1} \mid s_t, a_t^\epsilon\big) = \int T_\theta\big(s_{t+1} \mid s_t, a_t^\epsilon, a_t^{-\epsilon}\big)\, p\big(a_t^{-\epsilon} \mid s_t\big)\, \mathrm{d}a_t^{-\epsilon}, \tag{8.82}$$

with $p(a_t^{-\epsilon} \mid s_t) = \prod_{j \neq \epsilon} \Pi_j\big(a_t^j \mid \Omega_j(s_t)\big)$ for simultaneous moves, so that every agent still acts on its own partial observation. The whole pipeline, $s_t \xrightarrow{\Omega_\epsilon} o_t^\epsilon \xrightarrow{\Pi_\epsilon} a_t^\epsilon \xrightarrow{T_\theta} s_{t+1}$, is Eq. (8.71) with a mental component added to the state and a perspective added to the observation.

Evaluating Candidate Futures. Simulating each candidate gives a branch $\hat{s}_{t+1}^k$, and the predicted decision is the branch of highest value,

$$\hat{k} = \underset{k \in \{1,\ldots,K\}}{\arg\max}\ r_k. \tag{8.83}$$

In MENTIS, the training-free and fully inspectable baseline of MWM, the branch value r_k is a weighted combination of normalized scores for physical plausibility, mental consistency, and social appropriateness, and a binary safety and legality veto sets it to zero. Each formal component maps to a module: a state parser recovers $\hat{s}_t$, an observation generator implements Ω_ϵ, an action parser structures the candidates, a world-state transitor implements T_θ, and an evaluator assigns r_k. On situated decision scenarios presented as text, image, and sounding-video stories, experiments across eight LLM-based world models indicate that the explicit mental component is what makes a person's actual decision predictable [2352].

8.6.4 Discussions and Challenges

Representation-based World Models as an Alternative Foundation Model Paradigm. Representation world models may emerge as a parallel paradigm to current LLMs, functioning as large-scale foundational models trained through massive self-supervised representation learning to capture physical regularities, causal dynamics, and environmental constraints. Unlike LLMs, which learn the world indirectly through linguistic abstraction, representation world models aim to model the physical world directly from multimodal observations. Such models could serve video generation, action prediction in VLA systems, 3D and 4D understanding and generation, and decision-making in autonomous driving.

The Necessity of Multi-source Intelligence Integration. World models will likely require the integration of heterogeneous informational sources. On the one hand, LLMs embody language-driven intelligence, while world models emphasize physically grounded intelligence; rather than being mutually exclusive, a truly capable world model must incorporate both forms of reasoning. On the other hand, accurate world understanding requires far more than visual perception alone: audio, speech, tactile signals, force feedback, and even proprioception provide essential supplementary cues about the dynamics and structure of the physical environment. Yet the current research community focuses overwhelmingly on visual-only modeling. Building world models that holistically integrate multimodal sensory information remains a major challenge.

Unifying Physical Modeling and Mental Modeling. Another promising direction is the integration of physical and mental modeling within a unified world-modeling framework. Embodied agents operating in real environments must simultaneously reason about the physical evolution of the environment and the mental or behavioral responses of humans (or other agents). Recent work in multi-agent systems, social navigation, pedagogical reasoning, human-robot collaboration, and interactive LLM agents emphasizes the growing need for AI systems to build internal hypotheses about both external dynamics and human mental states. Mental World Modeling [2352] is an early formulation of such a joint state (Section 8.6.3.2). Ultimately, scalable general-purpose world models must simulate not only *what will happen* but also *what others believe, intend, desire, or misunderstand*.

Balance between Generative Fidelity and Physical Correctness. A central challenge for learning-driven world models is reconciling generative expressiveness with physical accuracy. High-capacity video generators often produce visually stunning but physically implausible outcomes, while models optimized for strict physical consistency may underperform in realism or diversity. Achieving stable long-horizon predictions without temporal drift or compounding errors remains unsolved. Designing architectures, training objectives, and constraints that jointly enforce physical laws, spatial coherence, and generative flexibility is a key open problem for the next generation of world simulators.

Evaluation, Benchmarks, and Verifiability of World Models. Unlike language models, which can be evaluated through perplexity, benchmarks, or task accuracy, world models lack standardized, reliable, and comprehensive evaluation protocols. Physical correctness, causal consistency, interactive fidelity, multimodal coherence, and long-horizon stability are difficult to quantify in a unified manner. Furthermore, world models that aim to simulate open-ended environments raise issues of verifiability, reproducibility, and safety: how can we assess whether a world model reliably generalizes to unseen real-world scenarios, or whether its internal dynamics exhibit hidden biases or unsafe behaviors? Establishing principled evaluation methodologies and shared benchmarks for both representation-based and generative world models is an urgent research direction.

8.7 Programmatic and Symbolic Spatial Generation



rogrammatic and symbolic spatial generation synthesizes 2D and 3D scene structure through explicit rules, executable programs, and symbolic geometric abstractions. It generates the structural backbone of a scene (spatial layout, hierarchical organization, geometric constraints, executable logic) rather than its raw appearance, which gives it controllability, compositional generalization, global spatial consistency, and interpretability.

Where the largely neural samplers of §8.1–§8.6 map a condition directly to pixels, voxels, or a neural latent, programmatic methods emit an intermediate *program* that a deterministic renderer executes into the final scene. This intermediate layer is what makes the structure auditable, editable, and compositionally reusable, properties that flat neural outputs lack. Three paradigm families mark the historical and technical progression of the field: *rule- and grammar-based procedural generation*, *symbolic graphics programs and domain-specific languages*, and *neuro-symbolic language-to-program synthesis* driven by large language models.

8.7.1 Task Definition

Let $\mathbf{c} \in \mathcal{C}$ denote a conditioning intent, typically a natural-language description, a layout specification, or a partial constraint set, and let $\kappa : \mathcal{C} \to \mathcal{C}'$ be its encoder. A programmatic generator G_θ emits a symbolic program ρ drawn from a graphics language or domain-specific language $\mathcal{L}$, which a deterministic executor R evaluates into a concrete scene $\hat{\mathbf{s}}$ (a raster image, vector graphic, UI specification, or 3D shape):

$$\rho = G_\theta\big(\kappa(\mathbf{c})\big), \qquad \hat{\mathbf{s}} = R(\rho), \qquad \rho \in \mathcal{L}. \tag{8.84}$$

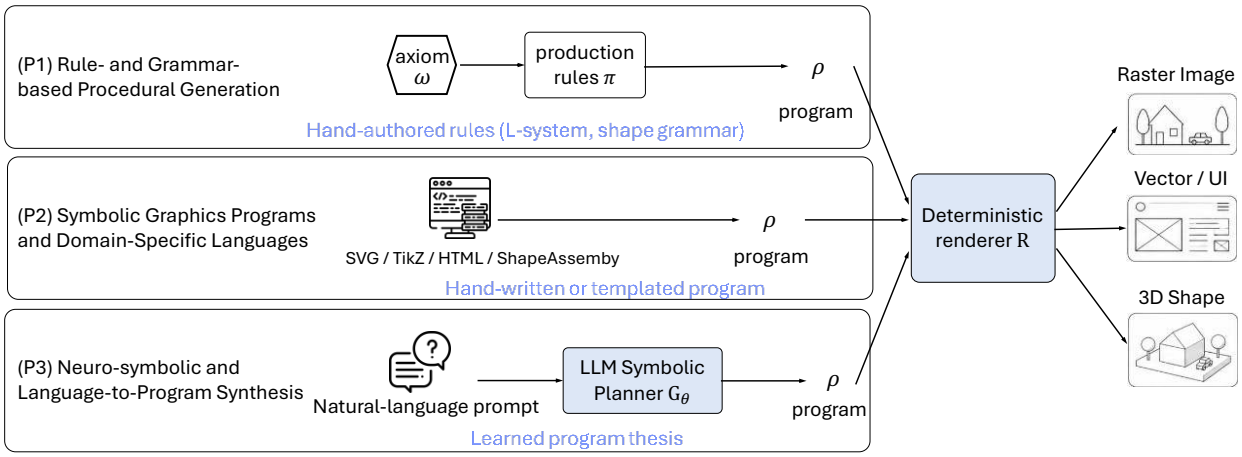


Figure 8.25: Three paradigm families for programmatic spatial generation, all producing a program ρ that a deterministic renderer R executes (Eq. (8.84)): (P1) rule- and grammar-based procedural generation, (P2) symbolic graphics programs and domain-specific languages, and (P3) language-to-program synthesis by an LLM planner.

This formulation inherits the conditional-sampler abstraction of Eq. (8.2) but replaces its learned decoder D (Eq. (8.1)) over a continuous latent space $\mathcal{Z}$ with an interpretable, discrete program ρ and an *exact, non-learned* renderer R. The neural generators of §8 are recovered as the limiting case in which R is itself a neural renderer and ρ a continuous latent, whereas the symbolic case constrains ρ to be a human-readable executable artifact. Two properties follow directly: errors are localizable to specific program statements, and structural constraints can be enforced or verified deterministically on ρ before rendering.

8.7.2 Methodology

Figure 8.25 schematizes the three paradigm families that instantiate Eq. (8.84), contrasting how each obtains the program ρ before the shared deterministic renderer R executes it.

Rule- and Grammar-based Procedural Generation. The earliest line of work, procedural content generation (PCG) [2353] in computer graphics and game development, synthesizes scenes such as dungeon maps, terrains, architectural plans, or UI layouts from hand-crafted rules or parameterized procedural programs. Classical formalisms instantiate G_θ as a fixed production system rather than a learned model. The Lindenmayer system (L-system) [2354], originally developed to model biological growth, employs parallel rewriting rules to simulate natural growth processes and fractal patterns, making it ideal for self-similar structures such as vegetation and complex terrain. Formally, the program evolves by repeated parallel application of a production set Π to an axiom ω,

$$\rho_0 = \omega, \qquad \rho_{t+1} = \Pi(\rho_t), \qquad \hat{\mathbf{s}} = R(\rho_T), \tag{8.85}$$

where the generated symbolic string is not the geometry itself but a sequence of Turtle Graphics commands that the interpreter R parses to draw the final shape in 2D or 3D space. Shape grammars, introduced by Stiny [2355], provide another influential symbolic mechanism by defining geometric primitives and rewrite rules for composing architectural or engineered shapes. These early paradigms demonstrated a fundamental insight: spatial structures can be synthesized by executing symbolic generative rules, rather than by directly predicting raw geometry.

Symbolic Graphics Programs and Domain-Specific Languages. Symbolic graphics programs (SGPs) encode spatial layouts and geometric configurations as formal programs that execute deterministically into images, vector graphics, UI designs, or 3D shapes, bridging abstract scene intent and concrete visual realization.

Common languages such as SVG [2356], TikZ/LaTeX [2357], and HTML/CSS [2358] already carry rich symbolic layout specifications in real-world use. In parallel, researchers have developed domain-specific languages (DSLs) tailored for fine-grained spatial and visual synthesis. Examples include hierarchical GUI-code generators [2359] that translate visual structures into programmatic UI specifications, and ShapeAssembly [2360], a 3D assembly language in which objects are represented as executable programs describing part composition, symmetry, and connectivity. Such DSLs make the advantages of explicit symbolic control concrete: precise geometry, modular design, verifiable constraints, and systematic compositionality.

Neuro-symbolic and Language-to-Program Synthesis. A more recent direction couples symbolic programs with learned models, so that G_θ becomes a trained synthesizer rather than a fixed rule set. The Scene Language [1614] unifies neural and symbolic components: a program specifying relational and hierarchical layout structure, natural-language labels summarizing entity semantics, and learned embeddings capturing visual identity. Executable structure, rather than raw geometry, thus becomes a primary representational layer for scenes. Large language models [43, 11] have accelerated the move toward language-to-program spatial generation: instead of predicting images or layouts directly, LLM-based systems act as symbolic planners that synthesize the program by approximately maximizing its conditional likelihood,

$$\rho^\star = \arg\max_{\rho \in \mathcal{L}} p_\theta\big(\rho \mid \kappa(\mathbf{c})\big), \tag{8.86}$$

mapping natural-language descriptions to executable layout programs [2361, 2362, 2363]. These programs then guide downstream renderers to produce UI designs, graphic layouts, architectural plans, or 3D scenes with precise structural fidelity, and users can edit or refine the underlying program with fine-grained control.

The three families differ along two cross-cutting axes. The first is the *representation* of ρ: a parallel string-rewriting system (grammar-based generation), an executable program in a general-purpose or domain-specific graphics language (SGPs and DSLs), or a hybrid structure that interleaves code, language, and learned embeddings (neuro-symbolic synthesis). The second is the *inference procedure* that produces ρ: hand-authored production rules, human-written or template-instantiated programs, or learned program synthesis and LLM-based planning. Across all three, the target modality (2D raster, vector graphics, UI, or 3D) is determined solely by the renderer R, which keeps the symbolic and rendering concerns cleanly separated.

8.7.3 Benchmarks

Evaluation in this area differs from the distributional metrics of §8 because the immediate output is a program, not an image, so quality is assessed along several complementary axes. **Program validity** measures whether the emitted ρ parses and executes without error in its target language, a prerequisite that purely neural generators do not face. **Rendered-output fidelity** then compares the executed scene $R(\rho)$ against a reference, using pixel- or perceptual-level similarity for raster and vector targets and geometric agreement for 3D shapes. **Constraint satisfaction** checks whether explicitly specified spatial relations, counts, alignments, or symmetries hold in the rendered result, a property that symbolic methods are designed to guarantee. **Controllability and editability** assess how reliably localized edits to ρ produce the intended localized changes in the scene without unintended global drift. Concretely, these axes are exercised across vector-graphics and SVG synthesis [2356], UI and GUI code generation [2359, 2358], 3D program induction over part-structured shapes [2360], evaluated against part-hierarchy benchmarks such as PartNet [2364], and full scene-program synthesis [1614, 2363], where the executable program serves simultaneously as the prediction and as the verifiable ground for measurement.

Table 8.14: The seven spatial-generation subfields of this chapter, all instances of the conditional latent-generator backbone of §8.1.1.

Subfield	Output space	Dominant paradigms	Decoder / renderer
§8.1 2D image	Pixel grid (raster)	GAN; autoregressive; latent diffusion; flow matching	Learned pixel decoder
§8.2 3D object/scene	Mesh / NeRF / 3DGS / latent	Native 3D diffusion; score distillation; multi-view lift; feed-forward reconstruction	Differentiable / volumetric renderer
§8.3 Video	Frame sequence (clip latent)	Inflated T2I backbones; spatial / temporal control; editing	Spatio-temporal decoder
§8.4 Motion	Pose / trajectory sequence	Regression; diffusion samplers; editing operators	Kinematic / skeletal decoder
§8.5 4D	Time-indexed 3D representation	Score distillation; generate-then-reconstruct	Deformable / per-frame renderer
§8.6 World model	Latent state / future observations	Representation; generative; mental world models	Latent transition / observation predictor
§8.7 Symbolic program	Executable program in a language $\mathcal{L}$	Procedural grammars; SGP/DSL; LLM program synthesis	Deterministic, non-learned executor

8.7.4 Challenges and Trends

The field has moved from entirely hand-authored rules toward neuro-symbolic integration with large models, driven by the demand for generators that are more controllable and interpretable than end-to-end neural methods.

Generalization. Automatically synthesized programs still struggle with novel constraints and complex semantic specifications [2365]. Program synthesis has to infer the structural logic of a scene rather than memorize surface patterns if the generator is to extrapolate to compositions and constraint combinations absent from training.

Efficiency and differentiability. Neuro-symbolic programs that are both efficient and fully differentiable [2366] would let a rendering-level loss propagate back through the program, so that the whole pipeline, from symbolic planning to neural rendering, could be trained and fine-tuned end to end.

Embodied interaction. Procedurally synthesized interactive environments already supply large-scale embodied worlds [1621, 2367, 2368], and the next step is for agents to generate or validate symbolic scene structure through active exploration rather than from static datasets alone, yielding symbolic representations of physical spaces that can be checked against what the agent observes.

8.8 Summary and Outlook

This chapter has surveyed spatial reconstruction and generation across seven subfields that, taken together, span an axis of increasing output structure and dimensionality: 2D images (§8.1), 3D objects and scenes (§8.2), video (§8.3), embodied motion (§8.4), 4D space-time content (§8.5), world models (§8.6), and interpretable symbolic programs (§8.7). Despite their different output domains, the subfields are unified by a single conditional-generation backbone, established in §8.1.1: an encoder κ maps a condition $\mathbf{c}$ into a latent code, a generator G_θ samples structure from that code, and a decoder or renderer turns the sampled structure into a concrete artifact (Eq. (8.1), Eq. (8.2)). What distinguishes the subfields is not this backbone but the *output space* and the *decoder* that closes it: a pixel decoder for images, a volumetric or splatting renderer for 3D, a spatio-temporal decoder for video, a kinematic sampler for motion, a time-indexed 3D representation for 4D, a state-transition predictor for world models, and an exact, non-learned program executor for symbolic generation. Table 8.14 consolidates this view.

A shared backbone, specialized by output space. The progression from §8.1 to §8.7 is best read not as seven disjoint problems but as one generative abstraction instantiated against progressively richer codomains,

and each section casts its own subtasks as conditional specializations of that common sampler. The symbolic case (§8.7) is the limiting instance in which the decoder degenerates to an exact program executor, trading the expressiveness of a learned renderer for interpretability and verifiable control.

The 2D prior as a reusable engine. The image generators of §8.1 recur throughout the chapter not as terminal systems but as priors to be distilled or lifted: 3D synthesis distills frozen 2D diffusion through score distillation and multi-view consistency (§8.2), video models inflate text-to-image backbones along the temporal axis (§8.3), and 4D pipelines compose both (§8.5). A single advance in the 2D substrate therefore propagates outward across every higher-dimensional subfield, which concentrates research leverage but also propagates the substrate's failure modes.

Physical plausibility and fragmented evaluation. Across subfields, the recurring failure mode is not low visual fidelity but violated physics and geometry, manifest as multi-view inconsistency in 3D, temporal drift and flicker in video, foot-skating and interpenetration in motion, and dynamic implausibility in 4D. Evaluation remains correspondingly fragmented: each subfield has evolved its own metric tuples, and all are migrating toward learned and vision-language-model judges as scalar similarity scores prove insufficient for spatially structured output. A unified, physically grounded evaluation protocol that spans these subfields remains an open goal.

Outlook. These observations point toward convergence rather than further fragmentation. World modeling (§8.6) already acts as a conceptual capstone that absorbs reconstruction, generation, and dynamics into a single predictive substrate, while programmatic and symbolic generation (§8.7) supplies the controllability and interpretability that purely neural pipelines lack; their integration, neural generators steered and verified by symbolic structure, is a natural trajectory for the field. The capabilities surveyed here also form the generative substrate for the applications of Part IV, where embodied agents, digital twins, and interactive environments consume generated spatial content rather than producing it for its own sake.

Part IV

Applications

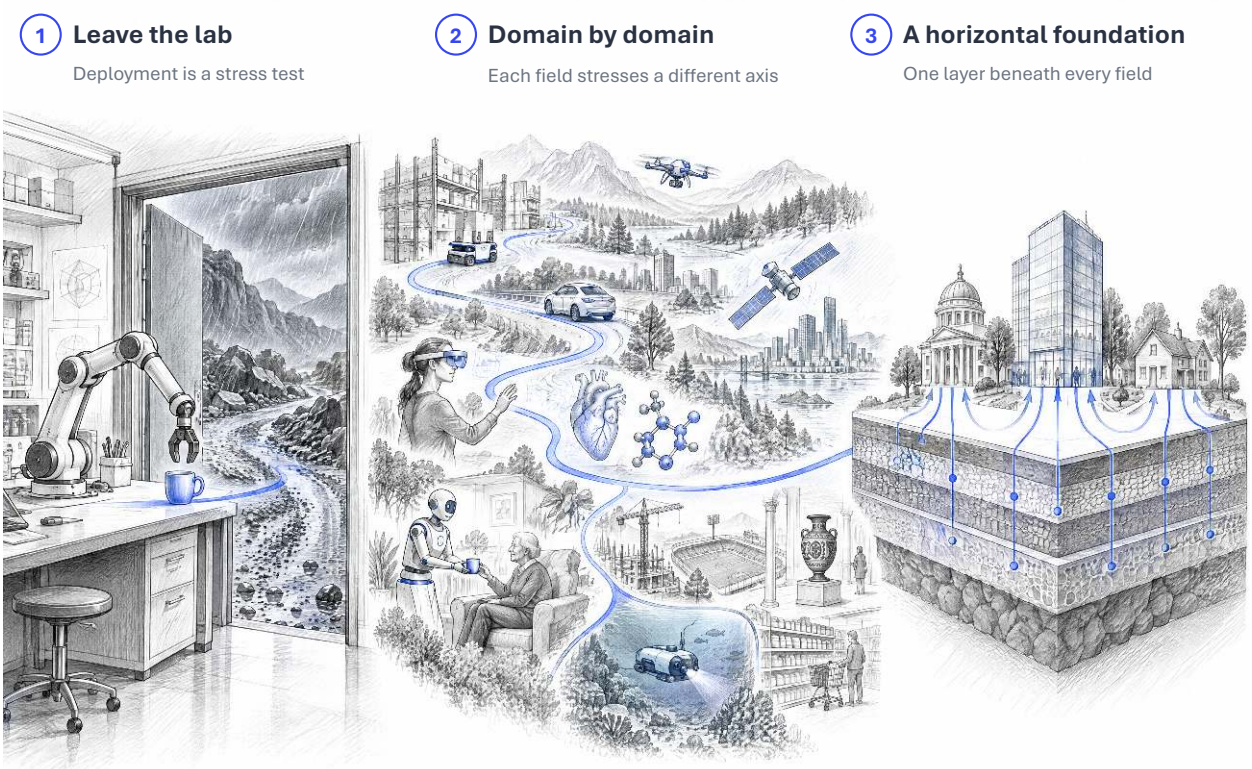


Figure IV: Part IV at a glance. (1) Leave the lab: deployment tests spatial intelligence under latency, safety, sensor failure, and the cost of error. (2) Domain by domain: embodied agents, extended reality, geo-spatial and scientific settings, assistants, and further fields, each stressing a different axis. (3) A horizontal foundation: one shared layer beneath science, industry, and daily life, with gaps found in the field feeding back into it.

The preceding parts built spatial intelligence as a set of capabilities; this part asks what those capabilities are worth once they leave the laboratory. Applications are not a victory lap over Parts II and III but a stress test: a method that reconstructs a scene cleanly on a benchmark must, in the field, do so under a latency budget, a safety constraint, a sensor that fails, or a cost of error measured in more than accuracy. What a domain demands of spatial intelligence, and what it reveals about the gaps that remain, is visible only at deployment.

This part is therefore organized by domain rather than by capability, because the interesting content lives in the delta each domain adds over a general treatment. The tour runs from embodied agents that must act in the physical world, through the extended-reality systems that render it back to a user in real time, to the geo-spatial and scientific settings that model it at the scale of planets and of cells, to the assistants that inhabit a person's own environment, and out into a long tail of further fields. Each stresses spatial intelligence along a different axis: real-time action, on-device perception, metric fidelity, human-centered interaction. Read together, they make the case that spatial intelligence is not one more application but a horizontal foundation on which applications across science, industry, and daily life are increasingly built.

Chapter 9

Spatial Intelligence for Embodied Systems

Embodied systems are the arena in which spatial intelligence stops being a matter of analysis and becomes a matter of consequence. An embodied agent does not merely describe a scene; it occupies one, and every perception it forms and every inference it draws is ultimately answerable to an action it must take in the physical world, under real time budgets, sensor noise, and the cost of being wrong. This chapter turns from the capabilities of the previous part to their deployment, examining how perception, reasoning, and generation are brought to bear inside agents that move, manipulate, and assist.

What makes this more than a restatement is that embodiment changes the problem. The same spatial competence looks different in a manipulator closing a grasp, a car threading traffic, a wearable inferring its user's intent, a drone holding position in wind, or a fleet of robots coordinating on a warehouse floor: each fixes its own sensors, latencies, safety margins, and notion of success. The chapter is organized by these domains rather than by capability, because the interesting content lives in the domain-specific delta, what each setting demands of spatial intelligence that a general treatment does not reveal.

9.1 Robotic Manipulation

9.1.1 Introduction

Robotic manipulation has advanced rapidly, driven by strides in perception, learning, and control. Yet a significant gap remains between competent laboratory systems and robust, general-purpose robots capable of operating in unstructured settings. A central bottleneck is that manipulation is fundamentally spatial: every decision (where to reach, how to establish contact, how to route a trajectory through clutter, or how to use a tool) hinges on perceiving, representing, and reasoning about three-dimensional structure under uncertainty. Manipulation is thus a direct test of the *spatial intelligence* developed throughout Parts I–III: it demands that systems go beyond semantic recognition to build actionable 3D world models, ground tasks in geometric constraints, predict contact outcomes, and plan behaviors that respect the physical laws of the environment. The perception, reasoning, and generation capabilities that such systems presuppose are surveyed in Chapter 6, Chapter 7, and Chapter 8; this section focuses not on re-deriving those capabilities but on how they are composed, grounded, and constrained for physical manipulation.

9.1.2 System Overview

A manipulation system maps raw multimodal observations and a task instruction into executable, constraint-satisfying actions, and this section organizes that mapping into four strategic **pivots**, each a fundamental shift in how systems bridge perception and physical action. **(i) Grounding Action in Spatial Anchors** moves beyond raw pixels to vision priors and sparse spatial tokens (keypoints, dense correspondences) that stay consistent across viewpoints and instances; **(ii) Inferring Functional Action Possibilities** shifts from object classification to extracting affordances and tool-use capabilities, bridging perception with contact-rich execution; **(iii) Encoding Spatial Reasoning in Generalist Models** transitions from single-task policies to foundation models (VLMs/VLAs) that integrate explicit 3D representations and reasoning traces for cross-embodiment generalization; and **(iv) Satisfying Geometric and Logical Constraints** advances from simple trajectory generation to constrained optimization and neuro-symbolic planning (TAMP), ensuring actions respect strict safety limits and long-horizon logical structure. The four pivots are complementary and mutually reinforcing rather than strictly sequential: anchors and affordances structure the perceptual input, generalist models supply broad priors, and constraint-based planners check feasibility under the modeled geometric, physical, and logical assumptions. Figure 9.1 situates them as complementary capabilities within a closed-loop perception-to-action system, and the remainder of this section treats them in this order.

9.1.3 Pivotal Task-1: Grounding Action in Spatial Anchors

This pivot reframes visuomotor learning from operating over raw pixels to reasoning through explicit **spatial anchors** (sparse semantic keypoints, dense correspondences, point tracks, or affordance-based contact points) that compress high-dimensional observations into spatial tokens tied to object- or task-centric reference frames and designed to remain comparatively stable across viewpoint, instance, and clutter. As Figure 9.2 indicates, the approach follows a three-step logic: large-scale vision priors first *produce* the anchors, policies are then *conditioned* on them, and the payoff is read off as a *generalization gap* between seen and unseen instances.

The dominant methodology leverages large-scale vision priors to generate, propagate, or retrieve these anchors, then trains standard visuomotor learners on the resulting structured inputs. A common strategy prescribes anchor points in a few reference frames and propagates them across large datasets via off-the-shelf correspondence models: P3-PO [2369] propagates manually specified keypoints using dense features from DINO [2370], DINOv2 [958], or diffusion-based representations [2371, 2372, 2373] (these visual representations are surveyed in Chapter 4). Relatedly, Robo-ABC [2374] retrieves affordance contact points from an external memory and transfers them to novel scenes via dense semantic correspondence, while Track2Act [2375] learns long-term point tracks from Internet-scale video and uses them to infer object-centric transformations that serve as open-loop manipulation plans. Once anchors are obtained, policy learning conditions action prediction directly on them, through keypoint-conditioned behavior cloning (KALM [2376]), diffusion policies in object-relative frames (AffordDP [2377]), and shared human–robot 3D keypoint spaces (Point Policy [2378]). Across these variants, anchoring actions to spatial structure reduces perceptual ambiguity and improves robustness under distribution shift.

Because the central claim is out-of-distribution generalization, evaluation pairs high-fidelity simulation with real-robot deployment rather than relying on either alone. In simulation, affordance-centric methods such as AffordDP [2377] run on IsaacGym [2379] over articulated objects from GAPartnet [2380] whose functional parts (handles, knobs, levers) vary widely in geometry, while imitation suites such as Robomimic [2381] and RoboTwin [224] isolate trajectory tracking and dual-arm coordination, with P3-PO [2369] using them for controlled spatial-anchoring ablations. Real-world suites then stress in-the-wild generalization on articulated (drawers, microwaves), deformable (towel folding), and tool-use (sweeping) tasks, as in Point Policy [2378] and Track2Act [2375], and a zero-shot cross-category protocol (train on one knife, test on spatulas or novel knives) is formalized by Robo-ABC [2374] to measure how well retrieved anchors transfer without extra supervision.

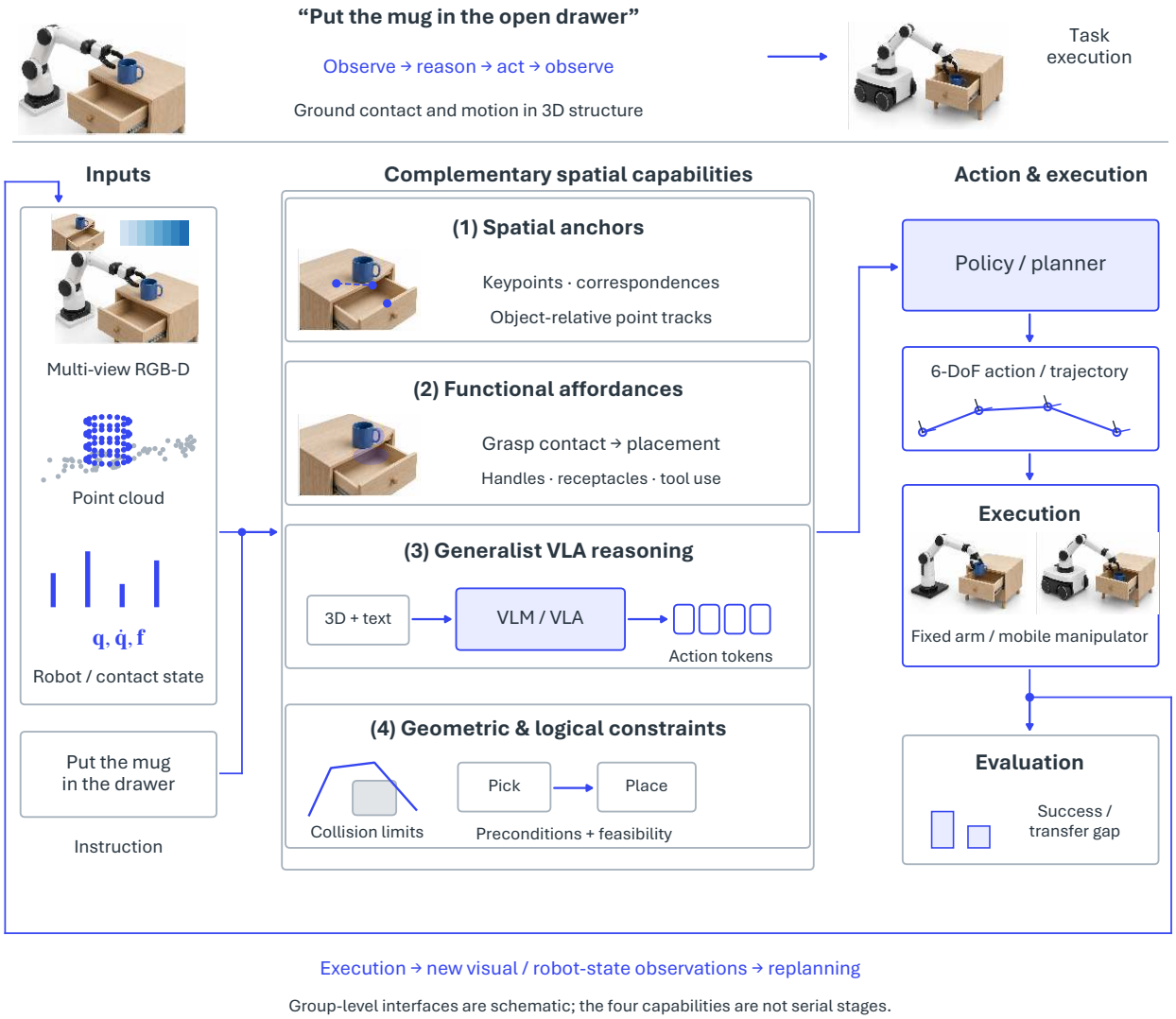


Figure 9.1: Spatial intelligence for robotic manipulation, illustrated by placing a mug in an open drawer: four complementary spatial capabilities link multimodal inputs to 6-DoF actions, and execution returns new observations for replanning. Evaluation bars and depth/point-cloud graphics are schematic, not measured results.

Performance is reported as Task Success Rate and, to isolate the contribution of anchoring, a Generalization Gap between seen and unseen instances. Spatially grounded policies outperform pixel baselines under shift in these studies: P3-PO reports large gains on cluttered, novel-object scenes [2369], and Robo-ABC improves both affordance-retrieval accuracy and real-robot success without heavy robot-data collection, binding demonstrations to reusable spatial structure more effectively than classic one-shot imitation [2382].

9.1.4 Pivotal Task-2: Inferring Functional Action Possibilities

This pivot shifts from general object recognition to extracting **object affordances**, *i.e.*, the specific action possibilities implied by an object's functional properties, and uses them as the intermediate representation that bridges raw perception and contact-rich execution. Whereas Pivotal Task-1 treats contact points as viewpoint-stable *anchors* for conditioning a policy, affordances here are studied as *functional possibilities* that

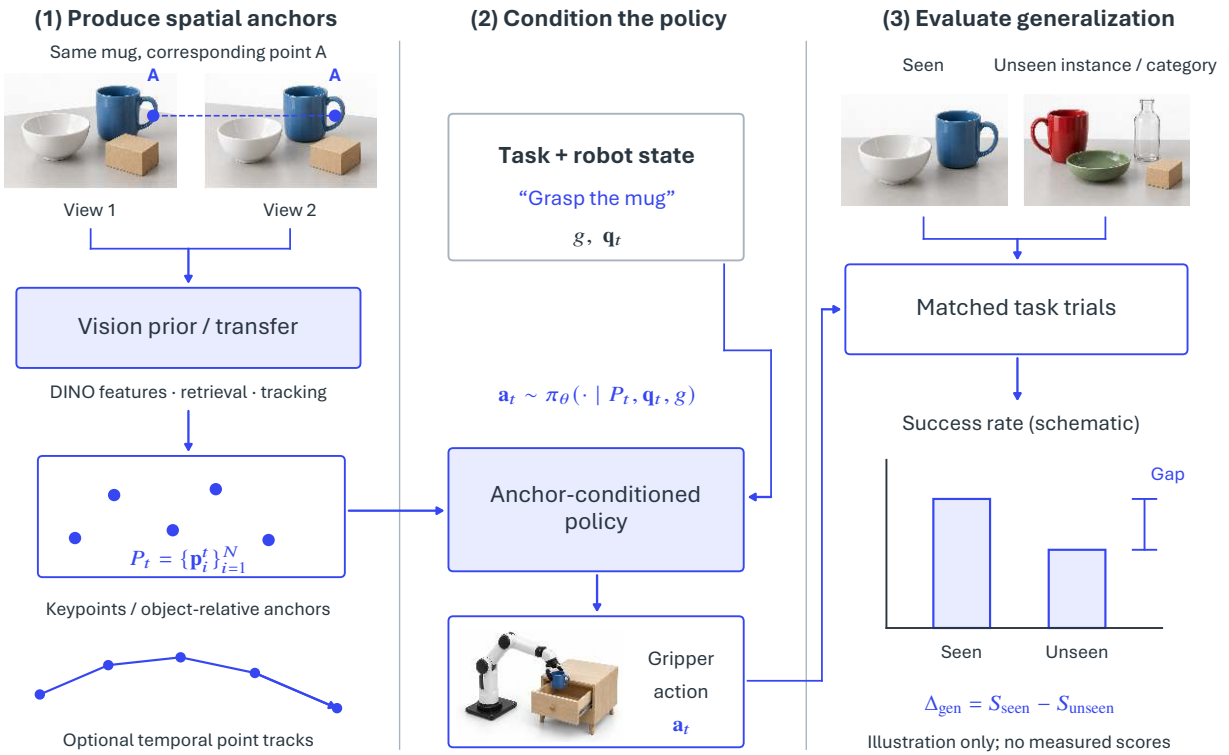


Figure 9.2: Grounding action in spatial anchors: (1) anchors P_t produced or transferred from vision priors across views, (2) an anchor-conditioned policy, and (3) generalization measured as the seen–unseen success gap Δ_{gen}. Scenes, correspondences, and score bars are schematic, not measured results.

specify what action an object supports. As Figure 9.3 shows, the thread runs from how affordances are *represented*, through the *actions* they drive (grasping, placement, tool use), to *outcome-oriented* benchmarks that score achieved function rather than imitated trajectory.

To capture functional properties beyond whole-object categorization, prior work builds object-centric representations (part-level segmentation [2383], dense visual descriptors [335], and keypoint-based encodings [547, 2384, 337]) that directly support action synthesis, and more recent vision–language models predict spatial affordance points directly from open-vocabulary prompts (RoboPoint [1494], MOKA [2385]). These representations drive learning-based 6-DoF grasping [2386, 2372, 2387, 2275, 2388, 2389] and stable object placement [2390, 2391, 267, 2388, 2392] by grounding generation in functional structure rather than raw geometry, and a specialized strand further models tool–behavior relationships, either learning tool–action–effect pairs [2393] or predicting specific affordances such as pulling [2394], so as to infer end-effector poses that achieve the intended functional outcome.

Benchmarks for affordance methods emphasize understanding object functionality over geometric matching: the property under test is *functional* generalization, not trajectory imitation. For grasping, YCB-based real-world evaluations [1184], GraspNet-1Billion [1183], and simulation suites such as Dex-Net [336] score 6-DoF stability and contact validity across novel instances and viewpoints; placement benchmarks such as RLBench [1346] and ManiSkill [2395] test stable stacking and insertion in cluttered, articulated scenes, and M2T2 [2388] explicitly evaluates valid placement poses conditioned on object geometry and scene context. For articulated objects and tools, Where2Act [337] and PartNet-Mobility [2396] score the prediction of actionable parts and interaction parameters for pulling, pushing, and rotating, while tool-use tasks judge whether the inferred end-effector pose produces the intended causal effect (drawing, hooking, pulling) rather than reproducing a specific trajectory. Evaluation therefore centers on outcome-oriented success: stable grasps that lift and hold unseen objects under pose, scale, and appearance variation; placement success, pose

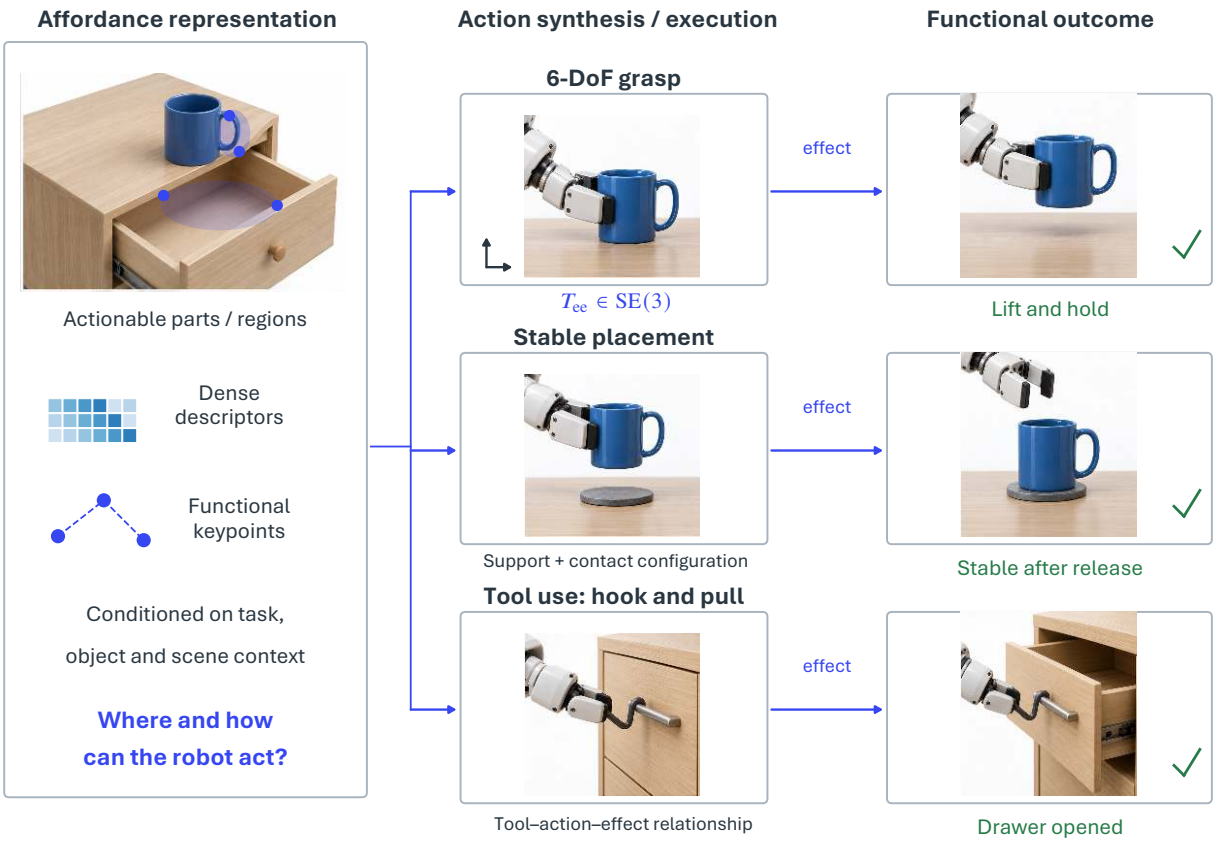


Figure 9.3: Affordances as functional action possibilities: a task- and context-conditioned affordance representation (left) supports three parallel task examples (6-DoF grasp, stable placement, hook-and-pull tool use), each judged by its functional outcome. Green checks mark intended outcomes in these illustrations, not measured results.

accuracy, and post-release stability under clutter and occlusion; and, for articulated and tool interactions, whether the predicted parameters yield the desired functional result.

9.1.5 Pivotal Task-3: Encoding Spatial Reasoning in Generalist Models

This pivot moves from specialized, single-task policies toward **generalist robotic foundation models**. Existing Vision–Language–Action (VLA) models such as RT-2 [97] and OpenVLA [99] show strong semantic understanding but often lack the spatial common sense needed for precise manipulation or robust generalization to novel viewpoints, so the central challenge is to strengthen spatial grounding during pre-training, enabling explicit reasoning about 3D geometry and generalization across environments and embodiments [229]. The general foundation-model and spatial-reasoning machinery these policies inherit is treated in Chapter 7; here we examine only its manipulation-specific adaptation, which Figure 9.4 groups into three families: explicit 3D representations, view-invariant representations, and grounded reasoning traces.

The first family builds explicit 3D representations: models such as 3D-LLM [1141] and 3D-VLA [201] bridge point clouds and large language models, with 3D-VLA adding a 3D generative pre-training objective that lets the model reason about action outcomes in a consistent geometric space, effectively "imagining" future states. The second groups view canonicalization and spatial action encoding as distinct mechanisms: OG-VLA [2397] projects multi-view observations into a point cloud and re-renders canonical orthographic views to reduce sensitivity to camera placement, while SpatialVLA [2398] uses Ego3D position encoding for observations and **Adaptive Action Grids** to discretize spatial movement actions; adapting to new setups

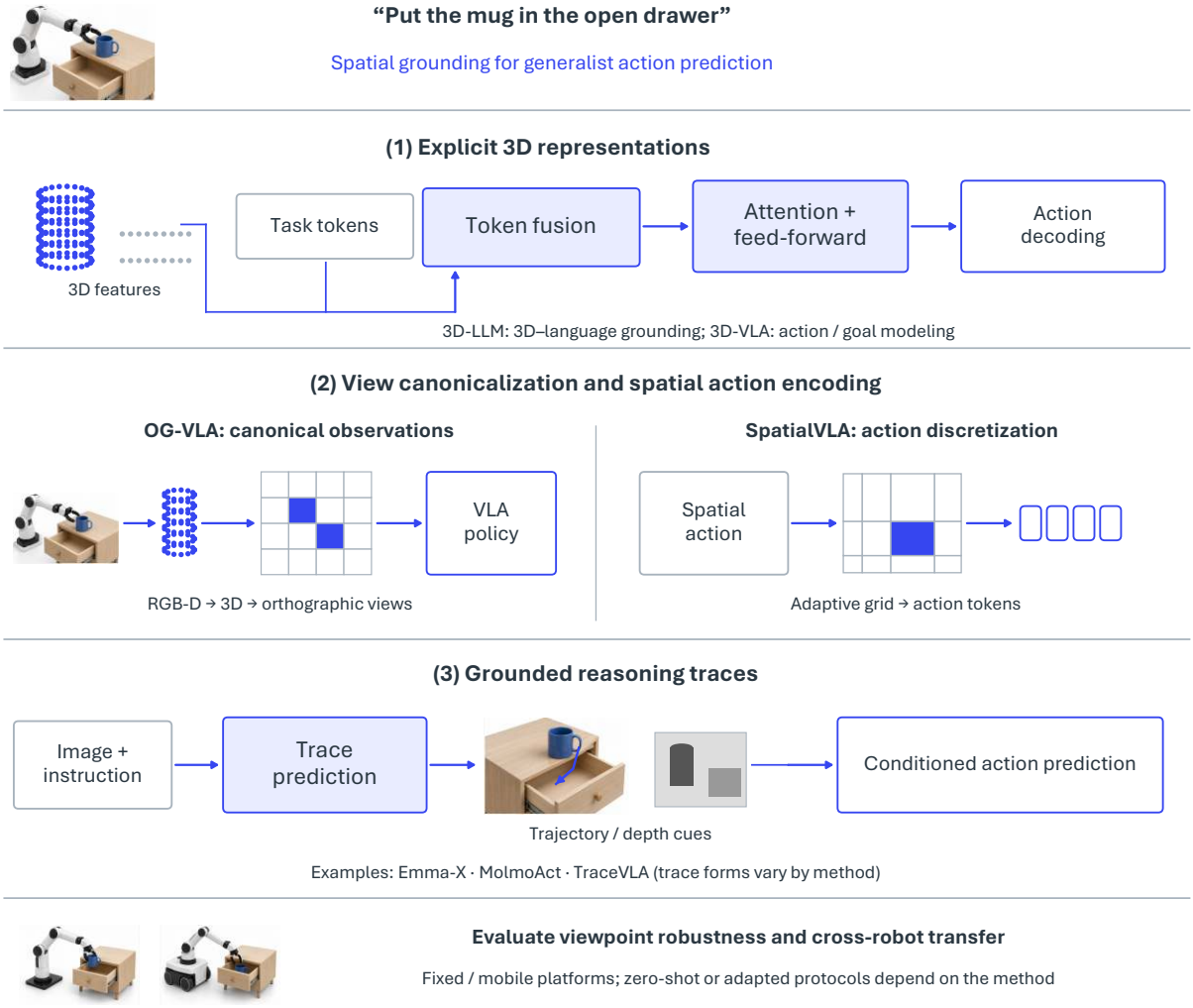


Figure 9.4: Spatial grounding mechanisms for generalist robot policies: (1) explicit 3D features fused with task tokens (3D-LLM, 3D-VLA), (2) view canonicalization (OG-VLA) and spatial action discretization (SpatialVLA), two separate methods rather than stages of one, and (3) grounded reasoning traces that condition action prediction. The bottom row shows evaluation of viewpoint robustness and cross-robot transfer; all maps and overlays are schematic.

can involve re-discretizing the grids. The third injects spatial reasoning through grounded reasoning traces: "thinking" models such as Emma-X [2399], MolmoAct series [2400, 2401], closely related to TraceVLA [1485], replace direct pixel-to-action regression with intermediate, grounded outputs (2D motion trajectories or relative depth maps) that explicitly condition the final action prediction.

These approaches are evaluated under method-specific zero-shot and adaptation protocols across embodiments, viewpoints, and environments, with key metrics including multimodal goal-generation quality, robustness to extrinsic camera perturbations, and skill transfer across robot morphologies such as from a fixed-base Franka arm to a mobile manipulator [1577, 2402]. The reported gains point the same way: canonical orthographic views keep grounding consistent under viewpoint change (OG-VLA), spatially aligned action grids share 3D structure across setups through discretized movement (SpatialVLA), and intermediate visual or depth-aware reasoning traces act as a regularizer on action prediction (Emma-X, MolmoAct).

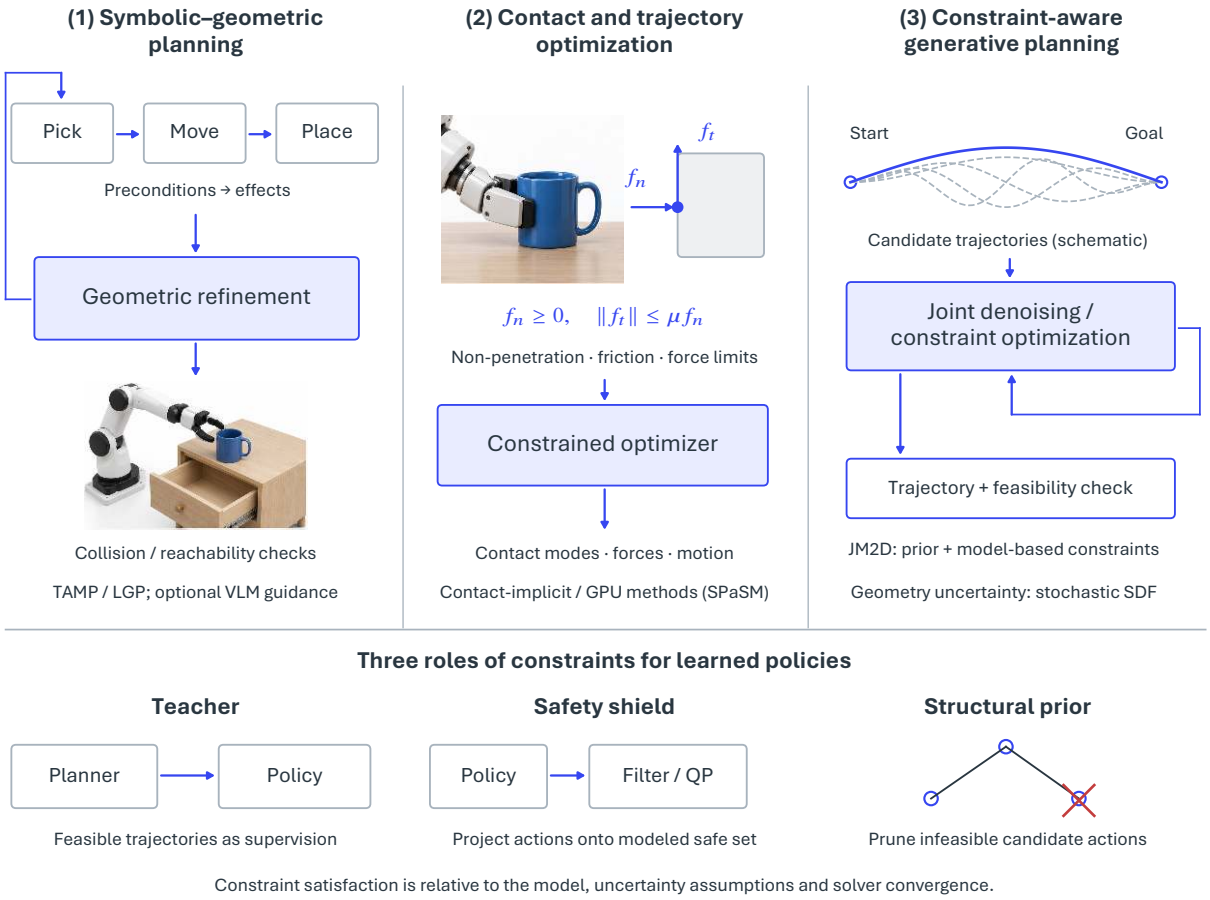


Figure 9.5: Geometric and logical constraints in manipulation: (1) symbolic–geometric planning with revision after infeasibility, (2) contact and trajectory optimization (inset: normal force f_n into the object, tangential force f_t along its surface), and (3) constraint-aware generative planning. The bottom row shows three roles constraints play for learned policies.

9.1.6 Pivotal Task-4: Satisfying Geometric and Logical Constraints

This pivot moves planning from heuristic trajectory generation toward formal **constrained optimization** that enforces geometric feasibility, fundamental physical laws (non-penetration, friction cones, force limits), and high-level logical task structure. By casting manipulation as multi-objective optimization, these methods seek to satisfy modeled hard constraints while optimizing soft task objectives, spanning both high-frequency control and long-horizon decision-making. Figure 9.5 organizes the thread along three fronts (symbolic–geometric planning, contact and trajectory optimization, and constraint-aware generative planning) before synthesizing the roles constraints play in learning-based systems.

Methodological progress has moved from purely symbolic planners toward neuro-symbolic, learning-augmented optimization frameworks that unify discrete logic with continuous physics. Classical Task and Motion Planning (TAMP) and Logic-Geometric Programming (LGP) [1267] provide the foundation by jointly reasoning over symbolic task structure and geometric feasibility, and recent systems augment them with Large Language Models and Vision-Language Models for scalability in open-ended domains; for example, VIZ-COAST [2403] uses the spatial reasoning of VLMs to predict refinement failures *before* expensive geometric optimization, substantially reducing search complexity on long-horizon problems. At the level of physical interaction, contact-implicit optimization discovers contact schedules without explicit

Table 9.1: Representative benchmarks, methods, and headline metrics for the four manipulation pivots, drawn from simulation and real-robot settings; metrics are reported on out-of-distribution object instances and configurations where the underlying work permits.

Pivot	Representative Benchmark(s)	Representative Method(s)	Headline Metric
P1: Spatial anchors	IsaacGym [2379] with GAPartNet [2380], Robomimic [2381], RoboTwin [224] + real-world	P3-PO [2369], Point Policy [2378]	Task SR / Generalization Gap
P2: Affordances	GraspNet-1B [1183], Dex-Net [336], Where2Act [337], PartNet-Mobility [2396]	M2T2 [2388]	Grasp / Placement SR
P3: Generalist VLA	ManiSkill [2395], Octo eval [1577]	SpatialVLA [2398], 3D-VLA [201]	Zero-shot cross-embodiment SR
P4: Constraints	FMB [2408]	SPaSM [2404], JM2D [2407]	Constraint satisfaction / long-horizon SR

mode enumeration but is traditionally too slow for real-time control; GPU-parallelized systems such as SPaSM [2404] compile constraint evaluation and gradient computation into CUDA kernels, accelerating joint optimization of object placements and robot trajectories. Beyond deterministic constraints, learning-based and generative models capture uncertainty and complex structure: differentiable physics engines [2405] let gradients flow through physical interactions, stochastic implicit neural signed distance functions [2406] model geometric uncertainty directly within the planning manifold, and diffusion-based planners such as JM2D [2407] couple model-free diffusion with model-based constraint optimization during sampling, rather than relying only on a sequential plan-then-filter interface.

Such systems are benchmarked on multi-stage re-grasping under uncertainty, forceful tool use, and tight-tolerance sequential interaction, since what they promise is reliability over long, contact-rich horizons; suites such as the Functional Manipulation Benchmark (FMB) [2408] scoring not only task completion but constraint satisfaction under partial observability, external disturbance, and compounding execution error. In these systems constrained optimization plays three roles: it acts as a *teacher*, generating physically valid trajectories that supervise imitation and offline reinforcement learning; as a *safety shield*, where model-based projections map unconstrained neural or diffusion policy outputs onto constraint-satisfying sets at execution time [2409]; and as a *structural prior* that prunes physically infeasible regions of the action space and curbs compounding error in long-horizon tasks. Together these roles keep learned and generative planners grounded in geometric and physical reality, reducing physically implausible outputs and supporting interpretable, feasible plans under the modeled assumptions; real-world execution still depends on state estimation, model accuracy, solver convergence, and disturbance handling.

Table 9.1 consolidates the principal benchmarks, representative methods, and headline evaluation focus across the four pivots, situating the spatially grounded methods surveyed above within a common evaluation landscape.

9.1.7 Established Tools and Systems

Progress in spatially grounded manipulation has been inseparable from a maturing ecosystem of simulators, open software stacks, shared datasets, and, increasingly, commercial robot platforms that serve as common development and deployment substrates.

Simulation and physics engines. GPU-parallel engines such as NVIDIA IsaacGym and the newer Isaac Sim / Isaac Lab[1] [2379], articulated-object simulators such as SAPIEN [2396] (with its PartNet-Mobility assets), and general-purpose physics engines including MuJoCo [1187], PyBullet[2], and the recent Genesis[3] provide

[1] https://developer.nvidia.com/isaac/sim
[2] https://pybullet.org
[3] https://github.com/Genesis-Embodied-AI/Genesis

high-throughput, contact-rich environments, while ManiSkill [2395], RLBench [1346], Robomimic [2381], and RoboTwin [224] package standardized task suites, demonstrations, and evaluation protocols that have become de facto testbeds for imitation and policy learning.

Open software stacks and datasets. On the software side, the Robot Operating System (ROS 2) with the MoveIt motion-planning framework[4] remains the de facto middleware for real-robot integration, and Hugging Face's LeRobot[5] packages datasets, pretrained policies, and training pipelines for low-cost reproduction. Large cross-embodiment efforts such as Open X-Embodiment[6], which aggregates demonstrations and the RT-X policies across dozens of robot platforms, together with open generalist checkpoints such as Octo [1577] and OpenVLA [99] and NVIDIA's GR00T humanoid foundation models[7], expose pretrained weights and fine-tuning interfaces that lower the barrier to transferring foundation-model policies onto new morphologies.

Commercial and humanoid platforms. On the deployment side, a rapidly growing set of companies is moving spatially grounded manipulation out of the laboratory. Boston Dynamics[8] (the electric Atlas and the Stretch warehouse robot), Tesla Optimus[9], Figure AI[10], 1X Technologies[11], Agility Robotics[12] (the Digit humanoid, piloted in logistics), Physical Intelligence[13] (the π_0 generalist policy), and Sanctuary AI[14] each pair learned manipulation with humanoid or mobile-manipulator hardware, while firms such as Covariant[15] target warehouse pick-and-place at commercial scale. These platforms are the clearest evidence that the spatial-intelligence capabilities surveyed above are crossing from benchmark tasks into economically deployed systems.

Together these tools convert the representation and reasoning machinery of Chapter 7 into reproducible, physically grounded manipulation pipelines; their principal limitation remains the fidelity gap between simulated and real contact dynamics, which the following discussion revisits.

9.1.8 Challenges

Despite rapid progress across all four pivots, bringing *spatial intelligence* to robust, general-purpose manipulation remains bounded by a set of intertwined difficulties that cut across perception, policy, and planning. They cluster into four threads: keeping a persistent and calibrated internal model of the scene; closing the loop between contact physics and active sensing; transferring skills across embodiments while sourcing the data to do so; and meeting the reliability budget of a deployed system.

Persistent, calibrated world models. The first requirement is a *persistent 3D belief*: an actionable, temporally consistent world model that survives heavy occlusion, articulation, deformation, and sensor drift, and that is maintained at control rates rather than reconstructed from scratch each frame. Persistence alone is not enough if the model cannot say how much it trusts itself. *Calibrated uncertainty* asks that distributional beliefs over geometry, contact state, friction, and articulation propagate through perception, policy, and planner, so that a decision reflects confidence rather than a bare point prediction and a low-confidence grasp can trigger caution instead of failure.

[4] https://moveit.ai
[5] https://github.com/huggingface/lerobot
[6] https://robotics-transformer-x.github.io
[7] https://developer.nvidia.com/isaac/gr00t
[8] https://bostondynamics.com
[9] https://www.tesla.com/we-robot
[10] https://www.figure.ai
[11] https://www.1x.tech
[12] https://agilityrobotics.com
[13] https://www.physicalintelligence.company
[14] https://sanctuary.ai
[15] https://covariant.ai

Contact modeling, sensing, and active perception. Manipulation is ultimately about contact, yet most spatial models still reason about collision-free geometry rather than the physics of touch. Progress on *contact modeling and sensing* hinges on tightly integrating vision with force and tactile feedback to reason about stick–slip, compliance, and incipient failure. Because contact is often unobservable until it happens, this couples naturally to *information-seeking control*: interleaving manipulation with active perception and probing (touch-to-disambiguate, look-while-acting) inside tractable belief-space planners. And because contact is precisely where simulators diverge from reality, *sim-to-real for contact* stays open, demanding better material, friction, and compliance models together with fast system identification for on-robot adaptation.

Cross-embodiment transfer and data at scale. Skills learned on one platform rarely move cleanly to another. *Cross-embodiment and cross-scene transfer* requires aligning action spaces and spatial tokens across grippers, cameras, and environments without hand-tuned frames or retraining. Sustaining that transfer is a data problem as much as an algorithmic one: *data and evaluation at scale* calls for "long-tail" data engines that cover rare geometries, materials, and tools with counterfactual and safety labels, and for benchmarks that measure calibration, persistence, recovery, and long-horizon competence rather than success rate alone.

Systems reliability. Finally, a deployed manipulator must hold together as a *system*: meeting latency budgets, absorbing calibration drift, detecting out-of-distribution geometry, and enabling safe fallback and continual learning without catastrophic forgetting. Closing these gaps will require tighter couplings among 3D world models, affordance predictors, language-grounded objectives, and constraint-aware planners, so that spatial representations are not only expressive but *actionable, calibrated, and safe* in unstructured settings.

9.2 Autonomous Driving

9.2.1 Introduction

Autonomous driving sits at the intersection of spatial intelligence, transportation engineering, and human–machine interaction, and it is one of the most demanding real-world testbeds for the spatial capabilities surveyed in Part III. A self-driving vehicle must continuously close a *perceive–localize–decide–act* loop: it has to recognize and track the surrounding traffic scene, estimate its own pose within a metric map, reason about how nearby agents will move, and commit to safe control actions, all under tight latency and safety budgets. Advances in deep learning, multi-sensor fusion, and vehicle-to-everything (V2X) communication have steadily pushed the boundaries of what vehicles can perceive, predict, and decide, while fundamental questions around robustness, trust, ethics, and regulation remain open. Perception and reasoning themselves are treated in Chapter 6 and Chapter 7; what is specific to driving is that each capability must hold up at speed, in all weather, and with lives at stake.

Figure 9.6 summarizes the SAE J3016 automation levels that frame this domain. The SAE J3016 standard [16] provides a widely adopted framework for defining automation levels, with most current production systems still operating at Level 2. Progress toward higher levels, especially full Level 5 autonomy, faces major technical [2410], ethical [2411], and financial [2412] challenges.

Deep learning has driven most of this progress, from Transformers [2413] and graph networks [2248] that model multi-agent interaction to lifelong learning for open-world adaptation [2414], but its opacity raises safety concerns compounded by sensor noise and the limited reach of vehicle-to-vehicle communication, and reaching Levels 4 and 5 depends as much on redundant hardware, large-scale data pipelines, and verifiable safety assurance as on model architecture.

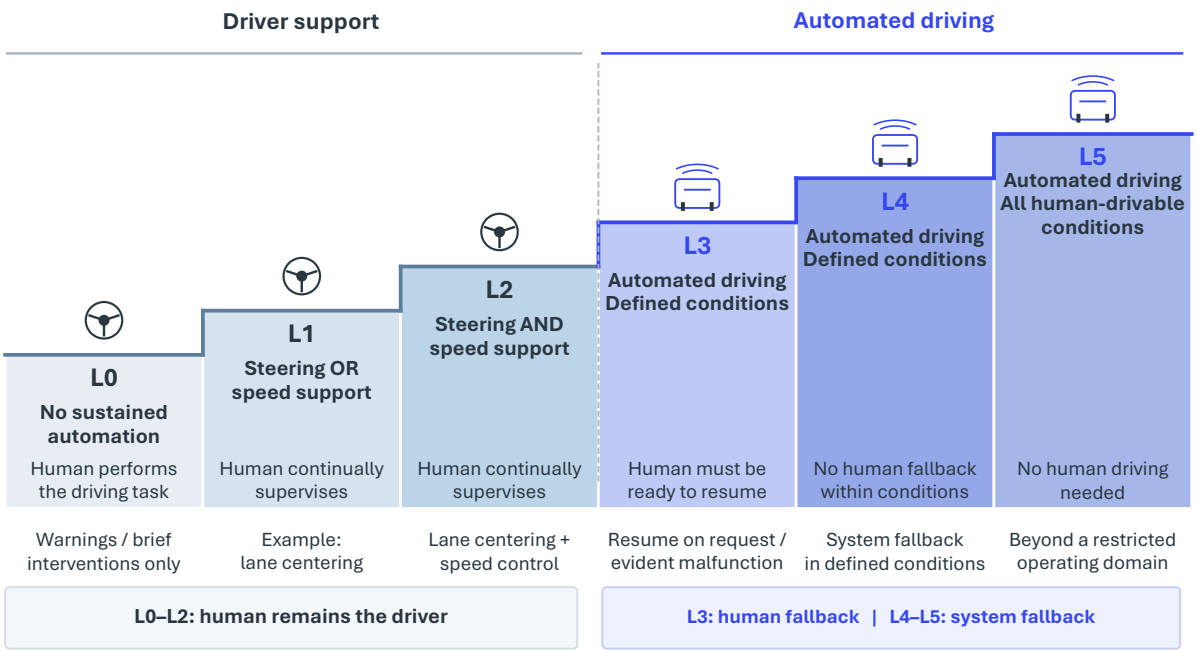


Figure 9.6: Functional distinctions among SAE J3016 driving-automation levels. Levels 0–2 retain a human driver; Level 3 requires a human to resume driving when necessary, whereas Levels 4–5 do not rely on human fallback within their operating scope. The stair heights indicate ordered categories, not measured safety, model complexity, or technology readiness.

[16] https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic

9.2.2 System Overview

Modern self-driving stacks realize the *perceive–localize–decide–act* loop as a modular pipeline, sketched in Figure 9.7. A front-end *perception* module converts raw, multi-modal sensor streams (camera, LiDAR, radar) into a structured model of the scene: what objects are present, where the drivable surface and lane geometry lie, and which agents are most safety-critical. A *localization and mapping* module places the ego-vehicle within a metric, typically HD-map-anchored, representation of that scene, so that perception outputs can be expressed in a stable world frame. A *decision and control* module then predicts how the scene will evolve and selects the steering, acceleration, and braking commands that are sent to the actuators and fed back into the environment, closing the loop. Each of the three stages is a distinct spatial-intelligence problem with its own representations, benchmarks, and failure modes, and they are the pivotal tasks below, in pipeline order.

9.2.3 Pivotal Task-1: Perception and Detection

Perception is the front-end of the pipeline: it turns raw camera, LiDAR, and radar streams into a structured, layered model of the scene that every downstream module consumes, reusing the detection, segmentation, and 3D scene-understanding machinery of Chapter 6 but specializing it to the driving Operational Design Domain (ODD). It is best read as a set of complementary structured representations, sketched in Figure 9.8: object detection localizes traffic participants, semantic and occupancy prediction labels the surrounding space, lane detection recovers the road geometry that constrains motion, and pedestrian recognition isolates the most safety-critical agents.

The base layer is **object detection**, which identifies and localizes obstacles and is the most direct determinant of collision safety [2415]. Its central practical constraint is the ODD: environmental factors such as poor weather degrade sensor visibility and model accuracy [2416], so recent work formalizes and iteratively validates the ODD through quantifiable environmental models and scenario catalogues, trusting a detector only within conditions it has demonstrably handled, though the absence of a standardized ODD specification still makes cross-system safety claims hard to compare. Robustness to adverse weather is therefore a first-class concern [2417, 2418, 2419]: rain, snow, fog, and glare corrupt LiDAR, radar, and camera signals alike, and because driving datasets under-represent these conditions, models trained on fair-weather data generalize poorly. Responses range from dedicated robust pipelines, such as multi-stage visibility enhancement coupled with deep CNN tracking [2420], to new sensing hardware, *e.g.*, frequency-modulated continuous-wave (FMCW) LiDAR and high-dynamic-range cameras that recover usable signal in glare at higher cost. Moving from the image plane into metric space, **3D object detection** localizes and tracks objects in the vehicle's surroundings [434], where the bird's-eye-view (BEV) representation has become dominant because it yields a unified top-down map well suited to planning. BEVFormer [2421] fuses multi-camera and temporal cues with spatiotemporal transformers to approach LiDAR-level accuracy on nuScenes, while Fast-BEV [2422] and SparseBEV [2423] trade marginal accuracy for the lightweight, sparse, query-based computation needed for real-time on-vehicle deployment, together tracing the field's balancing of accuracy, latency, and deployability.

Detections tell the planner *where objects are* but not *which space is free*; that dense layer is supplied by **semantic segmentation** and its 3D extension, occupancy prediction [2424, 2425]. The progression runs from 2D pixel labeling with FCN [127] and DeepLab [461] to volumetric reasoning: the Tri-Perspective View [2426] lifts RGB into a 3D representation rivalling LiDAR, SurroundOcc [2427] uses multi-camera 2D–3D attention to capture arbitrary shapes, and occupancy world models such as OccWorld [2333] extend this to forecasting how the 3D scene evolves, moving beyond the limits of bounding boxes. Because these dense predictors are data-hungry and domain-sensitive, robustness techniques such as the SePiCo pixel-contrast framework [2428] and GAN-based synthetic data [2429] are used to survive scene variability, occlusion, and weather shift. On top of the free-space layer, **lane detection** recovers the road geometry that defines the lateral reference for lane-keeping and planning; early CNN- and YOLO-style detectors trace markings well in clear conditions but are brittle on curved, worn, or occluded lanes, motivating successors that lift lanes

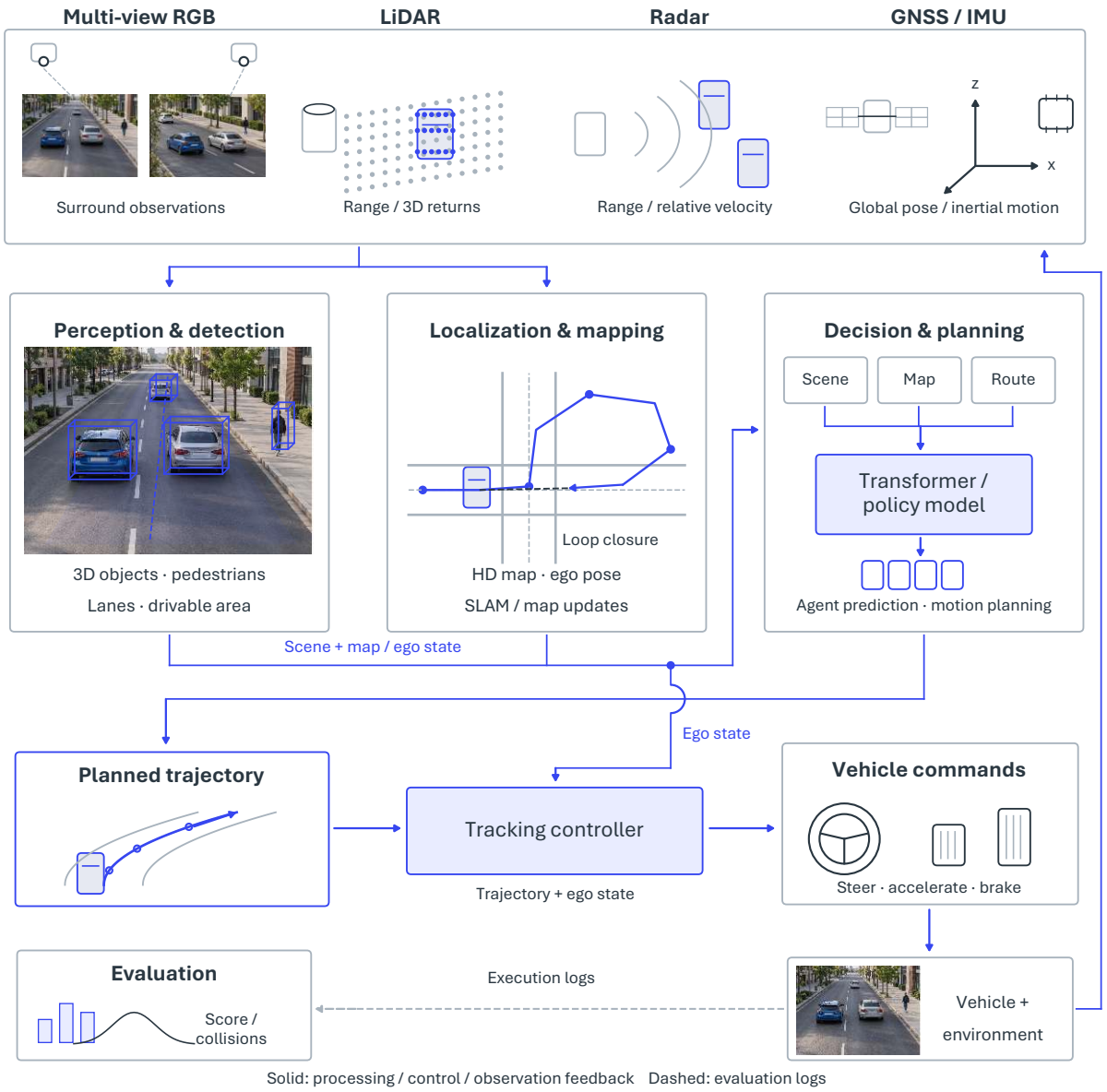


Figure 9.7: Closed-loop spatial intelligence for autonomous driving: complementary sensor streams feed perception and localization and mapping, whose scene and ego/map state drive decision, planning, and trajectory tracking; vehicle–environment interaction returns new observations. The controller also receives ego state; the curved wire crossing is not a junction. Dashed execution logs feed evaluation separately from control. Scenes, maps, and evaluation plots are illustrative, not measured results.

into 3D and use attention over global context to infer occluded segments. Finally, **pedestrian recognition** focuses on vulnerable road users whose behavior can be difficult to predict: CNN-, YOLO-, and SSD-based detectors deliver the required precision and latency, color–thermal fusion handles low-light scenes, and spatial–temporal models couple detection with motion prediction to anticipate crossing behavior.

Across all four layers, perception is benchmarked on large-scale driving datasets, KITTI [173], the Waymo Open Dataset [446], and nuScenes [445] for 2D/3D detection, Cityscapes [501] and BDD100K [711] for segmentation, dedicated lane benchmarks for lane detection, and CityPersons [2430] for pedestrians, and scored with mean Average Precision (mAP) and the nuScenes Detection Score (NDS), mean Intersection-over-Union (mIoU), and F1/accuracy, respectively (Table 9.2). Increasingly these metrics are reported on



Figure 9.8: Four complementary driving perception tasks feeding a shared scene model: (1) object detection, (2) semantics and 3D occupancy, (3) lane geometry, and (4) pedestrian/VRU tracking and motion prediction. Annotations and the dashed pedestrian path are illustrative, not measured predictions.

adverse-weather and out-of-distribution splits, precisely to measure behavior within and beyond the declared ODD.

9.2.4 Pivotal Task-2: Localization and Mapping

If perception answers *what is around the vehicle*, localization and mapping answer *where the vehicle is* and *what the persistent structure of the world is*. The two are tightly coupled: a metric, HD-map-anchored pose lets perception outputs be expressed in a stable world frame that planning can rely on, and the map itself is built and maintained from the same sensor streams. As Figure 9.9 shows, the problem decomposes into pose estimation against a known map (localization), the construction of that map (3D mapping and reconstruction), and the joint solution of both when no map exists (SLAM), with multi-sensor fusion the common thread that holds accuracy together when any single modality degrades.

Localization has moved from GPS-dependent positioning to learned matching of live camera or LiDAR observations against a pre-built map, reaching centimeter-level accuracy even in GPS-denied urban canyons; semantic cues such as roads and buildings further let a vehicle re-localize in changed environments without full remapping. Because no single sensor is reliable across occlusion, weather, and GPS dropout, **sensor-fusion localization** is the dominant paradigm: probabilistic estimators (particle filters, Kalman filters, and NDT scan matching) reconcile GNSS, IMU, and LiDAR, while learned vision–radar fusion compensates

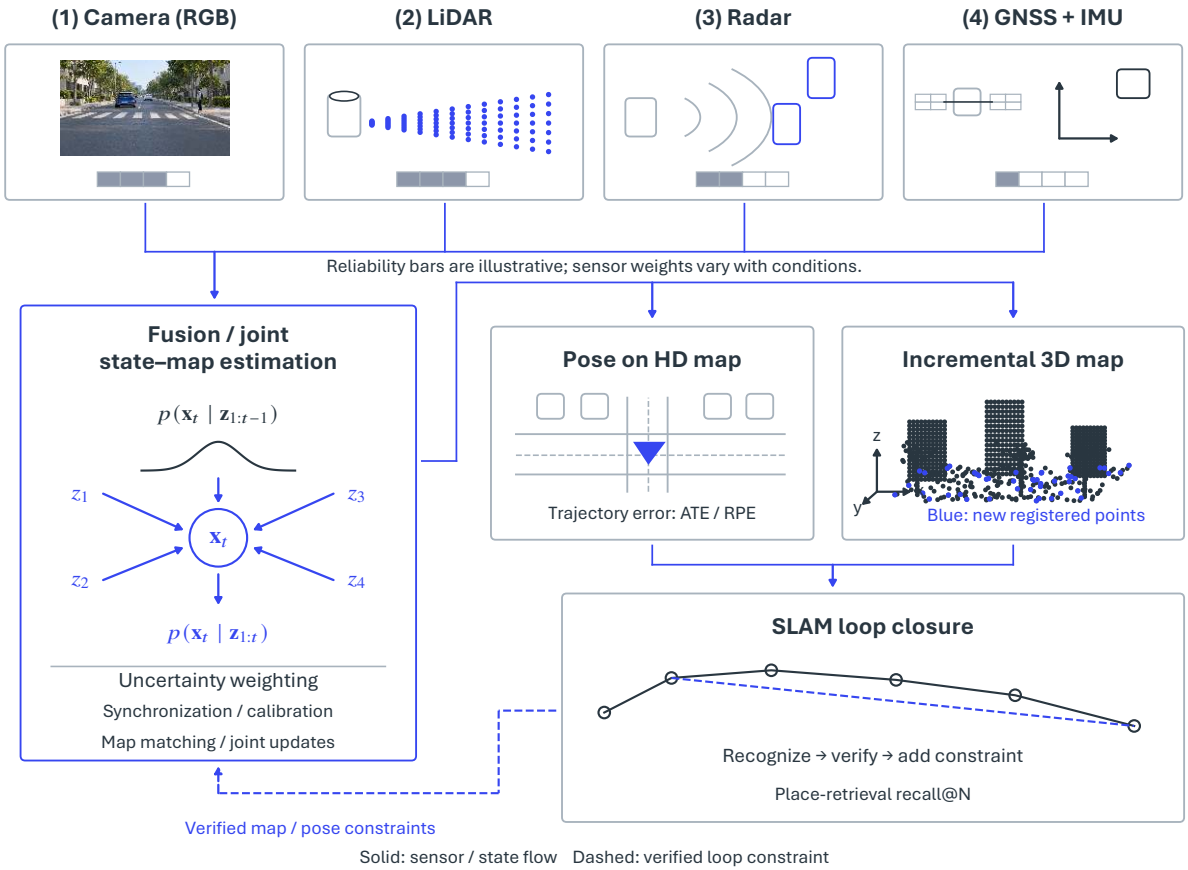


Figure 9.9: Localization and mapping through uncertainty-aware sensor fusion: camera, LiDAR, radar, and GNSS/IMU measurements update the state posterior, yielding a map-relative pose (evaluated by ATE/RPE), an incrementally updated 3D map, and loop closures that feed back only after verification. Pose and map are parallel products of joint estimation; only verified loop constraints return to the estimator. Recall@N evaluates place retrieval, not pose accuracy.

for modality-specific weaknesses in rain, fog, and glare. The recurring bottlenecks here are cross-sensor spatial/temporal synchronization and fault tolerance, which motivate Bayesian uncertainty weighting and online-adaptive fusion that recalibrates as conditions change.

3D mapping and reconstruction supply the persistent representation that localization matches against, turning LiDAR and stereo streams into metric models; for the place-recognition and loop-closure that keep maps globally consistent, transformer descriptors such as SeqOT [2431] match sequential LiDAR scans faster than the sensor frame rate, trained on richly annotated corpora such as ApolloScape [660]. The open problems are the cost of processing dense point clouds at city scale and the handling of occluded or incomplete returns, which favor incremental map updating over full re-mapping. When the map is unavailable, **SLAM** solves localization and mapping jointly, which is critical in GPS-denied corridors; modern driving-oriented vSLAM hybridizes classical geometry with learned features (for example, extending ORB-SLAM [393] with learned features, semantic labels, and learned depth to favor stable landmarks) and recovers from drift through loop closure with bundle adjustment.

These capabilities are evaluated on KITTI Odometry [173], Oxford RobotCar [1242], and ApolloScape [660] for long-horizon SLAM and place recognition, on nuScenes [445] and Waymo [446] for map-relative pose, and on Argoverse [2255] for the trajectory forecasting that localization feeds, *e.g.*, with the Motion Transformer [2247] (Table 9.2). The standard metrics, *i.e.*, Absolute and Relative Pose Error (ATE/RPE),

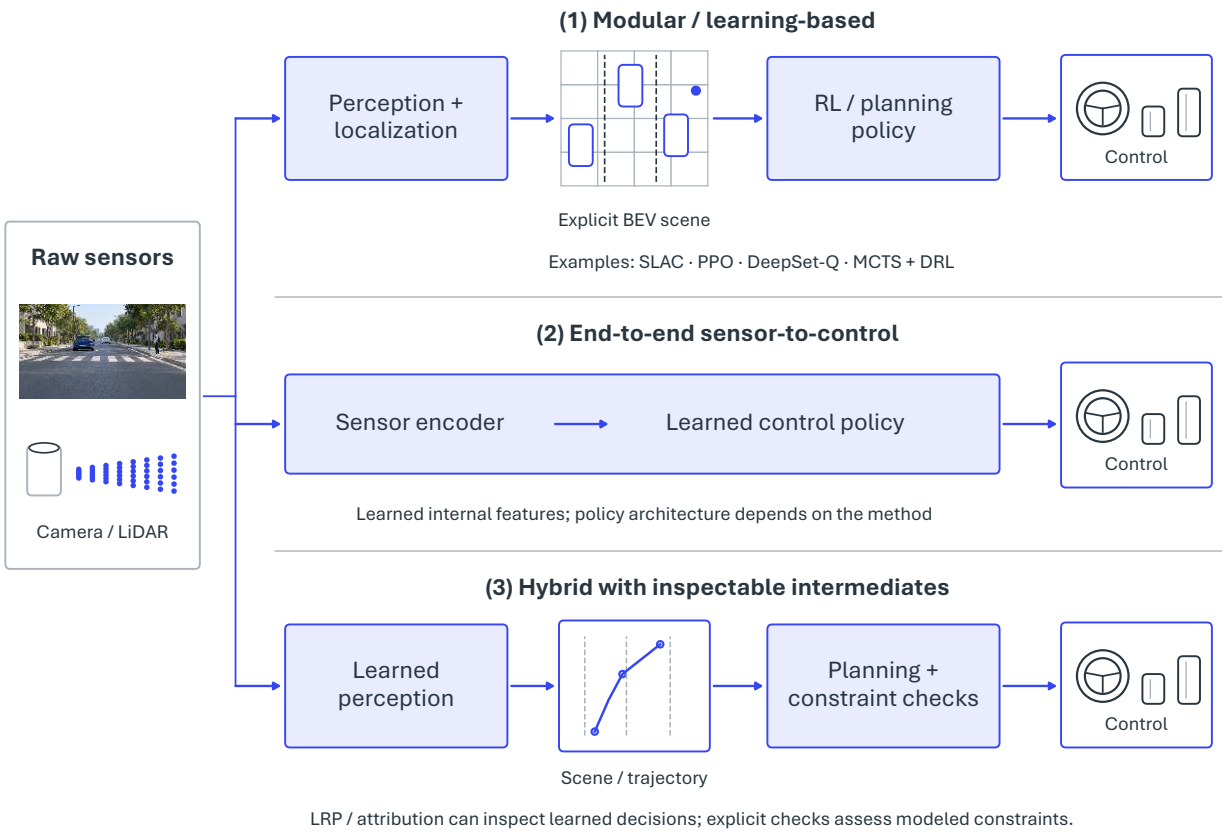


Figure 9.10: Three alternative architectures for driving decision and control: (1) modular pipelines with an explicit BEV scene, (2) end-to-end sensor-to-control learning, and (3) hybrids with inspectable scene or trajectory intermediates and explicit constraint checks.

recall@N for loop closure, and minADE/minFDE for forecasting, are reported under GPS-denied, adverse-weather, and sensor-degradation conditions to stress exactly the robustness the fusion machinery is built to provide.

9.2.5 Pivotal Task-3: Decision Making and Control

Decision making and control is the action end of the pipeline: it consumes the scene model from perception and the pose from localization, predicts how the scene will evolve, and commits to the steering, acceleration, and braking that close the loop, drawing on the spatial-reasoning and planning paradigms of Chapter 7. As Figure 9.10 indicates, two architectural philosophies dominate, differing in how much of the perceive-to-act mapping is learned end-to-end versus decomposed into interpretable modules, and they are increasingly converging.

The first philosophy is **learning-based decision making**, centered on (deep) reinforcement learning, which acquires driving policies through trial-and-error interaction and is well suited to the sequential, interactive nature of driving. Representative methods illustrate its design axes: the stochastic latent actor-critic (SLAC) [2432] learns compact latent states from high-dimensional images, PPO [1413] within a nonlinear control framework handles sharp turns and obstacle avoidance, and the DeepSet-Q architecture [2433], building on permutation-invariant set encoders [1372], ingests a variable number of surrounding agents for high-level lane-change decisions. Hybrids with explicit planning, such as Monte Carlo tree search guided by DRL or scene-representation transformers that forecast neighbor intentions, improve sample efficiency and situational awareness in dense traffic. The binding difficulty is that safety-critical episodes are rare in logged

data, which pushes the field toward high-fidelity simulation for rare-event coverage and toward encoding hard safety constraints directly in the policy rather than trusting the reward signal alone. The second philosophy, **end-to-end decision making**, learns the mapping from sensors to trajectory or control directly, removing hand-designed interfaces between modules. Early instances learned state-to-action policies with DDPG [2434] in simulators such as TORCS, and multi-task backbones such as HybridNets [2435] share computation across detection and segmentation without yet closing the loop to control. Current systems are trained by imitation on large driving logs: TransFuser [2436] fuses camera and LiDAR features with transformer attention and predicts waypoints for a closed-loop agent, and UniAD [2437] chains tracking, mapping, motion forecasting, and occupancy prediction through query interfaces inside one network optimized for planning. Driving world models such as GAIA-1 [2330] generate future multi-view scenes conditioned on ego actions, giving such policies a learned simulator for training and evaluation. Opacity is the price of end-to-end training: attribution methods such as layer-wise relevance propagation (LRP) [2438] support interpretation but do not by themselves establish safety or regulatory compliance, which is why hybrid designs keep inspectable scene or trajectory intermediates together with explicit constraint checks, whose guarantees extend only as far as the modeled constraints.

Closed-loop driving policies are evaluated chiefly in simulation, *i.e.*, the CARLA urban simulator [2326] (NoCrash, Longest6), the nuPlan planning benchmark [2439], and highway-interaction simulators built on HighD/INTERACTION (Table 9.2), because rare, safety-critical events cannot be exercised safely on public roads. The headline metrics are the CARLA Driving Score (route completion penalized by infractions), collision and intervention rates, and route completion, complemented by comfort proxies such as jerk and, for the open-loop forecasting sub-problem, minADE/minFDE.

Table 9.2: Representative benchmarks for spatial intelligence in autonomous driving, covering perception, localization and mapping, motion prediction, and decision making.

Pivotal task	Benchmark	Representative method	Headline metric
Perception (3D detection)	nuScenes	BEVFormer [2421]	NDS / mAP
Perception (3D detection)	Waymo Open	SparseBEV [2423]	mAP / latency
Perception (segmentation)	Cityscapes	SePiCo [2428]	mIoU
Perception (occupancy)	nuScenes Occ3D	SurroundOcc [2427]	mIoU
Localization & mapping	KITTI Odometry	Deep / semantic vSLAM	ATE / RPE
Prediction (forecasting)	Argoverse	Motion Transformer (MTR) [2247]	minADE / minFDE
Decision & control	CARLA (Longest6)	TransFuser [2436]	Driving Score
Decision & control	nuScenes (open-loop planning)	UniAD [2437]	L2 error / collision rate
Decision & control	nuPlan	Learned planners	Closed-loop score

9.2.6 Established Tools and Systems

The systems on the road today populate a deployment ladder whose rungs are the SAE levels of Figure 9.6, but what separates one rung from the next is rarely a single breakthrough in perception accuracy. It is, instead, two coupled engineering choices: how wide an Operational Design Domain (ODD) the system claims, and where legal responsibility for the driving task sits. Three tiers are in genuine production, *i.e.*, supervised Level 2 assistance, certified Level 3 "eyes-off" automation, and driverless Level 4 robotaxis, and, as the following progression shows, moving up the ladder consistently means adding sensing and compute redundancy while *narrowing* the domain in which the autonomy guarantee holds.

9.2.6.1 Large-scale Production ADAS (SAE Level 2 / 2+)

The defining property of Level 2 is that the system performs continuous lateral and longitudinal control, yet the human remains the legal driver and the immediate fallback; consequently, every Level 2 product is fundamentally an exercise in bounding the ODD and enforcing driver attention rather than in achieving

autonomy, and the deployed systems differ mainly in how they strike that balance. One design philosophy gates hands-free operation to pre-validated, HD-mapped limited-access highways and polices it with in-cabin driver monitoring: **GM Super Cruise**[17], **Ford/Lincoln BlueCruise**[18], and **BMW Driving Assistant Professional**[19] all keep the vehicle centered, regulate following distance, and offer assisted lane changes, but escalate alerts and disengage the moment attention drifts or the car leaves a supported segment; their reliability is a direct product of this deliberately narrow envelope. A second philosophy trades map dependence for scale: **Tesla Autopilot**[20] and **FSD (Supervised)**[21] rely on a camera-first vision stack refined through fleet-wide over-the-air training, which extends assistance to a far broader and less-bounded set of scenarios but, lacking comparable geofencing, still demands constant supervision and degrades on poor markings, construction zones, and adverse weather. At the conservative end, **Toyota Safety Sense** declines hands-free operation altogether, functioning as an always-on crash-avoidance layer (automatic emergency braking, lane-keeping, and adaptive cruise) that intervenes only to prevent or mitigate collisions. Despite these differing emphases, the binding limitation is identical across all of them: a bounded ODD coupled with a mandatory human fallback, and it is precisely that fallback which the next tier begins to remove.

9.2.6.2 Certified "Eyes-off" Conditional Automation (SAE Level 3)

Level 3 is the first tier at which, inside a defined envelope, the driver may legally divert attention and the manufacturer assumes liability for the driving task. This shift in responsibility, rather than any new perception capability, is what dictates the engineering. **Mercedes-Benz Drive Pilot**[22] and **Honda Traffic Jam Pilot**[23] purchase the "eyes-off" guarantee by pairing complementary sensing (cameras, radar, and LiDAR with high-precision, HD-map-anchored localization) with redundant braking, steering, and power paths that retain control through a single-component failure. Both, tellingly, keep the envelope deliberately small (low-speed or traffic-jam flow on mapped, divided highways), and both center on the same safety mechanism: as conditions approach the ODD boundary, the system issues a timed takeover request and, absent a driver response, executes a minimal-risk maneuver such as a controlled stop with hazard signals. The narrowness of the Level 3 envelope is therefore not a marketing concession but the direct cost of guaranteeing safe behavior once the human is no longer continuously in the loop: a cost the next tier pays in full by removing the driver entirely.

9.2.6.3 Driverless Robotaxis and Autonomous Ride-hailing (SAE Level 4)

Level 4 removes the human from the loop entirely within a geofence: there is no driver to hand back to, so the vehicle must resolve every in-domain situation itself or bring itself to a safe state, with remote operators supplying only high-level guidance rather than moment-to-moment control. To meet this far stricter bar, **Waymo One**[24] and **Baidu Apollo Go**[25] converge on essentially the same recipe: a dense, overlapping multi-sensor suite (lidar, radar, and camera vision) for redundant coverage, HD-map-anchored localization, and conservative prediction-and-planning that favors lawful, cautious maneuvers through dense urban traffic. Each makes the problem tractable not by claiming universal competence but by tightly restricting where and when it operates and validating performance city by city before extending the geofence. Across the three tiers, every step up the ladder buys wider autonomy by narrowing the operating domain and

[17] https://www.gm.com/innovation/autonomous-driving
[18] https://www.lincoln.com/technology/bluecruise/
[19] https://www.bmwgroup.com/en/news/general/2023/driver-assistance-systems.html
[20] https://www.tesla.com/en_gb/support/autopilot
[21] https://www.tesla.com/en_gb/fsd
[22] https://www.mercedes-benz.co.uk/passengercars/technology/autonomous-driving.html
[23] https://global.honda/en/tech/Traffic_Jam_Pilot_TJP/
[24] https://waymo.com/
[25] https://www.apollogo.com/

adding redundancy, until at Level 4 the system finally internalizes the fallback that Levels 2 and 3 delegated to the driver.

9.2.7 Challenges

Every tier of the deployment ladder works because it limits where it operates, and general, trustworthy autonomy requires lifting those limits. Four threads, each cutting across perception, localization, and decision, stand in the way: justifying life-critical decisions (*safety and explainability*), dependable behavior across the long tail of road, weather, and traffic conditions (*resilience and robustness*), adaptation to genuinely novel situations from little prior data (*unfamiliar road scenarios*), and continual learning on deployed vehicles without eroding mastered behavior (*enhanced autonomy in complex scenarios*).

Safety and Explainability. Because driving decisions are life-critical, the inability to explain why a model selected a given maneuver is a first-order obstacle to certification and public trust. Post-hoc attribution methods such as layer-wise relevance propagation [2438] and Shapley-value attribution expose which inputs drove a decision, supporting both engineering debugging and regulatory audit, while causal-inference models go further by separating mere correlation from the actual influence of factors such as road state and a maneuver's effect on surrounding agents. The unresolved difficulty is that these explanations remain largely retrospective and local to individual decisions; converting them into guarantees that provably hold across the full operating domain is still an open problem.

Resilience and Robustness. Robustness to the long tail of driving conditions is commonly pursued through curriculum learning, where a policy is trained on a graded sequence from simple navigation to dense mixed-traffic interaction, which improves sample efficiency and training stability relative to confronting hard cases directly. Its effectiveness, however, hinges on a scenario curriculum that faithfully spans the deployment distribution and on objectives that keep safety and comfort constraints satisfied at every difficulty level. The practical risk is that, absent a benchmark which deliberately surfaces rare but critical events, strong curriculum performance on common scenes can mask failures precisely on the cases that determine real-world safety.

Unfamiliar Road Scenarios. A distinct difficulty is generalization to genuinely novel situations (an unseen sign, an irregular work zone, an atypical vehicle) where data-hungry models degrade but human drivers adapt from a single exposure. Approaches that target this gap include probabilistic program induction in a Bayesian framework, which represents a new concept as a compact program inferred from few examples, and the injection of compositional, causal, and meta-learning priors so that, for instance, a system can infer that a fallen obstacle will trigger sudden lane changes in adjacent traffic. The standing obstacle is validating such few-shot behavior to a safety standard, since the very rarity of these events makes their failure modes hard to measure and certify.

Enhanced Autonomy in Complex Scenarios. Sustained operation in changing environments motivates lifelong learning [2414], in which a deployed system keeps acquiring new knowledge (a new traffic pattern or road layout) without retraining from scratch and without overwriting prior competence. The central technical barrier is catastrophic forgetting, which is addressed by memory-augmented architectures and continual-learning regularizers that protect previously learned parameters, together with transfer mechanisms that reuse knowledge across tasks and cities. The harder, still-open requirement is to perform such updates on-vehicle under real-time and verification constraints, so that an update improving one scenario cannot silently regress a safety-critical behavior elsewhere.

9.3 Wearable AI System

9.3.1 Introduction

Wearable devices have become a central sensing modality for embodied and spatial intelligence, delivering continuous, fine-grained measurements of human motion, physiology, and physical interaction. Because they are worn on or integrated into the body, they access proprioceptive and kinesthetic cues, joint rotations, muscle activations, contact forces, and locomotion patterns, which exteroceptive sensors such as RGB-D cameras or LiDAR cannot capture directly. We use the term broadly, covering inertial measurement units (IMUs), surface electromyography (EMG) sensors, smart textiles, haptic interfaces, and wearable cameras: body-mounted technologies that together model human dynamics and embodied interaction in real-world settings. It is this egocentric, body-centric vantage point that makes wearables valuable for spatial intelligence, letting a system reason about how a person moves through, reads, and acts upon their surroundings, and it has driven rapid adoption across health monitoring, sports analytics, rehabilitation [2440], industrial safety, and human–robot collaboration.

The signals wearables provide are especially valuable because they are anticipatory. Joint kinematics and muscle activity reveal what a person is doing and, crucially, what they are about to do next, which is what underpins intent recognition, action forecasting, and safe coordination in shared spaces. Wearable IMU arrays, for example, recover whole-body skeletal motion in cluttered or occluded settings where optical capture breaks down [2441, 2442], and EMG surfaces pre-activation patterns that precede any visible movement [2443].

These devices are often bracketed with head-mounted XR hardware, yet their role in spatial intelligence differs in kind. XR headsets integrate sensing and display for immersive visualization and mixed-reality interaction, leaning on outward-facing cameras for SLAM and environment mapping. The wearables treated here look inward instead, supplying the biomechanical and physiological state, high-frequency, continuous, and on-body, that an XR headset does not observe.

The spatial-intelligence capabilities that wearables ultimately serve, pose estimation, action recognition, localization, and trajectory prediction, are formulated in Part III (spatial perception in Chapter 6, spatial reasoning in Chapter 7). This section does not re-derive them; it concentrates on the body-centric sensing layer, the devices that acquire embodied signals, the systems assembled around them, and the tasks, methods, and failure modes that surface once spatial intelligence must be grounded in continuous on-body measurement, spanning long-horizon activity and gait analysis, intention prediction for human–robot collaboration, and real-time safety in industrial and logistics settings.

9.3.2 Wearable Devices

Body-centric sensing spans a diverse device ecosystem, summarized in Table 9.3.

Motion and Body-State Sensing. Inertial measurement units (accelerometers, gyroscopes, and often magnetometers) are the most widely deployed body-worn sensors. Consumer watches and bands such as the Apple Watch[26], Samsung Galaxy Watch[27], Fitbit[28], and Xiaomi Band stream inertial data for activity and gait monitoring, while research and industrial systems place IMUs at specific anatomical sites and fuse them into full-body kinematics [2442, 2444]; because they need no line of sight, they keep working under occlusion and poor lighting. Full-body motion-capture suits such as Xsens[29], Rokoko Smartsuit Pro[30], and Noitom[31]

[26] https://www.apple.com/watch/
[27] https://www.samsung.com/global/galaxy/galaxy-watch/
[28] https://www.fitbit.com/
[29] https://www.movella.com/products/xsens
[30] https://www.rokoko.com/
[31] https://www.noitom.com/

combine many IMUs, sometimes with pressure or stretch sensors, into accurate skeletal pose outside the studio, which has made them standard in biomechanics, sports, robotics, and character animation [2445]. Smart clothing embeds conductive fabrics, stretch sensors, or distributed pressure sensors in garments for unobtrusive, long-duration monitoring of posture, joint angles, and whole-body contact [2446], and physiological sensors (heart rate, galvanic skin response, respiration), foot-mounted inertial or pressure sensors, and ear-worn IMUs add workload, ground-contact timing, and head orientation [2440].

Neuromuscular, Egocentric, and Interactive Wearables. Electromyography (EMG) armbands and high-density arrays sense the muscle activations that precede visible movement, giving early cues for gesture recognition and intent estimation [2443], and combining EMG with inertial data makes them more robust to arm posture and fatigue [2447]. Wearable cameras add the exteroceptive, first-person view: egocentric RGB, depth, or fisheye streams supply scene geometry and activity context that can be fused with on-body signals [681, 2448]. Two device families also act on the body. Haptic gloves such as HaptX[32] and SenseGlove[33] and haptic vests such as bHaptics[34] return tactile, vibration, or force feedback during manipulation, teleoperation, and training [2449], and exoskeletons combine IMUs, torque and force sensors, and joint encoders with actuation, estimating user intent in real time to assist or rehabilitate movement [2450], which makes them a direct test of embodied spatial intelligence.

Table 9.3: Wearable body-centric sensing devices, by sensed signal and recovered spatial or biomechanical quantity. Haptic and exoskeletal wearables are partly *actuators* that return force and tactile signals to the wearer.

Device family	Primary signal	Spatial / biomechanical quantity recovered
IMU wearables	accel., gyro., (magnet.)	orientation, acceleration, gait, sparse-to-full body kinematics
Motion-capture suits	multi-IMU (+ pressure/stretch)	full-body skeletal pose in unconstrained settings
EMG / gesture bands	neuromuscular activation	muscle intent, hand/arm gestures, pre-movement cues
Smart clothing / textile	distributed pressure, strain	posture, joint angles, whole-body contact patterns
Egocentric cameras	first-person RGB / depth	scene geometry, activity, environmental context
Haptic / force-feedback	tactile, vibration, force (I/O)	contact rendering, spatial alerts (feedback channel)
Exoskeletons	IMU, torque, force, joint enc.	joint torque, user effort and movement intent
Physiological / other	HR, GSR, respiration; foot press.	workload, exertion, ground contact, head orientation

9.3.3 System Overview

Taken together, the devices above form the front end of a single body-centric perception-to-action loop that defines the wearable AI system as a whole. Raw on-body signals are first *sensed* (inertial, neuromuscular, tactile, and egocentric-visual streams), then *modeled* into geometric and biomechanical state (joint kinematics, body pose, contact forces), *interpreted and predicted* into higher-level behavioral and intentional descriptions (activities, gestures, short-horizon motion forecasts), and finally fed back to the wearer or to a collaborating agent through *action* channels (haptic cues, exoskeleton assistance, safety alerts, or shared-control commands). The loop then closes: the consequences of feedback alter the wearer's motion, which the sensors observe anew.

Figure 9.11 schematizes this loop and situates the three pivotal tasks that organize the remainder of this section as complementary stages of it. **Pivotal Task-1 (Embodied Motion Sensing and Pose Reconstruction)** occupies the sense-to-model stage, turning body-centric signals into stable geometric descriptions of motion. **Pivotal Task-2 (Behavioral Understanding and Intention Modeling)** occupies the interpret-and-predict stage, mapping reconstructed motion and neuromuscular cues to recognized activities and anticipated

[32] https://haptx.com/
[33] https://www.senseglove.com/
[34] https://www.bhaptics.com/

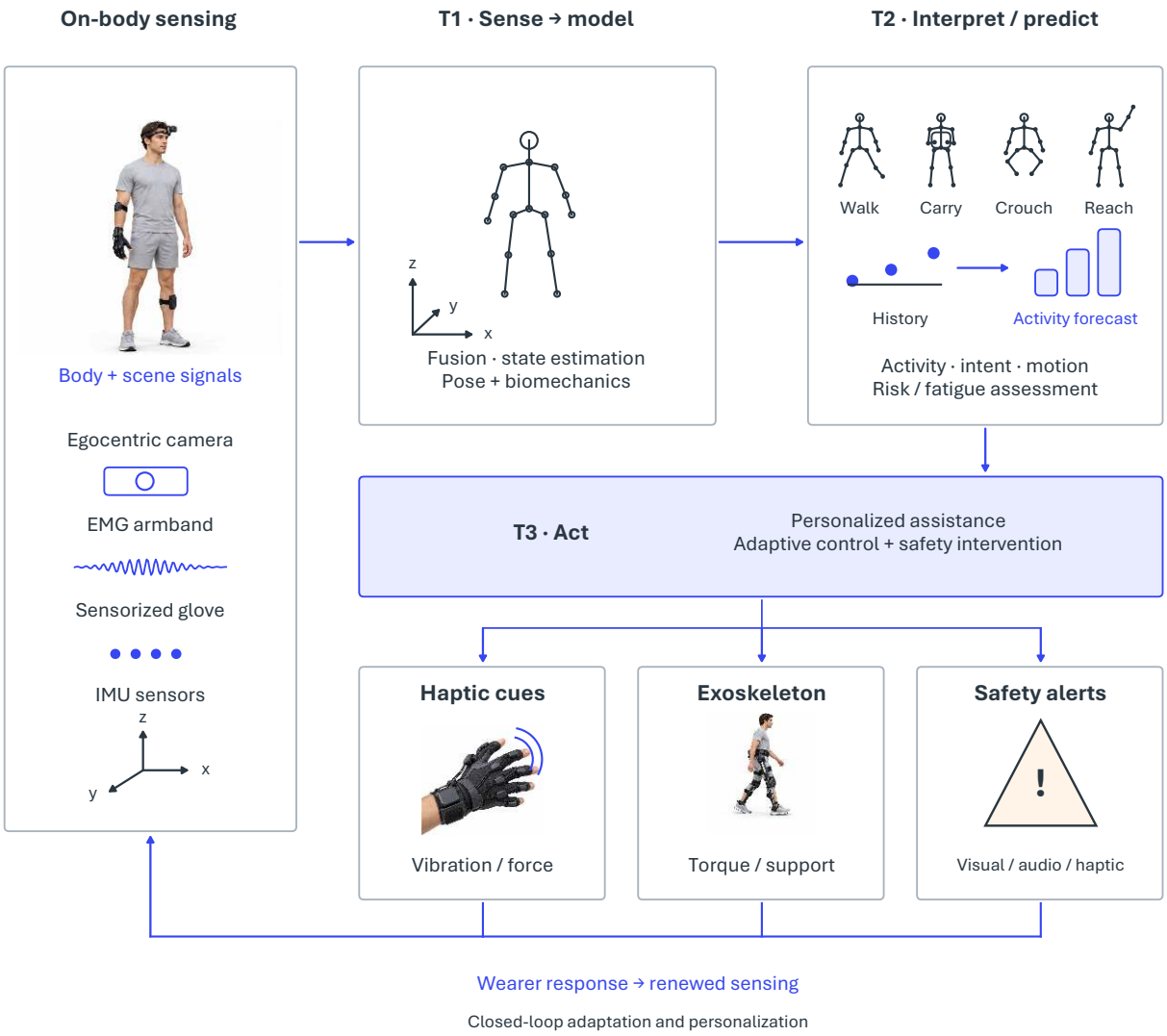


Figure 9.11: The wearable AI system as a body-centric loop: on-body signals are modeled into kinematic and biomechanical state (T1), interpreted into activity and intent (T2), and acted on through haptic, exoskeleton, and safety channels (T3), used separately or together. Device and prediction illustrations are schematic.

actions. **Pivotal Task-3 (Safe Collaboration and Embodied Interaction)** closes the loop at the action stage, converting interpreted human state into coordinated, safe, and assistive responses in shared physical environments. Because all three stages must run under the power, compute, and reliability constraints of body-worn hardware, lightweight on-device inference is a cross-cutting concern that recurs throughout.

9.3.4 Pivotal Task-1: Embodied Motion Sensing and Pose Reconstruction

The first stage of the wearable loop turns raw body-centric signals into a geometric description of how the wearer moves. It is a bottom-up reconstruction problem: per-segment inertial and biomechanical measurements must yield joint angles, segment velocities, and ultimately a full-body pose that remains stable over long horizons and in settings where exteroceptive capture fails. This task subsumes full-body skeletal pose estimation, fine-grained limb tracking, real-world locomotion and gait analysis, and body-centric localization in GPS-denied spaces; the underlying pose-estimation formulations are developed in Chapter 6,

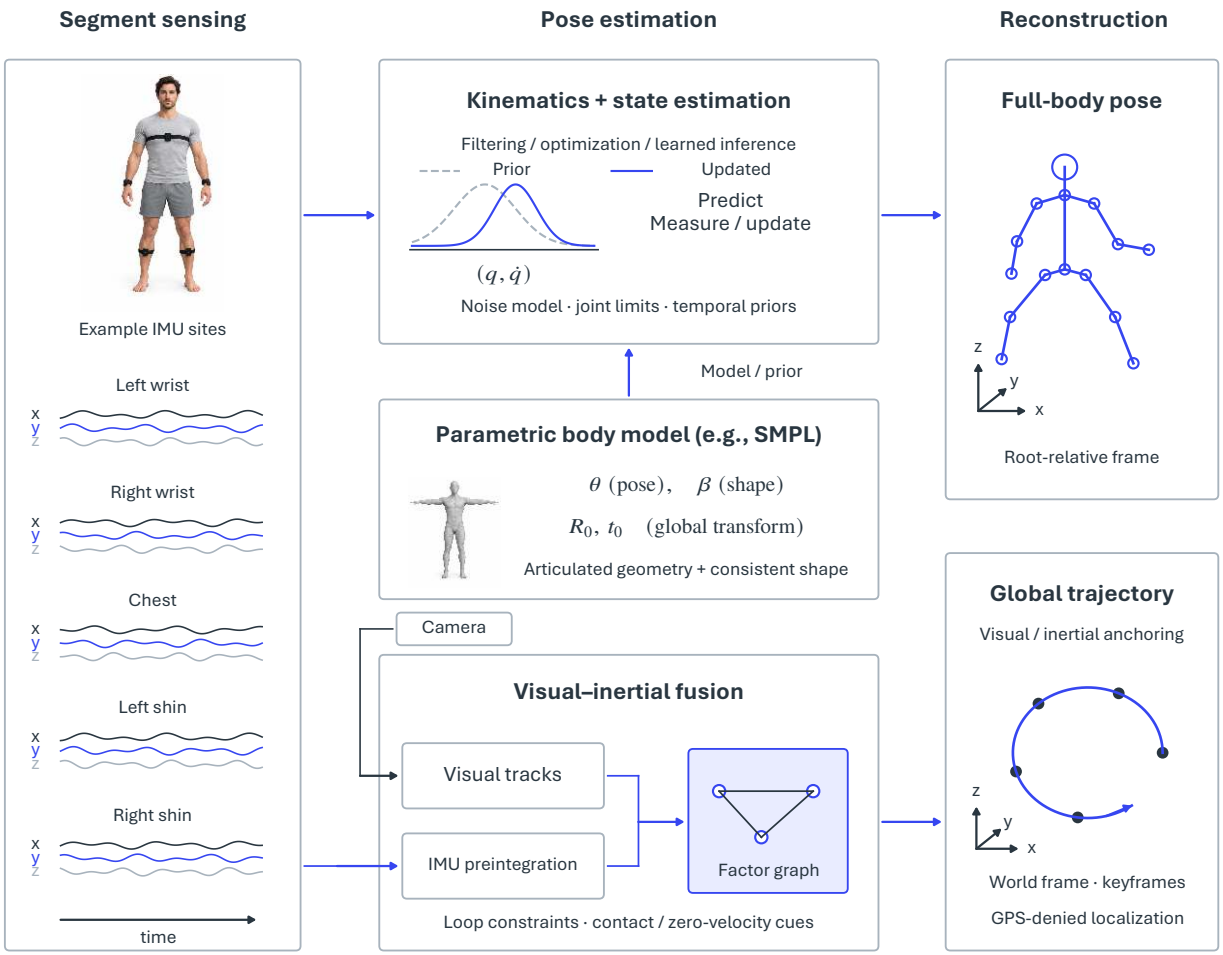


Figure 9.12: Motion sensing and pose reconstruction from body-worn IMUs: estimators supported by a parametric body model recover root-relative full-body pose, while visual–inertial fusion anchors a world-frame trajectory. Contact or zero-velocity updates apply only when their assumptions hold, and metric scale requires sufficient observability; signals and trajectories are illustrative.

so the emphasis here is on reconstruction from on-body sensing. Figure 9.12 illustrates this reconstruction stack from segment-level signals to full-body pose.

From signals to kinematics. Reconstruction begins with kinematic and dynamic modeling, which provides the physical grounding for interpreting inertial and pressure-based signals. Forward and inverse kinematics infer joint angles and segment orientations from inertial measurements, compensating for drift and sensor noise through optimization-based filtering and biomechanical constraints. Classical recursive estimators such as extended and unscented Kalman filters incorporate kinematic priors and temporal smoothness to improve stability in real-world settings [164, 2451], while articulated-body dynamics and parametric body models such as SMPL [338] enforce anatomical plausibility for complex motions such as rapid locomotion or limb coordination. Multi-IMU fusion against musculoskeletal models extends these ideas to full-body pose estimation in challenging environments where optical motion-capture systems fail [2441, 2444, 2452], and

recent systems fold in physics-based dynamics and inter-sensor ultra-wideband ranging to tighten accuracy and suppress jitter [2453].

Learning-based reconstruction and sensor fusion. Where analytic models struggle with sparse or noisy sensing, data-driven approaches learn the mapping from raw inertial streams to pose directly [2441, 2444], and sensor fusion integrates heterogeneous modalities for drift-compensated estimates. IMU-and-vision fusion resolves ambiguities in inertial tracking by leveraging environmental context from egocentric cameras [681, 2448], while drift-compensation techniques, including zero-velocity updates and constraint-based corrections, stabilize long-horizon wear. For body-centric localization, sparse-sensor systems couple inertial pose estimation with simultaneous localization and mapping, so that the wearer's global trajectory is recovered together with body pose rather than left to drift freely [2454]; visual-inertial odometry that fuses the same inertial signals with first-person imagery offers a complementary route through GPS-denied indoor spaces, assessed on dedicated benchmarks [935].

Benchmarks and evaluation. The kinematic, learning-based, and fusion methods above are measured against the same yardsticks: large-scale motion-capture corpora such as AMASS [2209], which provides a canonical surface-and-skeleton reference for full-body pose, alongside dedicated inertial-pose datasets for sparse-IMU reconstruction [2445, 2441], while body-centric localization is assessed separately on visual-inertial odometry benchmarks [935]. Reconstruction quality is reported as mean per-joint position error (MPJPE) and joint-angle error, and localization as absolute trajectory error and accumulated drift (Table 9.4). Read against the needs of real deployment, these numbers reveal a clear gap. On curated benchmarks, sparse-IMU systems now recover *root-relative* full-body pose to within roughly five centimeters of optical ground truth [2452], but global translation still drifts over long sequences unless it is anchored by SLAM or ranging [2454], and even the root-relative accuracy falls off in the wild, where fast or unseen motions and loose sensor placement push errors past the tolerance that downstream control or clinical gait analysis can absorb. The pose the next stage consumes is thus dependable on average yet brittle at exactly the fast, atypical movements that matter most.

9.3.5 Pivotal Task-2: Behavioral Understanding and Intention Modeling

Once motion has been reconstructed, the second stage of the loop interprets it: recognizing what the wearer is doing and anticipating what they will do next. Continuous neuromuscular and inertial cues feed activity recognition and segmentation and then short-horizon intention prediction, even in daily-life settings where visual systems face severe occlusion or field-of-view limits. Because the recognition and reasoning formulations themselves are treated in Chapter 7, the focus here is on how high-frequency wearable streams are modeled to support behavioral inference. Figure 9.13 sketches the path from multimodal time-series and body-graph representations to recognized activity and predicted intent.

Sequence models for wearable time-series. The availability of large-scale wearable recordings has driven deep models tailored to high-frequency temporal streams from IMU, EMG, and multimodal sensors. Recurrent networks, temporal convolutional networks, and Transformers are widely applied to gait classification, gesture recognition, and activity segmentation [2455]: temporal convolutions capture multi-scale dependencies for long-horizon pattern detection [2456], while attention-based architectures learn complex correlations across sensor channels [141]. Because labeled wearable data are scarce and expensive, self-supervised pretraining on unlabeled motion recordings has become the dominant route to transferable representations [2457]: masked-reconstruction encoders such as LIMU-BERT learn compact IMU features from raw streams [2458], and multimodal contrastive objectives such as IMU2CLIP align inertial signals with video and language, pointing toward wearable foundation models that generalize across tasks and users [2459].

Graph-based body representations. Because human anatomy is structured as a kinematic tree, graph neural networks offer a natural representation for wearable-derived motion [599], encoding relational dependencies between joints and limbs directly in graph space. Such formulations fuse IMU placements

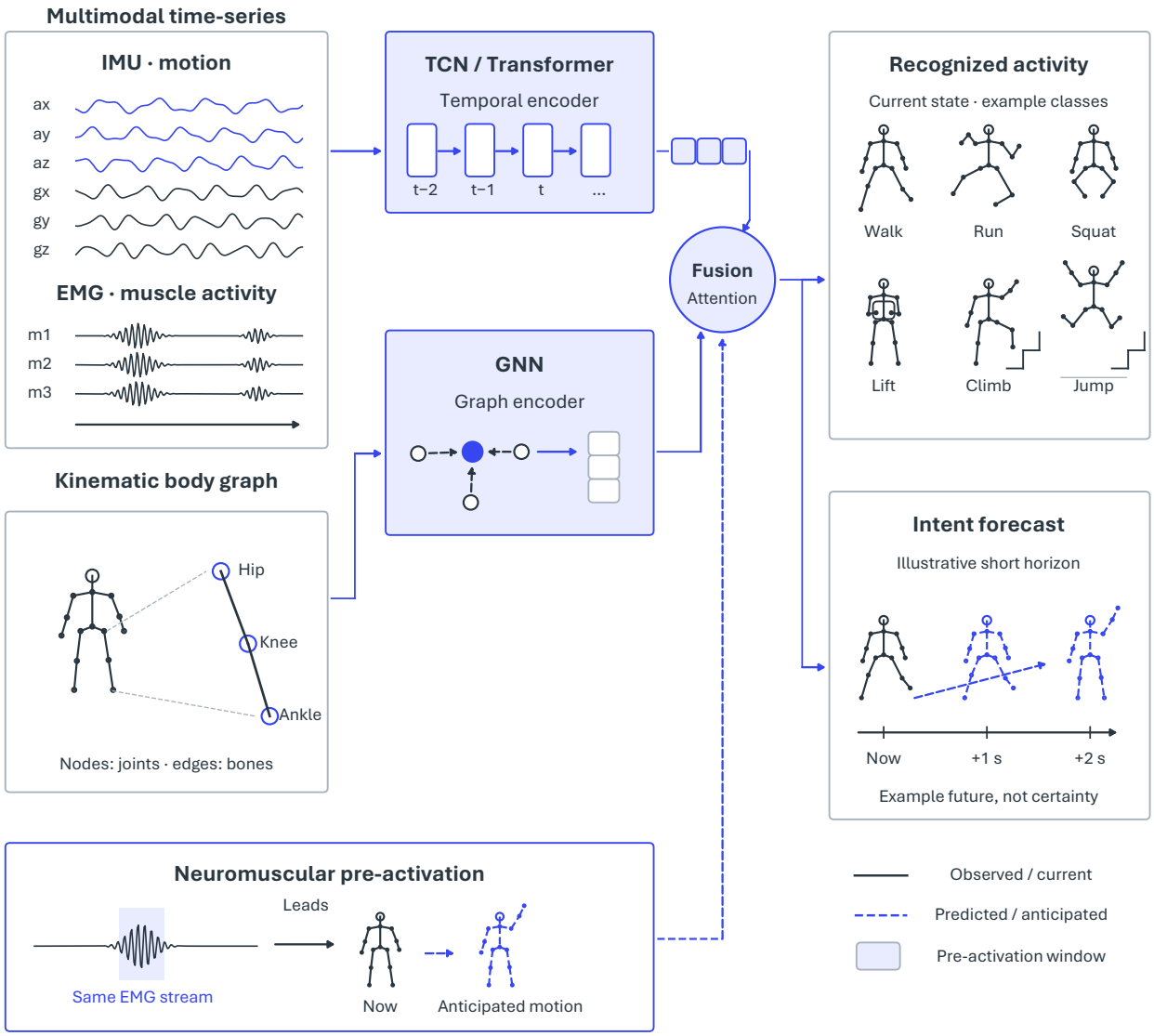


Figure 9.13: Behavioral understanding and intention modeling: IMU and EMG sequences and a kinematic body graph pass through temporal and graph encoders, fused by attention to recognize current activity and forecast short-horizon intent. The EMG pre-activation inset draws anticipatory cues from the observed stream, not from future measurements.

across body segments, propagate temporal features along skeletal edges, and remain robust to missing or corrupted sensor channels; combining graph convolution with temporal attention jointly models spatial structure and temporal evolution for activity recognition and biomechanical analysis [2460].

Neuromuscular cues and multimodal fusion for intent. Beyond recognizing observed motion, EMG-based wearables detect subtle neuromuscular activations that precede visible movement [2443], providing predictive cues for early intention estimation and responsive human-robot collaboration. Fusing EMG with inertial data sharpens fine-grained gesture and anticipatory-command decoding [2447], while fusing IMU with egocentric video grounds activity recognition in environmental context [913, 681]. A recent milestone shows how far this can scale: a surface-EMG wristband trained across more than a thousand users decodes gestures, clicks, and handwriting from neuromuscular signals alone and, critically, generalizes to new wearers without per-user calibration [2461], addressing the inter-user variability that has long limited EMG interfaces. Physiological

and pressure cues additionally support inference of fatigue, balance, and abnormal gait over extended periods.

Benchmarks and evaluation. Taking the reconstructed motion of the previous stage as input, the sequence, graph, and fusion models above are evaluated on wearable human-activity-recognition corpora [2462, 2463], EMG gesture datasets [2443], and egocentric activity datasets such as Ego4D [681] and EPIC-KITCHENS [2464], with recognition reported as accuracy and macro-F1 and intention prediction as anticipation accuracy and prediction lead time (Table 9.4). Here the gap to maturity is one of generalization rather than peak score. Balanced locomotion corpora such as PAMAP2 are close to saturated, yet the same architectures score markedly lower on cluttered, null-dominated sets such as Opportunity and lose accuracy again under cross-user and cross-environment evaluation; reliable intention forecasting, meanwhile, still extends only to short, well-rehearsed horizons. This falls well short of the open-world, anticipatory understanding that the safe-collaboration stage below depends on, where the system must read unfamiliar users in unfamiliar settings.

9.3.6 Pivotal Task-3: Safe Collaboration and Embodied Interaction

The third stage closes the loop, converting interpreted human state into coordinated, safe, and assistive responses in shared physical environments. Contact- and force-level sensing of human–environment interaction supports real-time inference of effort and hazard, which in turn drives feedback for collaboration and shared autonomy. The collaboration and shared-control reasoning that this task builds on is developed in Chapter 7 and, for the manipulation setting specifically, in Section 9.1; here the emphasis is on the wearable feedback and safety layer. Figure 9.14 depicts this closing stage, from contact sensing to coordinated safety response.

Sensing interaction and effort. Haptic and tactile wearables capture contact forces, surface interactions, and manipulative behaviors, supplying the signals needed to model human–environment interaction and object handling [2465, 2446]. Exoskeletons equipped with torque and force sensors use these signals to infer user effort and detect hazardous conditions, adjusting assistive strategies accordingly [2450]; probabilistic fusion that integrates biomechanical priors and uncertainty estimates is central to robustness in such safety-critical settings.

Coordination, safety, and feedback. In industrial and shared-autonomy scenarios, on-body inertial and proximity sensors contribute to safety monitoring by detecting collision risks, unsafe postures, and sudden motion anomalies, while haptic channels return spatial alerts and force feedback that close the sensorimotor loop [2449]. Combined with the intention forecasts of Pivotal Task-2, these signals form the basis for predictive safety mechanisms and shared-control frameworks that anticipate human motion rather than merely reacting to it.

On-device constraints. Because feedback must be timely, the action stage is where the wearable system's power, memory, and compute limits bite hardest. Model compression (pruning, quantization, and distillation) [2466] and edge-optimized architectures [2467] enable real-time inference on wearable microcontrollers or smartphone-class processors, and on-device adaptation mitigates inter-user variability without relying on remote servers, keeping safety-critical computation local and low-latency.

Benchmarks and evaluation. Closing the loop, the interaction-sensing, safety, and on-device methods above are assessed on human-object-interaction datasets such as HOI4D [2465] and, for assistive control, on exoskeleton and teleoperation testbeds, with metrics spanning contact- and force-estimation error, collision-avoidance and safety-violation rates, and intention-anticipation lead time (Table 9.4). Evaluation here is the least mature of the three stages: it remains fragmented across hardware-specific testbeds with few shared protocols, so reported gains are hard to compare and seldom stress real safety guarantees. Current systems perform convincingly in instrumented demonstrations, but closed-loop, certified safety under the full variability of human partners, the bar that genuine shared autonomy must clear, is still an open target



Figure 9.14: Safe human–robot collaboration with wearables: fused contact, force, and proximity signals, together with intent forecasts (T2) and robot motion context, drive motion prediction and risk evaluation, which select glove, exoskeleton, or cobot responses. Prediction envelopes and risk thresholds are schematic, not measured safety guarantees.

rather than a measured result. The robustness, generalization, and safety gaps that recur at each stage of the loop are what the challenges below take up directly.

9.3.7 Established Tools and Systems

Commercial wearables have converged on the head and the wrist, and what separates current products is less their sensors than where perception runs and whether anything is shown back to the wearer. Full augmented-

Table 9.4: Representative benchmarks for wearable spatial-intelligence tasks, grouped by pivotal task.

Pivotal Task	Benchmark / Dataset	Representative method / use	Headline metric
T1: Motion & Pose	AMASS [2209]	Full-body pose reference (SMPL surface)	MPJPE
T1: Motion & Pose	TotalCapture [2445]	Video+inertial full-body pose	MPJPE
T1: Motion & Pose	DIP-IMU [2441]	Sparse-IMU pose reconstruction (DIP, TransPose, PIP)	Joint-angle / position error
T1: Localization	TUM-VI [935]	Visual-inertial odometry (body-centric)	ATE / drift
T2: Activity	PAMAP2 [2462]	Wearable activity recognition (deep recurrent / TCN)	Accuracy / macro-F1
T2: Activity	Opportunity [2463]	On-body activity recognition (DeepConvLSTM)	Weighted / macro-F1
T2: Activity	Ego4D [681]	Egocentric activity recognition	Accuracy / F1
T2: Gesture	Ninapro [2443]	EMG gesture / movement classification	Accuracy
T2: Gesture	emg2qwerty [2468]	Large-scale surface-EMG typing	Character error rate
T3: Interaction	HOI4D [2465]	Human-object interaction sensing	Contact / interaction acc.

reality headsets are covered with the XR platforms in Section 10.1; here the focus is on everyday-wearable glasses, neural input bands, and the research platforms and chips behind them.

AI glasses without a display. By volume, the dominant wearable AI device is a pair of ordinary-looking glasses carrying a camera, microphones, open-ear speakers, and an IMU, with a multimodal assistant running on the paired phone or in the cloud. Meta's line with EssilorLuxottica[35] defines the category: the second-generation Ray-Ban Meta (2025) records 3K video from a 12-megapixel camera and runs up to eight hours per charge, and the Oakley Meta Vanguard adapts the design to sport with a 122-degree camera. Display-less models made up 96% of AI-glasses shipments in the first half of 2026, almost all of them Meta's[36], while Xiaomi's AI glasses (2025) and Alibaba's Qwen-based Quark AI glasses[37] (November 2025) lead a fast-growing Chinese segment. Google and Samsung previewed Gemini-based Android XR glasses[38] with Gentle Monster and Warby Parker in May 2026, with the first display-less models due in the fall. Spatially, these devices answer questions about what the wearer is looking at one camera frame at a time; most keep no persistent map of the surroundings, so the continuous body and scene tracking of Pivotal Tasks 1 and 2 is still largely absent from shipping products.

Display glasses and neural input. Adding a display makes the glasses an interface, and input then becomes the hard problem. Meta Ray-Ban Display[39] (September 2025) puts a small monocular display, with a 20-degree field of view, in the right lens and ships with the Meta Neural Band, a surface-EMG wristband that decodes finger gestures and handwriting; it is the commercial form of the generic neuromotor interface described in Pivotal Task-2 [2461]. In waveguide display glasses, Rokid[40] led shipments in the first half of 2026, ahead of Meta and Even Realities[41], whose glasses favor a minimal text display over cameras.

Research platforms and on-glass compute. Much of the egocentric data behind the methods of this section was recorded on research glasses rather than consumer products. Meta's Project Aria [2469][42] provides glasses with RGB, tracking, and eye-tracking cameras, IMUs, and spatial microphones, together with machine-perception services for trajectories, eye gaze, and hand tracking; its second generation (2025) adds a heart-rate sensor and a contact microphone in the nosepad, alongside on-device perception. The platform underlies Aria Digital Twin [2448]; Ego-Exo4D [2470], which pairs egocentric Aria recordings of

[35] https://www.meta.com/ai-glasses/
[36] Counterpoint Research, global AI glasses shipments, H1 2026.
[37] https://www.quark.cn/
[38] https://www.android.com/xr/
[39] https://www.meta.com/ai-glasses/meta-ray-ban-display/
[40] https://global.rokid.com/
[41] https://www.evenrealities.com/
[42] https://www.projectaria.com/

skilled activities with synchronized exocentric cameras over 1,422 hours; and Nymeria [2471], which adds full-body ground-truth motion from a motion-capture suit to 300 hours of everyday activity, the kind of supervision the pose reconstruction of Pivotal Task-1 needs. Ego4D [681] gathered its 3,670 hours from several head-mounted cameras, and remains the main benchmark for egocentric activity understanding. On the compute side, Qualcomm's Snapdragon AR1+ Gen 1[43] (2025) runs a one-billion-parameter language model on the glasses without a phone, a first step toward the on-device inference that the latency and power constraints of Pivotal Task-3 call for.

9.3.8 Challenges

Wearable sensing is coupled to the dynamics and variability of the human body in a way that external sensing is not, and that coupling is the source of its hardest problems. The four below track the loop of the System Overview: signal quality at the sense stage, robustness under partial observation, fusion across heterogeneous modalities, and generalization across users.

Signal Instability and Drift. On-body signals are inherently unstable. IMU drift, EMG noise, and calibration shifts arise continuously from motion artifacts, sensor displacement, and soft-tissue deformation, and they strike precisely the quantities that pose estimation, trajectory tracking, and dynamics reconstruction integrate over time, so small per-step errors compound into large ones across a long recording.

Incomplete and Intermittent Sensing. Sensing is rarely continuous in practice. Sweat, motion, and wireless dropouts cause brief losses of contact and stretches of degraded signal, leaving the system to reconstruct reliable motion from partial and sometimes corrupted segment-level data, a demand that clean-benchmark evaluation rarely exercises.

Heterogeneous Multimodal Fusion. The modalities that make wearables powerful are also mismatched. IMU, EMG, pressure, and egocentric-visual streams differ in sampling rate, latency, noise, and coordinate frame, and fusing them under real-world variability, without letting one noisy channel corrupt the whole estimate, remains an open integration problem rather than a solved preprocessing step.

Inter-User Variability. Finally, the body itself is the confound. Anatomy, muscle-activation patterns, and movement style vary enough between people that a model tuned on a small or homogeneous population transfers poorly, and the effect is sharpest exactly where wearables matter most, in motion reconstruction, intent prediction, and gait analysis, where individual biomechanics dominate the sensor signature. Personalization, domain adaptation, and larger, more diverse training corpora are the levers for closing this gap, and how far they succeed will decide whether wearable spatial intelligence moves from convincing demonstrations to dependable deployment.

[43] https://www.qualcomm.com/products/mobile/snapdragon/xr-vr-ar

9.4 Aerial Agents

9.4.1 Introduction

Aerial agents (satellites, high-altitude platform stations (HAPS), and unmanned aerial vehicles (UAVs)) acquire rich, dynamic, large-scale observations from the sky and near space and, increasingly, perceive, decide, and act on them autonomously through onboard processing and adaptive control (Table 9.5). The aerial domain is a demanding testbed for the spatial capabilities surveyed in Part III: it stresses perception across extreme scale and viewpoint changes (Chapter 6) and decision-making under partial observability and tight latency budgets (Chapter 7), across platform tiers that differ in altitude, resolution, endurance, and latency by orders of magnitude.

Satellites supply the global tier. Navigation constellations, namely GPS [2472, 2473] within the broader Global Navigation Satellite System (GNSS) [2474], Russia's GLONASS [2475], the European Union's Galileo [2476], and China's BeiDou [2477, 2478], provide the positioning, navigation, and timing on which vehicles [2479], smartphones [2480], network synchronization [2481], and every lower aerial tier rely. Earth-observation satellites operate continuously over years, which suits long-term, wide-area monitoring that neither ground sensors nor short-endurance UAVs can provide: seasonal vegetation dynamics [2482], wildlife migration [2483], river-basin ecosystem health [2484], deforestation [2485, 2486], desertification [2487, 2488], biodiversity loss [2489], and climate indicators such as land surface temperature, ocean color, and atmospheric composition [2490, 2491], together with ecological conservation [2492, 2493], multispectral and hyperspectral resource discovery [2494, 2495], precision agriculture [2496, 2497] and crop-yield estimation [2498, 2499], and the detection of algal blooms [2500] and pollutant leakage [2501].

Table 9.5: Comparison of satellite, HAPS, and UAV platforms.

Parameter	Satellite	HAPS	UAV
Altitude	500–36,000 km	20–50 km	< 10 km
Spatial Resolution	m to sub-m	dm to cm	cm-level
Temporal Coverage	Hours–days revisit	Persistent (regional)	Real-time (local)
Coverage Area	Global/regional	Regional	Localized
Endurance	Years	Months	Hours
Maneuverability	Minimal	Limited	High
Data Latency	High	Medium	Low
Payload Capacity	High	Moderate	Low
Onboard Processing	Limited	Moderate	High
Atmospheric Impact	Minimal	Low	High
Deployment Speed	Slow	Medium	Fast
Cost Structure	High (low per area)	Medium	Low (high per area)

Differently, UAVs, commonly known as drones, have gained widespread popularity due to their agility, low operational cost, and capability for autonomous operation. Equipped with onboard sensors and processing units, UAVs can perceive their surroundings, plan flight paths, and adapt to dynamic conditions in real time. This unique perspective allows UAVs to capture high-resolution imagery and video from flexible vantage points, enabling frequent revisits and reduced occlusions in complex environments. Compared to satellites, UAVs excel in capturing fine-grained, localized data, and can be rapidly deployed to areas of interest, including hazardous or otherwise inaccessible regions. They now serve aerial photography [2502, 2503], cinematography [2504, 2505], environmental mapping [2506, 2507, 2508], precision agriculture [2509, 2510], geolocalization [2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519], and last-mile delivery [2520, 2521].

Filling the altitude gap between satellite systems and UAVs, emerging platforms such as stratospheric balloons, solar-powered high-altitude platform stations, and small satellite constellations (*e.g.*, CubeSats [2522]) are increasingly being integrated into aerial observation networks. These high-altitude platform stations (HAPS) offer complementary advantages: HAPS [2523] provide persistent regional coverage with higher

spatial resolution than satellites, while orbital systems [2524] enable global, long-term monitoring at varying temporal and spectral scales. Together, they form a multi-tiered aerial ecosystem capable of capturing heterogeneous multimedia data, from visible and thermal imagery to hyperspectral and LiDAR scans, across spatiotemporal scales unattainable by any single platform alone. For instance, solar-powered HAPS such as Airbus Zephyr [2525] or HAWK30 by AeroVironment/HAPSMobile [2526] can operate autonomously in the stratosphere (about 20 km altitude) for weeks, offering persistent observation and communication relay services with sub-meter resolution, which is superior to most Earth observation satellites and at lower cost than maintaining UAV fleets.

A growing trend is the coordination of heterogeneous aerial agents into cooperative networks, where satellites offer global context, HAPS serve as communication relays or edge computing nodes, and UAVs execute localized missions, enabling real-time, closed-loop intelligence. The utility of these systems is further amplified by advances in artificial intelligence, including machine learning, computer vision, big data analytics, and transfer learning [2527, 2528, 2529]. These technologies empower intelligent processing of massive aerial datasets, enabling automated image and video analysis [2530, 2531, 2532, 2533, 2534, 2535], real-time object detection and tracking [2536, 2537], predictive modeling for urban dynamics [2538, 2539], natural language-based control interfaces [2540], and city-scale digital twin simulations [2541]. Data privacy, airspace integration, energy efficiency, and operational robustness remain open, and are taken up in the Challenges below.

9.4.2 System Overview

An aerial agent realizes a sensing-to-action loop that spans the multi-tier ecosystem of Table 9.5: satellites supply global context and positioning, HAPS act as persistent regional relays and edge-computing nodes, and UAVs execute localized, low-latency missions at the point of interest. Across this stack, spatial intelligence is exercised through three complementary stages, which constitute the pivotal tasks of this section. Cross-view perception anchors aerial and ground observations to a common geographic frame, so that what a drone sees can be matched to a map or a street-level query (Pivotal Task-1); remote-sensing detection and environmental understanding turn that imagery into structured scene knowledge about objects, layouts, and human activity (Pivotal Task-2); and autonomous navigation and decision making convert perception into flight plans and control actions in unseen environments (Pivotal Task-3). Figure 9.15 sketches this pipeline, with a feedback path from action back onto sensing that ties the three stages into one closed loop rather than a feed-forward chain.

9.4.3 Pivotal Task-1: Ground-Air Collaborative Perception and Visual Geolocalization

A foundational requirement of any aerial system is to place its observations on the map: the integration of ground-level and aerial views for cross-modal understanding enables visual geolocalization, determining the geographic location of a query image by matching it against a reference database. This task anchors the perception stage of the system and is particularly challenging due to the large modality gap between ground street-view images and overhead (aerial or satellite) imagery, an instance of the cross-view correspondence problem treated more generally in Chapter 6. For instance, the *University-1652* dataset [2506] has become a benchmark for this task, offering ground, drone and satellite image triplets of campus buildings, thus facilitating research on cross-view matching and localization. Such capabilities are critical for urban search-and-rescue, augmented reality, and infrastructure monitoring, where precise alignment between ground observations and aerial maps is essential. Progress has come from bridging the modality gap, encoding spatial structure into a viewpoint-invariant representation, and exploiting multiple views per location (Figure 9.16).

Bridging the modality gap. Most extant research frames geolocalization as a cross-view image retrieval problem, where the core challenge is to learn a viewpoint-invariant representation that effectively bridges the

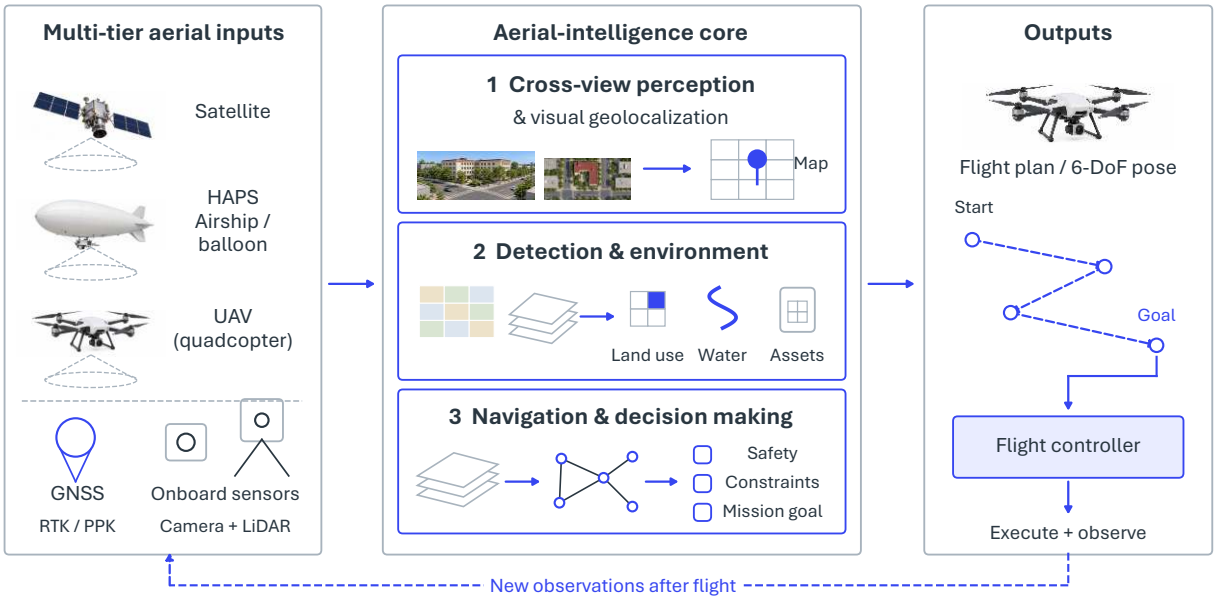


Figure 9.15: Spatial intelligence for aerial agents as a closed loop from multi-tier aerial sensing (satellite, HAPS, UAV, GNSS, onboard sensors), through three complementary core capabilities, to flight plans with 6-DoF pose targets and downstream flight control. The controller executes the plan; subsequent observations close the sensing loop. The core comprises cross-view geolocalization, detection and environmental understanding, and navigation and decision making. Land-use, water, and infrastructure layers are parallel examples. Platform and scene imagery is illustrative.

disparity between images captured from different perspectives. Recent advances in deep learning have led to the widespread application of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) for extracting robust visual features. One prominent line of inquiry employs metric learning to construct a shared latent space for images acquired from diverse platforms. Workman *et al*. [2542] show that a CNN pre-trained on the Places dataset [596] can, by itself, yield highly discriminative features without explicit fine-tuning for the task. The integration of contrastive loss, which minimizes the distance between positive pairs in the feature space, further enhances geolocalization performance [2543, 2544]. Liu *et al*. [2545] introduce the Stochastic Attraction and Repulsion Embedding (SARE) loss, which minimizes the Kullback-Leibler divergence between the learned and actual data distributions, and more recently Deuser *et al*. [2546] show that hard negative sampling strategies also significantly influence the final discriminative power.

Encoding spatial structure. Another significant body of work addresses the problem of spatial misalignment in ground-to-aerial matching, injecting geometric priors into the representation. Vo *et al*. evaluate various network architectures and propose an orientation regression loss to train an orientation-aware network [2547]. Zhai *et al*. [2548] utilize semantic segmentation maps to facilitate semantic alignment, while Hu *et al*. [2549] incorporate a NetVLAD layer to extract more discriminative features. Subsequently, Liu *et al*. [2511] propose a Siamese Network to explicitly integrate spatial cues, such as orientation maps, into the training process, and Shi *et al*. [2513] introduce a spatial-aware layer to improve localization performance. Given that each location has multiple training images from different viewpoints, Zheng *et al*. [2506] train a multi-branch CNN as a base model. Building upon this multi-branch architecture, Wang *et al*. [2507] further partition the feature map into multiple segments to improve matching discriminability, while Lin *et al*. [2508] incorporate keypoint detection into the learning process. More recently, researchers resort to ViT structures: L2LTR [2550] adopts a hybrid CNN-ViT architecture, leveraging CNNs to extract high-level semantic features and ViTs to encode them into global representations; GeoDTR [2551] further enhances geometric reasoning by introducing a ViT-based layout extractor that generates geometry-aware attention maps; and TransGeo [2552] advances the paradigm with a two-branch ViT model and a two-stage coarse-to-fine learning strategy.

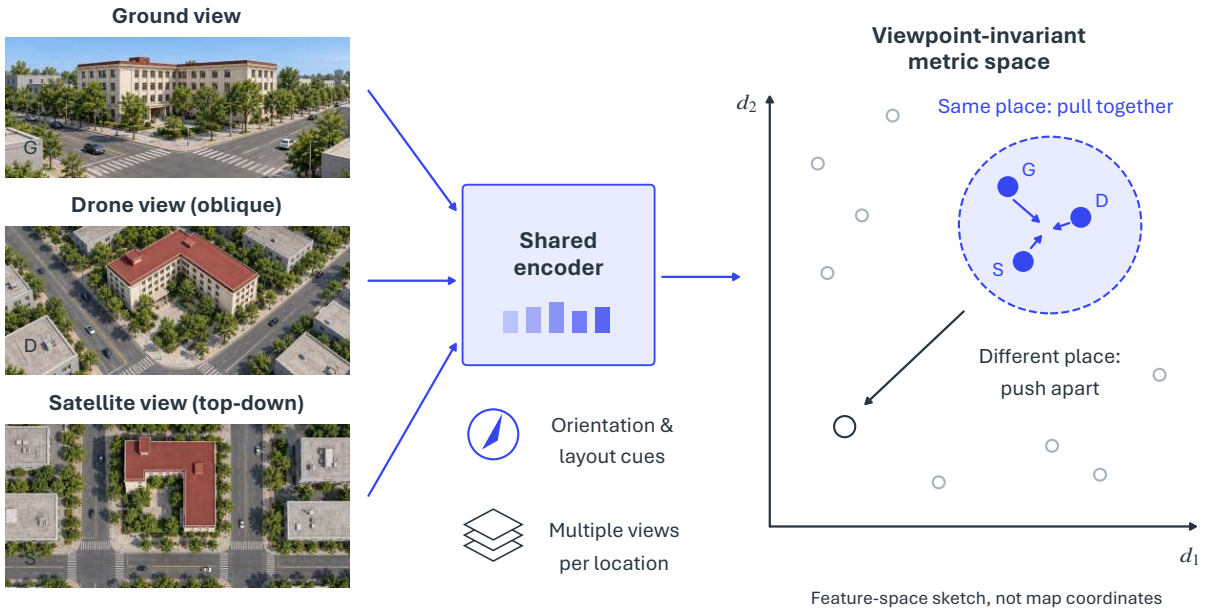


Figure 9.16: Cross-view visual geolocalization as viewpoint-invariant metric learning: the illustrated shared encoder embeds ground, drone, and satellite views so that same-place (positive) pairs cluster and different-place negatives separate. Architecture variants may instead use view-specific branches. Arrows within the feature-space sketch indicate attraction and repulsion during representation learning, not geographic motion. Synthetic views illustrate a common site rather than a calibrated image triplet.

Table 9.6: Comparison of visual geolocalization datasets. R@K: recall at top K; AP: average precision; PR: precision-recall.

Dataset	Univ.-1652 [2506]	SUES-200 [2553]	CVUSA [2548]	CVACT [2511]	Lin *et al.* [2544]	Tian *et al.* [2554]	Vo *et al.* [2547]
#Train imgs.	701×71.6	120 loc.	35.5k×2	35.5k×2	37.5k×2	15.7k×2	900k×2
Platform	Drone, Ground, Sat.	Drone, Sat.	Ground, Sat.	Ground, Sat.	Ground, 45° Aerial	Ground, 45° Aerial	Ground, Sat.
Imgs./loc.	54+16.6+1	drone + 1	1+1	1+1	1+1	1+1	1+1
Target	Building	Diverse	User	User	Building	Building	User
Eval.	R@K, AP	R@K, AP	R@K	R@K	PR, AP	PR, AP	R@K

From image pairs to multi-view benchmarks. The maturation of these methods is closely tied to the datasets that score them (see Table 9.6). The majority of prior geolocalization datasets are based on image pairs, designed for matching images from two distinct platforms such as terrestrial cameras and satellites. One of the earliest contributions [2544] leverages public sources to create 78,000 image pairs spanning a 45-degree bird's-eye perspective and the ground. In a similar vein, Tian *et al.* [2554] compile image pairs for urban localization, positing that buildings are crucial for this task and thus incorporating building detection into their pipeline. The CVUSA [2548] and CVACT [2511] datasets instead address matching panoramic ground-view images with satellite-view images, which is particularly useful for user localization when GPS data is unavailable. A key distinction is one of objective: the latter two focus on localizing the user who captures the photograph, while the former focus on localizing a specific target within the image, so that multiple views of the target are more beneficial for the target-localization datasets (Lin *et al.*, Tian *et al.*) because they encourage the model to learn the target's structure. However, most existing datasets typically provide only two views of a given location. The University-1652 dataset [2506] addresses this by including a greater number of views per target, thereby fostering more robust viewpoint-invariant features; SUES-200 [2553] follows a similar multi-view strategy with drone-captured real-world data. Wang *et al.* [2527] further extend University-1652 to multi-weather evaluation (rain, fog, and other complex conditions), while

Chu *et al*. [2540] introduce linguistic annotations and a language-based query setting for a more intuitive user experience.

Metrics. These benchmarks are scored by retrieval metrics that reflect their gallery structure. Several datasets [2511, 2548, 2547] adopt Recall@K, defined as 1 if a correct match appears within the top K retrieved images; Recall@K is highly sensitive to the rank of the first correct match and is best suited for test sets containing only a single ground-truth match in the gallery. In contrast, datasets such as University-1652 [2506] and SUES-200 [2553] include multiple ground-truth matches from various viewpoints, where Recall@K fails to comprehensively reflect retrieval performance; they therefore also employ Average Precision (AP), as used by Lin *et al*. [2544] and Tian *et al*. [2554]. AP, defined as the area under the Precision-Recall (PR) curve, considers all ground-truth images in the gallery, and recent evaluations usually report both Recall@K and the mean Average Precision (mAP) across all queries.

9.4.4 Pivotal Task-2: Remote Sensing Detection and Environmental Understanding

Once observations are geolocalized, the next stage turns aerial imagery into structured scene knowledge. Aerial platforms are integral to large-scale environmental monitoring and public safety, facilitating fine-grained object detection in complex scenes, a specialization of the detection and segmentation machinery of Chapter 6 to the aerial viewpoint. Prominent datasets such as *DOTA* [2555], *DOTA-v2* [2555, 2556], and *VisDrone* [2557] provide comprehensive annotations for detecting various objects, including vehicles and pedestrians, under a wide range of weather, illumination, and scale conditions. These benchmarks have spurred the development of robust detection models specifically engineered to address challenges inherent to aerial perspectives, such as extreme aspect ratios, dense object clustering, and diminutive object sizes. A significant application is found in UAV-based wildlife monitoring, where systems are deployed for animal census in conservation areas [2558], highlighting the utility of aerial technologies in promoting ecological sustainability. Figure 9.17 relates the aerial-specific difficulties of orientation, scale, and density to the detectors built for them and to the human-centric understanding that aerial video enables.

Beyond object detection, as shown in Table 9.7, with the proliferation of UAVs in smart-city and other human-centric applications, human behavior understanding [2531] has emerged as a critical and formidable research challenge, including person re-identification and action recognition. For instance, the UAV person re-identification (re-ID) task [2559] is designed to address cross-drone identity matching amid substantial viewpoint and illumination variations, thereby motivating the creation of view-invariant feature representations. In conjunction with resources like *VisDrone*, which contains multi-target tracking sequences from mobile UAVs, these datasets facilitate the design of intelligent video analytics systems capable of performing long-term human behavior analysis and anomaly detection in dynamic environments.

Remote sensing object detection is primarily challenged by the prevalence of arbitrarily oriented objects and substantial scale variations [2555, 2556, 2560, 2561, 2562, 2563], and existing research can be broadly organized along two corresponding fronts. **Detection of arbitrarily oriented objects.** A significant line of work focuses on developing specialized oriented object detection (OOD) methods. Early approaches often augment horizontal bounding boxes (HBBs) with an angle parameter, employing techniques such as feature refinement in the detector neck [2564, 2565], extraction of rotated regions of interest (RoIs) [2566, 2567], or designing specific OOD heads [2568, 2569, 2570]. However, these methods frequently suffer from boundary discontinuity and representation ambiguity due to their reliance on angle parameters. To mitigate these issues, subsequent research has explored more flexible *object representations*: some methods describe oriented objects using points [2571, 2572] or gliding vertexes [2573], while others model them with Gaussian distributions [2574, 2575, 2576] and design corresponding loss functions [2577]. **Handling substantial scale variations.** Another research direction designs specialized backbones and feature extraction strategies to cope with large scale variations in remote sensing images (RSIs). A common strategy is multi-scale feature learning, achieved through data augmentation [2578, 2579, 2580], multi-scale feature

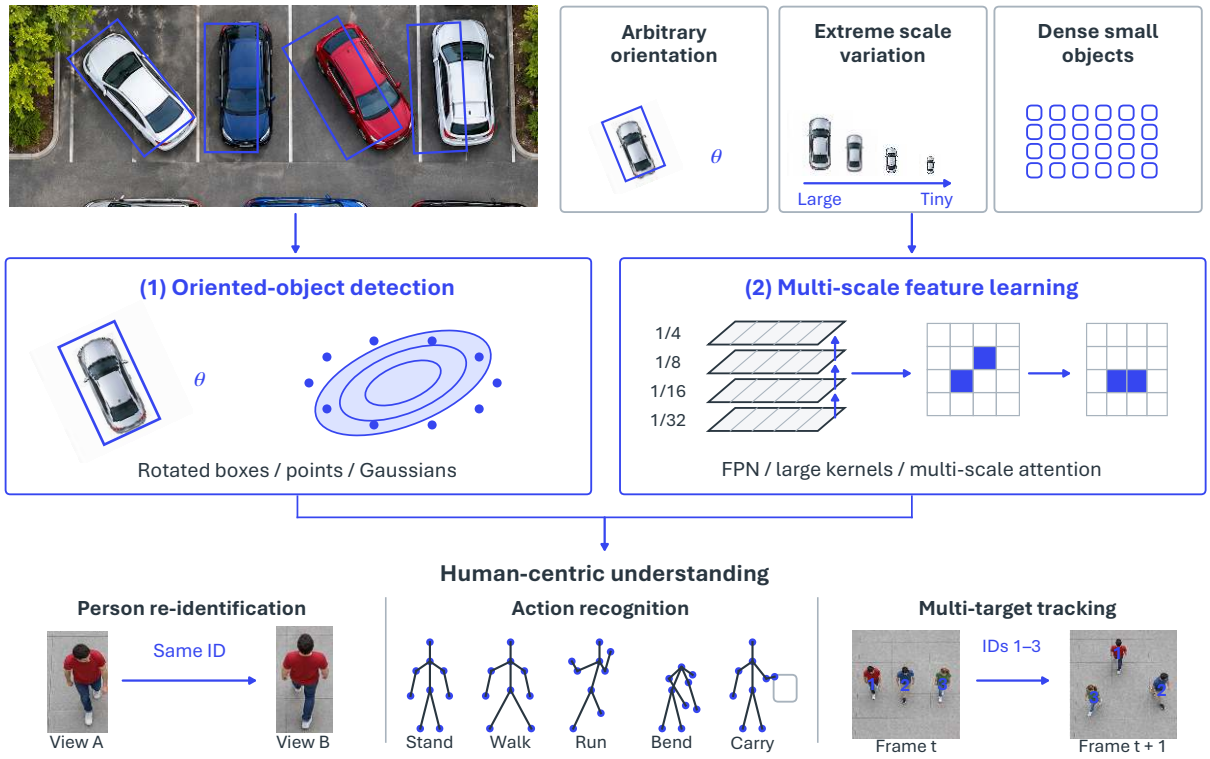


Figure 9.17: Remote-sensing detection from the aerial viewpoint: arbitrary orientation, extreme scale variation, and dense small objects (top) motivate (1) oriented-object detection and (2) multi-scale feature learning, which provide complementary representations for human-centric understanding (bottom). Person re-identification matches identities across views; action examples are alternative classes, not a prescribed sequence; tracking maintains identities across frames. Synthetic scenes illustrate these tasks.

integration [2581, 2582, 2583, 411], enhanced feature pyramid networks (FPNs) [597, 2584, 2585, 2586], or multi-scale anchor generation [2587, 2588, 2589]. Other works focus on ensuring equivalent receptive fields for objects of different orientations [2569, 2590]. More recently, some methods [2591] have employed large kernel convolutions [237, 2592, 2593] to enlarge the spatial receptive field for large objects, though this can introduce background noise for smaller ones, while others harness multi-scale convolution kernels and attention to address scale variance [2594, 2595, 2596, 2597].

These method fronts are evaluated on a complementary set of benchmarks and metrics. The VisDrone dataset [2557] provides a comprehensive benchmark for drone-captured imagery, featuring urban and rural scenes with challenges like varying altitudes and occlusions; DOTA-v2 [2555, 2556] focuses on oriented object detection in high-resolution satellite images, emphasizing rotated bounding boxes for objects like planes and ships; and UAV-Human [2531] targets human behavior understanding from UAVs, with multi-modal sequences for pose and action analysis. Complementary datasets, such as UAVDT [2598] for traffic surveillance and iSAID [2599] for instance segmentation, broaden the scope of aerial detection tasks and collectively enable cross-dataset fine-tuning to mitigate domain gaps. Evaluation extends standard metrics to address orientation, scale, and density: the primary metric, mean Average Precision (mAP), is computed as the area under the precision-recall curve, typically at an Intersection over Union (IoU) threshold of 0.5 (mAP@0.5) or across 0.5:0.95 for stricter assessment [440], and for oriented detection, as in DOTA-v2, rotated IoU variants ensure alignment with object headings [2555]. Per-class and per-size Average Precision exposes imbalances, and small objects ($< 32 \times 32$ pixels) remain the main source of error on drone imagery [2557]. Additional metrics include precision, recall, and F1-score to assess false positives (*e.g.*, background clutter) and misses (*e.g.*, occlusions), while Frames Per Second (FPS) is critical for real-time UAV deployment,

Table 9.7: Comparison of UAV-based human and object datasets.

Feature	UAV-Human [2531]	DOTA-v2 [2555, 2556]	VisDrone [2557]
Primary Focus	Human Behavior Understanding	Object Detection	Object Detection / Tracking / Crowd Counting
Data Type	Videos & Images	Images	Videos & Images
#Images	22,476 frames for pose estimation, 41,290 frames for re-ID, 22,263 frames for attribute recognition	11,268	10,209
#Videos	67,428	N/A	288
#Categories	155 action categories	18 object categories	10 object categories
#Instances	119 subjects for action recognition, 1,144 identities for re-ID	1,793,658	> 2.6 million annotated bounding boxes
Sensor Modalities	RGB, Depth, Infrared (IR), Fisheye, Night-Vision	RGB (from Google Earth, aerial) & Grayscale (from satellites)	RGB
Image Resolution	1920×1080 (RGB), 640×576 (Depth,IR), 640×480(Fisheye, Night-Vision),	800×800 to 20,000×20,000 pixels	2000×1500 (image), 3840×2160 (video)
Annotation Format	Bounding boxes, 17 keypoints for pose	Oriented Bounding Boxes (Quadrilaterals)	Bounding boxes with attributes (*e.g.*, occlusion)
Scene Diversity	Urban and rural, daytime and nighttime	Wide area, diverse landscapes (ports, airports, cities, etc.)	14 different cities, urban and rural, various weather and lighting
Platform	Drone (DJI Matrice 100)	Satellite, Aerial Platforms	Drone

targeting 30+ FPS [2600]. Video-specific metrics like Multiple Object Tracking Accuracy (MOTA) are used for dynamic scenes in datasets like UAV-Human [2531]. Modern real-time detectors of the YOLO family [2601] are increasingly adapted to the aerial setting, though aggregated aerial mAP still trails their COCO-level performance owing to small-object and orientation hardness, keeping open gaps such as multi-modal fusion and zero-shot adaptation active.

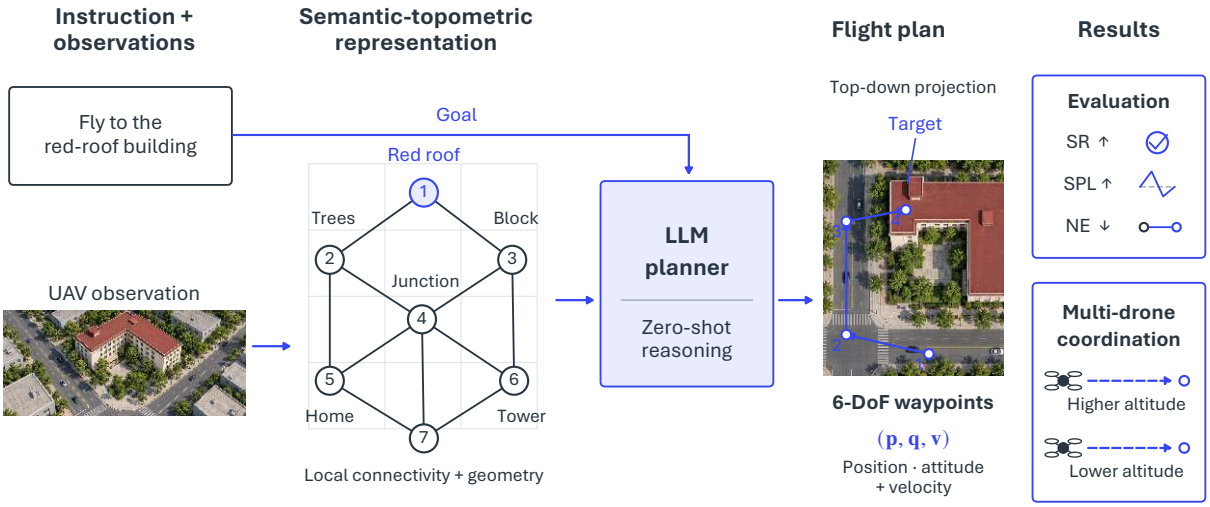


Figure 9.18: Autonomous aerial navigation from a language instruction: a UAV observation is grounded into a semantic-topometric graph; the instruction and graph condition an LLM planner that produces pose waypoints with velocity. The displayed route is a top-down projection, with attitude and altitude encoded in the waypoint state. Multi-drone planning is an extension with coordinated altitude separation. SR: success rate; SPL: success weighted by path length; NE: navigation error. Metric glyphs indicate evaluation criteria, not measured scores; scenes and graph geometry are schematic.

9.4.5 Pivotal Task-3: Autonomous Aerial Navigation and Decision Making

The final stage closes the loop, converting perception into action. Traditional UAV geolocalization addresses basic aerial navigation by matching a query (*e.g.*, a satellite image or textual description) to a database of historical flight paths or geographic points [2506, 2540]. However, a critical distinction exists between such perception-oriented tasks and the overarching goal of autonomous navigation: the latter requires complex decision-making and control to bridge perception to action, drawing on the spatial-reasoning and planning paradigms of Chapter 7. In practical scenarios, the objective is often for an aerial agent to autonomously locate a destination, particularly an unexplored target, through environmental exploration and, when available, collaboration with a map. This goal introduces significant challenges in generalization to unseen environments, including novel objects, unmapped regions, and interaction with the environment. For instance, recent benchmarks adapted from general Vision-Language Navigation (VLN) have begun to address this by requiring models to output a detailed flight plan with 6-DoF control signals [2602]. Figure 9.18 follows the problem from the perception-to-action gap to foundation-model-driven VLN and multi-agent extensions.

Table 9.8: Comparison of recent aerial VLN datasets.

Dataset	Yr	Env.	Size	Modalities	Key Features
UAVA [2603]	2020	Indoor simulated (MP3D)	90 traj.	RGB, depth, flow, ego/exo views	Photorealistic UAV assist; vision/robotics tasks; gamified traj.
AVDN [2604]	2022	Outdoor simulated (xView)	3,064 traj. with dialog	RGB, dialog	Async human dialogs; attn. for interactive nav.
AerialVLN [2605]	2023	Outdoor simulated (cities)	8,446 traj. and 25,338 instructions	RGB, lang. instr.	Synthetic VLN; landmark nav in urban scenes.
CityNav [2606]	2024	Real-world (UK cities)	32,637 traj./instr.	3D point clouds, RGB, geo-lang	Real 3D scans (4.65 km^2); human demos; scalable VLN.
UAV-Need-Help [2602]	2024	Outdoor simulated (OpenUAV)	12,149 traj.	RGB, lang. instr.	Continuous 6-DoF traj.; target-oriented.
OpenFly [2607]	2025	Outdoor real/sim (18 scenes)	100k traj.	3D point clouds, RGB, lang. instr.	Large-scale; real-to-sim transfer.
AirV2X [2608]	2025	Outdoor simulated (CARLA+AirSim)	6.73 hours of UAV-ground traj.	3D point clouds, RGB	V2X collab; drone strategies; reduces occlusions.

From visual perception to language-guided planning. Recent years have witnessed significant progress in UAV navigation, with a primary focus on visual perception [2609, 2610, 2611, 2612, 2613, 2614, 2615] and collision avoidance [2616, 2617]. Building on this aerial-perception foundation, Vision-and-Language Navigation (VLN) for UAVs remains an emerging area of research. To address it, recent efforts have leveraged Large Multi-modal Models (LMMs) to harness general knowledge for navigation planning: STMR [2618] builds upon AerialVLN [2605] with a zero-shot LLM-based framework utilizing a Semantic-TopoMetric Representation for enhanced spatial reasoning. A parallel line reframes UAV autonomy as a full agentic loop rather than one-shot instruction following. AeroVerse [2619] assembles a simulator, real-virtual pre-training corpora, and five downstream tasks, *i.e.*, scene awareness, spatial reasoning, navigational exploration, task planning, and motion decision, to train aerospace embodied world models, and proposes SkyAgentX as a unified perception-reasoning-navigation-planning agent; AirVista-II [2620] pursues zero-shot dynamic-scene semantic understanding via agent-based task scheduling and adaptive keyframe selection; and PRPSearcher [2621] casts aerial visual object search over the CityAVOS benchmark as a perception-reasoning-planning problem solved by a multimodal LLM agent. Complementary systems such as NeLV [2622] operationalize the language-to-flight pipeline end to end, chaining an LLM parser, route and path planners, and a control platform across short-, medium-, and long-range missions. Beyond single-agent setups, some

works have begun to explore multi-drone systems; for instance, Wu *et al*. [2623] deploy two drones flying at different altitudes to facilitate planning.

Datasets and simulators. As shown in Table 9.8, several foundational frameworks and datasets have been established. One of the early works, *i.e.*, UAVA [2603], leverages the panorama images in the Matterport3D dataset [340] to render the indoor environment, while harnessing the camera viewpoints as physically available flight anchors to generate trajectories. AerialVLN [2605] introduces a large-scale dataset with an effective baseline, and further infrastructure contributions include datasets such as CityNav [2606], AVDN [2604], and OpenFly [2607]. Notably, UAV-Need-Help [2602] provides a realistic UAV dynamics simulator, *i.e.*, OpenUAV, alongside its evaluation protocol; OpenUAV is further used in HaL-13k [2623] to simulate two drones at different altitudes. Additional observation agents, such as road infrastructures and vehicles with depth sensing, have also been incorporated to enrich environmental understanding [2608].

Metrics. Evaluation metrics for UAV aerial navigation, especially VLN, measure how well a model navigates a drone based on visual inputs and language instructions. Common metrics include Success Rate (SR), which checks if the UAV reaches the target location within a specified distance, and Success weighted by Path Length (SPL), which balances success with how efficiently the drone travels (shorter paths score higher). Path Length (PL) measures the total distance traveled, while Navigation Error (NE) calculates the distance between the final position and the goal. Together these metrics assess both accuracy (reaching the target) and efficiency (taking an optimal route), providing a clear way to compare model performance across datasets like UAVA, AerialVLN, and CityNav.

9.4.6 Established Tools and Systems

Beyond the methods and benchmarks above, the aerial domain rests on a real-world layer of simulators, platforms, and infrastructure that turns research prototypes into deployable systems; we group it into three tiers.

Simulation environments and benchmarks. Because flight data is costly and safety-critical to collect, high-fidelity simulation underpins much of the field. The OpenUAV simulator released with UAV-Need-Help [2602] supplies realistic UAV dynamics and a continuous 6-DoF evaluation protocol, and is reused to simulate multi-altitude drone teams in HaL-13k [2623]; OpenFly [2607] targets large-scale real-to-sim transfer across diverse scenes; and AerialVLN [2605] pairs a city-scale synthetic environment with a navigation baseline. Photorealistic indoor rendering for UAV tasks is provided through Matterport3D [340] in UAVA [2603], while V2X-style multi-agent simulation combines CARLA and Microsoft AirSim[44] in AirV2X [2608].

Commercial platforms and vendors. On the hardware side, consumer and enterprise drones from DJI[45] (whose Matrice series collects datasets such as UAV-Human) and autonomy-focused vendors such as Skydio[46] dominate the UAV tier. In the stratospheric tier, solar-powered HAPS such as Airbus Zephyr [2525] and AeroVironment/HAPSMobile's HAWK30 [2526] provide persistent regional coverage, complemented by small-satellite constellations built on CubeSat platforms [2522] and broader HAPS programs [2523].

Connectivity and positioning infrastructure. Aerial autonomy ultimately depends on shared positioning and communication backbones: the GNSS constellations introduced above (GPS [2472], GLONASS [2475], Galileo [2476], and BeiDou [2477]) supply global timing and localization, while space-based broadband relays such as Starlink [2624] and orbital observation systems [2524] increasingly serve as the connectivity layer for coordinated, real-time aerial operations.

[44] https://microsoft.github.io/AirSim/
[45] https://www.dji.com/
[46] https://www.skydio.com/

9.4.7 Challenges

Aerial autonomy is limited less by any single model than by the gaps between platforms, scales, and operating conditions. Four threads follow the pipeline in order: fusing heterogeneous platforms into a coherent situational model, perceiving small objects and complex activities reliably, navigating unseen and GPS-denied environments, and meeting the ethical and operational constraints under which any of this can be deployed.

Heterogeneous-Platform Fusion. A primary obstacle lies in integrating aerial agents with heterogeneous platforms for robust environmental perception. Coordinating a UAV with ground-based sensors or satellite systems to construct a cohesive situational-awareness model remains complex, particularly in dynamic or cluttered settings where the platforms differ in resolution, latency, and viewpoint (Table 9.5). Reconciling these asynchronous, multi-resolution streams into a single consistent representation, without one modality's noise corrupting the fused estimate, is the standing difficulty, and it motivates the V2X-style cooperative-perception efforts that incorporate road infrastructure and ground vehicles as additional observers.

Small-Object and Complex-Activity Understanding. Aerial perception is defined by extremes of scale and density. The recognition of small objects, such as identifying debris in disaster zones or distinguishing wildlife in ecological surveys, demands high-resolution imaging and detectors robust to dense clustering, which often strain onboard computational resources. Understanding complex activities, such as discerning human behavior during search-and-rescue missions, compounds the problem by requiring nuanced spatio-temporal and contextual analysis that current systems struggle to deliver consistently. Closing this gap calls for detectors and activity models that remain accurate at extreme scale variation while fitting a UAV's tight compute and power budget.

Navigation in Unseen and GPS-denied Environments. Autonomous operation in unseen or GPS-denied environments, such as dense urban canyons or subterranean spaces, further complicates the action stage, as agents must rely on real-time mapping and obstacle avoidance without pre-existing data. The core requirement is generalization: a navigation policy must extend to novel objects, unmapped regions, and genuinely interactive scenarios, rather than replaying matched trajectories. This is precisely the gap that foundation-model-driven VLN and multi-altitude cooperative planning aim to narrow, but validating such behavior to a safety standard in the open world remains open.

Privacy, Airspace, and Operational Robustness. Beyond perception and navigation, several ethical and operational challenges persist. Data-privacy concerns arise when aerial agents capture high-fidelity multimedia data, potentially compromising individual rights without stringent safeguards, motivating privacy-preserving approaches such as federated learning. Airspace integration poses another hurdle, as regulatory frameworks struggle to accommodate the proliferation of aerial agents alongside traditional aviation, necessitating harmonized protocols to ensure safety. Energy efficiency remains a critical bottleneck, since prolonged missions demand lightweight yet powerful power sources and motivate renewable solutions such as solar-powered propulsion. Finally, operational robustness is challenged by environmental uncertainties, such as extreme weather or electromagnetic interference, that can disrupt system reliability. Advances in artificial intelligence and sensor fusion that address these constraints, rather than perception accuracy alone, are what will let aerial agents move from bounded demonstrations toward dependable, large-scale deployment.

9.5 Warehouse and Industrial Logistics

9.5.1 Introduction

Warehouse and industrial logistics is one of the most mature real-world testbeds for the spatial capabilities surveyed in Part III: it demands perception-to-action pipelines that interpret complex, dynamic environments such as dense shelving systems populated with diverse stock keeping units (SKUs), pallets, forklifts, and human workers. These systems construct and maintain spatial memory representations (semantic maps, object and pose states) to support downstream planning and manipulation for tasks including inventory counting, order picking, pallet handling, and coordinated fleet operations. Perception and reasoning themselves are treated in Chapter 6 and Chapter 7; logistics adds throughput targets, engineered but crowded layouts, and people sharing the floor with fleets of robots. Performance is judged in terms of throughput, order accuracy, safety, and operational cost, especially under high SKU diversity, visual clutter, elongated aisles, and human-robot collaboration. Dense shelving and manipulation in clutter remain the hardest cases, as the Amazon Picking Challenge made plain [2625], and they demand that perception, pose estimation, mapping, and multi-agent planning be integrated rather than tuned in isolation.

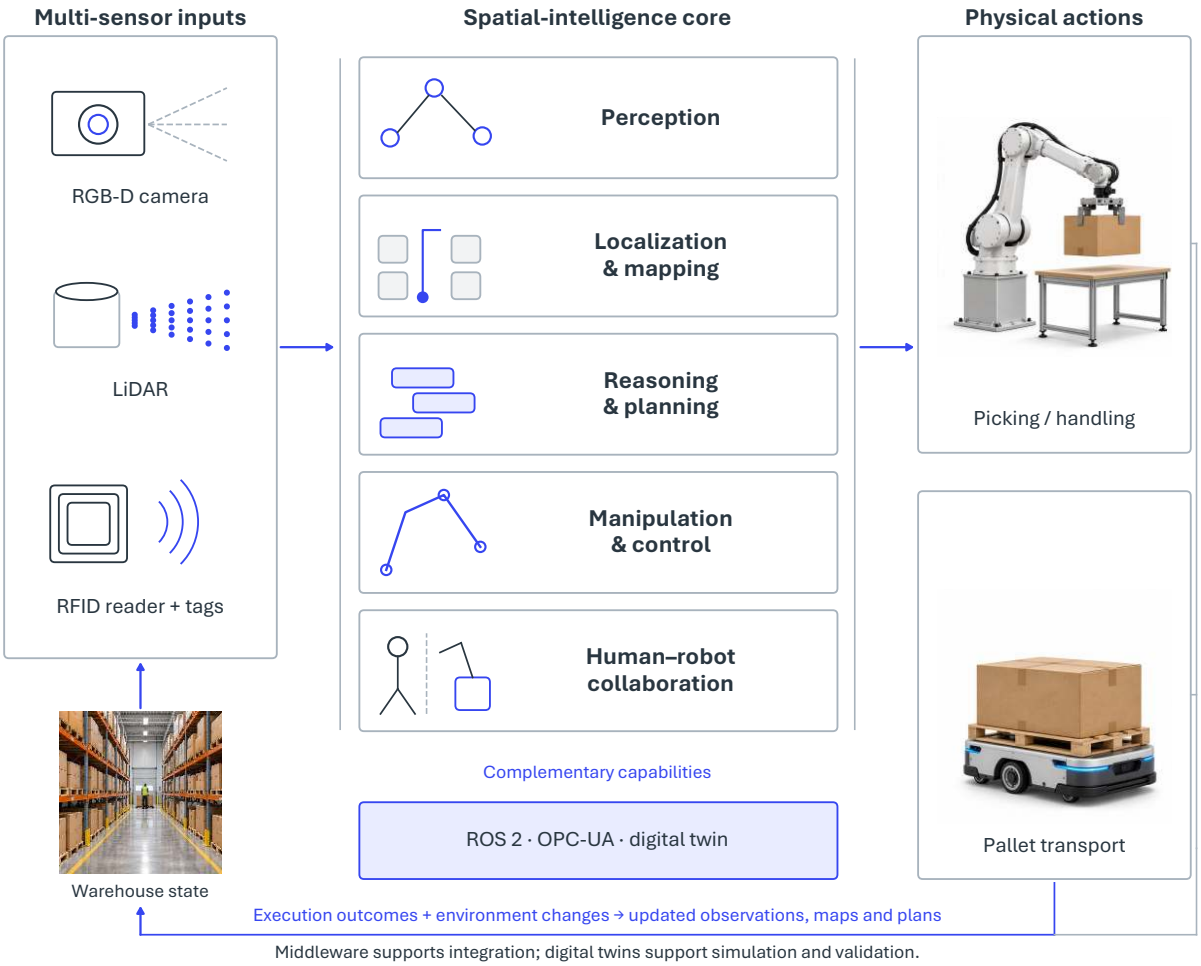


Figure 9.19: Spatial intelligence for warehouse and industrial logistics: RGB-D, LiDAR, and RFID observations support five complementary capabilities for picking and pallet transport, and execution outcomes update maps and plans. Scene images and diagrams are illustrative.

9.5.2 System Overview

The complete warehouse automation pipeline, sketched in Figure 9.19, integrates five complementary capabilities into a closed loop. **Perception** converts raw multi-sensor streams into semantic object information (what is present, where, and in what pose); **localization and mapping** grounds that information in a metric spatial map and keeps it consistent over time; **reasoning and planning** consume these representations to allocate tasks and generate optimized fleet schedules; **manipulation and control** physically execute the resulting picking and transport actions; and a **human-robot collaboration and safety** layer adapts behavior dynamically in mixed human-robot environments. Integration is achieved via middleware such as ROS 2 and industrial protocols like Open Platform Communications Unified Architecture (OPC-UA), while digital twin platforms such as NVIDIA Omniverse[47] support simulation and validation before deployment. The five capabilities are the pivotal tasks below, taken in pipeline order.

9.5.3 Pivotal Task-1: Perception and Recognition

Perception constitutes the sensory foundation of the pipeline: it transforms raw sensor data into actionable knowledge, reusing the detection, segmentation, and pose-estimation machinery of Chapter 6 but specializing it to the warehouse Operational Design Domain. Core capabilities include detecting and classifying objects (packages, pallets, tools), estimating their 3D poses, and tracking their motion and state over time. Unlike outdoor environments, warehouses impose distinct perceptual challenges: dense clutter, heavy occlusion, visually similar items, reflective packaging, narrow aisles, and continually changing layouts, which together make robust perception a central and ongoing research focus [2626, 130]. Formally, perception is a structured mapping from a synchronized multimodal stream $X_t = \{I_t^{\text{RGB}}, I_t^{\text{Depth}}, P_t^{\text{LiDAR}}, R_t^{\text{RFID}}\}$ (together with calibration parameters and a prior map M) to a scene graph $G_t = \{o_i, c_i, \mathbf{b}_i, \mathbf{p}_i, \Sigma_i\}$, in which each detected object o_i carries a class label c_i, a bounding box or segmentation mask $\mathbf{b}_i$, an estimated pose $\mathbf{p}_i$, and an uncertainty measure Σ_i that grounds subsequent manipulation, navigation, and decision making. Figure 9.20 sketches this representation stack.

Detection and recognition form the base layer. Convolutional and transformer-based architectures, *i.e.*, YOLO [130], RetinaNet [412], and the transformer-based DETR with deformable attention [415], have steadily improved accuracy and efficiency under clutter, and are scored on COCO [440], whose mean Average Precision (mAP) broken down by object size is particularly informative for the small-SKU and barcode-label cases that dominate warehouse failure modes. **Pose estimation** lifts these 2D detections into the metric space that manipulation requires: PoseCNN [737], DenseFusion [740], and CosyPose [746] fuse visual and geometric cues to infer six-degree-of-freedom (6D) orientation and position even under occlusion, and are evaluated on YCB-Video [737] for textured household and industrial objects and on T-LESS [768] for textureless, industry-relevant parts, with Average Distance (ADD) the standard accuracy measure and MOTA/MOTP reporting tracking quality in dynamic scenes. The Amazon Picking Challenge [2625] couples perception with manipulation in real warehouse scenarios, stressing generalization across novel object sets and thereby anticipating the manipulation task of §9.5.6 below. Representative benchmarks across the section are consolidated in Table 9.9. In industrial practice, perception pipelines are increasingly standardized and modularized; the deployed software stacks that realize them are surveyed in the Established Tools and Systems subsection.

9.5.4 Pivotal Task-2: Localization and Mapping

If perception answers *what is around the robot*, localization and mapping answer *where the robot is* and *what the persistent structure of the warehouse is*, forming the spatial backbone on which fleet coordination rests and drawing on the same scene-construction techniques as Chapter 6. Automated guided vehicles

[47] https://www.nvidia.com/en-gb/omniverse/

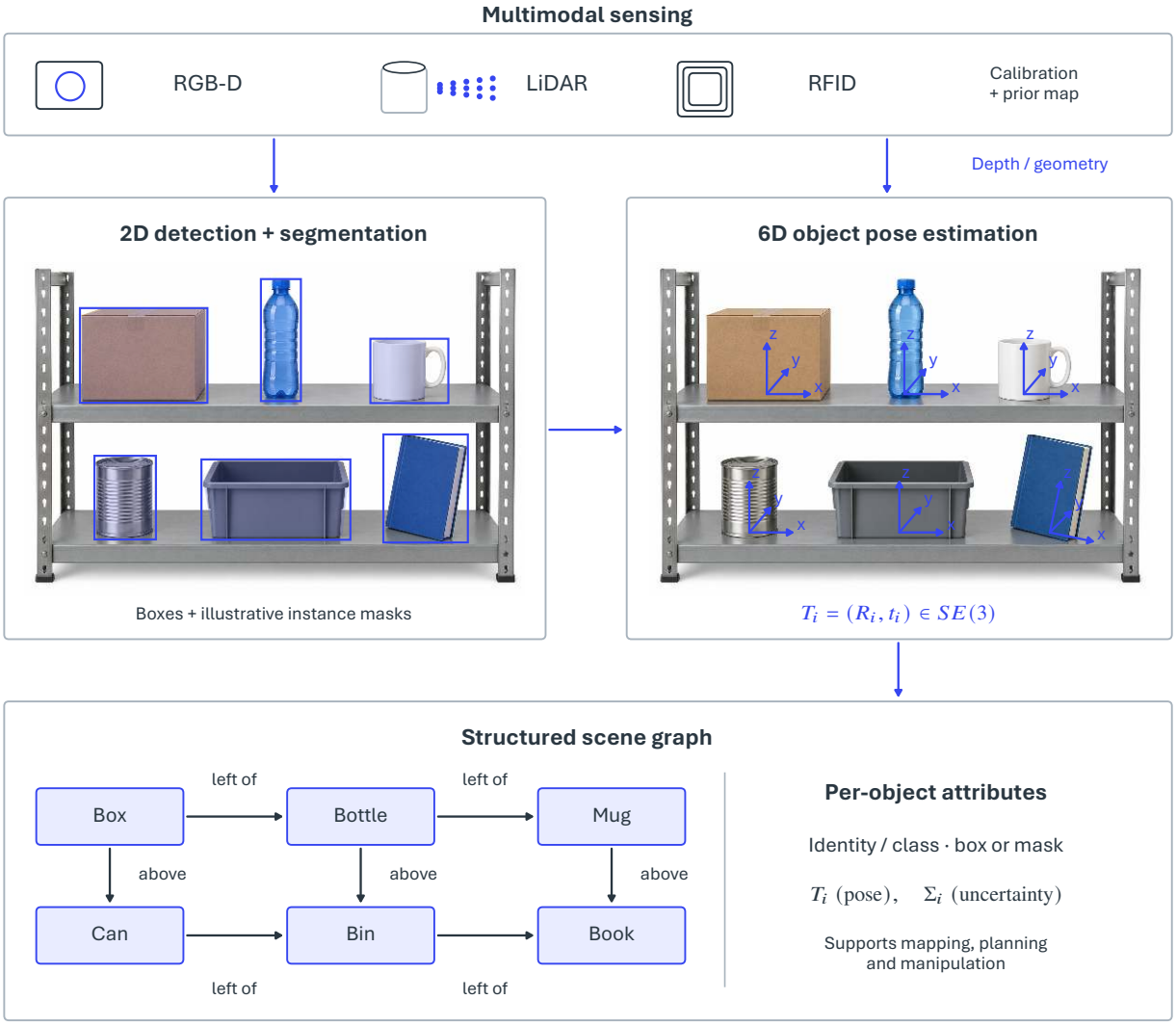


Figure 9.20: The warehouse perception stack: 2D detection and segmentation, 6D object pose T_i, and a structured scene graph whose nodes carry identity, class, box or mask, pose, and uncertainty Σ_i. RFID can support identity association; masks and axes are schematic, not calibrated estimates.

(AGVs) and autonomous mobile robots (AMRs) require centimeter-level accuracy to navigate narrow aisles, dock precisely at shelves, and operate safely alongside workers. The problem is conceptually the SLAM problem of autonomous driving, yet it faces warehouse-specific constraints: highly repetitive structures (rows of identical shelves), dynamic obstacles (workers, pallets, forklifts), and low-light or GPS-denied indoor conditions. Formally, a sequence of odometry and sensor measurements $Z_{1:t} = \{\text{IMU}, \text{LiDAR}, \text{cameras}, \text{wheel encoders}, \text{UWB beacons}\}$, optionally supplemented by a prior map M_0 or fiducial anchors, is fused into a pose graph $\mathcal{X}_t = \{x_1, \ldots, x_t\}$ that estimates the trajectory and an evolving semantic map M_t fusing geometric structure with storage zones and restricted areas. Beyond navigation, this map underpins the digital twins (continuously updated spatial models used for simulation, optimization, and predictive planning) that increasingly govern modern facilities [2627]. Figure 9.21 casts the task as a multi-sensor fusion problem.

A family of **SLAM algorithms** has been tailored to these conditions. ORB-SLAM3 [736] extends visual SLAM with inertial and multi-camera support for accurate visual-inertial localization; LIO-SAM [2628] provides real-time LiDAR–IMU fusion through factor-graph optimization that stays robust under fast motion

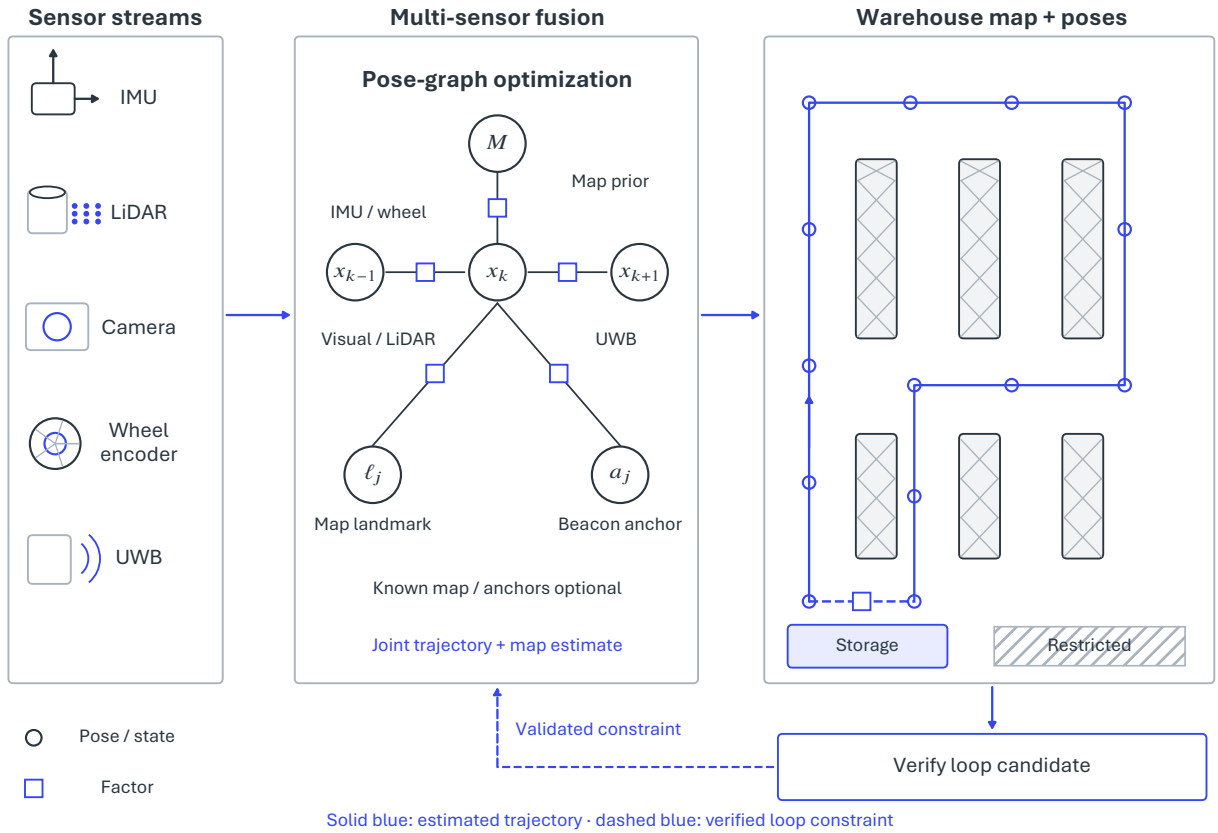


Figure 9.21: Localization and mapping in the warehouse as a factor-graph fusion problem: IMU, LiDAR, camera, wheel-encoder, and UWB measurements constrain pose states, optionally with a prior map or beacon anchors, producing poses over warehouse geometry, storage zones, and restricted areas. Repetitive shelving can cause false place matches, so only verified loop candidates add constraints.

and partial occlusion; and RTAB-Map [2629] supports online RGB-D mapping and loop-closure detection with memory management suited to long-term operation, all expressible within the unified factor-graph framework of GTSAM [2630]. **Multi-modal fusion** is the common thread that holds accuracy together when any single modality degrades, and is also the dominant emerging research direction, alongside learning-based estimation (neural depth prediction, differentiable bundle adjustment, and learned place-recognition embeddings) and the migration of SLAM computation to the edge for fleet-scale latency. These capabilities are evaluated on KITTI [653] for outdoor visual and LiDAR odometry and loop closure, on the indoor visual-inertial EuRoC MAV dataset [733] whose challenging motion and illumination mirror warehouse conditions, and on the high-fidelity Replica reconstructions [339] for dense mapping and semantic labeling, with Absolute and Relative Pose Error (APE/RPE) and loop-closure precision/recall the standard metrics and, for warehouse use, semantic-zone IoU and map completeness (Table 9.9).

9.5.5 Pivotal Task-3: Reasoning and Planning

Reasoning and planning form the decision-making core of warehouse automation, transforming the perceptual and localization outputs into task assignments and motion plans, and instantiating the spatial-reasoning and planning paradigms of Chapter 7. This layer determines which robot executes which task, how tasks are sequenced, and how routes and schedules are optimized for safe, efficient, coordinated operation. In contrast to autonomous driving, where planning typically focuses on individual vehicles, warehouse reasoning must address multi-agent coordination across fleets of dozens or hundreds of robots in constrained indoor space [2631], maximizing throughput and minimizing latency while maintaining safety under dynamic obstacles,

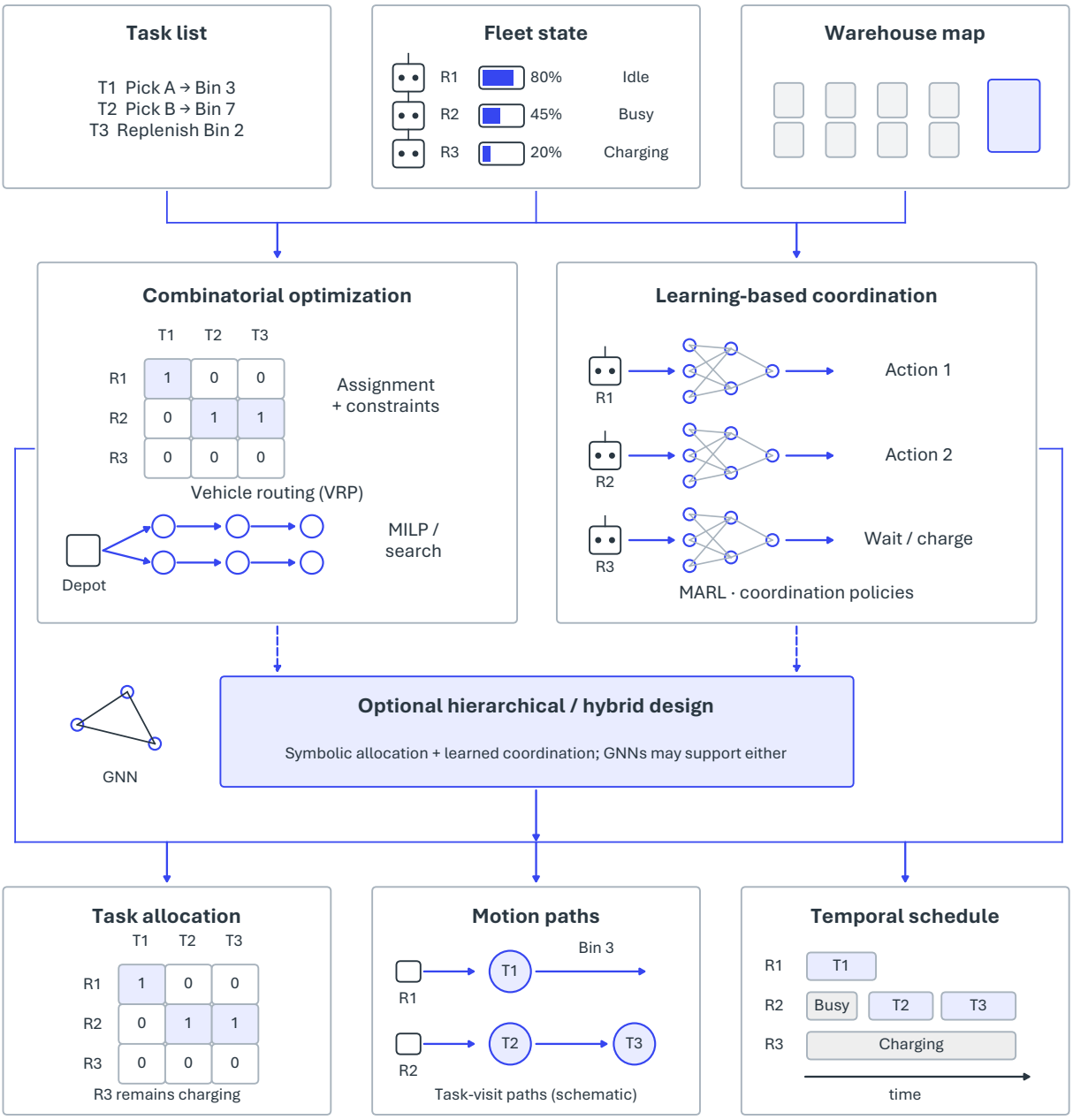


Figure 9.22: Two families for warehouse reasoning and planning over shared task, fleet, and map inputs: combinatorial optimization (assignment and routing) and learning-based coordination (per-agent policies), optionally combined in hierarchical or hybrid designs. Allocations, paths, and schedules are illustrative.

human interactions, and stochastic task arrivals. Formally, the perceived states G_t, a task list $\mathcal{T}$, the fleet state F_t (positions, capabilities, battery levels), and the warehouse map M_t are mapped to a task-allocation matrix A, optimized motion paths $\mathcal{R}$, and temporal schedules π that jointly specify when and how tasks execute. Figure 9.22 contrasts the two solution families.

The first family is **combinatorial optimization**. Traditional methods (the Hungarian algorithm, branch-and-bound, and mixed-integer linear programming) remain widely used for task allocation and scheduling in small- to medium-scale deployments, while Vehicle Routing Problem (VRP) formulations [2632] and their multi-depot and time-window variants are central to modeling fleet movement. The second family is **learning-**

based coordination: reinforcement learning (RL) and multi-agent RL (MARL) handle dynamic environments and real-time uncertainty [2633, 1435], with toolkits such as Google OR-Tools [2634] and PyMARL enabling rapid experimentation across both families. Emerging research couples the two: hierarchical and hybrid frameworks combine symbolic reasoning with learned components, graph neural networks and differentiable optimization accelerate multi-agent coordination and transfer policies across layouts, and digital-twin integration enables simulation-informed, predictive re-planning, increasingly under multi-objective criteria that balance throughput, energy, and human safety. Planning systems are benchmarked on Solomon's VRP suite [2635] for static routing and scheduling and on the Open Motion Planning Library (OMPL) [1272] for continuous-space motion planning, complemented by synthetic multi-robot testbeds; common metrics are throughput (items/hour), task-completion latency, total travel distance or time, collision rate, and re-planning latency (Table 9.9).

9.5.6 Pivotal Task-4: Manipulation and Control

Manipulation constitutes the physical execution layer, enabling robots to interact directly with objects, containers, and infrastructure such as bins, shelves, and conveyors. In industrial logistics this typically involves robotic arms performing grasping, placing, sorting, or palletizing, as well as AGVs docking to pallets or workstations. Manipulation bridges high-level symbolic or geometric plans with continuous motor control, and therefore demands both precision and robustness to the inherent variability in packaging, geometry, and material properties [35]; the grasping and control techniques here connect directly to the robotic-manipulation treatment of §9.1 and the planning paradigms of Chapter 7. Formally, estimated object poses $\mathbf{p}_i$ and task descriptions (grasp targets, placement goals, trajectory constraints) are mapped to low-level motor commands $\mathbf{u}_t$ that ensure stable grasping, precise placement, and compliant, collision-free motion. Figure 9.23 contrasts modular grasp-and-plan systems with learned visuomotor control.

Grasp synthesis anchors the methodology: Dex-Net combines synthetic grasp data and analytic robustness metrics to train a grasp-quality predictor for candidate selection [336], while learned visual control can predict grasp success from camera images and candidate task-space motions, then select actions for adaptive manipulation [35]. **Motion planning** then realizes the chosen grasp: sampling-based algorithms such as RRT* [1223] remain fundamental for generating collision-free trajectories in high-dimensional configuration spaces, exposed through frameworks like MoveIt [2636]. These capabilities are benchmarked on the Amazon Picking Challenge [2625], which established a standardized testbed for picking under clutter and uncertainty, and on the YCB Object and Model Set [1184], a diverse collection of everyday items with physical models for grasping and manipulation; grasp success rate, cycle time per pick, placement accuracy, and task-completion rate are the headline metrics (Table 9.9). Hybrid control frameworks that combine analytical models with data-driven adaptation are emerging as the practical route to generalizing across variable geometries and grasp contexts, complemented by tactile sensing and force control for slip detection and contact regulation.

9.5.7 Pivotal Task-5: Human-Robot Collaboration and Safety

Even in highly automated warehouses, human workers remain indispensable for complex, dexterous, or exception-handling tasks that require contextual understanding and flexible decision making, so the final stage of the pipeline is safe and efficient human-robot collaboration (HRC). HRC systems must balance productivity with stringent safety guarantees, enabling mixed teams (robots handling repetitive or heavy-load work, humans managing fine-grained, judgment-based activities) to share workspaces without physical or operational interference [2637]. The task maps multimodal human-detection and intent signals (pose, gesture, gaze, and trajectory data from vision, LiDAR, wearables, or motion capture) to adaptive robot behaviors such as dynamic speed modulation, path re-routing, stop-and-wait actions, or task reallocation in response to predicted human movement. Figure 9.24 sketches this predict-then-adapt loop.

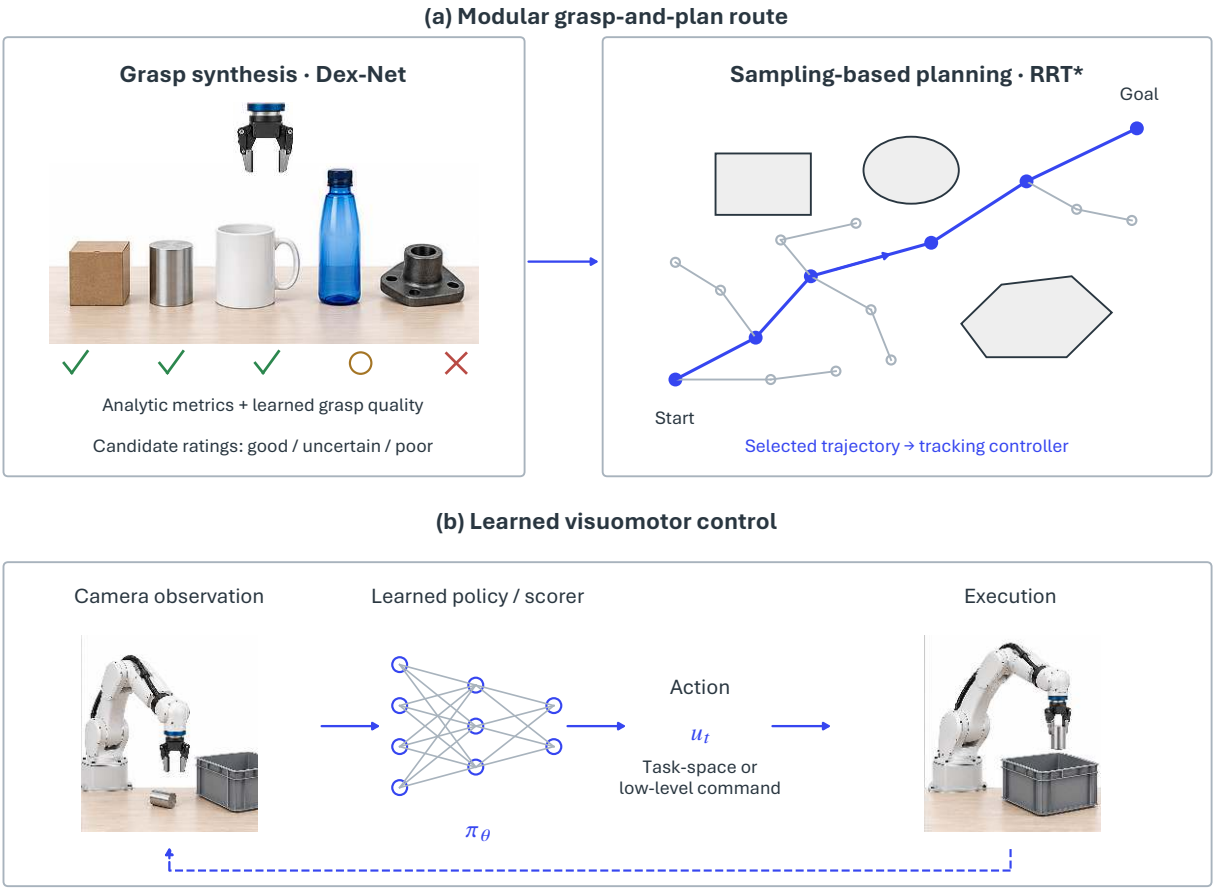


Figure 9.23: Warehouse manipulation through (a) a modular route, where grasp candidates scored by analytic metrics and learned quality (Dex-Net) feed sampling-based planning (RRT*) and trajectory tracking, and (b) learned visuomotor control from camera observations. Action spaces vary across systems, and the routes can be combined.

Intent recognition lets robots anticipate human motion: sequence models based on hidden Markov models or recurrent networks predict trajectories and proactively adjust behavior [2638], while **proxemics-aware planning** maintains socially acceptable distances and re-routes trajectories under human proximity [2639], with multimodal sensing (vision-based detection fused with wearable or RFID localization) improving reliability. These principles are embodied in force- and speed-limited collaborative robots and safety-rated AMRs (see Established Tools and Systems), governed by international standards such as ISO 10218 and ISO/TS 15066. Evaluation combines quantitative safety metrics with qualitative interaction assessment: human-motion corpora such as Human3.6M [550] supply sequences for intent prediction, and key indicators include intervention frequency (safety-triggered stops per hour), near-miss rate, human-detection accuracy, and compliance with ISO-defined safety zones.

9.5.8 Established Tools and Systems

The pivotal tasks above are realized in production through a layered ecosystem of simulators, middleware, optimization services, and robot hardware that together constitute the real-world delta over the methods of Part III.

Simulation and middleware form the integration substrate. Digital-twin and physics platforms such as NVIDIA Omniverse Isaac Sim [2327] and Gazebo provide realistic environments for developing and validating perception, localization, and planning pipelines before deployment, while perception SDKs such as

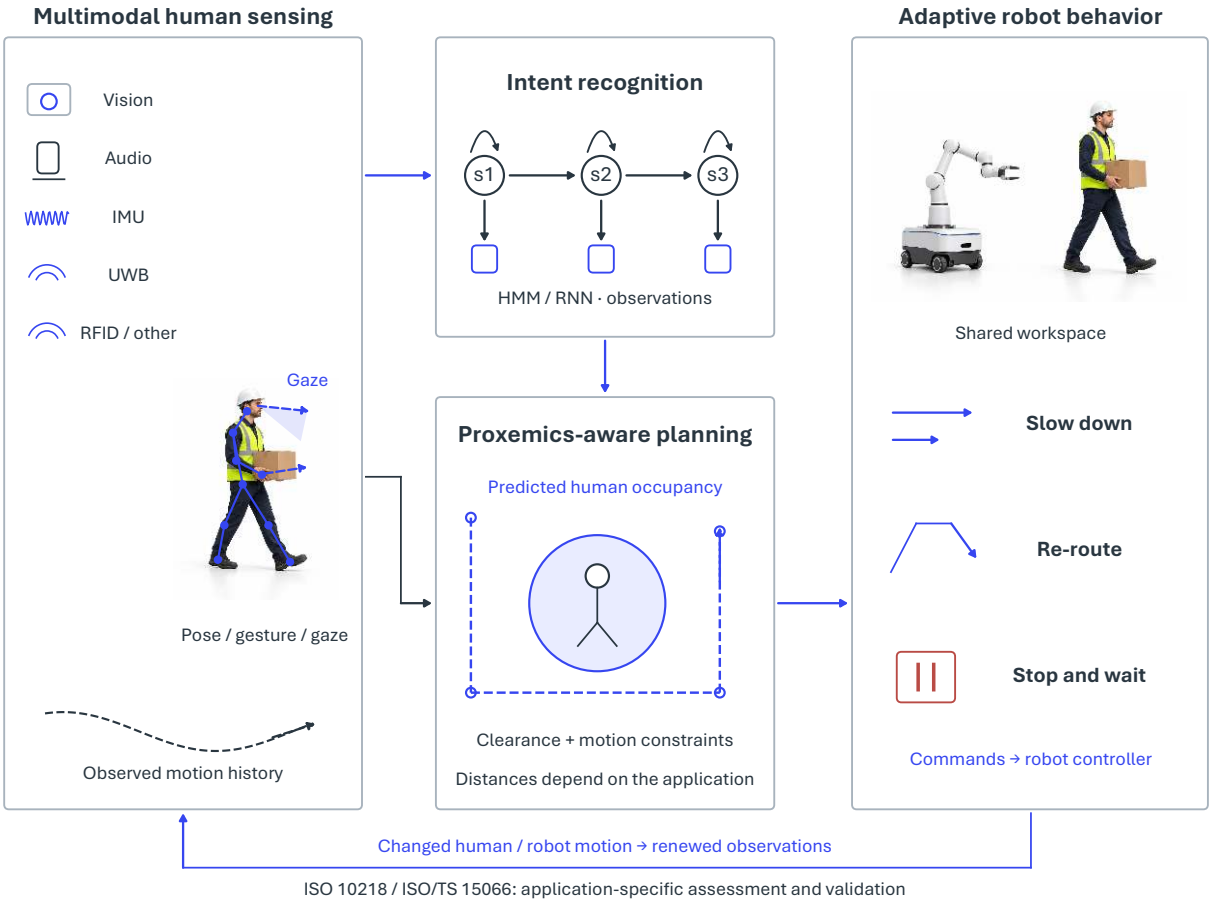


Figure 9.24: The human–robot collaboration loop: multimodal human sensing and intent recognition inform proximity-aware planning, which selects slowing, rerouting, or stopping. Clearance regions and routes are illustrative, not certified safety zones.

Table 9.9: Representative benchmarks for spatial intelligence in warehouse and industrial logistics, spanning the five pivotal tasks.

Pivotal task	Benchmark	Representative method	Headline metric
Perception (detection)	COCO [440]	YOLO [130] / DETR [415]	mAP
Perception (6D pose)	YCB-Video [737]	DenseFusion [740]	ADD
Perception (textureless pose)	T-LESS [768]	CosyPose [746]	ADD / VSD
Localization & mapping	KITTI [653]	Visual / LiDAR SLAM	APE / RPE
Localization & mapping	EuRoC MAV [733]	ORB-SLAM3 [736] (visual-inertial)	APE
Mapping (dense)	Replica [339]	RTAB-Map [2629]	IoU / completeness
Reasoning & planning	Solomon VRP [2635]	OR-Tools [2634]	total cost / distance
Reasoning & planning	OMPL [1272]	Sampling-based planners	planning time / success
Manipulation	Amazon Picking [2625]	Integrated perception–grasping team pipelines	pick success / time
Manipulation	YCB Object Set [1184]	Dex-Net [336]	grasp success rate

NVIDIA Isaac, the Intel RealSense SDK [260], and Zebra RFID/barcode platforms [2640] supply end-to-end sensing support in real facilities. Interoperability across heterogeneous hardware is provided by ROS [2641], ROS 2, and ROS-Industrial, with commercial localization stacks such as SLAMcore[48] and BlueBotics ANT[49] delivering optimized multi-sensor fusion, map management, and fleet coordination for AGVs and AMRs.

[48] https://www.slamcore.com/
[49] https://bluebotics.com/

Warehouse management and fleet orchestration sit at the enterprise layer. Large-scale optimization services such as SAP Extended Warehouse Management[50], Blue Yonder WMS[51], and Manhattan Associates[52] integrate task allocation and scheduling with enterprise resource planning, while open toolkits such as Google OR-Tools [2634] and PyMARL support the underlying combinatorial and learning-based planners.

Robot hardware and manipulation close the loop physically. High-throughput pick-and-place arms from ABB[53], FANUC[54], and KUKA[55] are widely deployed and integrated through ROS-based middleware, mobile manipulators such as those from Fetch Robotics[56] unify navigation and grasping, and motion-planning frameworks such as MoveIt [2636] and NVIDIA cuMotion provide inverse kinematics and trajectory optimization. For shared workspaces, collaborative robots from Universal Robots[57] and AMRs from Locus Robotics[58] provide the safety-rated platforms on which the HRC layer is built.

9.5.9 Challenges

Deployed warehouse systems work because they fix the SKU sets, layouts, and operating conditions they handle, and each stage of the pipeline has its own obstacle to relaxing those limits: small-object and cluttered perception, perceptual aliasing in repetitive aisles, multi-agent scalability under uncertainty, manipulation of variable objects, and the safety–throughput trade-off in human–robot collaboration.

Cluttered and Small-Object Perception. Barcode labels and compact SKU packages often occupy only a few pixels in high-resolution images, leading to frequent misdetections, while occlusion and clutter compound the difficulty as items overlap within bins and shelves. A further persistent issue is domain shift: models trained on synthetic or generic data degrade under real-world variation in lighting, reflective surfaces, and packaging. Research responses include domain adaptation and self-supervised learning for generalization [2642], active perception that deliberately adjusts viewpoints or sensing to resolve occlusion [2643], and continual learning that lets perception adapt over time without exhaustive retraining, increasingly combined with multi-sensor fusion and neuro-symbolic representations for higher-level causal and semantic reasoning.

Perceptual Aliasing in Repetitive Environments. Localization and mapping are degraded by dynamic obstacles (humans, forklifts) that generate transient or misleading landmarks corrupting loop-closure hypotheses [2627], and above all by perceptual aliasing: repetitive shelving and low-texture surfaces produce ambiguous place recognition and false data associations in feature-based SLAM [736, 2629]. High-speed AGV and AMR operation further demands low-latency, sub-centimeter pose updates that sensor delays, occlusion, and monocular scale drift can readily violate. Robust multi-modal fusion, learning-based estimation, and tighter integration of SLAM with semantic perception and digital twins are the leading mitigations.

Scalability and Uncertainty in Multi-agent Planning. The combinatorial complexity of multi-robot coordination grows exponentially with the number of agents and tasks, rendering exact optimization intractable for large fleets, while real-world uncertainty (new order arrivals, blocked aisles, unpredictable human movement) necessitates adaptive re-planning and decentralized decision making under partial observability [2631, 2633]. Hierarchical and hybrid planners, graph-neural and differentiable coordination, and simulation-informed multi-objective optimization are the active directions toward scalable, resilient fleet behavior.

[50] https://www.sap.com/products/scm/extended-warehouse-management.html
[51] https://blueyonder.com/
[52] https://www.manh.com/
[53] https://new.abb.com/products/robotics
[54] https://www.fanuc.com/
[55] https://www.kuka.com/
[56] https://fetchrobotics.com/
[57] https://www.universal-robots.com/
[58] https://locusrobotics.com/

Manipulation of Variable and Deformable Objects. Many SKUs exhibit deformable, reflective, or transparent surfaces that interfere with perception and grasp stability [1184, 336], and deformable handling remains especially difficult because grasp dynamics and contact points resist rigid-body modeling [35]. Tactile sensing and force control improve slip detection and grasp robustness but face cost and calibration barriers in industrial deployment, motivating hybrid control that combines analytical models with data-driven adaptation for generalization across geometries and grasp contexts.

Safety–Throughput Trade-off in Collaboration. Robust human detection under occlusion, variable lighting, and cluttered aisles continues to hinder reliable intent estimation [2639, 2638], and the trade-off between throughput and safety remains unresolved: conservative margins reduce efficiency while aggressive planning risks unsafe interaction [2637]. Emerging work emphasizes predictive and explainable safety models that justify decisions in human-interpretable terms, and adaptive safety frameworks that fuse learned intent models with certified control policies for dynamic risk assessment and shared autonomy on the warehouse floor.

Chapter 10

Spatial Intelligence for Virtual Realities

Virtual realities are the media in which spatial intelligence becomes constitutive rather than optional: an extended-reality (XR) device cannot draw a single pixel until it knows where it is, what surrounds it, and where the user is looking and reaching. This makes XR the most demanding real-time deployment target for the perception, reasoning, and generation capabilities of the previous part, specialized to run egocentrically, on-device, and within a hard latency budget.

Beneath everything in XR sits the platform itself, the runtime that localizes the device, maps and registers the surrounding world, senses the user's body, and composites the result in real time; layered on top is the application that has driven the medium hardest, entertainment, spanning live AR and VR experiences and virtual production as well as the offline, high-fidelity reconstruction and performance capture that supply them with authored content.

10.1 AR/VR/XR

10.1.1 Introduction

Augmented, virtual, and mixed reality (collectively, extended reality or XR) form a continuum of media in which a head-worn display couples the user's egocentric viewpoint to a computational world that is registered to, or substituted for, the physical surroundings. What distinguishes XR from screen-based media is that spatial intelligence is not an optional feature but a constitutive requirement: before any pixel can be drawn, the system must know where the headset is, what geometry surrounds it, and where the user is looking and reaching. XR is therefore the most demanding real-time deployment target for the spatial capabilities surveyed in Part III. The perception machinery (detection, depth, pose, reconstruction) of Chapter 6, the spatial reasoning and planning of Chapter 7, and the neural reconstruction and rendering of Chapter 8 all reappear here, but specialized to run egocentrically, on-device, and within a hard latency budget.

This section treats the XR *medium itself* as a spatial-computing platform: the runtime engine that localizes the device, maps and registers the world, senses the user's body, and presents a composited result. The neighboring section on entertainment (§10.2) builds *on top of* this engine, treating entertainment as an application domain: its real-time AR/VR experiences and virtual production, together with the offline, high-fidelity reconstruction and performance capture that supply them with authored content; here we focus on the platform that makes such experiences run at interactive rates. Concretely, a headset must continuously answer three coupled questions, *i.e.*, where am I (6-DoF self-localization), what is around me (geometry and semantics), and how do I fuse virtual content into that reality so that it appears physically present, while simultaneously sensing the user and refreshing the display. Early systems answered the localization question

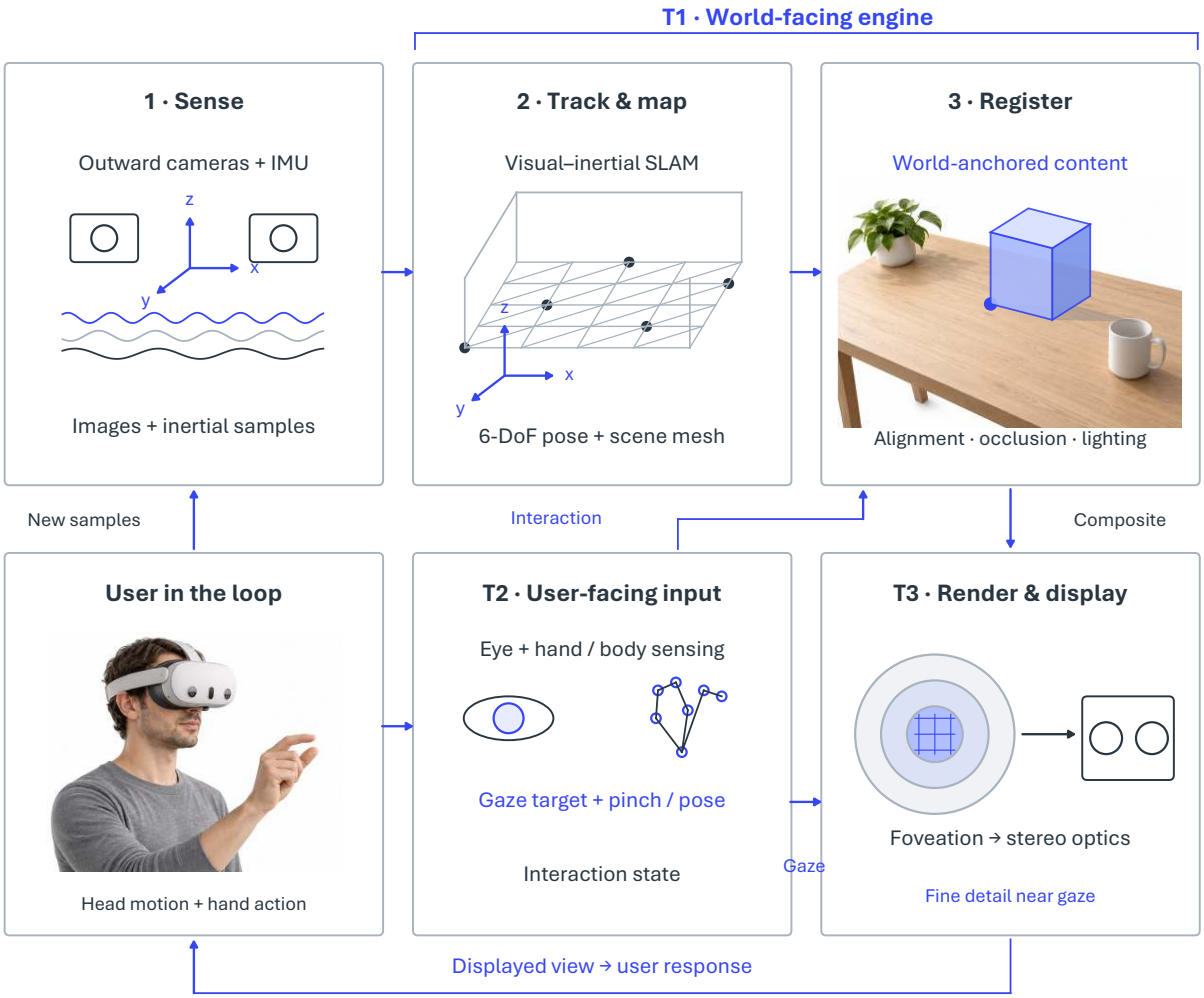


Figure 10.1: Spatial intelligence for XR as a closed real-time loop: world-facing sensing, tracking and mapping, and registration (T1), user-facing input (T2), and rendering and display (T3). The latency values are illustrative end-to-end targets, not per-stage budgets.

with outside-in tracking, using external base stations or cameras (SteamVR Lighthouse, the original Oculus constellation), but the dominant modern paradigm is inside-out tracking, in which visual–inertial SLAM runs entirely on the device, as in Meta Quest, Apple Vision Pro, and Microsoft HoloLens. The three pivotal tasks below correspond to the stages of this loop.

10.1.2 System Overview

As sketched in Figure 10.1, an XR device closes a tight real-time loop. Egocentric sensors first **sense** the device and the user (outward cameras and an inertial measurement unit for head pose; inward-facing cameras for eyes and outward- or downward-facing cameras for hands and body). The system then **tracks and maps** the environment, recovering a 6-DoF pose and a persistent geometric model, and **registers** virtual content into that model with correct anchoring, occlusion, and lighting. Finally, it **renders and displays** the composite back through the headset optics, after which the user moves and the loop repeats. The pipeline shares an end-to-end motion-to-photon budget; roughly 20 ms at a display refresh of 90 Hz or higher are the commonly cited targets, though the right values depend on device and application, and exceeding them degrades presence and comfort. The three pivotal tasks map onto this loop directly: Task-1 is the world-facing engine

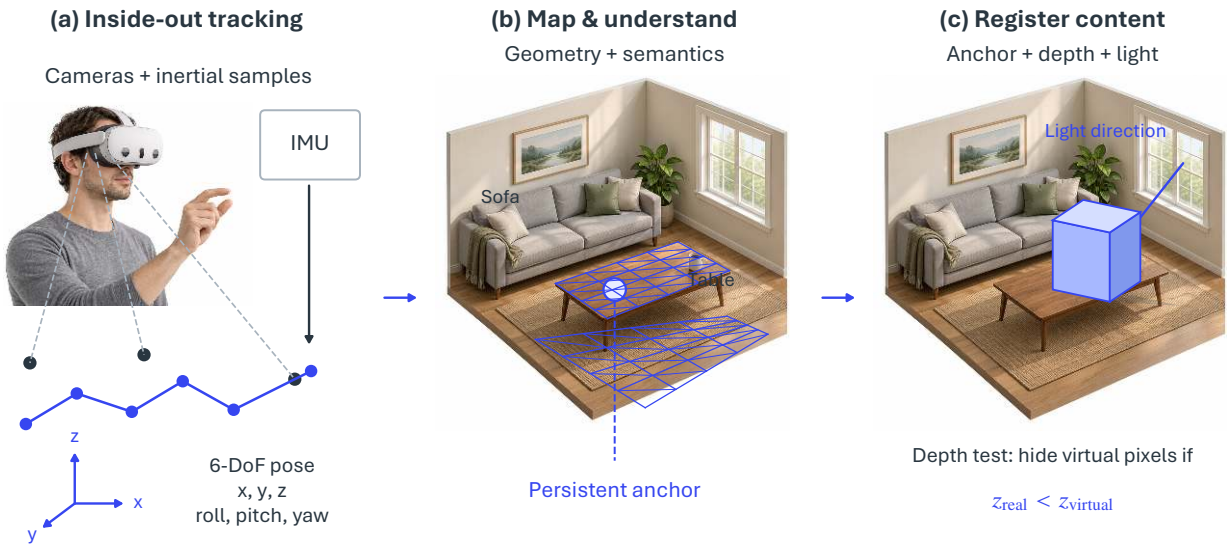


Figure 10.2: The XR world-facing engine: (a) inside-out visual–inertial tracking of 6-DoF pose, (b) mapping of geometry, semantic labels, and persistent anchors, and (c) registration of virtual content with depth-tested occlusion and matched lighting. The room and shading are illustrative, not measured reconstructions.

(track, map, register), Task-2 is the user-facing input (egocentric sensing and interaction), and Task-3 is the user-facing output (display and rendering).

10.1.3 Pivotal Task-1: Spatial Tracking, Mapping, and Registration

The first task is the medium's spatial backbone: it answers where the device is, what surrounds it, and how virtual content is pinned to the world so convincingly that the illusion holds. It proceeds in three steps: localize the device, model the environment, and register synthetic content into that model. Figure 10.2 sketches this progression from inside-out tracking to a persistent scene map and finally to anchored, occluded, and relit virtual content.

Inside-out 6-DoF tracking. The foundational layer estimates the headset's six-degree-of-freedom pose at high rate by fusing visual features with inertial measurements. Feature-based visual SLAM with loop closure and relocalization remains the academic standard, exemplified by the ORB-SLAM lineage and its visual-inertial extension ORB-SLAM3 [736], while tightly-coupled visual–inertial odometry such as VINS-Mono [2644] underpins many mobile and headset stacks; direct (photometric) methods such as DSO [719] and the earlier LSD-SLAM [718] offer an alternative that does not depend on sparse keypoints. More recent learned dense systems, including DROID-SLAM [95] and transformer-based geometric matchers [728], trade computation for robustness in low-texture and fast-motion regimes. These methods are evaluated chiefly on EuRoC [733] and TUM-VI [935], with absolute trajectory error (ATE) as the headline metric, but the binding constraints in XR are drift accumulation, recovery from tracking loss (the kidnapped-device problem), and the power and thermal envelope of on-device inference.

Spatial mapping and scene understanding. The tracked stream is consolidated into a usable model of the environment: dense depth and mesh, planar surfaces (floor, wall, table), and semantic labels. Real-time volumetric fusion in the KinectFusion tradition [114] provides the geometric substrate, on top of which plane and mesh extraction expose the surfaces that gameplay and user interfaces attach to. (The offline, high-fidelity reconstruction of such scenes as navigable content is treated in §10.2 and Chapter 8; here the map is a means

to localization and registration, not an end in itself.) Semantic and instance understanding is benchmarked on indoor datasets such as ScanNet [448] and Matterport3D [340], with mean intersection-over-union (mIoU) for segmentation, and increasingly on mobile-captured data with laser ground truth such as ARKitScenes [669] and high-quality reconstructions such as Replica [339] for depth and geometry evaluation.

Persistent and shared anchors. For content to survive across sessions and to be experienced jointly by co-located users, the map must be persistent and shareable. Cloud-hosted and peer-to-peer world maps let a device relocalize against a previously built model, and at city scale a Visual Positioning System anchors content to real landmarks in well-mapped areas. The open problems are relocalization under appearance change (lighting, season, rearranged furniture), the memory footprint of on-device maps, and the privacy implications of uploading scans of private spaces, a thread we return to in the challenges.

Registration and realistic compositing. Finally, virtual content must be registered both geometrically and photometrically so that it appears to occupy the scene. Geometric registration pins objects to anchors and planes in the SLAM frame; depth-based occlusion lets real geometry correctly hide virtual pixels, as in the per-pixel depth produced for mobile interaction by DepthLab and the ARCore Depth API [2645]; and lighting estimation predicts the dominant light direction and an ambient environment map from a single image so that virtual objects are shaded and shadowed consistently [2646]. Small errors here are perceptually unforgiving: a few centimeters of depth error at an object boundary, or a mismatched shadow, immediately reveals the composite as synthetic, which is why robust occlusion and real-time inverse lighting remain active research problems.

10.1.4 Pivotal Task-2: Egocentric Human Sensing and Spatial Interaction

The second task closes the loop from the user's body back into the system: it senses the hands, eyes, and posture from inside-out cameras and turns those signals into spatial intent. Low-level egocentric estimates of hands, gaze, and body are fused into selection, manipulation, and locomotion. As Figure 10.3 indicates, the same headset sensors that localize the device are repurposed to track the user, and the resulting estimates drive a gaze-and-gesture interaction model.

Hand tracking. Articulated 3D hand pose (commonly 21 joints, often a full mesh) is regressed from headset cameras under heavy self-occlusion and truncation. The standard hand model is MANO [2647], and on-device pipelines such as MediaPipe Hands [2648] combine a palm detector with a landmark regressor to run in real time. Methods are benchmarked on FreiHAND [561] and the two-hand InterHand2.6M [562], with 3D mean per-joint position error (MPJPE) in millimeters as the headline metric; robust two-hand interaction and egocentric truncation remain the hard cases.

Eye and gaze tracking. Estimating 3D gaze from infrared eye images serves three roles: calibration, gaze as a pointing modality, and as the driver for foveated rendering (Task-3). Appearance-based estimation in the iTracker and GazeCapture tradition [2649] established that gaze can be regressed from commodity cameras, and headset-specific datasets such as OpenEDS [2650] provide near-eye infrared imagery with dense eye-region segmentation, with the later OpenEDS2020 track adding gaze prediction (angular error in degrees). The decisive interaction consequence is that gaze targets quickly but lacks a natural trigger, a long-standing difficulty known as the Midas-touch problem, which motivates the gaze-plus-pinch pattern discussed below.

Full-body pose from sparse signals. Driving a full-body avatar from a headset and two controllers is an underconstrained inverse problem, since the lower body is unobserved and must be synthesized. Recent methods infer plausible full-body motion from these three tracked points, including the transformer-based AvatarPoser [2651] and the diffusion-based AGRoL [2652], while a parallel line recovers pose from sparse body-worn inertial sensors, as in DIP [2441], TransPose [2444], and the physics-aware PIP [2452]. These models are trained on large motion corpora such as AMASS [2209] and evaluated with MPJPE and temporal jitter. This sparse, live setting is distinct both from the wearable-IMU body sensing of §9.3 (which instruments

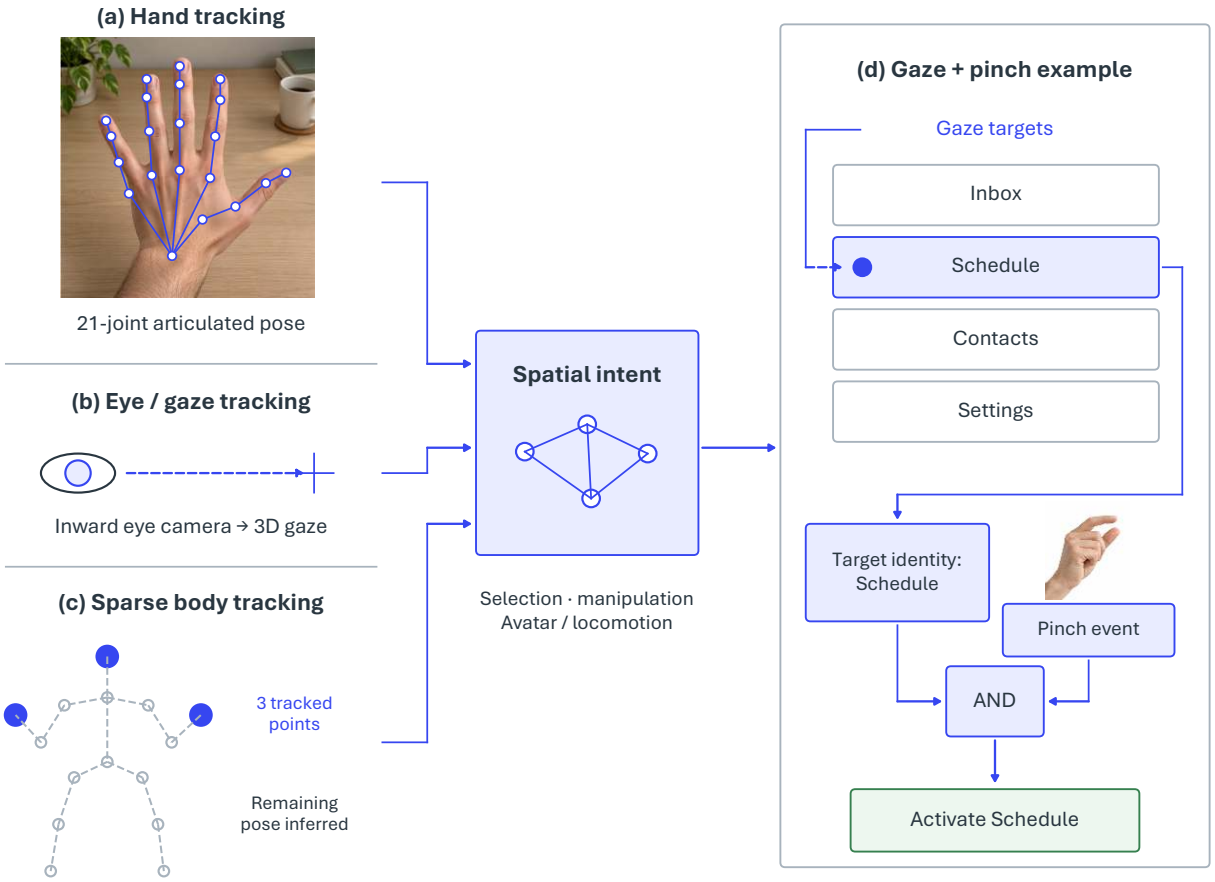


Figure 10.3: Egocentric human sensing for XR interaction: (a) articulated hand tracking, (b) gaze from inward-facing eye cameras, (c) full-body pose inferred from three tracked points, and (d) gaze-plus-pinch selection, in which gaze supplies the target and a pinch confirms it.

the body more densely) and from the studio performance capture that drives authored avatars in §10.2; the XR-specific challenge is plausibility from radically incomplete observation.

Multimodal interaction and locomotion. The recovered signals are fused into spatial actions. The foundational idea of combining a deictic gesture with a complementary modality dates to Bolt's "Put-That-There" [2653], and the modern instantiation decouples targeting from confirmation: the eyes select while a hand pinch or a voice command commits, sidestepping the Midas-touch problem. Locomotion through a virtual space larger than the physical room is handled by redirected walking, which applies imperceptible rotational and translational gains to steer the user within a confined area [2654], with teleportation as the common non-natural fallback. Spatial user interfaces, in turn, must choose a reference frame (world-locked, body-locked, or head-locked) and increasingly anchor panels to scene semantics so that content lands on appropriate surfaces. Learning these interaction behaviors at scale is increasingly grounded in large egocentric corpora such as Ego4D [681], which supply the first-person video of hands, objects, and activities that egocentric-understanding models are trained and benchmarked on.

10.1.5 Pivotal Task-3: Spatial Display and Real-Time Rendering

The third task presents the composited world back to the user within the perceptual budget that the previous two stages must respect. Three questions decide it: where to spend rendering effort, how to hide system latency, and how to display the result comfortably. Figure 10.4 previews this pipeline, from gaze-contingent shading through latency compensation to the optics that place the image before the eye.

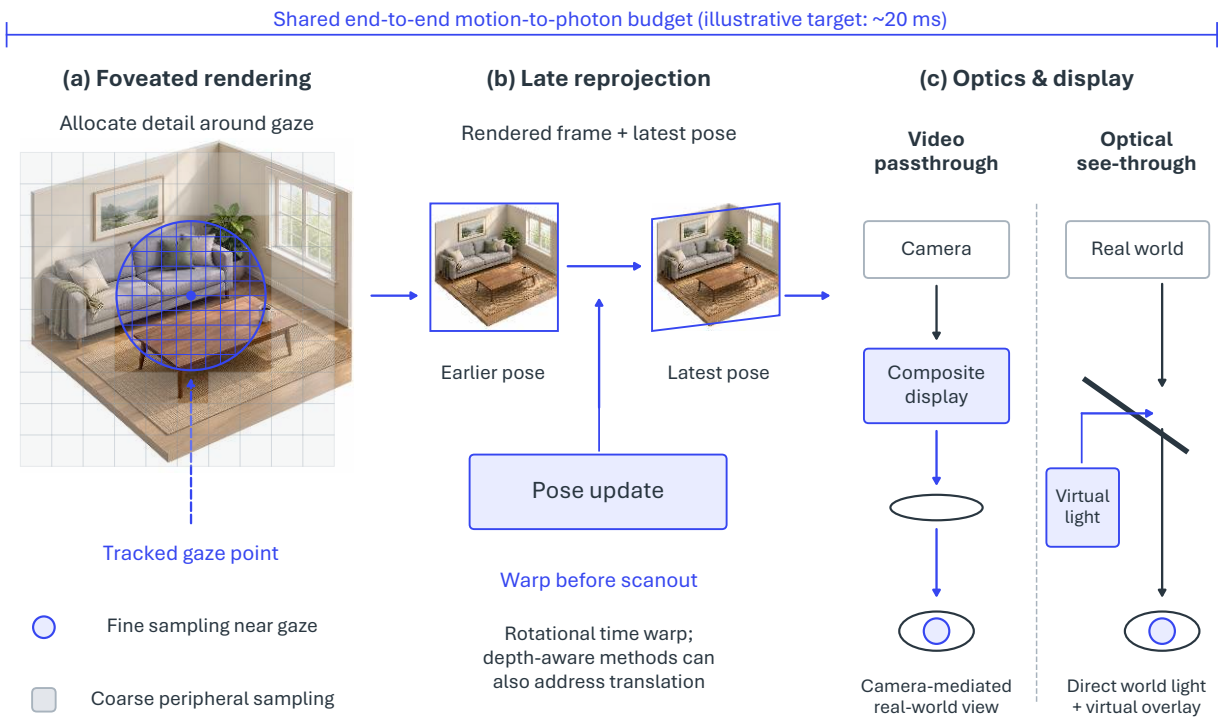


Figure 10.4: The XR display and rendering pipeline: (a) gaze-contingent foveated rendering, (b) late reprojection of a rendered frame to the latest pose, and (c) alternative video-passthrough and optical see-through display paths. The motion-to-photon target is illustrative and applies end to end.

Foveated rendering. Because acuity falls off sharply outside the fovea, rendering effort can be concentrated at the gaze point. Foveated 3D graphics established multi-resolution shading driven by eye tracking [2655], and subsequent work added temporal stability and contrast preservation to make the quality reduction imperceptible [2656]. Eye-tracked foveated rendering now ships on consumer headsets, materially reducing shading cost on thermally constrained hardware.

Latency compensation. Even an efficient renderer cannot meet the motion-to-photon budget on every frame, so the last stage warps the rendered image to the most recent head pose. Asynchronous time warp applies a rotational reprojection at display refresh, asynchronous space warp adds depth-aware frame extrapolation, and head-mounted optical-see-through systems use per-pixel late-stage reprojection [2657]. These techniques keep the world stable under fast head motion and are essential to comfort.

Passthrough versus optical see-through. Two display regimes trade off differently. Video passthrough mixed reality (as on Meta Quest 3 and Apple Vision Pro) renders the real world from cameras, granting full control over occlusion and relighting at the cost of camera-to-display latency, parallax, and limited dynamic range; optical see-through (as on HoloLens 2 and Magic Leap 2) presents the real world with zero latency at its true focal depth (though the virtual overlay still incurs render latency and a fixed focal plane) but cannot occlude real light or match real-world brightness. The choice propagates back into the registration problem of Task-1.

Depth-correct rendering and visual comfort. Conventional stereo displays fix the focal plane while vergence varies with content, producing the vergence-accommodation conflict and the eye strain that limits session length. Varifocal, light-field, and holographic near-eye displays aim to restore correct focus cues [2658], but remain largely at the prototype stage.

Neural rendering for telepresence. Finally, photoreal social presence requires driving a faithful avatar of a remote user from headset-mounted cameras that see only oblique crops of the face. The codec-avatar line,

beginning with deep appearance models [2659], learns a person-specific decoder that is driven on-device, a direct specialization of the neural rendering of Chapter 8 to the constrained, real-time XR setting.

Table 10.1: Representative benchmarks for spatial intelligence in XR, spanning the three pivotal tasks.

Pivotal task	Benchmark	Representative method	Headline metric
Tracking	EuRoC [733]	ORB-SLAM3 [736]	ATE
Tracking	TUM-VI [935]	VINS-Mono [2644]	ATE
Mapping	ScanNet [448]	dense fusion / semantics	mIoU
Mapping	ARKitScenes [669]	mobile RGB-D mapping	depth abs-rel
Hand tracking	InterHand2.6M [562]	InterNet [562]	3D MPJPE
Body (sparse)	AMASS [2209]	AvatarPoser [2651]	MPJPE / jitter
Gaze	OpenEDS [2650]	appearance-based gaze regressor	angular error
Interaction	Ego4D [681]	egocentric understanding	task-specific

10.1.6 Established Tools and Systems

A mature ecosystem implements these tasks across mobile AR, standalone headsets, and the web, and it is organized into developer frameworks, cross-platform standards, hardware, and the companies that drive them.

SDKs and standards. At the device level, **Apple's ARKit**[1] and **Google's ARCore**[2] expose plane detection, anchors, depth, light estimation, and people occlusion, with ARCore's Geospatial API localizing against world imagery for outdoor anchoring; **Vuforia**[3] specializes in robust image and object targets for industrial AR. Cross-engine abstraction layers such as **Unity AR Foundation**[4] and the **Meta Presence Platform and Spatial SDK**[5] present the common subset (planes, anchors, meshes, occlusion, hand and body tracking) over multiple devices, while **8th Wall and WebXR**[6] bring markerless AR into the browser. These abstractions matter because XR content is expensive to author and must migrate across devices and sessions; this is the role of cross-platform standards, principally **OpenXR**[7] as a unifying runtime API and glTF together with OpenUSD and **USDZ**[8] as asset and scene-description formats. Without a shared runtime and shared asset formats, the ecosystem fragments into incompatible walled gardens.

Hardware. Current headsets divide along the passthrough versus optical-see-through axis. **Apple Vision Pro**[9] (2024) and **Meta Quest 3 and 3S**[10] use high-resolution video passthrough, with the Quest line the consumer volume leader; **Microsoft HoloLens 2**[11] (optical see-through, with production ended in 2024) and **Magic Leap 2**[12] target enterprise, and **Varjo**[13] and **Pico**[14] serve high-fidelity simulation and the Asian consumer market respectively. Lightweight true-AR glasses, such as **Snap Spectacles**[15] and **Meta's Orion and Project Aria** research prototypes, signal the all-day-glasses horizon toward which the field is converging.

[1] https://developer.apple.com/augmented-reality/
[2] https://developers.google.com/ar
[3] https://developer.vuforia.com
[4] https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.0/
[5] https://developers.meta.com/horizon/develop/spatial-sdk
[6] https://immersiveweb.dev
[7] https://www.khronos.org/openxr/
[8] https://openusd.org
[9] https://www.apple.com/apple-vision-pro/
[10] https://www.meta.com/quest/
[11] https://www.microsoft.com/en-us/hololens/
[12] https://www.magicleap.com
[13] https://varjo.com
[14] https://www.picoxr.com
[15] https://www.spectacles.com

Companies and platforms. Meta leads consumer volume and the most aggressive glasses research bet; **Apple** defines the premium spatial-computing tier and the USD content pipeline; **Google** supplies world-scale localization through ARCore and Android XR; **Microsoft** has pivoted from first-party HoloLens hardware toward OpenXR software and enterprise cloud services; **Niantic**[16] operates a city-scale Visual Positioning System for persistent, multi-user AR; and Snap, Varjo, and ByteDance's Pico round out the social-AR, high-fidelity, and consumer-standalone niches.

10.1.7 Challenges

XR runs every spatial capability egocentrically and on-device under a perceptual real-time budget, and its open problems follow from that constraint: holding the motion-to-photon budget, fitting perception into the compute and thermal envelope of glasses, keeping tracking and registration robust in the wild, sharing persistent maps without compromising privacy, and making extended wear visually comfortable.

Latency and the Motion-to-Photon Budget. Meeting the motion-to-photon targets of the System Overview is increasingly difficult as pipelines add wireless or split rendering, cloud offload, and learned perception in the loop, each of which contributes its own delay that the reprojection stage of Task-3 can only partially mask.

On-Device Compute, Thermal, and Power. All-day glasses face a hard energy and heat envelope: a lightweight form factor leaves little room for the silicon that SLAM, hand and eye tracking, and scene understanding demand. The frontier is partitioning perception across on-device accelerators, companion devices such as wristbands or pucks, and edge offload, without violating the latency or thermal budget.

Robust Tracking and Registration in the Wild. Visual–inertial SLAM and relocalization degrade in dynamic scenes, textureless rooms, repetitive structures, low light, and outdoors, exactly the conditions that curated benchmarks such as EuRoC and ScanNet underrepresent [733, 448]. Persistent Visual Positioning Systems push toward city-scale anchoring, but globally drift-free, cross-device localization, together with convincing occlusion and lighting registration, remains unsolved.

Persistent, Shared Maps and Privacy. Co-located multi-user AR requires a shared, persistent spatial map, an "AR cloud", which collides directly with privacy: always-on egocentric cameras continuously scan private spaces and bystanders. Designing maps that support shared anchoring while minimizing the retention of raw imagery, and that admit meaningful consent, is both a research and a policy frontier.

Visual Comfort. Beyond latency, fixed-focus stereo optics decouple vergence from accommodation, causing eye strain that limits session length, and poorly designed locomotion induces cybersickness. Varifocal and holographic displays, gaze-contingent depth-of-field, and accurate interpupillary-distance matching are promising but immature, leaving comfort a gating factor for mainstream adoption.

Because XR subjects the perception, reasoning, and generation capabilities of Part III to egocentric, on-device, real-time constraints, it exposes failure modes that offline settings rarely reveal.

[16] https://www.nianticspatial.com

10.2 Entertainment

10.2.1 Introduction

In entertainment, spatial intelligence is the ability to perceive and reconstruct 3D environments, objects, and human bodies, and to embed virtual content into the physical world so that it is geometrically and physically consistent. Spatial entertainment (AR games, VR experiences, location-based attractions, and virtual production for film) differs from screen-based media in that it must compute camera pose, scene geometry, and user state continuously and within strict latency limits, without sacrificing aesthetic quality or narrative coherence; the perception and reconstruction machinery involved is treated in Chapter 6 and Chapter 8.

The field has followed the same arc as the capabilities it borrows: commodity depth sensing made real-time dense mapping practical for AR games, neural radiance and Gaussian representations made captured sets photoreal and navigable, parametric body models made full-body avatars routine, and world-scale visual positioning let location-based games such as Pokémon GO anchor characters to real landmarks across sessions and players. Each of these threads returns in the pivotal tasks below.

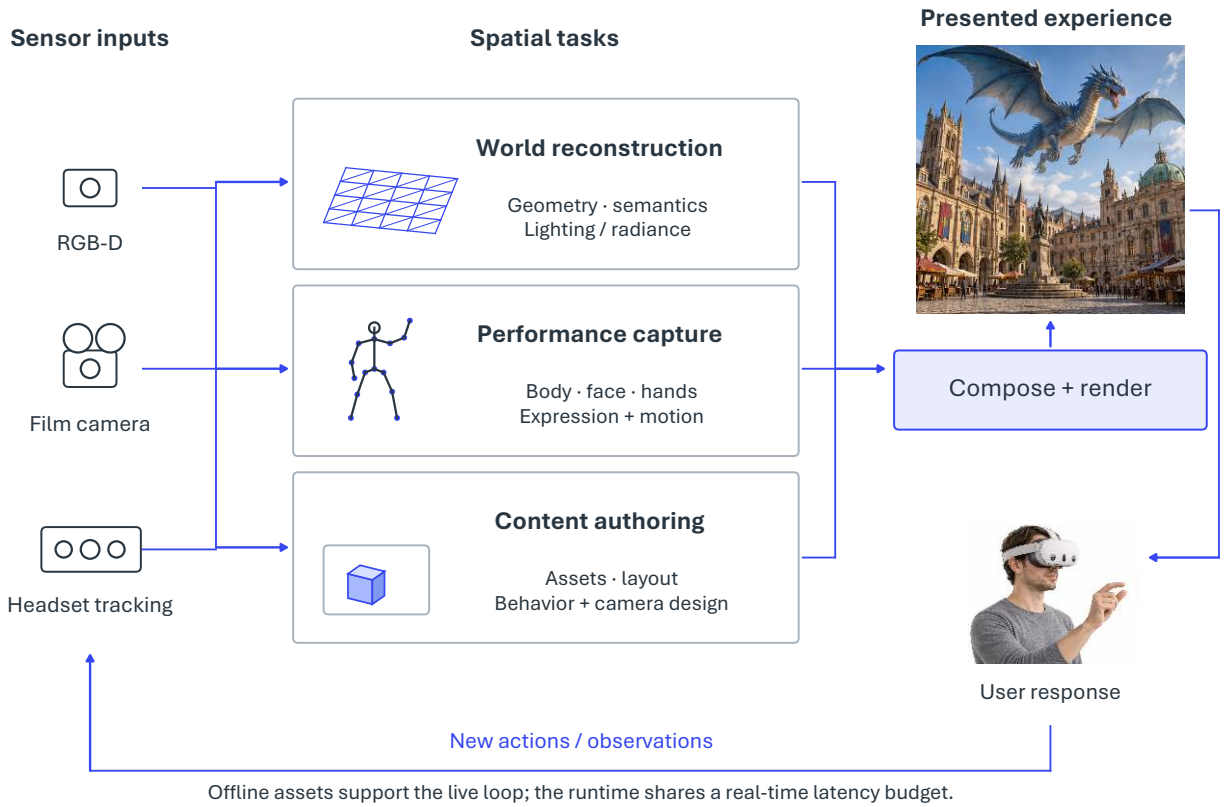


Figure 10.5: Spatial intelligence for entertainment as a real-time closed loop: sensor inputs feed world reconstruction, performance capture, and content authoring, whose composited render is presented back to the user.

10.2.2 System Overview

Entertainment sits on top of the XR platform of §10.1: it does not re-implement that runtime but builds on it, and its distinctive demand is production-quality spatial content, whether rendered live or authored offline. The domain spans two regimes. Real-time interactive experiences (AR games, VR titles, location-based

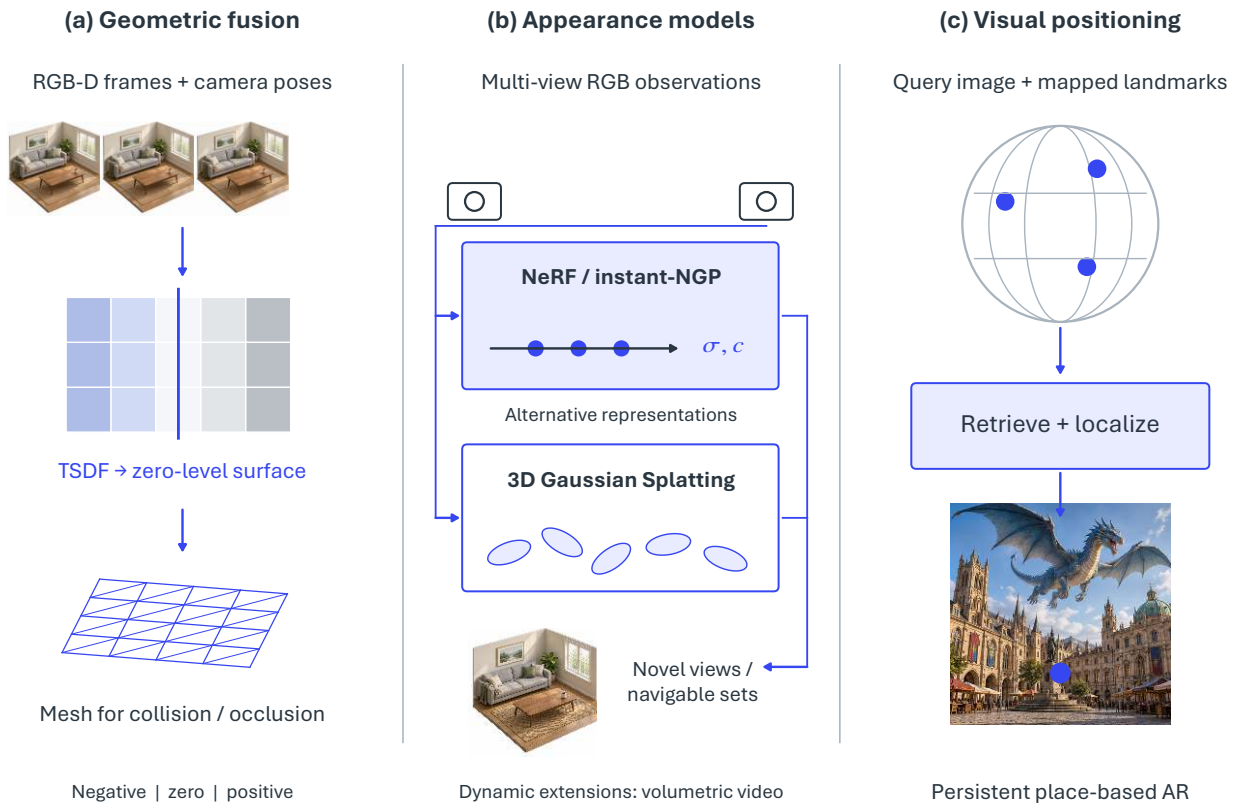


Figure 10.6: Complementary capabilities for entertainment reconstruction, not sequential stages: (a) TSDF fusion of posed RGB-D frames into a mesh, (b) radiance-field or 3D Gaussian appearance models for view synthesis, and (c) visual positioning for persistent place-based content. Samples, surfaces, and place markers are schematic.

attractions, and live virtual production) close the tight sensing-to-display loop sketched in Figure 10.5, whereas offline high-fidelity creation (scanning sets, capturing performers, building volumetric footage) supplies those experiences with assets. Across both, three spatial tasks recur as the frequent bottlenecks. **World reconstruction** builds the geometric and radiance model of the environment, supplying localization and occlusion for live experiences and navigable captured content for authored ones; **embodied performance capture** digitizes the user or performer into a controllable avatar; and **spatial content authoring** decides where virtual elements appear and how cameras move so that the result is both consistent and expressive. The interactive regime runs all three under hard latency budgets, while offline authoring trades that latency for fidelity.

10.2.3 Pivotal Task-1: World Reconstruction and Spatial Mapping

The first task builds and maintains an accurate, consistent model of the environment that supports localization, occlusion, and physics for live experiences and navigable content for authored ones, specializing the dense-reconstruction and scene-understanding techniques of Chapter 6 and Chapter 8 to interactive rates. Where §10.1 treated mapping as a means to on-device localization, here the reconstruction is itself the deliverable: the navigable set or stage that players and film-makers inhabit. Figure 10.6 contrasts geometric fusion, neural and point-based scene representations, and world-scale localization as complementary capabilities rather than obligatory sequential stages.

Geometric reconstruction forms the base: geometric SLAM and dense methods fuse depth or RGB data into volumetric maps, as in KinectFusion [114] and the reconstructions underlying datasets such as ScanNet [448],

while DynamicFusion extends this to moving users by tracking non-rigid deformations [2660]. **Neural and point-based representations** such as NeRF, instant-NGP, and 3D Gaussian Splatting then represent scene radiance to produce compact, high-fidelity reconstructions that render at interactive rates [119, 120, 121]; dynamic and 4D extensions that deform or time-index the primitives carry this to moving scenes, so a performance or an entire set can be captured as free-viewpoint volumetric video for later replay [2277, 191]. At **global scale**, visual positioning systems (VPS) combine local reconstructions into a worldwide map that lets a mobile device localize from a single image, as in Niantic's Lightship platform. These methods are evaluated on indoor datasets such as ScanNet and Matterport3D, which provide large-scale reconstructions with semantic labels for scene understanding [448, 340], and on NeRF-style view-synthesis benchmarks (*e.g.*, synthetic Blender scenes) for photometric fidelity; the latter, however, rarely capture entertainment-specific constraints such as rapid camera motion or cluttered living rooms, so practitioners often fall back on custom internal datasets (Table 10.2). The standard metrics are absolute trajectory error (ATE) for localization, surface reconstruction error for geometry, and PSNR/SSIM for neural rendering [119, 121], but in entertainment these are complemented by interaction-centric measures: the accuracy of occlusion masks, the physical plausibility of collisions, and user reports on realism (for instance, whether a virtual character convincingly hides behind a sofa).

10.2.4 Pivotal Task-2: Embodied Performance Capture and Avatar Control

The second task estimates full-body pose, shape, and facial expression to drive avatars in real time, specializing the human pose and motion machinery of Chapter 6 (and the motion-generation methods of Chapter 8) to live interaction. Unlike the sparse three-point body tracking of §10.1, driven from a headset and two controllers, entertainment capture targets studio-grade, fully-observed performance. As Figure 10.7 indicates, approaches range from skeleton-based joint prediction to surface-based model fitting and full volumetric capture.

Skeleton-based methods predict joint positions from sensors, while **surface-based** approaches fit parametric models such as SMPL directly to video [338], with datasets like Human3.6M and AMASS supplying the supervised motion priors that make this robust [550, 2209]. **Volumetric capture** systems such as Holoportation go further, streaming textured 3D avatars into VR/AR scenes by combining reconstruction with tight time synchronization [221]. More recent Gaussian-splatting avatars learn drivable, photorealistic humans from multi-view capture, with rigged 3D Gaussians giving expressive control over head avatars [2661] and pose-dependent Gaussian maps modeling full-body avatars with dynamic clothing [2662], narrowing the gap between a captured performance and a real-time renderable asset. These capabilities are benchmarked chiefly on Human3.6M, which offers millions of 3D poses with calibrated cameras [550], and on AMASS, which provides over forty hours of motion data for learning priors [2209]; commercial scenarios remain underrepresented, however, since common cases such as players using controllers in small rooms or performers in costumes lack sufficient public benchmark data (Table 10.2). Quantitative evaluation uses metrics like mean per-joint position error (MPJPE), but for interactive entertainment latency and temporal stability are decisive, as even minor jitter breaks the sense of embodiment, and user studies show that enjoyment depends on the alignment between perceived and actual motion [2663]; production systems therefore pair algorithmic metrics with feedback from expert animators.

10.2.5 Pivotal Task-3: Spatially-Aware Content Authoring and Virtual Production

The third task helps creators decide where virtual elements appear and how cameras move, turning the reconstructed world and captured performers into a coherent, expressive scene, and connecting to the spatial-reasoning and planning paradigms of Chapter 7. Figure 10.8 previews the spectrum from SDK-level authoring primitives to camera-tracked LED-wall virtual production and AI-assisted cinematography.

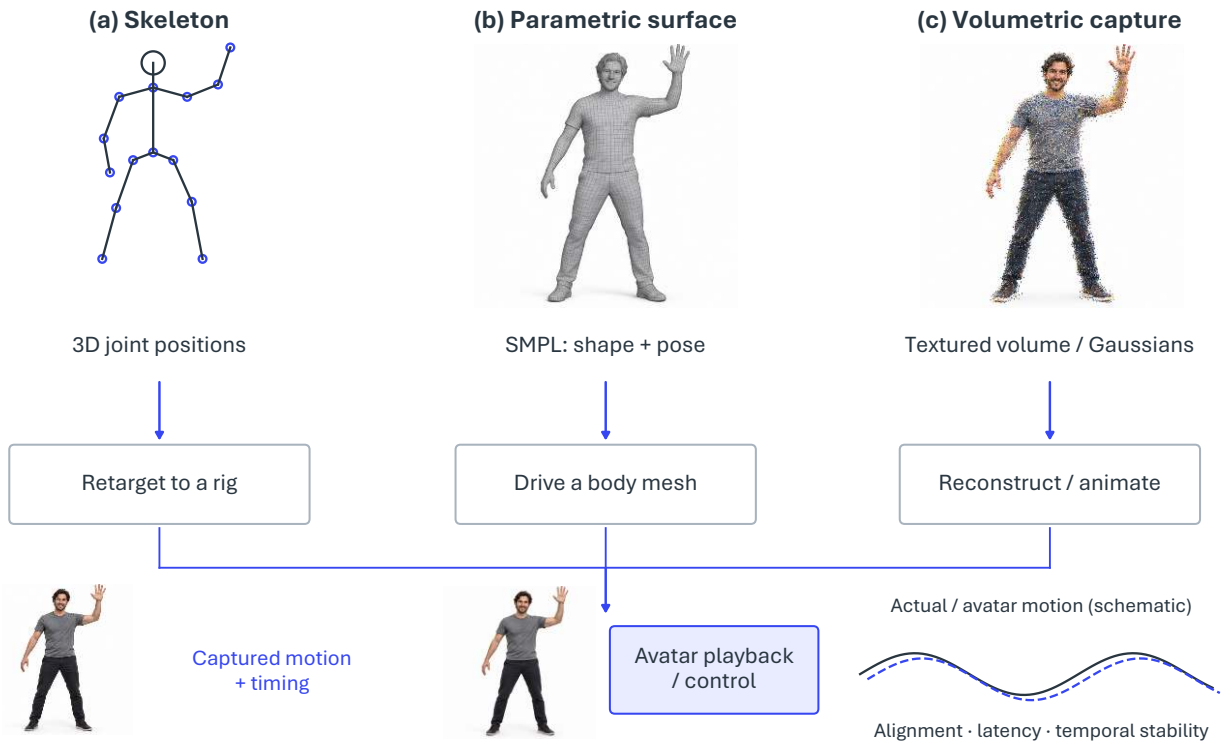


Figure 10.7: Three routes to performance capture, each able to drive avatar playback or control: (a) skeleton joints retargeted to a rig, (b) a parametric surface (SMPL) driving a body mesh, and (c) volumetric capture for reconstructed or animatable appearance. Performer illustrations and motion traces are qualitative, not measured scores.

Engine-level authoring provides designers with high-level primitives such as plane detection and anchors through AR SDKs in real-time engines like Unity and Unreal. **Virtual production** links physical cameras to 3D scenes on LED walls so that lighting and parallax respond correctly to camera movement, and **intelligent cinematography** research is exploring AI-assisted camera planning that respects film grammar while adapting to dynamic user actions [2664]. Standardized benchmarks for spatial authoring are rare: some studies evaluate camera planning on synthetic storyboards, but most assessment happens internally at studios, and virtual-production workflows are typically tested on complex reference sequences rather than public datasets, which makes objective comparison of authoring tools difficult (Table 10.2). Evaluation accordingly mixes system metrics such as rendering latency with expert judgment on cinematic quality, with intelligent-cinematography studies often convening expert panels to analyze framing and continuity [2664]; for AR/VR tools usability is paramount, and HCI evaluations measure how quickly a user completes a level-design task or how rich the resulting spatial layouts are under different tool abstractions. Generative world models are beginning to bypass authoring by synthesizing playable worlds directly: GameNGen [1818] simulates the game DOOM interactively in real time with a diffusion model, and Genie 3 [157] generates navigable environments that respond to user actions, so that creating content starts to mean conditioning a world model (§8.6).

10.2.6 Established Tools and Systems

A robust set of tools implements these tasks for games, film, and live events. At the engine level, Unity's AR Foundation[17] offers a unified API over ARKit, ARCore, and other SDKs, exposing essential features

[17] https://docs.unity3d.com/Packages/com.unity.xr.arfoundation

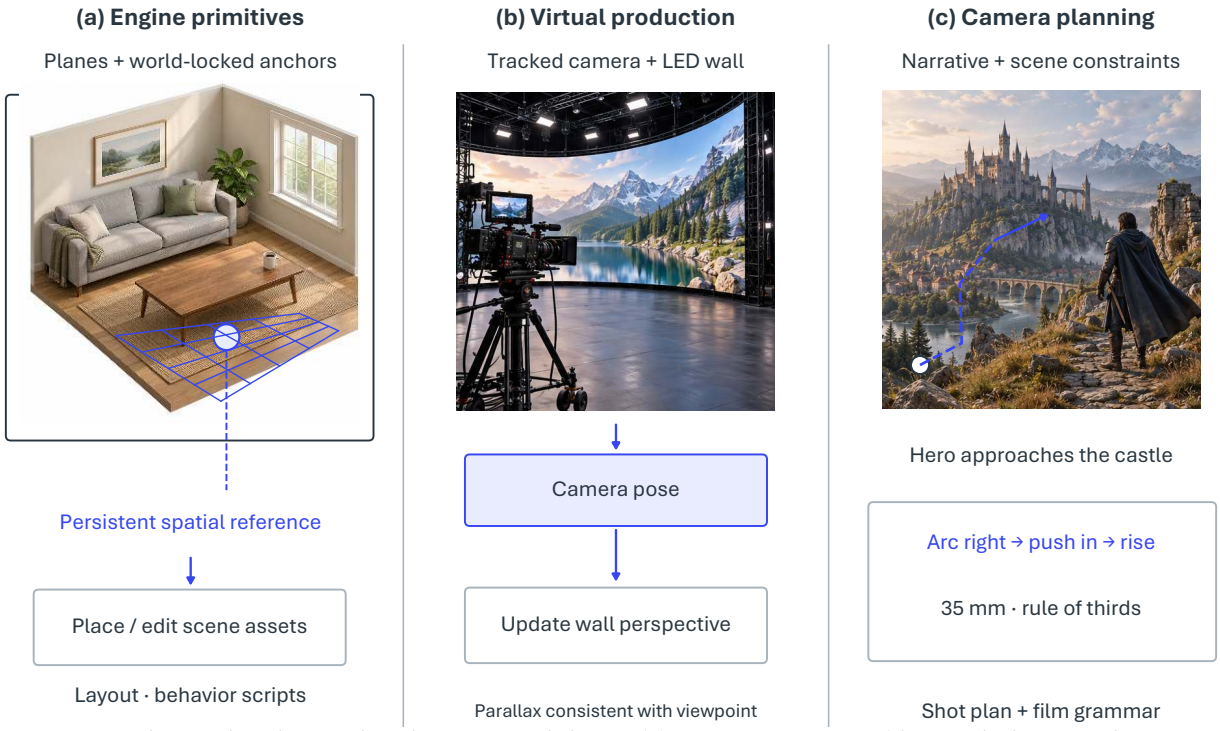


Figure 10.8: Three related spatial authoring capabilities: (a) engine primitives (detected planes and persistent anchors) for asset placement, (b) virtual production, where tracked camera pose updates the LED-wall perspective, and (c) camera planning from narrative and scene constraints to a shot plan. The scene, camera path, and shot settings are illustrative.

Table 10.2: Representative benchmarks for spatial intelligence in entertainment, spanning the three pivotal tasks.

Pivotal task	Benchmark	Representative method	Headline metric
World reconstruction	ScanNet [448]	Dense TSDF fusion [114]	recon. error
World reconstruction	Matterport3D [340]	Dense SLAM	recon. error / mIoU
View synthesis	NeRF-synthetic [119]	3D Gaussian Splatting [121]	PSNR / SSIM
Performance capture	Human3.6M [550]	SMPL fitting [338]	MPJPE
Performance capture	AMASS [2209]	Motion-prior learning	MPJPE / jitter
Content authoring	(no public benchmark)	Intelligent cinematography [2664]	expert panel / latency

such as point clouds and depth meshing, while Google's ARCore Geospatial API[18] adds global localization, allowing applications to attach anchors to outdoor landmarks with meter-level accuracy. These abstractions let designers build AR gameplay around real-world surfaces without managing low-level sensor fusion.

Niantic's Lightship platform packages spatial mapping and multiplayer synchronization into turnkey services: Reality Blending[19] in Pokémon GO uses learned depth to let virtual creatures move behind real-world obstacles, and Pokémon Playgrounds[20] extends this to persistent AR, where players leave virtual items at specific locations for others to find, with Niantic's Visual Positioning System (VPS) keeping that content anchored consistently across users and times.

In VR gaming, consumer headsets with inside-out tracking enable room-scale interaction; games like Beat Saber use precise 6-DoF tracking to create rhythm gameplay that doubles as exercise, and studies show a

[18] https://developers.google.com/ar/develop/unity-arf/geospatial/developer-guide
[19] https://nianticlabs.com/news/realityblending-announcement
[20] https://nianticlabs.com/news/pokemon-playgrounds

strong link between spatial design choices and player exertion, underscoring the connection between tracking quality and user experience [2663]. Virtual production stages have likewise transformed film and sports broadcasting: productions such as *The Mandalorian* (built on ILM's StageCraft) and sports studios such as ESPN's Catalyst Stage[21] use LED volumes driven by tracked cameras, where spatial intelligence involves calibrating physical cameras to digital scenes and maintaining low-latency tracking. Finally, neural capture tools inspired by NeRF and 3D Gaussian Splatting are entering the pipeline, letting artists scan a location with a phone and import navigable 3D assets into game engines [120, 121]; while this bridges scouting and production, it introduces new challenges around lighting control and integration.

10.2.7 Challenges

Entertainment asks for production-quality spatial content at interactive rates, and its open problems stem from that demand. The tension between real-time performance and visual fidelity is the most immediate; close behind come robustness once the system leaves controlled spaces, the cognitive load that authoring places on creators, and the human-factors and ethics questions that surface the moment these systems scan real users and real rooms.

Real-time Performance versus Fidelity. Neural representations have improved visual quality, but many methods remain too computationally heavy for mobile or standalone VR. While techniques such as 3D Gaussian Splatting achieve real-time rendering on high-end hardware [121], porting them to thermally constrained devices is difficult, and entertainment applications must additionally handle variable network conditions while guaranteeing stable frame rates, since frame drops that are acceptable in research benchmarks can cause physical discomfort in VR.

Robustness in the Wild. Existing benchmarks such as ScanNet focus on standard indoor spaces, yet entertainment experiences often occur in unpredictable environments such as festivals, dim clubs, or crowded stadiums. World-scale AR platforms address this by training on millions of crowdsourced scans, which raises questions about coverage bias (particularly toward urban areas) and the difficulty of maintaining maps as the world changes; designing models that remain robust under these shifts is an open research area.

Authoring Complexity. Although modern SDKs are powerful, creating high-quality spatial experiences imposes a high cognitive load, as creators must simultaneously manage geometry, interaction, cinematography, and technical constraints. Intelligent-cinematography tools attempt to assist by suggesting camera paths [2664], but these are not yet standard in creative workflows; there is a pressing need for tools that let designers specify high-level intentions (for example, "the character should hide behind the sofa") which the system then realizes via underlying planning modules.

Human Factors and Ethics. Spatial systems frequently scan users and private spaces. While datasets like AMASS demonstrate the utility of large-scale motion capture, collecting similar data in the wild raises privacy concerns, and from an HCI perspective issues such as cybersickness and fatigue limit the duration of VR and AR sessions, while physically active titles introduce comfort and safety considerations that spatial layout must respect. Addressing these problems requires a collaborative approach spanning algorithm design, user-feedback systems, and clear data governance.

[21] https://espnpressroom.com/us/press-releases/2023/06/catalyst-stage-disneys-new-immersive-espn-studio-opens-storyte

Chapter 11

Spatial Intelligence for Geo-spatial Systems

Geo-spatial systems operate on spatial structure at the scale of assets, cities, and the planet, where the task is to build a faithful model of a physical system and keep it aligned with live observation. This chapter applies the spatial intelligence of the previous part to that task, across three settings that share the same closed loop between model and world but differ in scale and stakes.

The digital twin binds a metric, physically grounded replica to a specific asset through a live bidirectional loop, so that the physical system and its model track each other. GeoAI widens the frame to the planet, where streams of Earth-observation data drive perception and decision-making over terrain, infrastructure, and environmental change. Aerospace systems carry the same loop off the planet, where a spacecraft or rover must perceive, reason about, and act on its surroundings with little or no ground support.

11.1 Digital Twins

11.1.1 Introduction

A **digital twin** is a 3D representation faithful enough to stand in for its physical counterpart in analysis and operation. Building one requires more than accurate geometric reconstruction and photorealistic appearance under varying illumination; it requires realistic modeling of physical properties and interaction, so that the replica behaves as the original would. What separates a digital twin from any other 3D asset, however, is not fidelity alone but *coupling*. Following the now-standard taxonomy of Kritzinger *et al*. [2665], a *digital model* carries no automated data exchange with the physical object, a *digital shadow* carries an automated one-way flow from the physical to the virtual, and only a *digital twin* closes the loop with an automated, bidirectional flow: the virtual model is continuously synchronized to the physical asset, simulated forward to predict, and used to act back upon it.

A digital twin therefore reuses, but specializes, much of the machinery surveyed earlier in this handbook. The perception stack of Chapter 6 supplies the captures from which a twin is built, and the neural reconstruction and generation methods of Chapter 8 supply its geometry and appearance; what the twin adds is metric accuracy, physical grounding, and a persistent correspondence to a specific real object over its lifecycle. Entertainment (§10.2) builds the same high-fidelity captures as creative *content*; here they become an *operational engineering model* wired to a live counterpart, and the city- and planet-scale twins that recur below connect to the GeoAI methods of §11.2. The concept itself traces from Grieves' product-lifecycle

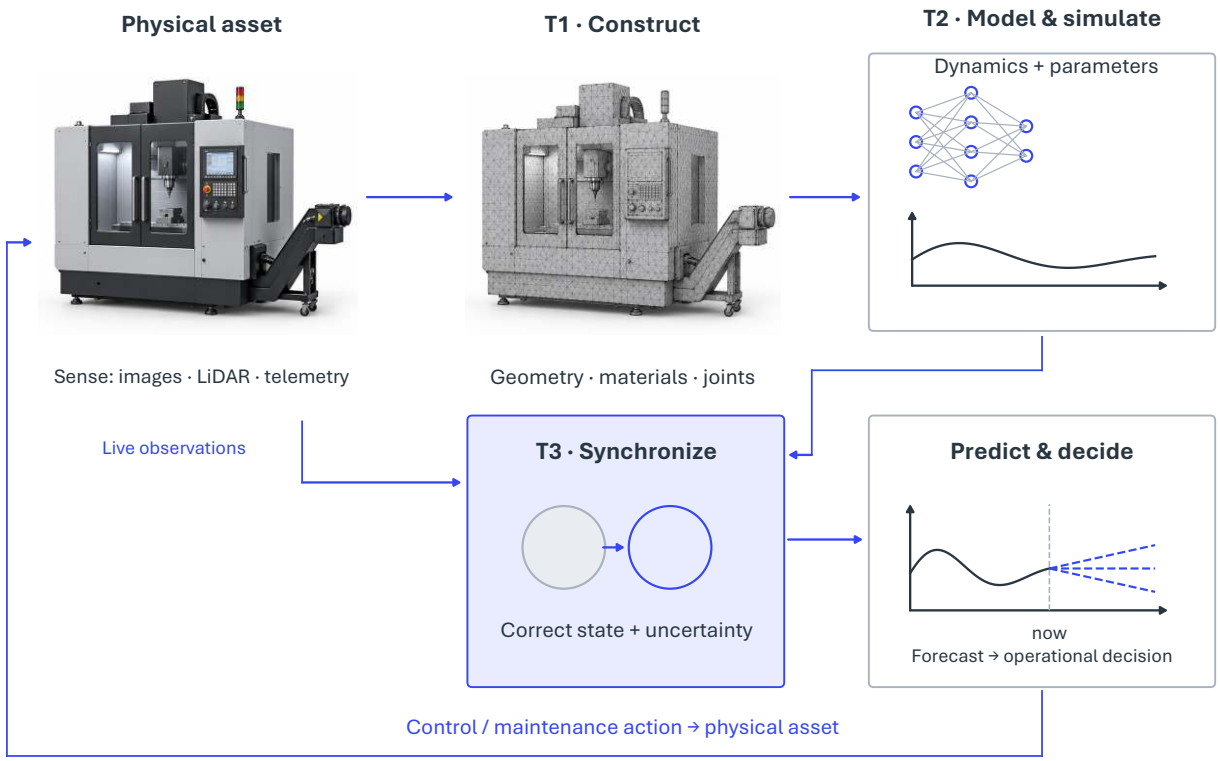


Figure 11.1: The digital twin as a closed loop around its physical asset: construction (T1) and modeling (T2) establish the replica, synchronization (T3) corrects its state from live observations, and forecasts drive decisions through a control path separate from the observation path.

lectures, formalized with Vickers [2666], through its aerospace codification for NASA and U.S. Air Force vehicles [2667], to its industrial maturation [2668, 2669], where the five-dimension model of Tao *et al*. makes the data connections between physical entity, virtual entity, and services a first-class element of the architecture.

11.1.2 System Overview

Figure 11.1 draws the twin as a closed loop around its physical asset. Sensing (images, LiDAR, RGB-D scans, onboard IoT telemetry) feeds the construction of a replica that carries geometry, appearance, materials, and physical properties; the replica is simulated forward, kept synchronized by live data assimilation, and queried to predict, and its decisions operate the asset, which is then sensed anew. The three pivotal tasks map onto this loop directly. Task-1 builds the twin (sense and construct), Task-2 makes it behave (model and simulate), and Task-3 keeps it synchronized and turns it to use (assimilate, predict, and act). Table 11.1 collects the benchmarks on which the first two are evaluated; synchronization has no shared benchmark.

11.1.3 Pivotal Task-1: High-Fidelity Twin Construction

The first task builds the twin from sensor captures, and it must deliver four properties that a generic one-shot reconstruction does not guarantee. The twin must be *metric* (correct at real-world scale), *relightable* (its appearance separable from capture-time illumination), *physically grounded* (carrying mass, material, and friction), and *actionable* (its movable parts and joints encoded so a simulator can drive it). Figure 11.2 sketches the progression from raw captures through geometry and material recovery to a simulatable model.

Geometry and appearance acquisition. The earliest replicas were built by photogrammetry: multiple overlapping photographs of an object, terrain, or structure are captured, and a Structure-from-Motion algorithm recovers camera poses and a sparse metric point cloud, as systematized in COLMAP [115], after

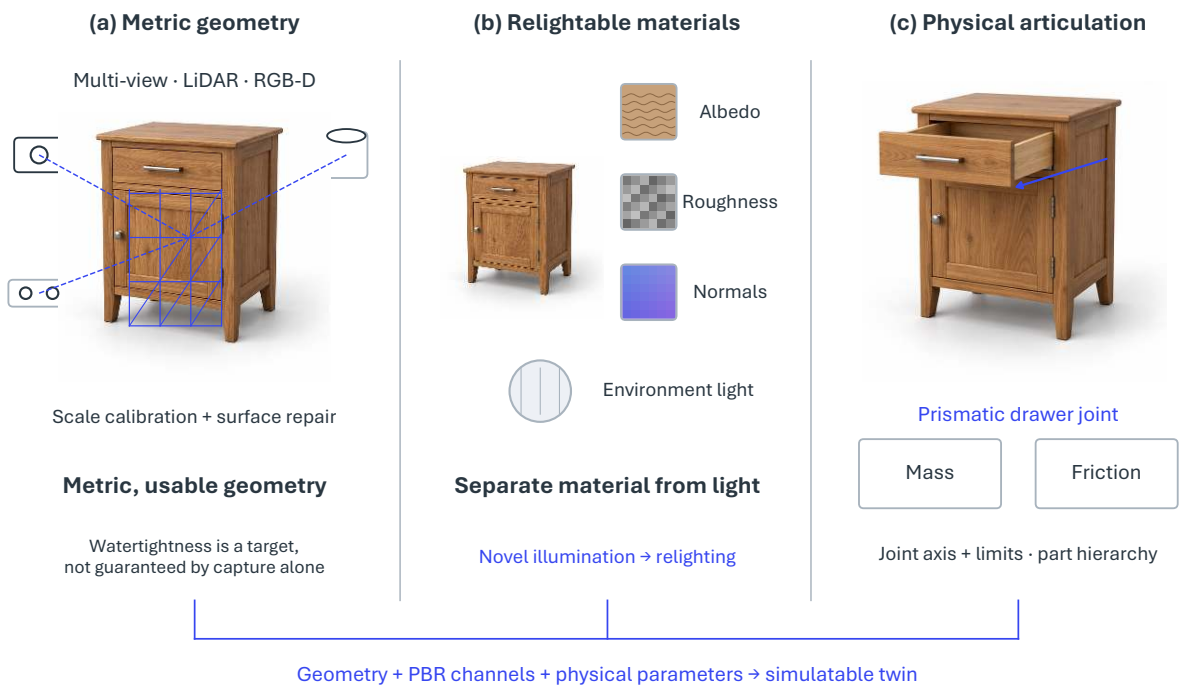


Figure 11.2: Three components of high-fidelity twin construction: (a) metric geometry from multi-view, LiDAR, or RGB-D capture; (b) relightable material channels separated from illumination; and (c) physical articulation, here a prismatic drawer joint with mass and friction. Material swatches and mesh overlays are schematic.

which dense multi-view stereo and meshing produce a surface. Real-time RGB-D fusion in the KinectFusion tradition [114] yields metric, watertight geometry over the observed region, and neural implicit-surface methods such as NeuS [288] and the detail-preserving Neuralangelo [2670] recover watertight meshes as the zero-level-set of a learned signed-distance field. A second family optimizes for view synthesis rather than surface quality: NeRF [119], its hash-grid acceleration Instant-NGP [120], and the explicit, real-time 3D Gaussian Splatting [121] render novel views with high fidelity but historically produced no clean surface and bake appearance into the representation; surface-aligned variants such as SuGaR [1845] and 2D Gaussian Splatting [1844] narrow this gap by extracting watertight meshes directly from a Gaussian field. Because a twin requires metric scale and an extractable surface, the photogrammetric and implicit-surface lines are load-bearing, with meshing operators such as Poisson surface reconstruction [298] and Marching Cubes [105] converting a field or point set into geometry; the radiance-field family enters as an appearance and speed substrate, treated in full in Chapter 8.

Inverse rendering and relightable materials. A captured asset whose appearance is baked at scan time only reproduces the object under the one illumination it was scanned in, because each pixel entangles albedo, geometry, and lighting. A twin, by contrast, lives in a simulator where viewpoint and illumination change every frame, so its construction must recover separated physically based rendering (PBR) channels (albedo, roughness, normals) together with an estimate of the environment lighting. Inverse-rendering methods supply exactly this factorization: NeRFactor decomposes a captured radiance field into shape, reflectance, and a single environment light [292], while the differentiable, engine-ready NVDiffRec jointly extracts a triangle mesh, an SVBRDF, and a lighting estimate directly optimizable for downstream rendering [679]. The same factorization has been ported onto explicit Gaussians, as in Relightable 3D Gaussians, which attaches normals and a BRDF to each primitive and ray-traces shadows for real-time relighting [1985]. Relightability

is what lets a twin be composited under novel lighting and rendered consistently inside a physics or sensor simulation.

Physical-property and articulation estimation. A static mesh is viewable but not operable: it has no notion of which surfaces move, with what effort, or under what dynamics. Closing this gap requires a movable-part inventory and the kinematic joints (revolute or prismatic, with axes and limits) that a physics engine can integrate, together with physical parameters such as mass and friction. Ditto recovers part geometry and an articulation model jointly from before-and-after observations of an interaction [2671], and URDFormer predicts a full URDF, the standard interactive description, from a single real image, turning a photograph into a simulatable articulated twin [2672]. Dynamical parameters can be identified directly in a differentiable field representation, as in PAC-NeRF, which estimates geometry and physical parameters together through a differentiable material-point simulator [2673], and PhysGaussian, which integrates Newtonian dynamics into a Gaussian representation so that what is reconstructed can also be simulated [2674]. Such methods lean heavily on articulated-object datasets, principally the PartNet-Mobility models exposed through the SAPIEN simulator [2396], the fine-grained part hierarchy of PartNet [2364], and the cross-category, actionable parts of GAPartNet [2380]. The resulting articulated, parameterized model is the input to the behavioral simulation of Task-2.

Datasets and metrics. Construction quality is measured against scanned-object and scene corpora (Table 11.1). At the object level, the recent Digital Twin Catalog provides a large set of digital-twin-quality scans with millimeter geometry, photorealistic materials, and an explicit relighting evaluation [2675], complementing the real-scanned OmniObject3D [1779], the simulation-ready Google Scanned Objects [1778], the artist-authored, PBR-annotated ABO [2676], the large Objaverse collection [1764], and the in-the-wild Common Objects in 3D [2677]. At the scene level, twins are evaluated on ScanNet [448], Matterport3D [340], the photorealistic Replica reconstructions [339], and the commodity-LiDAR ARKitScenes [669], with metric-geometry benchmarks such as Tanks and Temples supplying laser-scan ground truth [1173]. Geometry is reported with Chamfer distance, F-score at a stated threshold, and normal consistency; appearance with PSNR, SSIM, and LPIPS; and, for twins specifically, an added relighting error that penalizes appearance that fails to transfer across illumination.

11.1.4 Pivotal Task-2: Physics-Based Modeling and Behavioral Simulation

The second task makes the constructed replica behave. Geometry is not behavior: a twin must reproduce its asset's dynamics, not merely its shape, so the static model of Task-1 is coupled to a physics, behavioral, or surrogate engine and run forward as an *executable* model whose parameters are fit to the real asset (Figure 11.3).

Physics engines. The behavioral substrate of a twin is a simulator that integrates its dynamics. MuJoCo provides accurate rigid-body contact dynamics and is the de facto standard for model-based control [1187]; the open-source PyBullet offers a widely used free alternative spanning rigid and soft bodies [1188]; and SAPIEN specializes in part-level interaction with large articulated objects [2396], the natural runtime for the actionable twins of Task-1. GPU-accelerated platforms couple photorealistic rendering with physics at scale, exemplified by NVIDIA Isaac Sim[1] on the Omniverse stack and its successor Isaac Lab[2], with newer GPU-first simulators such as Genesis[3] and the accelerator-native MuJoCo MJX pushing throughput further; deformable phenomena such as cloth and fluids are handled by unified particle and material-point (MPM) solvers.

Differentiable simulation and system identification. A twin is only behaviorally faithful if its physical parameters match the real asset, which makes parameter estimation central. Differentiable simulators

[1] https://developer.nvidia.com/isaac/sim
[2] https://developer.nvidia.com/isaac/lab
[3] https://genesis-embodied-ai.github.io/

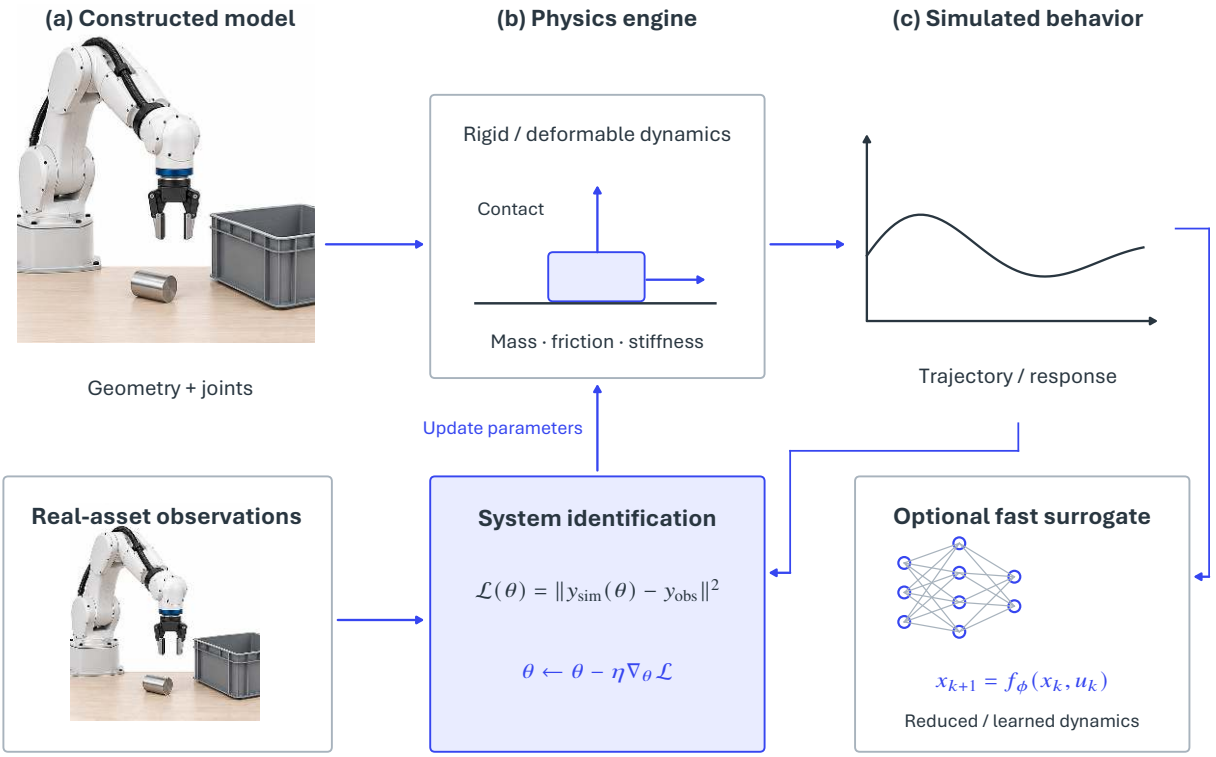


Figure 11.3: From a constructed replica to a behaving twin: (a) constructed model, (b) physics engine, and (c) simulated behavior, with system identification fitting the physical parameters θ to real-asset observations and an optional fast surrogate f_ϕ. Response traces are illustrative, not measured results.

expose gradients of simulated outcomes with respect to physical parameters, turning system identification into gradient-based optimization: DiffTaichi compiles differentiable physical simulators from high-level code [2678], Brax provides massively parallel differentiable rigid-body dynamics on accelerators [2405], and gradSim couples differentiable simulation with differentiable rendering to backpropagate from video pixels all the way to mass, friction, and elasticity [2679]. A twin can thus be calibrated to its counterpart from observation rather than hand-tuned.

Surrogate and physics-informed models. High-fidelity simulation is often too slow to run in the synchronization loop of Task-3, so twins frequently substitute a reduced or learned surrogate. Physics-informed neural networks embed governing partial differential equations directly in the training loss, solving forward and inverse problems with scarce data [2680], a strategy situated within the broader programme of physics-informed machine learning [2681]. Reduced-order modeling, identified early as a key enabling technology for digital twins, compresses a high-fidelity simulation to real-time evaluation [2682], and the hybrid-twin paradigm couples such a physics-based reduced model with a data-driven correction term to absorb unmodeled effects [2683].

Real-to-sim for policy learning. A faithful behavioral twin is also the substrate on which robot policies are trained and validated before deployment, which connects this task to the manipulation methods of §9.1 and the reasoning of Chapter 7. The reality gap between simulation and the physical world is bridged by randomizing visual appearance [2684] and dynamics [2685] during training, and recent real-to-sim-to-real pipelines build a twin of the workspace from a quick scan and use it to robustify learned policies, as in RialTo [2686]. A complementary line argues that an exact twin is not always necessary: ACDC instead generates geometrically and semantically similar "digital cousins," trading exact correspondence for robustness and zero-shot transfer [2687], sharpening by contrast what a true digital twin provides.

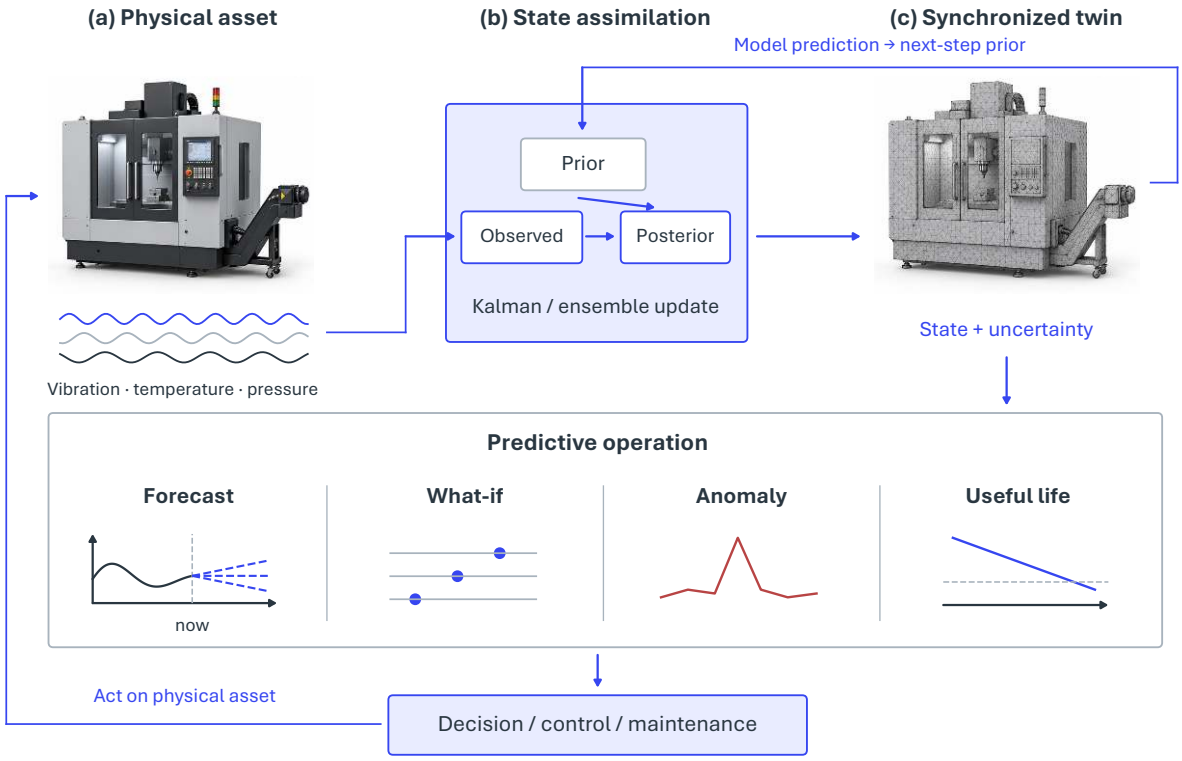


Figure 11.4: Real-time synchronization and predictive operation: (a) live measurements from the physical asset are (b) assimilated with the model prior into a posterior state of (c) the synchronized twin, which feeds forecasts, what-if, anomaly, and useful-life outputs. The upper feedback path propagates the updated state into the next model prior; the lower path returns decisions or controls to the physical asset. Dashed forecast branches are possible futures, not observed trajectories; all analytical outputs are schematic.

11.1.5 Pivotal Task-3: Real-Time Synchronization and Predictive Operation

The third task is the one that earns the name *twin*: it maintains the automated bidirectional coupling between model and asset that defines a twin [2665], and turns that coupling to use. Figure 11.4 previews the pipeline, from a live sensor stream through data assimilation to prediction and a control signal fed back to the physical asset.

Data assimilation and state estimation. Keeping the twin synchronized is a state-estimation problem: the model's latent state must be continually corrected toward the incoming measurements. The classical recursive estimator is the Kalman filter [2688], and for the nonlinear, high-dimensional systems typical of physical twins the ensemble Kalman filter provides a Monte-Carlo data-assimilation scheme that propagates an ensemble of states to forecast error statistics [2689]. At the level of an entire engineered system, a probabilistic graphical model casts data-driven model adaptation as Bayesian state estimation with end-to-end uncertainty quantification [2690], so that predictive twins can be assembled and updated at scale and their predictions carry calibrated confidence rather than point estimates.

Predictive operation across domains. The synchronized twin is queried to anticipate rather than merely to monitor, and the applications follow the asset. In manufacturing, the twin couples machine geometry to live telemetry for prognostics and health management, estimating remaining useful life and scheduling maintenance before failure [2691]. At urban scale, a city digital twin fuses a semantically rich 3D model with sensor and simulation layers to support planning and what-if analysis, as in the Herrenberg study [2692] and national programmes built on CityGML foundations [2693], a layer that connects directly to the GeoAI

methods of §11.2. In healthcare, patient-specific anatomical twins driven by physiological models support precision diagnosis and treatment planning, with cardiology an early exemplar [2694] and a growing body of work surveyed across the field [2695]; the medical twin is developed in §12.1. In aerospace, where the paradigm was first codified, a structural twin mirrors an as-flown airframe to predict fatigue [2667]; and in agriculture, field and crop twins fuse geospatial layers with growth models for precision management [2696].

Table 11.1: Representative benchmarks for twin construction and simulation. Synchronization (Task-3) has no shared benchmark.

Pivotal task	Benchmark / dataset	Representative method	Headline metric
Construction (object)	DTC [2675]	neural reconstruction + PBR	geometry + relighting error
Construction (object)	OmniObject3D [1779]	multi-view reconstruction	Chamfer / F-score
Construction (materials)	ABO [2676]	NVDiffRec [679]	material / PSNR
Construction (scene)	ScanNet [448]	KinectFusion-style RGB-D fusion [114]	Chamfer / F-score
Construction (metric)	Tanks and Temples [1173]	MVS / neural surfaces	F-score
Construction (articulation)	PartNet-Mobility [2396]	Ditto [2671] / URDFormer [2672]	joint / part error
Construction (city-scale)	GauU-Scene / MatrixCity	CityGaussianV2 [2697]	PSNR / geometry F-score
Simulation (real-to-sim)	real-world manipulation tasks	RialTo [2686] / ACDC [2687]	policy success rate

11.1.6 Established Tools and Systems

Capture, authoring, and render runtimes. The construction stage is served by spatial-capture products such as **Matterport**[4] for interiors and sites, the mobile scanner **Polycam**[5], the photogrammetry pipelines **RealityScan**[6] (formerly RealityCapture) and Agisoft Metashape[7], and, as an integrating substrate, NVIDIA Omniverse with **OpenUSD**[8], which provides the scene description and real-time rendering on which twins are assembled. Game and render engines such as **Unreal Engine**[9] and Unity, the latter packaged for industry as **Unity Industry**[10], serve as the interactive runtimes for the resulting models.

Industrial twin platforms. The operational layer is dominated by enterprise platforms that wire CAD geometry to live telemetry. **Siemens Teamcenter**[11] drives product twins from a PLM backbone; **Ansys Twin Builder**[12] couples reduced-order models to 3D physics; Dassault's 3DEXPERIENCE markets sensor-driven "virtual twin" experiences[13]; and cloud services including **PTC ThingWorx**[14], **AWS IoT TwinMaker**[15], **Azure Digital Twins**[16], and **Bentley iTwin**[17] maintain live twin graphs of factories, buildings, and infrastructure, with **GE Vernova**[18] specializing in energy-asset twins.

[4] https://matterport.com/
[5] https://poly.cam/
[6] https://www.realityscan.com/
[7] https://www.agisoft.com/
[8] https://www.nvidia.com/en-us/omniverse/
[9] https://www.unrealengine.com/
[10] https://unity.com/products/unity-industry
[11] https://plm.sw.siemens.com/en-US/teamcenter/
[12] https://www.ansys.com/products/digital-twin/ansys-twin-builder
[13] https://www.3ds.com/virtual-twin
[14] https://www.ptc.com/en/products/thingworx
[15] https://aws.amazon.com/iot-twinmaker/
[16] https://azure.microsoft.com/en-us/products/digital-twins
[17] https://www.bentley.com/software/itwin-platform/
[18] https://www.gevernova.com/software/

City-scale systems and interoperability standards. At geo-scale, procedural and streaming platforms such as **Esri ArcGIS CityEngine**[19], **Cesium with 3D Tiles**[20], **Google's photorealistic 3D Tiles**[21], and **NVIDIA Earth-2**[22] build and serve city- to planet-scale twins, with national programmes such as Virtual Singapore as flagship deployments. What keeps such twins portable across this fragmented tooling are open standards: OpenUSD for scene description, IFC (ISO 16739) for building information, CityGML for semantic city models, and OGC 3D Tiles for streaming massive geospatial geometry, alongside the framework and terminology standards ISO 23247 and ISO/IEC 30173. The data layer feeding these geo-scale twins is the subject of §11.2.

11.1.7 Challenges

A twin must be high-fidelity, live, and scalable at once, and these three demands pull against one another. The open problems follow from that tension: scaling construction from an object to a city, keeping the twin current, validating that it is accurate rather than merely photorealistic, coupling several physical domains in real time, and making twins interoperable without exposing the spaces they capture.

Scalability to Large and City Scale. Building a twin of a single object is now routine; building one of a city is not. City twins must convert heterogeneous source data into semantically structured models (CityGML levels of detail) and stream them efficiently (3D Tiles), and the reconstruction itself must scale, motivating divide-and-conquer pipelines for large-scene Gaussian splatting [2697]. Data integration and cross-format interoperability remain the named bottlenecks at this scale.

Synchronization Latency and Data Assimilation. A twin is only as useful as it is current, and maintaining real-time correspondence across an edge-fog-cloud hierarchy is hard: assimilation must keep pace with the asset's dynamics, and the "liveness gap" between the asset's true state and the twin's estimate degrades every downstream prediction. Scaling data assimilation to high-dimensional physical systems without violating the update budget is an open problem.

Model Fidelity, Validation, and the Photorealism-Accuracy Tradeoff. Verification, validation, and uncertainty quantification are the maturity frontier for twins used in consequential decisions. A specific tension recurs: methods such as vanilla Gaussian splatting optimize visual fidelity, not metric accuracy, so a photorealistic reconstruction can still fail a metric tolerance check; surface-aligned variants [1845, 1844] narrow but do not close this gap.

Multi-Physics and Multi-Scale Coupling. Real assets couple mechanics, thermics, fluids, and electromagnetics across scales, and no universal model spans them; coupled systems risk solver instability, and the reduced-order and physics-informed surrogates [2682, 2680] that make twins real-time must do so without discarding the cross-domain interactions that matter.

Standardization, Interoperability, Privacy, and Security. The proliferation of scene and asset formats (USD, IFC, CityGML, 3D Tiles) keeps twins from composing across vendors, and the always-on capture that a live twin requires raises acute concerns: scanning private interiors and urban space implicates bystanders, and a compromised twin server can leak an asset's dynamics, configuration, and credentials [2669]. Designing twins that interoperate while minimizing retained raw data is both a research and a policy frontier.

[19] https://www.esri.com/en-us/arcgis/products/arcgis-cityengine/overview
[20] https://cesium.com/
[21] https://developers.google.com/maps/documentation/tile/3d-tiles
[22] https://www.nvidia.com/en-us/high-performance-computing/earth-2/

11.2 GeoAI

11.2.1 Introduction

Geospatial artificial intelligence (GeoAI) applies machine learning to geographic data (satellite and UAV imagery, LiDAR, GNSS traces, and IoT sensor streams) to model and act on processes that unfold over terrain, cities, and the Earth system. It makes the geospatial domain a large-scale deployment target for the perception and scene-understanding capabilities surveyed in Part III: the detection, segmentation, and remote-sensing techniques of Chapter 6 (and the aerial and remote-sensing setting of Section 9.4) are here specialized to Earth-observation data and decision workflows at planetary scale.

11.2.2 System Overview

A GeoAI workflow couples heterogeneous geospatial data to models that turn raw observations into spatially explicit decisions (Figure 11.5). Following the application taxonomy of the review by Boutayeb *et al*. [2698], we treat six domains as pivotal tasks: precision agriculture, the urban built environment, environmental monitoring, water resources, disaster management, and public health. The six share their inputs, largely the free Sentinel, Landsat, and MODIS archives, and until recently they shared a modeling pattern as well: a task-specific supervised network or tree ensemble trained for one region and one target, which the foundation models at the end of the section are beginning to replace. Table 11.2 samples representative data sources, methods, and reported results across the six.

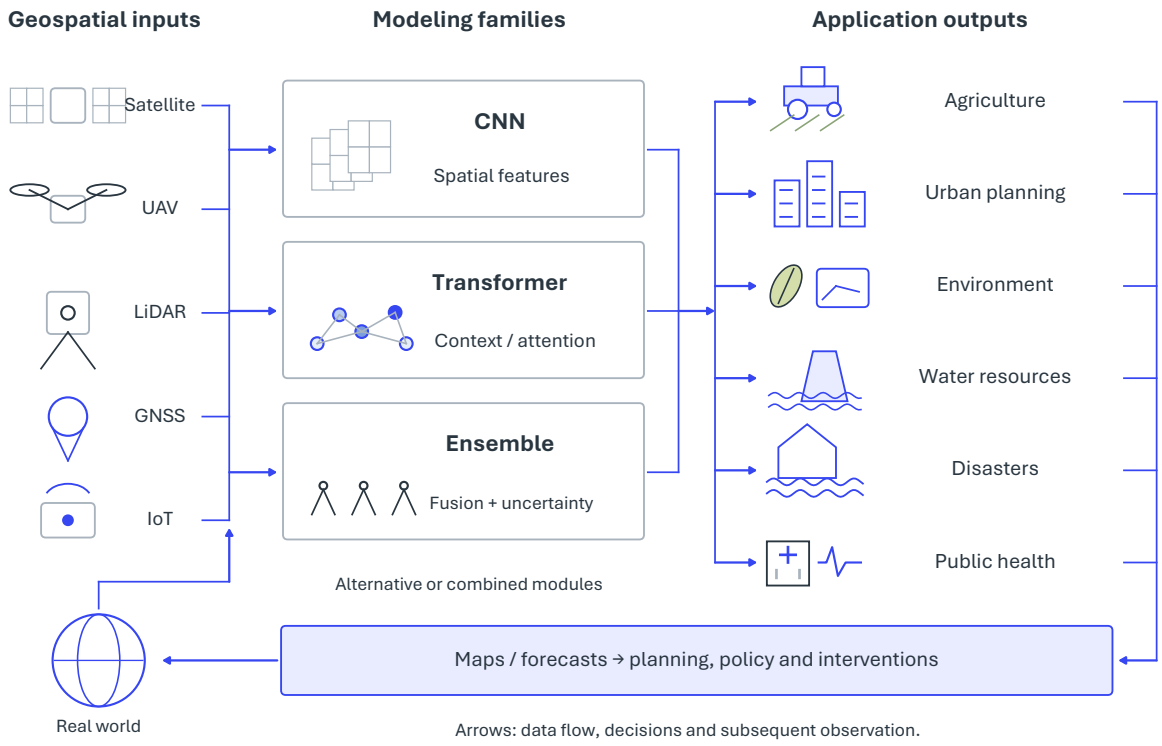


Figure 11.5: GeoAI as a data-to-decision loop: satellite, UAV, LiDAR, GNSS, and IoT observations feed alternative or combined CNN, transformer, and ensemble models, whose outputs serve six application domains and act back, through planning and policy, on the sensed world.

11.2.3 Pivotal Task-1: Precision Agriculture

Precision agriculture tailors irrigation, fertilization, and pest control to the conditions of each field, or each plant, instead of managing a field uniformly. GeoAI supplies the maps and predictions this requires through the complementary mapping, yield-estimation, and irrigation tasks of Figure 11.6.

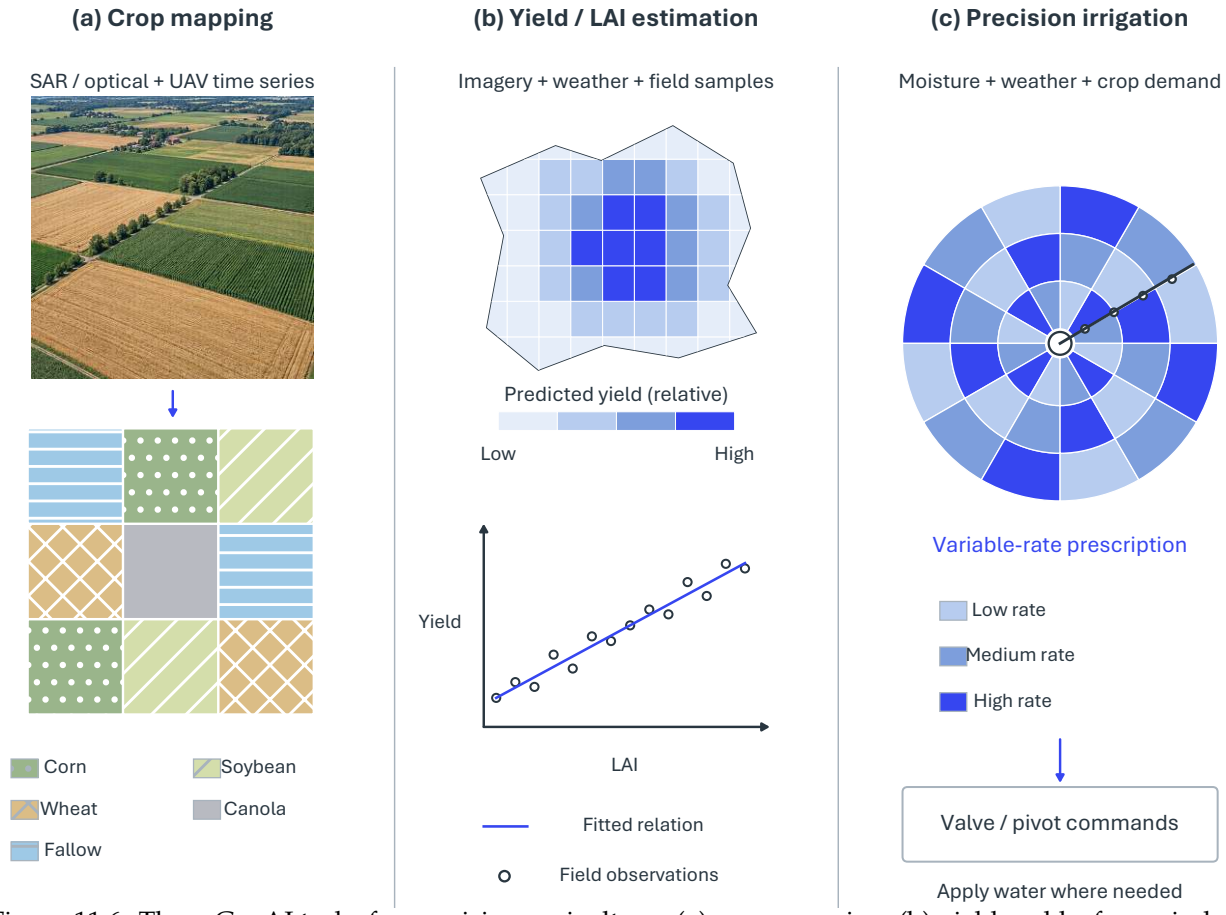


Figure 11.6: Three GeoAI tasks for precision agriculture: (a) crop mapping, (b) yield and leaf-area-index (LAI) estimation, and (c) variable-rate precision irrigation. Maps and fitted points are schematic.

Crops Mapping. Crop mapping turns a time series of satellite or UAV imagery into a per-pixel crop-type label, and the accuracy gains in recent work track the richness of the temporal signal. Han *et al*. [2699] combine a ResNet feature extractor with cross-attention and spatio-temporal self-attention over Sentinel-1 SAR series, reaching 96% overall accuracy on the Brandenburg benchmark. 3D convolutional networks over Sentinel-2 [2700] and Landsat [2701] series treat time as a third convolution axis, and fusing Sentinel-1 SAR with Sentinel-2 optical data maps rice phenological stages within the season [2702]. Multi-season and SAR–optical inputs consistently beat single-date classification, in part because SAR sees through the cloud that interrupts optical series. At the other end of the resolution range, UAV imagery supports row-level segmentation, where U-Net outperforms LinkNet and PSPNet on sugarcane crop lines [2703].

Yields Prediction. Yield prediction regresses a continuous productivity estimate from the same imagery, and the usual design has two stages: extract a vegetation index or phenotype, then regress. Random forests over Sentinel-2, RTK GPS, and LiDAR variables predict cotton yield [2704], and an ensemble over Landsat-8 NDVI and agro-meteorological data reports R^2 0.99 for tea yield on its test set [2705]. UAV imagery earns its extra cost where the target is a fine-grained phenotype: Mask R-CNN extracts wheat-spike phenotypes

that feed a yield regressor [2706], and a linear mixed-effect model on fixed-wing UAV multispectral imagery estimates rice leaf-area index and biomass [2707].

Precision Irrigation. Irrigation closes the loop, turning maps and estimates into water-delivery decisions [2708]. One line of work detects irrigation infrastructure at scale, such as center-pivot systems in Sentinel-2 series with Mask R-CNN [2709] or irrigation types in NAIP aerial images with a U-Net and transfer learning [2710]. The other controls water locally: a drone system fuses RGB, thermal-infrared, and GPS data to find the driest zones of a field and sets valve flow with a random forest [2711].

11.2.4 Pivotal Task-2: Urban Planning, Logistics, and Transport

The second task turns GeoAI on the built environment. Figure 11.7 presents five complementary tasks on a shared urban context, from built structure and temporal change to services, transport, and green infrastructure.



Figure 11.7: GeoAI for the built environment: (a) building and land-use extraction from overhead and street-view imagery; (b) temporal urban change; (c) sensor-supported digital twins and operational services; (d) mobility, routing, and infrastructure analysis; and (e) canopy, parks, green roofs, and biosecurity monitoring. Tasks are complementary rather than a mandatory serial pipeline. Shared map geometry, temporal trends, and the city illustration are schematic; arrows denote time, data flow, or travel direction within the corresponding panels.

Urban Data Extraction. Land use and built form are classified from overhead and street-level imagery, and fusing the two views resolves ambiguities that neither settles alone. A fine-tuned EfficientNet classifies the

21 land-use classes of the UC Merced set [2712], while a dual-branch network that pairs dilated convolutions on Google Earth imagery with an attention-based multi-view network on Street View detects informal urban villages [2713].

Urban Dynamics. Temporal models capture how urban form changes over time. An LSTM with attention simulates urban densification from land-use rasters of 2000, 2010, and 2020 [2714], and a pix2pix GAN predicts land-use and land-cover change by conditioning Landsat imagery on planning indicators [2715]. A static quantity such as urban solar exposure needs no temporal model; a feed-forward network on building geometry, occupancy, and orientation suffices [2716].

Smart Cities. The service layer turns these representations into live operation: video georeferencing and small-object detection inside an urban digital twin [222], LSTM parking-occupancy prediction served to a mobile application [2717], and detection of commercial signage in Street View imagery to map urban commerce [2718].

Logistics and Transport. Movement through the city splits into two problems with different tools. Flow forecasting is temporal: bus arrival times are predicted from a traffic-density matrix with recurrent and convolutional models [2719], a hybrid GRU–LSTM forecasts traffic from OpenStreetMap and GIS features [2720], IODRNN decomposes its outputs incrementally to remove the lag in GNSS-based forecasts [2721], and itinerary recommendation adds a denoising autoencoder with collaborative learning [2722]. Infrastructure mapping is a segmentation problem: open LiDAR point clouds inventory rockfall-prone road slopes [2723], SegNet on RapidEye imagery updates rural road networks [2724], and U-Net with PointNet++ performs panoptic segmentation of railway environments from multimodal LiDAR [2725].

Green Infrastructure and Bio Security. Vegetation monitoring moves from coverage to function to threat, and each step demands finer spatial and spectral resolution than the last. A semi-supervised DeepLabv3+ maps urban tree cover nationwide from high-resolution satellite imagery [2726]; a Faster R-CNN tree inventory identifies seven canopy species and, through geographically weighted regression, their effect on land-surface temperature [2727]; and biosecurity surveillance of invasive forest pests uses multispectral aerial imagery, where an instance-segmentation model reached an F1-score of 79% against 58% for a tile classifier [2728].

11.2.5 Pivotal Task-3: Environmental Monitoring

The third task observes the Earth system itself, across the three fronts of Figure 11.8.

Climate Change. Long-term land-surface change is read mostly through vegetation records spanning decades of Landsat, Sentinel, and MODIS data, and the methods trade spatial detail against record length. Lou *et al*. [2729] map land use and land cover on Google Earth Engine with a deep network and find rapid greening of the Qinghai–Tibet Plateau from 1988 to 2018. Deep-learning-enhanced spatiotemporal fusion reconstructs long-term NDVI in the Danjiang River Basin [2730], random forests retrieve vegetation phenology from MODIS more accurately than rule-based methods [2731], and a bidirectional LSTM produces the GLASS 250 m leaf-area-index product (version 6) from MODIS reflectance [2732].

Atmospheric Chemistry. Atmospheric composition is retrieved by learning the map from satellite radiances and meteorology to ground-measured concentrations. Nitrogen dioxide is the most studied target, estimated hourly from Sentinel-5P, meteorology, and sensor data with XGBoost [2733] and daily across China with a random forest and K-means clustering over OMI data [9]. A convolutional autoencoder coupled to WRF-Chem models volcanic SO_2 dispersion [2734], AQNet fuses Sentinel-2 and Sentinel-5P to predict NO_2, O_3, and PM_{10} [2735], an LSTM forecasts five pollutants from ground-station series [2736], and a deep network derives aerosol absorption from the OMI ultraviolet aerosol index, validated against AERONET [2737].

Environmental Impacts. Impact assessment links land-use change to ecological outcomes, and it is judged not only on accuracy but on whether it can name the responsible driver, which is why explainability recurs

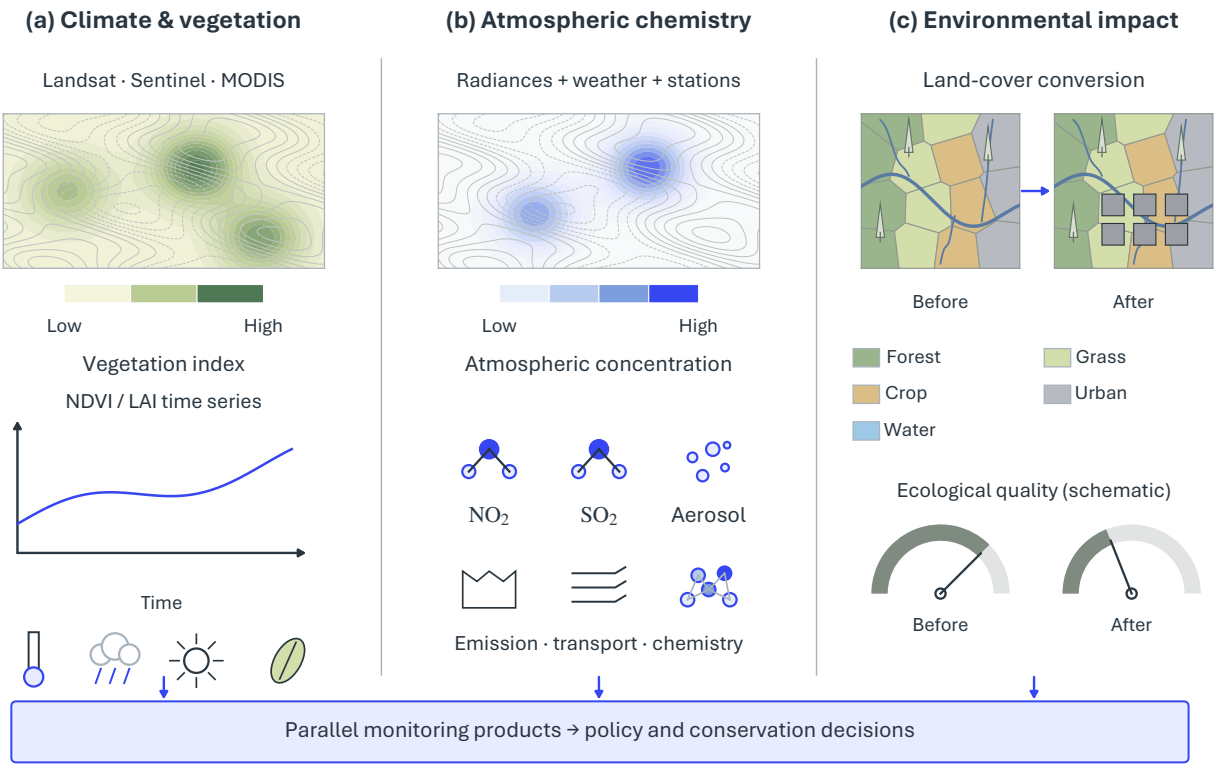


Figure 11.8: Three parallel GeoAI tasks for environmental monitoring: (a) climate and vegetation (NDVI/LAI trends), (b) atmospheric chemistry (NO_2, SO_2, aerosols), and (c) environmental impact of land-cover change. Maps and trends are illustrative, not measured changes.

here. Neural networks on multi-temporal Landsat land-cover maps assess urban ecological quality [2738]; multiscale geographically weighted regression, random forests, and SHAP relate agricultural land to $PM_{2.5}$ concentrations [2739]; and interpretable ensembles measure the effect of climate change on agricultural land suitability [2740].

11.2.6 Pivotal Task-4: Water Resources Management and Precipitation Forecasting

The fourth task addresses four complementary aspects of water resources in Figure 11.9.

Hydrological Modeling. Convolutional models dominate local surface geometry. Space-time image velocimetry on UAV video estimates river velocity and discharge, validated with LiDAR-derived terrain [2741]; U-Net extracts drainage ditches from LiDAR in low-relief terrain [2742]; quadtree decomposition with multi-scale CNNs extracts coastlines at lower cost than FCN and U-Net baselines [2743]; and a reflectance-transformation CNN maps bathymetry from Landsat and Sentinel-2 imagery [2744]. At basin scale, where streamflow depends on several precipitation products (MODIS, CHIRPS, MSWEP), super-ensembles that weight multiple regressors take over [2745].

Groundwater. No optical sensor sees the water table, so groundwater models work from proxy variables and gravimetry. Static delineation of recharge or shallow-groundwater zones uses tree ensembles and deep networks over hydrogeological and remote-sensing variables [2746, 2747, 2748]. GRACE/GRACE-FO measures total-water-storage changes; isolating groundwater requires auxiliary estimates of other water-storage components. Storage forecasting uses these gravity-derived anomalies, which a CNN–LSTM forecasts with an RMSE of 44.92 mm/month [2749] and which machine-learning regressors downscale to catchment resolution [2750].

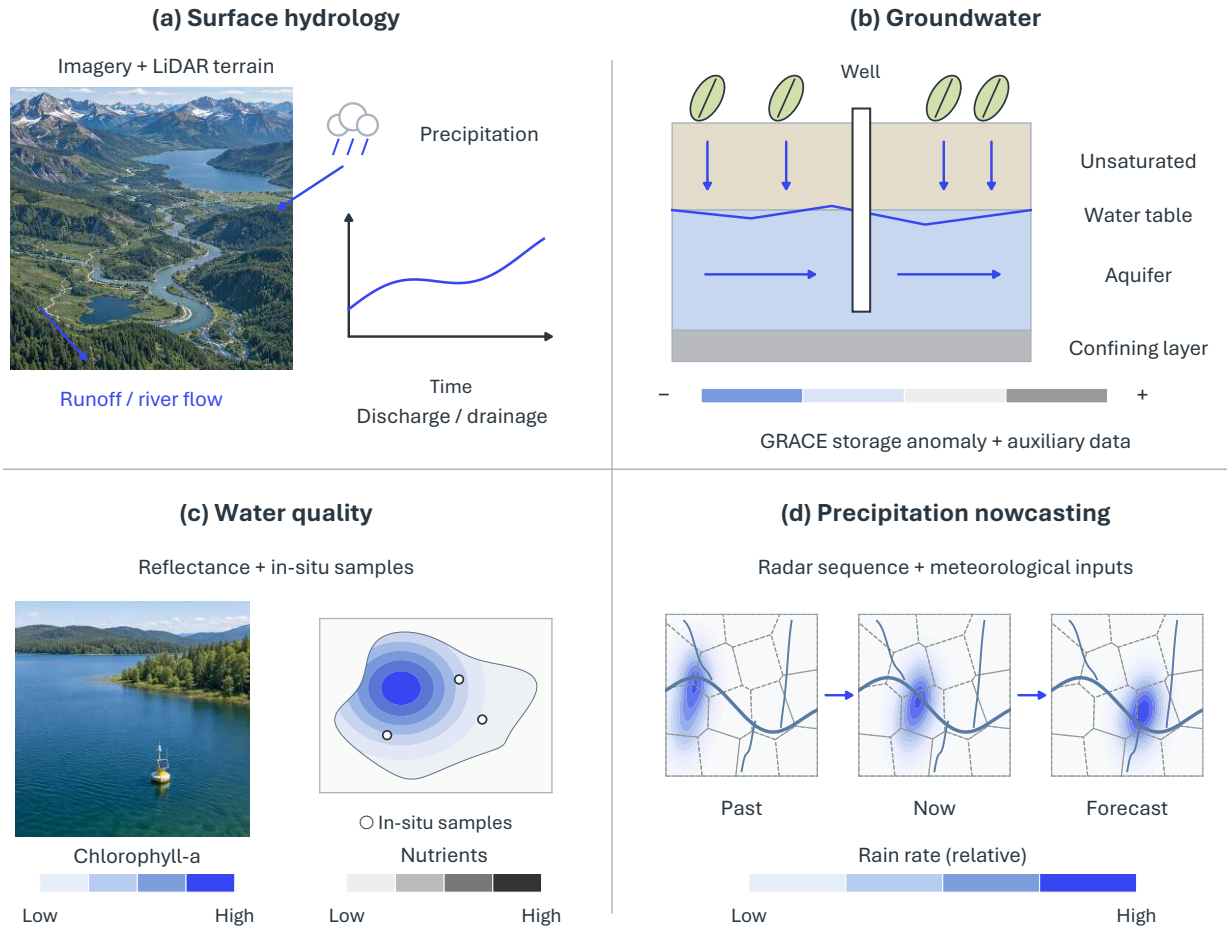


Figure 11.9: Four GeoAI tasks for water resources: (a) surface hydrology, (b) groundwater from GRACE storage anomalies and auxiliary data, (c) water quality calibrated against in-situ samples, and (d) radar-based precipitation nowcasting. The panels are parallel tasks, not successive stages of the water cycle; all values are schematic.

Water Quality. Water quality is regressed from spectral reflectance against in-situ samples, and accuracy follows the optical signature of the constituent. Pigments with a clear signature are tractable: fusion of data and classifiers through higher-order SVD estimates chlorophyll-a in Lake Nicaragua [2751], and on UAV multispectral and hyperspectral data of a karst wetland, transformers and gradient boosting predict phycocyanin, dissolved oxygen, and chlorophyll-a, interpreted with SHAP [2752]. Dissolved nutrients remain harder because their optical fingerprint is faint, whether estimated from MODIS with a spatiotemporal deep belief network [2753] or from Landsat-8 with an ANN [2754]. Optical water types themselves can be classified from Sentinel-2 with an SVM [2755].

Precipitation Forecasting. Precipitation nowcasting predicts the next radar frames from the current sequence. HST-AFP fuses radar echoes with weather forecasts through attention [2756], STIN adds humidity, wind, and temperature [2757], and PFST-LSTM adds pseudo-flow estimation to follow echo motion [2758]; at longer range, multi-model ensembles post-process numerical weather prediction [2759].

11.2.7 Pivotal Task-5: Disaster Management

The fifth task covers the disaster lifecycle (hazard or susceptibility mapping, event observation, and damage assessment) for earthquakes, floods, and forest fires, arranged as a grid in Figure 11.10.

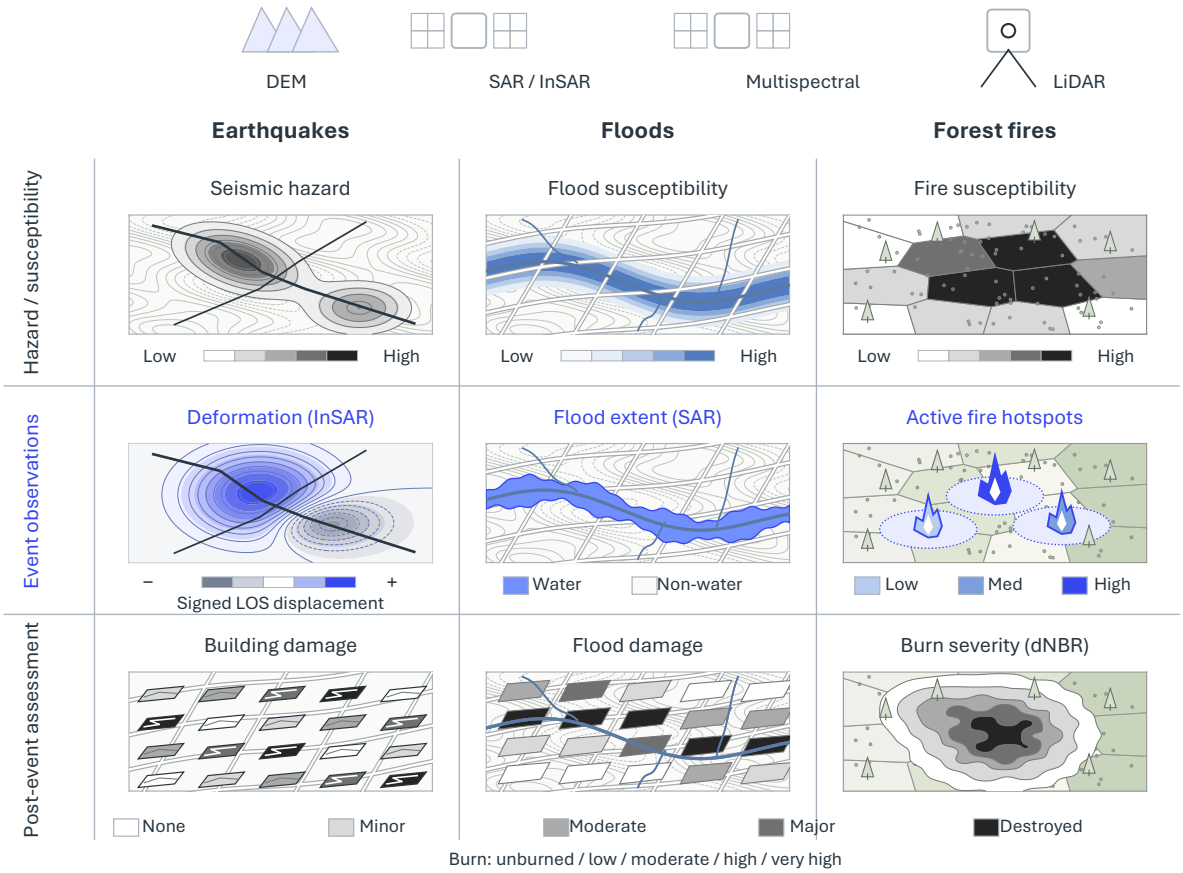


Figure 11.10: GeoAI for disaster management as a lifecycle-by-hazard grid: earthquakes, floods, and forest fires (columns) across hazard or susceptibility mapping, event observation, and post-event assessment (rows). All maps and severity scales are schematic.

Earthquakes. For earthquakes the lifecycle includes hazard mapping, event-related observation, and damage assessment; these tasks should not be equated with reliable prediction of an exact earthquake. A CNN over elevation, vector, and attribute data maps seismic risk in north-eastern India [2760], and encoder–decoder LSTMs [2761] and neural networks [2762] detect ionospheric anomalies in GNSS total-electron-content data around events such as the 2019 Pakistan earthquake. After the event, LiDAR, orthophotos, and multispectral imagery estimate building vulnerability [2763], and change detection on multi-temporal photogrammetric point clouds classifies structural damage into grades [2764].

Floods. Floods lean on radar. Forecasting fuses meteorological, hydrological, and crowdsourced data, either comparing classical learners under cross-validation [2765] or with a ConvLSTM that learns spatial rainfall patterns and their temporal dynamics jointly [2766]. Inundation mapping falls to SAR segmentation, which sees through cloud, as in an attention-augmented U-Net on dual-polarized Sentinel-1 imagery [2767].

Forest Fires. Fire has the fullest lifecycle. Susceptibility is predicted from drone imagery with temperature and humidity sensing [2768], or from elevation, vegetation indices, and hydrometeorology with a neuro-fuzzy system tuned by an artificial bee colony [2769]. Detection is trained on drone RGB and infrared imagery such as the FLAME 2 dataset [2770] and can be distributed across sites with federated learning [2771]. Burn scars are delineated afterward by a U-Net on PlanetScope RGB and near-infrared imagery, with F1 up to 0.94 [2772].

11.2.8 Pivotal Task-6: Healthcare

The sixth task brings GeoAI to public health, where spatial signals couple disease and environment (Figure 11.11).

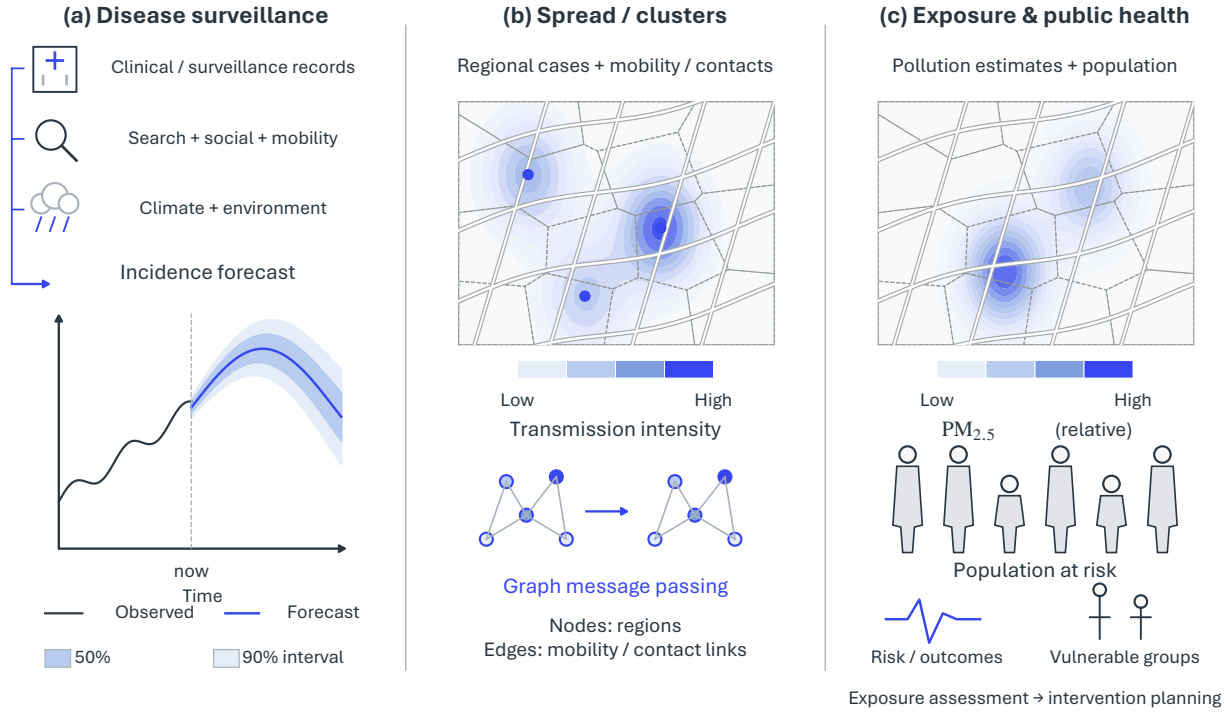


Figure 11.11: Three parallel GeoAI tasks for public health: (a) disease surveillance and incidence forecasting, (b) spatial spread modeled by graph message passing over regions, and (c) pollution exposure assessment. The panels are not causally linked; maps and intervals are illustrative, not surveillance results.

Infectious Diseases and Toxins. Disease surveillance gains most from leading indicators, since search queries and geotagged posts surface an outbreak before clinical reporting does. Influenza forecasting combines CDC surveillance data with Google Flu Trends and climate in a multi-stage LSTM [2773], with geotagged Twitter posts in a neural network tuned by an artificial-tree algorithm [2774], and with search frequencies and health records in a network-based regression that exploits synchrony across U.S. states [2775]; dengue in China is forecast from climate and weekly case counts, with support-vector regression performing best [2776]. FINDER shows the payoff most clearly: it geolocates web-search signals of foodborne illness to restaurants, and the restaurants it flagged were three times more likely to pose food-safety risks than those found by complaint-based inspection [2777].

Pandemics. A pandemic shifts the question to transmission across geography. Geospatial similarity learning predicts respiratory-virus spread in new regions [2778], and spatio-temporal regression tracks COVID-19 in Brazil [2779]. Detection networks catch what an epidemic curve misses: gatherings of people and vehicles in satellite imagery [2780], proximity between re-identified individuals in video [2781], and the drop in $PM_{2.5}$ during lockdown, measured from MODIS and Sentinel-5P [2782].

Public Health. Chronic exposure links where people are to what they breathe. On the mobility side, mobile health uses wireless devices for clinical and public-health purposes [2783], and the MD2K center turns mobile-sensor streams into markers of smoking, craving, and stress [2784]. On the pollutant side, a geographically weighted gradient-boosting machine [2785] and a neural network over MODIS aerosol optical depth and land-use features [2786] map daily $PM_{2.5}$ across China and the United States. Exposure assessment is only as sharp as the coarser of the two spatial scales.

Table 11.2: Representative data sources and methods for GeoAI, one or two per pivotal task. Numbers are each paper's own test set and are not comparable across rows; cells without a number name the metric the paper reports.

Pivotal task	Representative data source	Representative method	Headline metric
Precision agriculture	Sentinel-1 SAR	STMA [2699]	96% OA (Brandenburg)
Precision agriculture	Landsat-8 NDVI + meteorology	Ensemble [2705]	R^2 0.99
Urban planning	UC Merced land use	EfficientNet (fine-tuned) [2712]	accuracy
Environmental monitoring	Landsat, 1988–2018	Deep LUCC mapping on GEE [2729]	LUCC accuracy
Water resources	GRACE / precipitation	CNN–LSTM [2749]	RMSE 44.92 mm/month
Disaster (flood)	Sentinel-1 SAR	Attention U-Net [2767]	precision / recall / F1
Disaster (forest fire)	PlanetScope (RGB+NIR)	U-Net [2772]	F1 0.94
Healthcare	CDC / Google Flu / climate	Multi-stage LSTM [2773]	RMSE / MAPE

11.2.9 Established Tools and Systems

Cloud geospatial platforms provide planetary-scale storage and analysis next to the data. Google Earth Engine[23] offers a petabyte-scale archive of satellite imagery with server-side analysis, Microsoft's Planetary Computer[24] pairs open Earth-observation data with a Python/STAC API, and commercial providers such as Esri ArcGIS[25] and Descartes Labs[26] deliver hosted GeoAI pipelines for enterprise and government users.

Open-source GIS and ML toolkits carry the modeling layer. QGIS[27] is the de facto open desktop GIS, while deep-learning libraries tailored to raster geospatial data, such as TorchGeo[28] and Raster Vision[29], bridge PyTorch to satellite imagery, and Sentinel Hub[30] streamlines on-demand access and processing of Copernicus archives.

Open Earth-observation programmes supply the data foundation: the European Copernicus programme's Sentinel constellation and NASA/USGS Landsat provide free, calibrated, multi-decadal imagery, complemented by gravimetric (GRACE/GRACE-FO), atmospheric (OMI, Sentinel-5P), and high-resolution commercial (WorldView, PlanetScope) sources that recur throughout the methods above.

11.2.10 Toward Geospatial Foundation Models and GeoLLMs

Almost every method surveyed above is a task-specific supervised model: a CNN, random forest, LSTM, or GAN trained from scratch for one crop, one hazard, or one pollutant. That design is being displaced by the same pretraining-and-transfer paradigm that reshaped the rest of the field (Chapter 7), and the geospatial community has begun building its own foundation models. Masked-autoencoding backbones lead the shift: SatMAE [193] adapts the masked image model to the temporal and multi-spectral structure of satellite imagery, Scale-MAE [2787] makes the pretext task explicitly scale-aware so a single backbone serves the huge ground-sample-distance range Earth observation spans, and Prithvi [2788] packages such a backbone as a generalist model fine-tunable across flood mapping, burn-scar delineation, and crop segmentation. A single pretrained encoder thus stands in for the dozens of bespoke feature extractors that populate the tasks above.

The language side is moving just as fast. RemoteCLIP [2789] aligns remote-sensing imagery with text, giving open-vocabulary retrieval and zero-shot classification without per-class labels; GeoChat [2790] grounds a large vision-language model on overhead imagery so an analyst can query a scene, ask for

[23] https://earthengine.google.com/
[24] https://planetarycomputer.microsoft.com/
[25] https://www.arcgis.com/
[26] https://www.descarteslabs.com/
[27] https://qgis.org/
[28] https://github.com/microsoft/torchgeo
[29] https://rastervision.io/
[30] https://www.sentinel-hub.com/

region-level descriptions, and obtain referring localizations in dialogue; and SegEarth-R1 [2791] pushes further into pixel-level geospatial reasoning, generating segmentation masks from implicit, reasoning-heavy queries rather than fixed class names. These systems inherit the multimodal reasoning and world-model machinery of Part III and specialize it to Earth observation, pointing toward GeoAI workflows driven by natural-language instructions rather than hand-built per-task pipelines.

11.2.11 Challenges

Most of the methods above were trained and tested in one region on one sensor, and that pattern shapes the open problems: whether a model transfers across regions and sensors, how heterogeneous and uneven data are reconciled, whether pipelines run at the scale and latency that operations need, and what new failure modes the geospatial foundation models bring.

Generalization across Regions and Sensors. Few of the studies surveyed test a model outside the region, season, or sensor it was trained on, so their reported accuracies say little about transfer. Crop calendars, building styles, and spectral responses vary from place to place, and a crop map or flood segmenter trained on one country or one satellite can degrade sharply on another. Pretrained backbones such as Prithvi [2788] are motivated by exactly this gap, but systematic cross-region benchmarks remain rare.

Data Heterogeneity and Quality. Geospatial workflows combine imagery, sensor streams, vector layers, and crowdsourced inputs that differ in resolution, revisit time, and reliability, and the volume of these archives makes curation itself demanding [2698]. Open-access data needs quality control before it can serve as training labels. Geospatial data is also strategically sensitive, which raises questions of data sovereignty, and adversarial manipulation of inputs is a threat that none of the methods above addresses.

Scale and Latency. Operational services such as flood mapping during an event, precipitation nowcasting, and traffic forecasting must ingest new acquisitions continuously and update their models as satellites deliver data, which strains pipelines built around one-off training runs. When several models are chained in one workflow, validation and deployment cost grow with each link, pushing systems toward distributed, modular architectures.

Open Problems for Geospatial Foundation Models. The foundation models of the previous subsection bring problems of their own: pretraining that fuses radar, optical, and hyperspectral sensors at very different resolutions; evaluation benchmarks that test geospatial reasoning rather than image-level recognition; and grounding language output in physically consistent, georeferenced coordinates, which dialogue systems such as GeoChat [2790] do not yet guarantee.

11.3 Aerospace Engineering

11.3.1 Introduction

Aerospace engineering designs and operates systems that work within and beyond the Earth's atmosphere, and its missions are spatial problems at their core. Rendezvous and docking (RVD), on-orbit servicing (OOS), debris inspection, and planetary landing and roving all depend on estimating three-dimensional structure and relative motion, often of a non-cooperative target that carries no markers, while communication delays rule out continuous ground control. Both spacecraft autonomy (*e.g.*, guidance, navigation, and control) and human-in-the-loop operations (*e.g.*, situational awareness for ground controllers) therefore rest on spatial perception and reasoning over geometry, relative motion, semantics, and physical constraints.

Aerospace engineering is thus a demanding deployment target for the perception and reasoning capabilities surveyed in Part III: the detection, segmentation, pose-estimation, and 3D-reconstruction techniques of Chapter 6 and the planning and decision-making methods of Chapter 7 are here specialized to non-cooperative orbital targets, extreme illumination, and long-horizon autonomy under radiation-hardened compute and communication latency.

11.3.2 System Overview

The value of spatial intelligence in aerospace lies less in any single perception module than in the closed autonomy loop it serves (Figure 11.12): onboard and ground-based sensing (monocular and multi-view cameras, LiDAR, telemetry) feeds a perception-reasoning-decision core, whose commands change the scene that is sensed next. The four pivotal tasks treated below are complementary stages of this loop. **Satellite pose estimation** supplies the geometric anchor, recovering where a target is and how it is oriented; **spacecraft detection, segmentation, and parts recognition** enrich that anchor with structural semantics, identifying what the target is and which functional components it exposes; **operator agents** lift this perceptual state into high-level reasoning, planning, and human-facing decision support; and **planetary robotics** closes the loop on the surface, coupling perception, localization, and planning into physical exploration and sampling. Table 11.5, after Task-4, consolidates representative benchmarks and metrics across all four.

11.3.3 Pivotal Task-1: Satellite Pose Estimation

Vision-based satellite pose estimation predicts the 6-DoF relative position and orientation of a target spacecraft from monocular or multi-view imagery; it is the geometric anchor of the autonomy loop and a prerequisite for rendezvous, docking, and on-orbit servicing. It specializes the 6-DoF estimation problem of Chapter 6 to a hostile setting: harsh illumination, specular reflections, little texture, and a severe gap between the synthetic images models are trained on and the real images they meet in orbit. Figure 11.13 traces the pipeline from method to evaluation.

Traditional methods relied on hand-engineered features (*e.g.*, Harris corners, SIFT, SURF, and ORB) or model-based tracking [2792], which are sensitive to noise and domain shift, and deep learning has since become dominant [2793]. The first deep models regress the pose directly with CNNs [2794], as in the Spacecraft Pose Network (SPN) [2795], a GoogLeNet-based network [2796], and a VGG backbone [2797], trained on synthetic data and on laboratory imagery that simulates an on-orbit assembly operation. SPNv2 [2798] made multi-task learning the prevailing design: its shared encoder jointly predicts keypoints, a direct pose regression, and foreground segmentation, and it narrows the synthetic-to-real gap by online domain refinement, adapting the network's normalization layers on unlabeled target imagery at deployment. Hybrid pipelines estimate a coarse pose with a deep model and refine it with geometry-based optimization, keeping the final estimate physically consistent; early work classified discretized viewpoints with a CNN [2799], and the same coarse-to-fine pattern now extends to 6-DoF tracking with stereo event cameras [2800].

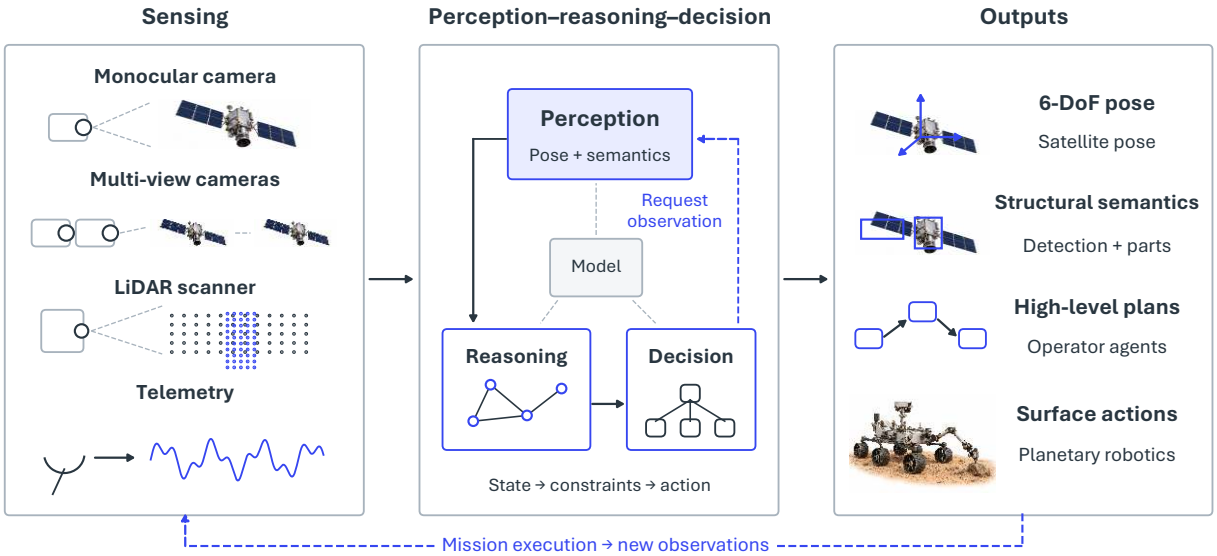


Figure 11.12: Aerospace spatial intelligence as a closed loop from onboard and ground sensing, through a perception–reasoning–decision core, to complementary pose, structural-semantic, planning, and surface-action outputs. Decisions may request additional observations; mission execution produces subsequent measurements. Pose and semantic estimates are information products rather than physical actions. Satellite views, point clouds, and rover imagery are schematic.

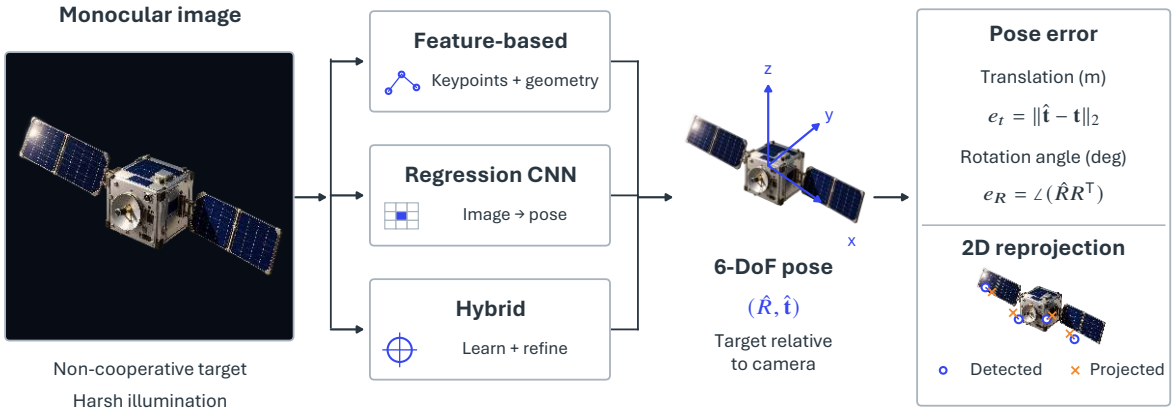


Figure 11.13: Satellite pose estimation: a monocular image of a non-cooperative target passes through alternative feature-based, regression, or hybrid pipelines to a target pose relative to the camera. Translation and relative-rotation errors compare estimated and reference poses; 2D reprojection compares detected image keypoints with projected model points. The satellite render and keypoint residuals are schematic; no numerical benchmark scores are implied.

SPEED [770] and SPEED+ [2801] are the standard testbeds (Table 11.3): both pair a large synthetic training set with laboratory hardware-in-the-loop test images, so that the synthetic-to-real gap is measured rather than assumed, and the Satellite Pose Estimation Challenge built on them fixed common training regimes, input formats, and metrics [2802]. CubeSat-CDT [2803] adds trajectories of a symmetric target, so that temporal methods and pose ambiguity can be evaluated. Evaluation reports translation and rotation error as defined for SPEED [770], often with 2D reprojection or keypoint error [2794]; results are given separately on synthetic and real test sets, and robustness to illumination, occlusion, and background clutter is probed explicitly [2798].

Table 11.3: Overview of existing pose estimation datasets. *sim* refers to simulated images and *lab* to laboratory data. SPARK's 11 classes are 10 spacecraft and one debris class.

Dataset	Image		Classes	Resolution	Rendering	Trajectories	Public
	Synthetic	Non-synthetic					
SPARK [2804]	150k	-	11	1024 × 1024	sim	✗	✓
SPEED [770]	15k	305	1	1920 × 1200	sim + lab	✗	✓
SPEED+ [2801]	60k	10k	1	1920 × 1200	sim + lab	✗	✓
URSO [2793]	15k	-	2	1080 × 960	sim	✗	✓
SwissCube [747]	50k	-	1	1024 × 1024	sim	✓	✓
Cygnus [2805]	20k	540	1	1024 × 1024	sim + real	✗	✗
Prisma12K [2806]	12k	-	1	752 × 580	sim	✗	✗
Prisma25 [2806]	-	25	1	-NA-	real	✗	✗
CubeSat-CDT [2803]	14k	8k	1	1440 × 1080	sim + lab	✓	✓

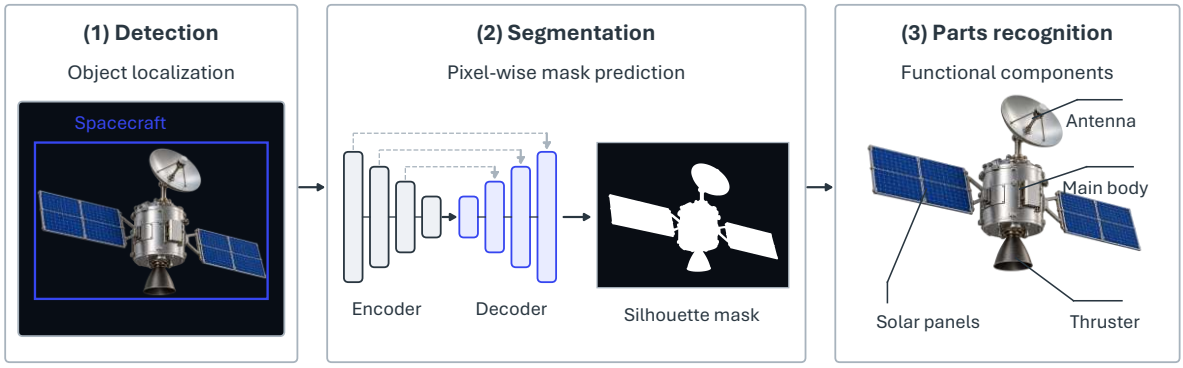


Figure 11.14: The "detect–segment–recognize" perception stack for non-cooperative spacecraft: (1) object localization, (2) encoder–decoder silhouette segmentation, and (3) recognition of solar panels, the main body, antenna, and thruster. The same synthetic target is used throughout. The silhouette is an illustrative foreground mask, not a measured prediction; arrows indicate increasing semantic detail rather than requiring every implementation to run these stages serially.

11.3.4 Pivotal Task-2: Spacecraft Detection, Segmentation, and Parts Recognition

Detection, segmentation, and parts recognition enrich the geometric anchor of Task-1 with structural semantics, telling what the target is and which functional components it exposes; proximity operations, on-orbit servicing, debris inspection, and collision avoidance all depend on this. The detection and segmentation machinery of Chapter 6 meets the obstacles of Task-1 here (strong illumination variation, specular metallic surfaces, little texture) and, in addition, a scarcity of annotated real imagery. The stack runs from coarse presence to fine structure: detection localizes the spacecraft, segmentation delineates its silhouette, and parts recognition resolves its components, as Figure 11.14 illustrates.

Convolutional and transformer architectures now carry all three stages [2807]. Detection is formulated as bounding-box or keypoint localization, with lightweight saliency-driven detectors for onboard compute, while encoder–decoder networks, pyramid attention modules, and hybrid 2D/3D architectures delineate silhouettes and component boundaries [2808]. Parts recognition (solar panels, antennas, thrusters, reflectors) often adds semantic priors or 3D geometric reasoning to stay robust under occlusion. Two recent lines reduce the dependence on manual labels: prompt-guided Segment Anything (SAM) automates the annotation of space targets [2809], and diffusion priors support few-shot segmentation and adaptation to unseen spacecraft [2810].

The datasets in Table 11.4 differ along a few axes. Labels range from whole-object masks (SwissCube [747], a pose dataset that also provides CubeSat masks, and SPARK [2804]) and object-level boxes (STAR-24K [2811]) to component labels, from a single component (the solar panels of the Tsinghua dataset [2812]) up to nine (Spacecraft-DS [2813]). Most are rendered (Unity3D for SPARK, Unreal Engine 4 for UESD [2814]), whereas Spacecraft-DS uses hardware-in-the-loop capture to narrow the synthetic-to-real gap. The Adelaide dataset [2815] fixed the common three-component scheme (solar panel, main body, antenna), which Shao *et al*. [2807] extended to video sequences rendered under three illumination conditions and several relative-motion modes; Zhao *et al*. [2808] carried the same labels to 3D point clouds, and NCSTP [2816] raised the scale to 200,000 images of satellites, debris, and rocks. Detection is scored by mean average precision and segmentation by IoU, as in Chapter 6.

Table 11.4: Summary of satellite recognition datasets. '-' indicates that the exact quantity is unknown; the Zhao *et al*. dataset consists of point clouds rather than images.

Dataset	#Images	#Satellite Models	Segmentation Level	#Components
SwissCube [747]	50,000	1	Global	0
UESD [2814]	10,000	33	Component	5
SPARK [2804]	150,000	10	Global	0
STAR-24K [2811]	24,451	-	Global	-
NCSTP [2816]	200,000	16	Component	4
Tsinghua Satellite Dataset [2812]	18,381	-	Component	1
Adelaide Satellite Dataset [2815]	3,117	-	Component	3
Satellite Component Video Dataset [2807]	32,402	13	Component	3
Spacecraft-DS [2813]	1,602	2	Component	9
3D component point clouds (Zhao *et al*.) [2808]	-	33	Component	3

11.3.5 Pivotal Task-3: Operator Agents

Operator agents lift the perceptual state recovered by Tasks 1 and 2 into reasoning, planning, and decision support, monitoring, interpreting, and controlling spacecraft systems to reduce operator workload and allow more autonomous mission execution. They are the aerospace instance of the planning and decision-making capabilities of Chapter 7. Rooted in rule-based and model-driven architectures, their role has expanded from health monitoring and scheduling to high-level control in dialogue with human experts, a history that runs from symbolic planning through multi-agent coordination to foundation-model reasoning (Figure 11.15).

Early operator-agent architectures were built on symbolic reasoning, constraint-based planning, and expert-system principles. The Remote Agent Experiment [2817] and early autonomous spacecraft-agent prototypes [2818] demonstrated onboard planning, diagnosis, and execution in deep-space missions. As autonomy requirements grew, multi-agent planning frameworks [2819, 225] were introduced to coordinate heterogeneous spacecraft, manage distributed sensing, and schedule jointly across satellite constellations. Large language models changed the design: recent work [2820, 2821] uses LLMs as operator agents that interpret telemetry, generate action plans, respond to anomalies, and follow natural-language mission specifications, and fine-tuned LLMs [2822, 2823, 2824, 2825] have been tested in simulators such as Kerbal Space Program (KSP) for guidance, navigation, and control. To meet the interpretability and safety requirements of mission-critical decisions, other studies integrate retrieval-augmented generation (RAG) [2826], domain knowledge graphs [2827], and structured symbolic reasoning, while hybrid architectures such as MCP-based multi-component agents [2828] combine small specialized models with foundation models to balance computational cost and reasoning depth, particularly for on-orbit servicing.

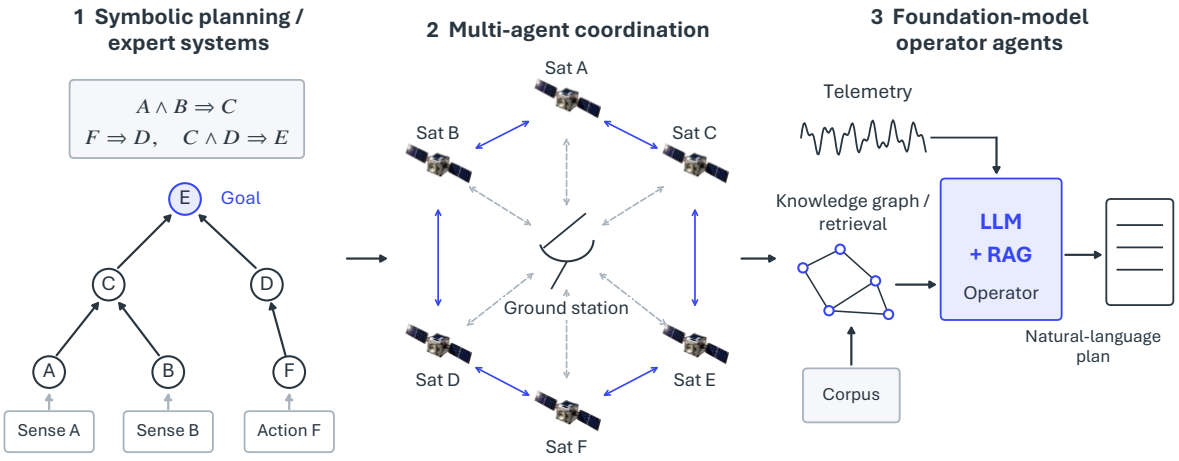


Figure 11.15: Three paradigms in the development of spacecraft operator agents: (1) symbolic planning and expert systems, (2) multi-agent coordination across a constellation, and (3) LLM- and RAG-based agents that combine telemetry with retrieved domain knowledge to propose natural-language plans. Inter-panel arrows indicate historical development, not required dataflow. In the rule example, observations and action outcomes support the goal; blue peer links and dashed ground links denote communication. Satellite imagery and rules are illustrative.

Evaluation environments followed the same path. The Remote Agent Experiment [2817] supplied real mission scenarios for onboard planning and diagnosis, and multi-agent planning studies [2819] supply constellation-level scheduling, resource-allocation, and communication constraints. LLM agents are tested in simulation, chiefly in KSP [2825] and in the KSPDG suite of SpaceGym [2829], a set of multi-agent pursuit–evasion differential games built in the KSP engine. Classical agents are scored on planning optimality, constraint satisfaction, fault-diagnosis accuracy, and task completion; LLM agents are additionally judged on adherence to safety constraints, robustness to shifts in telemetry or mission context, and the interpretability of their decisions, which is why uncertainty estimation and human-in-the-loop verification recur throughout this work.

11.3.6 Pivotal Task-4: Planetary Robotics

Planetary robotics closes the autonomy loop on the surface, coupling the perception, localization, and planning capabilities of Chapters 6 and 7 into exploration, navigation, and sampling on the Moon, Mars, and small bodies. Unlike orbital operations, surface missions face uneven and unstructured terrain, delayed and limited communication, restricted sensing, and long stretches of autonomy without continuous human supervision. Figure 11.16 layers the autonomy stack from perception and mapping through hierarchical decision-making to vision-guided manipulation.

Surface robots are organized around a mobility stack that couples perception, localization, mapping, planning, and control, with manipulation and science operations integrated at the task level. Feature tracking, visual odometry, and stereo reconstruction estimate rover motion and build local 3D models of rugged terrain, a capability flown since the Mars Exploration Rovers [2830]. The Enhanced AutoNav (ENav) system on NASA's Perseverance rover [2831] builds onboard 3D terrain maps, evaluates hazards, and plans safe local traverses under severe computational and communication constraints, and it evaluated terrain over a large majority of the rover's traversed distance. This autonomy remains embedded in a human-supervised workflow: ground operators specify science goals and route segments, and onboard autonomy resolves local navigation and hazard avoidance.

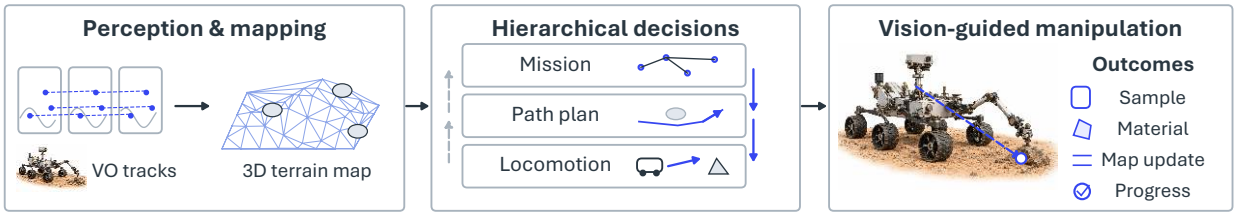


Figure 11.16: The planetary-robotics autonomy stack: visual-odometry feature tracks and 3D terrain mapping support hierarchical decisions and vision-guided manipulation. Mission reasoning handles goals, task decomposition, resources, and replanning; path planning uses global/local maps and constraints; reactive locomotion handles obstacles, stability, and terrain adaptation. Solid downward links convey plans, and dashed upward links convey status. Manipulation yields samples, material observations, environment updates, and task progress. Scenes and map geometry are illustrative.

Research and analogue-testbed systems extend these capabilities to multiple robots, where cooperative visual navigation and shared mapping improve coverage and robustness [2832]. Hierarchical frameworks [2833] decompose behavior across temporal and functional layers, balancing reactive locomotion and obstacle avoidance with deliberative path planning, task scheduling, and mission-level reasoning under safety, power, communication, and resource constraints.

Vision-guided manipulation supports sample collection and instrument deployment by coupling image-based pose estimation with position-controlled or compliant manipulation suited to contact with natural terrain [2834]. Learning-based methods are increasingly explored for terrain classification, slip prediction, and traversability estimation, but model-based perception, localization, and planning remain central: safety requirements are stringent, mission-representative labeled data is scarce, onboard computation is constrained, and purely data-driven methods are hard to validate for mission-critical use. Evaluation therefore combines physics-based simulators, mission-derived datasets, and laboratory or analogue-field testbeds that model illumination, dust, wheel–soil interaction, and communication delay, and it reports localization accuracy, map quality, hazard-aware traversal, task completion, and energy use.

Table 11.5: Representative benchmarks and headline metrics for the four pivotal aerospace tasks. The headline column names the standard evaluation axis, not a single best score.

Pivotal task	Benchmark	Representative method / system	Headline metric
Pose estimation	SPEED / SPEED+ [770, 2801]	Multi-task / domain-adaptive CNN [2798]	Translation + rotation error
Pose estimation	CubeSat-CDT [2803]	Cross-domain regression [2794]	2D reprojection error
Detection & parts	Adelaide / SPARK [2815, 2804]	Encoder-decoder segmentation [2808]	mAP
Detection & parts	NCSTP [2816]	SAM-based prompt segmentation [2809]	mAP / IoU
Operator agents	KSPDG / SpaceGym [2829]	Fine-tuned LLM agent [2825]	Maneuver success rate
Operator agents	Deep Space 1 flight experiment [2817]	Symbolic planner / diagnosis [2818]	Task completion
Planetary robotics	Mars rover traverses [2831]	Visual odometry [2830]	Localization drift
Planetary robotics	Hierarchical-autonomy suites [2833]	Layered decision-making [2833]	Task-completion / safety

11.3.7 Established Tools and Systems

Simulators and rendering engines. Because real orbital imagery is scarce and flight opportunities are expensive, most perception research is trained and validated in simulation. General-purpose game and physics engines have been repurposed as space testbeds: Unity3D and Blender drive the synthetic rendering pipelines behind the SPARK [2804] and satellite-component [2807] datasets, while the Kerbal Space Program[31] engine

[31] https://www.kerbalspaceprogram.com/

underpins LLM-agent guidance-navigation-control studies [2825]. For decision-making, SpaceGym[32] [2829] packages its OrbitDefender2D gridworld games and the KSPDG suite as standardized non-cooperative game environments, and robotics simulators such as NVIDIA Isaac Sim[33] and Gazebo[34] provide the wheel-soil and sensor models used to prototype planetary-rover autonomy.

Public challenges. The Satellite Pose Estimation Challenge [2802], run by Stanford's Space Rendezvous Laboratory and the European Space Agency on its Kelvins platform[35], set the protocols that pose estimation now follows, and its SPEED+ edition [2801] made the synthetic-to-real gap the explicit test.

Flight-proven autonomy systems. The NASA Remote Agent [2817], flown on Deep Space 1, remains the canonical reference for onboard planning, diagnosis, and execution. On planetary surfaces, rover visual odometry [2830] and the Enhanced AutoNav stack flown on Perseverance [2831] set the operational baseline against which hierarchical-autonomy and learning-augmented methods are measured, and current work pairs such flight-qualified stacks with foundation-model agents [2828].

11.3.8 Challenges

Orbital and planetary environments differ from terrestrial ones in ways that terrestrial data does not cover, and the open problems follow the pipeline: the data gap that limits perception, the compute budget that limits deployment, and the trustworthiness requirements that limit certification.

The Unresolved Sim-to-Real Domain Gap. Autonomous space systems, such as those performing on-orbit servicing or debris removal, are trained largely on simulated data, and simulators do not reproduce the high-dynamic-range lighting of orbit (*e.g.*, solar glare and deep shadow), intricate material reflections, or sensor degradation such as radiation noise. Models consequently lose accuracy on real imagery, as the hardware-in-the-loop test sets of the pose-estimation challenges show [2802, 2801], which keeps domain adaptation and domain randomization central to the field.

Computational Constraints and Real-time Operation. Spacecraft hardware is bound by size, weight, and power (SWaP) limits and must use radiation-hardened processors that are far less powerful than ground-based GPUs. Running 6-DoF pose estimation or SLAM at the update rates that proximity operations require on such computers calls for model compression, lightweight architectures, and hardware-aware optimization that preserve accuracy while cutting latency and power consumption.

Assuring Trustworthiness and Robustness in Extreme Conditions. Space missions are safety-critical, so perception must hold up under extreme visual ambiguity, such as estimating the pose of highly symmetric targets like CubeSats [2803] or navigating where standard reference signals are unavailable. Deployed systems also explain little about their own failures, so operators cannot easily trace a perception anomaly (*e.g.*, a pose error caused by sensor saturation) to its root cause in time to intervene. Formal verification and interpretable models that guarantee predictable behavior across operating extremes remain open problems.

[32] https://github.com/mit-ll/spacegym-kspdg
[33] https://developer.nvidia.com/isaac/sim
[34] https://gazebosim.org/
[35] https://kelvins.esa.int/

Chapter 12

Spatial Intelligence for Biomedicine and Natural Sciences

Biomedicine and the natural sciences are, at their core, spatial disciplines: their objects are structures in space, from organs and tissues down to cells and molecules, and understanding them means reasoning about shape, arrangement, and interaction in three dimensions. This chapter turns the spatial-intelligence capabilities of the previous part onto that setting, where in both of its cases the unit of value is the same: a metrically faithful 3D model of a specific physical system, kept in correspondence with real measurements.

In medicine, that model is patient-specific, and it drives a closed loop running from imaging through planning to image-guided intervention. In the natural sciences the same perception and reconstruction machinery operates several orders of magnitude smaller, at the molecular and cellular scale, where the structures reconstructed are the very objects of scientific explanation.

12.1 Medicine and Clinical Healthcare

12.1.1 Introduction

Anatomy is a three-dimensional arrangement of organs with shape, volume, and topology; disease is *localized*, so that a lesion is a position and an extent in the body frame rather than a label on an image; and intervention is fundamentally an act of navigation, guiding an instrument to a target through tissue that must not be harmed.

The perception machinery of Chapter 6 (detection, segmentation, depth, pose) and the reconstruction and generation of Chapter 8 reappear here, specialized to volumetric data, to registration across modalities and time, and to a regulatory setting in which a wrong answer harms a patient. The unit of analysis is the individual patient and the individual procedure: the molecular and cellular scale belongs to the natural sciences (§12.2), the generic digital twin, of which the patient-specific medical twin below is a specialization, to §11.1, and the population scale of spatial epidemiology to the GeoAI of §11.2.

12.1.2 System Overview

Figure 12.1 draws clinical spatial intelligence as a closed loop around the patient: multimodal imaging (computed tomography, magnetic resonance, ultrasound, endoscopy, pathology) is acquired and the anatomy perceived, a patient-specific 3D model is built and a plan formed, the plan is executed under image guidance,

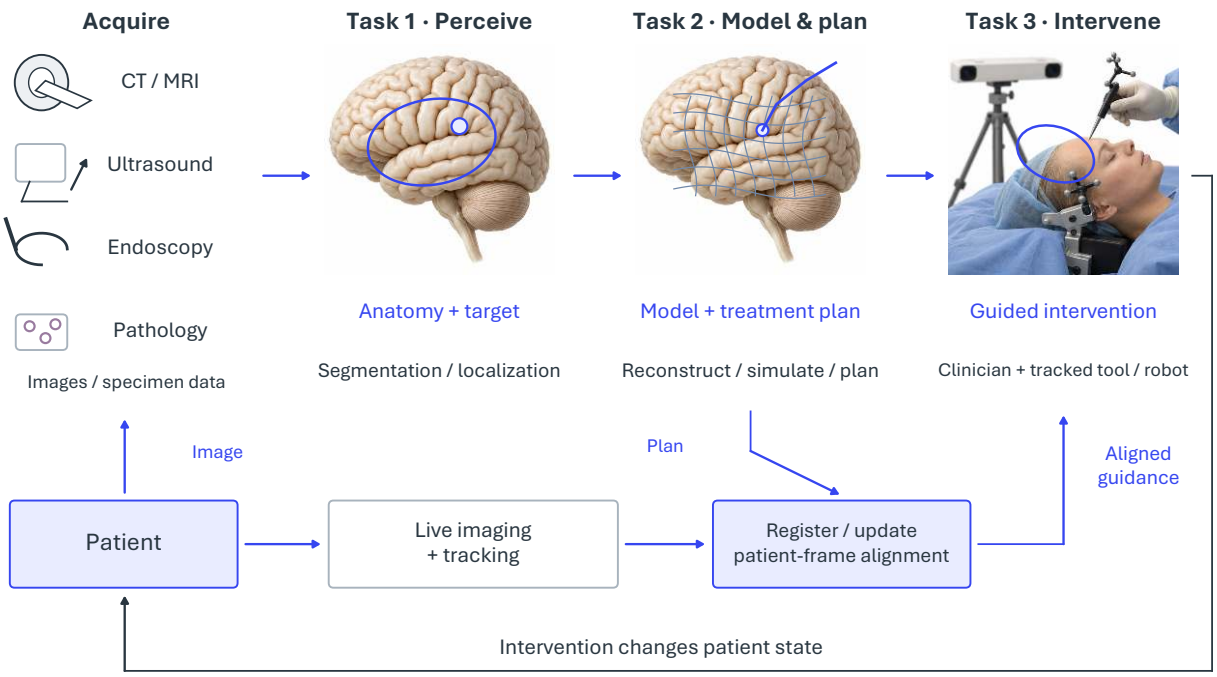


Figure 12.1: Clinical spatial intelligence with two connected information paths: acquired observations support anatomical perception and patient-specific planning, while live imaging and tracking update the alignment used for guided intervention. Physical intervention changes the patient state, prompting subsequent observation. Anatomical scenes are schematic; pathology may require specimen correspondence.

and the outcome is imaged anew. Registration connects the image, patient, and instrument frames throughout this pipeline; the recurring difficulty is keeping the virtual model aligned with the physical patient as tissue moves and deforms. The three pivotal tasks map onto the loop directly (Task-1 perceives, Task-2 models and plans, Task-3 acts), and Table 12.1 collects their benchmarks.

12.1.3 Pivotal Task-1: Volumetric Medical Image Perception

The first task perceives the anatomy. Medical perception is intrinsically volumetric: it converts a voxel grid into delineated 3D anatomical objects with measurable shape and volume, localizes disease in the body frame, and aligns observations across modalities and time. Figure 12.2 illustrates three complementary outputs: segmented organs, a localized candidate finding, and a cross-modal alignment; these tasks need not form a serial pipeline.

Volumetric segmentation. Segmentation is the spatial core of medical imaging, and its history is a move from two-dimensional slices to full volumes. The encoder-decoder U-Net [174] founded the area, and its volumetric successors, the 3D U-Net [268] and the V-Net [2835] (which introduced the now-ubiquitous Dice loss to handle voxel class imbalance), moved it into three dimensions. The self-configuring nnU-Net [2836] adapts preprocessing, architecture, and training to each dataset and remains the baseline that is hard to beat, while transformer encoders such as UNETR [2837] and Swin-UNETR [2838] add global context. At the whole-body scale, TotalSegmentator [2839] delineates more than a hundred anatomical structures in a single CT, functioning as a spatial atlas in routine use. These methods are evaluated by the Dice similarity coefficient for volumetric overlap and by boundary-geometry metrics, the 95th-percentile Hausdorff distance and the average symmetric surface distance, that capture shape errors which overlap alone misses [2840].

Medical foundation models. The recent shift mirrors the rest of vision: from per-task specialists to promptable foundation models that segment arbitrary anatomy from spatial prompts. The Segment Anything Model [138] catalyzed the move, and MedSAM [2841] adapted it by fine-tuning on more than a million

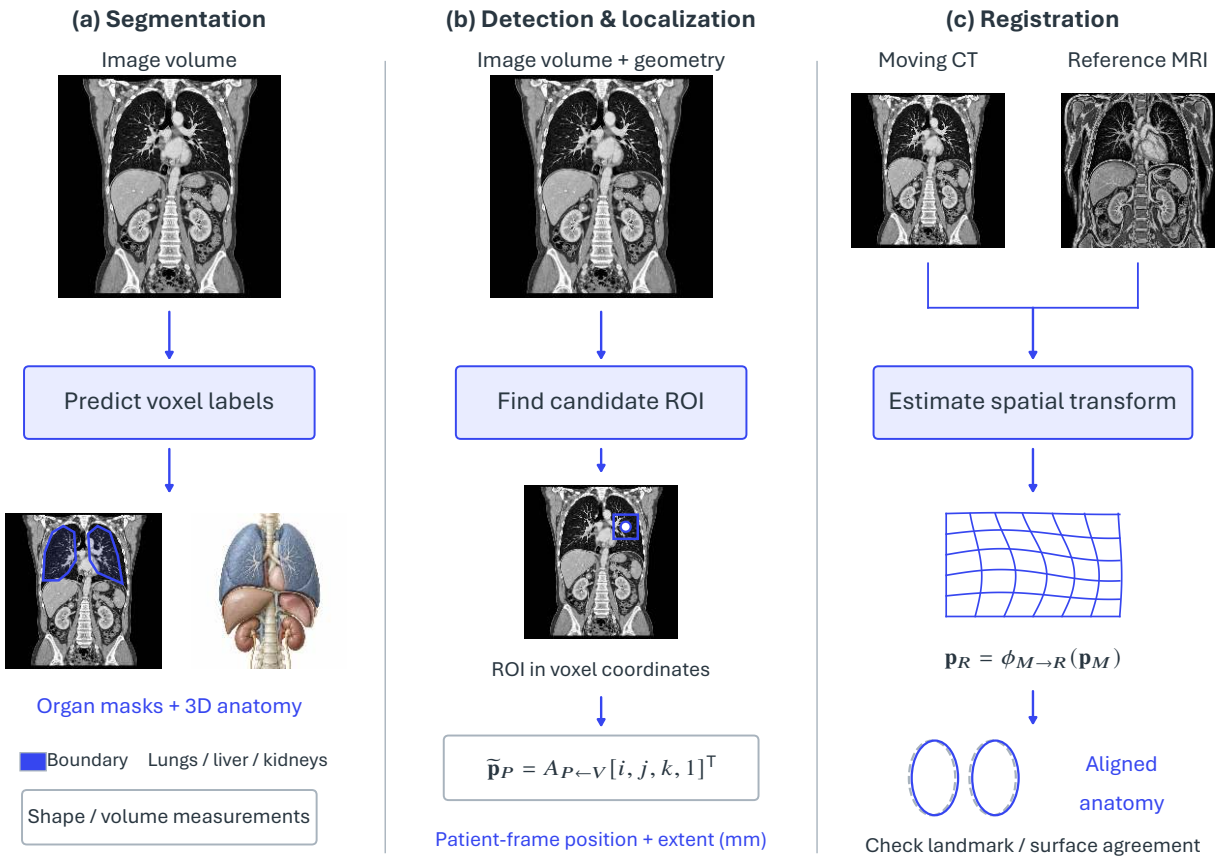


Figure 12.2: Three input-operation-output views of medical perception: (a) image volumes yield voxel labels, organ masks, and 3D anatomy; (b) detected candidate regions are mapped from voxel indices to physical patient coordinates using image geometry; and (c) moving and reference modalities jointly determine a spatial transform for anatomical alignment. Synthetic CT/MRI-style images and the hypothetical ROI illustrate task structure, not measured clinical findings or a validated registration result.

medical image–mask pairs. Because anatomy is volumetric, the decisive step is into 3D: SAM-Med3D [2842] replaces the slice-wise encoder with a fully three-dimensional one, and SegVol [2843] supports both spatial and textual prompts over hundreds of anatomical categories. Much of this work is built on the open MONAI framework [2844], whose medical-aware 3D transforms and model zoo have become standard infrastructure.

Detection and landmark localization. Detection localizes a finding in the body frame, and landmark localization fixes canonical anatomical coordinates; both are spatial tasks distinct from whole-image classification. Lung-nodule detection is anchored by the LIDC-IDRI reference database [2845] and the LUNA16 benchmark [2846], which standardized evaluation on the free-response operating characteristic (sensitivity against false positives per scan). Anatomical landmarks can be found by treating localization as a sequential spatial search, as in the reinforcement-learning agent of Ghesu *et al.* [2847] that navigates the volume to a target.

Registration as spatial alignment. Registration brings two images into a common coordinate frame and is foundational for multimodal fusion, atlas-based labeling, longitudinal tracking, and surgical planning. Learning-based methods reframe it as predicting a dense deformation field in a single forward pass: VoxelMorph [2848] established the unsupervised CNN formulation, and TransMorph [2849] added a transformer backbone. These build on classical diffeomorphic and intensity-based methods, principally the symmetric diffeomorphic SyN [2850] and the intensity-based toolbox elastix [2851], and are measured by the target registration error between corresponding landmarks. Segmentation and detection are benchmarked on

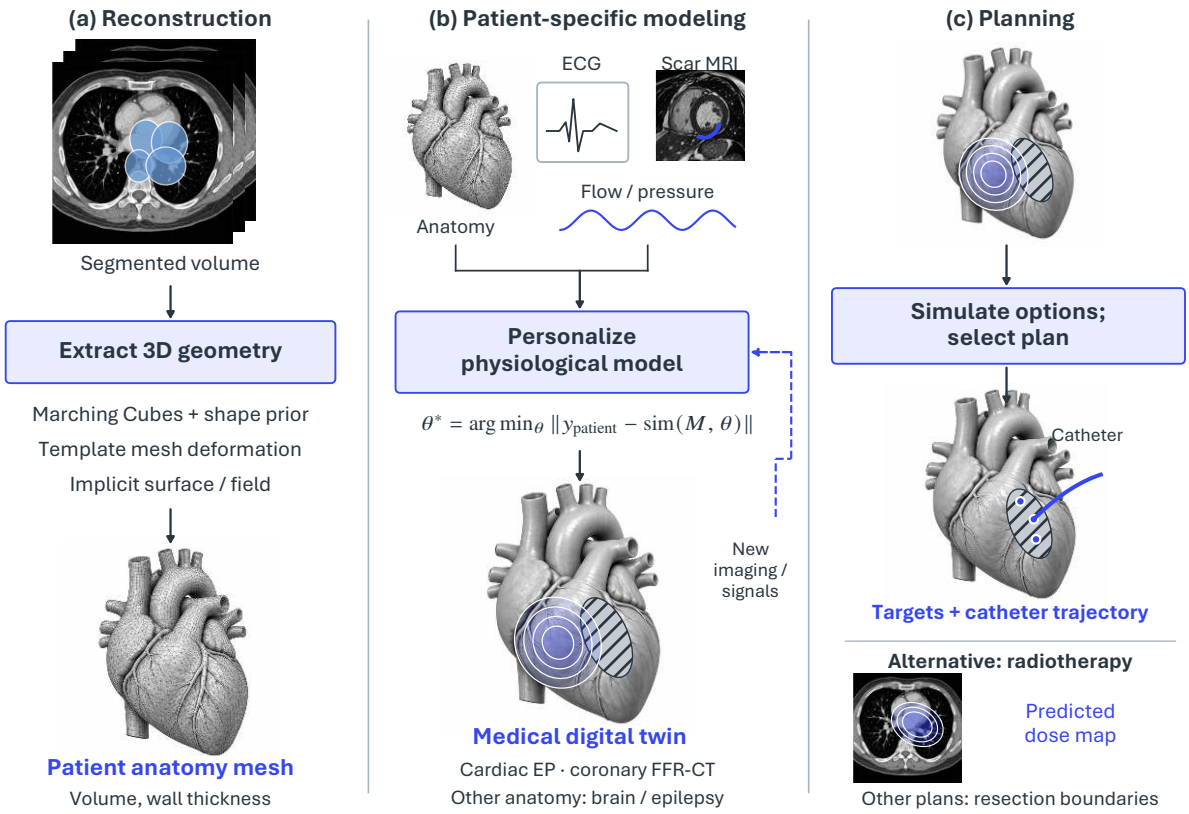


Figure 12.3: Anatomical reconstruction, modeling, and planning: a volumetric segmentation becomes a 3D anatomical mesh, is personalized into a medical digital twin (*e.g.*, a cardiac electrophysiology or blood-flow model), and is used to simulate outcomes and design a surgical or radiotherapy plan. New patient observations update model personalization. Cardiac targeting and radiotherapy dose prediction are alternative planning examples; synthetic anatomy and vector overlays illustrate the workflow rather than clinical results.

generalist suites such as the Medical Segmentation Decathlon [2852], the brain-tumor benchmark BraTS [2853], the abdominal multi-organ datasets AMOS [2854] and BTCV [2855], the kidney-tumor KiTS19 [2856], and the cardiac ACDC [2857].

12.1.4 Pivotal Task-2: Anatomical Reconstruction, Modeling, and Surgical Planning

The second task turns the perceived anatomy into a patient-specific 3D model that can be measured, simulated, and planned upon, bridging from pixels to clinical decisions. Figure 12.3 previews this progression from a segmentation to an anatomical mesh, a patient-specific simulation, and a surgical plan.

3D anatomical reconstruction. A segmentation becomes a usable model once it is turned into geometry. The classical route extracts an isosurface mesh by Marching Cubes [105], optionally constrained by a statistical shape model that encodes anatomical priors as a learned distribution over landmarks, an idea founded by Active Shape Models [518] and surveyed for medical use by Heimann and Meinzer [2858]. Deep methods deform a template mesh directly from the volume, as in Voxel2Mesh [2859] and the whole-heart reconstruction of Kong *et al*. [2860]. Implicit neural representations supply a continuous alternative, with DeepSDF [117] and Occupancy Networks [118] as the substrate (Chapter 8) and medical adaptations such as the neural attenuation fields of NAF [2861] reconstructing volumes from sparse-view cone-beam CT.

Patient-specific modeling and the medical digital twin. A geometric model becomes a *twin* when it is endowed with patient-specific physiology and kept faithful to a particular patient, a medical specialization

of the generic digital twin of §11.1. What is clinically specific is that the model is built from the patient's own imaging and physiological signals, governed by mechanistic physiology rather than CAD, and bound by regulatory credibility. Cardiac twins are the most mature instance: the vision was set out by Corral-Acero *et al*. [2694], the mechanistic backbone is the multiscale cardiac modeling reviewed by Niederer *et al*. [2862], and a personalized virtual heart built from scar imaging has been used to guide ablation of ventricular tachycardia [2863]. Hemodynamic twins compute fractional flow reserve from coronary CT by simulating blood flow [2864], and the paradigm extends to the brain, where a virtual epileptic patient estimates the epileptogenic zone to guide surgery [2865]. Such models also underwrite in-silico trials, in which virtual patient cohorts partially substitute for physical ones [2866], a route that regulators have begun to accept, as in the VICTRE imaging trial [2867]. The broader landscape is surveyed in recent reviews of digital twins for health [2695].

Surgical and treatment planning. The patient model is finally used to plan. Open platforms such as 3D Slicer [2868] assemble segmentation, registration, and 3D modeling into a planning workflow, and real-time finite-element engines such as SOFA [2869] support interactive surgical simulation and rehearsal on deformable tissue. In radiotherapy, planning is increasingly cast as a learning problem of predicting an achievable dose distribution from anatomy, standardized by the OpenKBP challenge [2870]. Across these settings the plan is a spatial object, a trajectory, a dose map, or a resection boundary, defined on the patient's own anatomy.

12.1.5 Pivotal Task-3: Image-Guided Intervention and Surgical Navigation

The third task brings the plan into the operating room and acts within the patient's live coordinate frame. The central problem is patient-to-image registration under real-world deformation and latency: the preoperative model must be aligned to the physical patient, and instruments tracked within that frame, in real time. Figure 12.4 previews registered preoperative anatomy and augmented-reality guidance alongside the complementary reconstruction of a deforming endoscopic scene.

Surgical navigation. Navigation aligns the preoperative model to the patient and tracks instruments within that frame, typically by optical tracking, which is sub-millimeter but requires line of sight, or electromagnetic tracking, which tolerates occlusion but is perturbed by metal. The defining difficulty is deformation: in neuronavigation, the brain shifts by up to centimeters once the skull is opened, invalidating the rigid preoperative registration, which is why intraoperative imaging and learning-based compensation are active concerns.

Augmented-reality guidance. Rather than divert the surgeon's gaze to a screen, augmented reality overlays the segmented anatomy directly on the surgical field, a use of the XR runtime treated in §10.1. The approach is most mature in spine surgery, where the Augmedics xvision system, the first FDA-cleared augmented-reality navigation platform, projects the patient's CT-derived spine onto a transparent head-mounted display for pedicle-screw placement [2871].

Endoscopic spatial perception. Minimally invasive surgery presents a hard perception problem: a monocular, moving camera views a deforming, specular, poorly lit scene. Surgical workflow and instruments are recognized from video, as standardized by the Cholec80 dataset and the EndoNet architecture for joint tool and phase recognition [2872]. Geometry is recovered by self-supervised depth and visual odometry, benchmarked by the SCARED challenge [2873] and the EndoSLAM dataset [2874], and the deforming surgical scene is reconstructed in 3D by dynamic neural rendering, as in EndoNeRF [2875], a clinical specialization of the neural reconstruction of Chapter 8.

Surgical robotics. Robotic systems execute the plan within the patient frame; their manipulation and control are the subject of §9.1, and the concern here is the spatial perception and autonomy they enable. The da Vinci Research Kit [2876] is the standard open research platform, and autonomy is commonly graded on a six-level scale from teleoperation to full autonomy [2877]. Most deployed systems remain teleoperated,

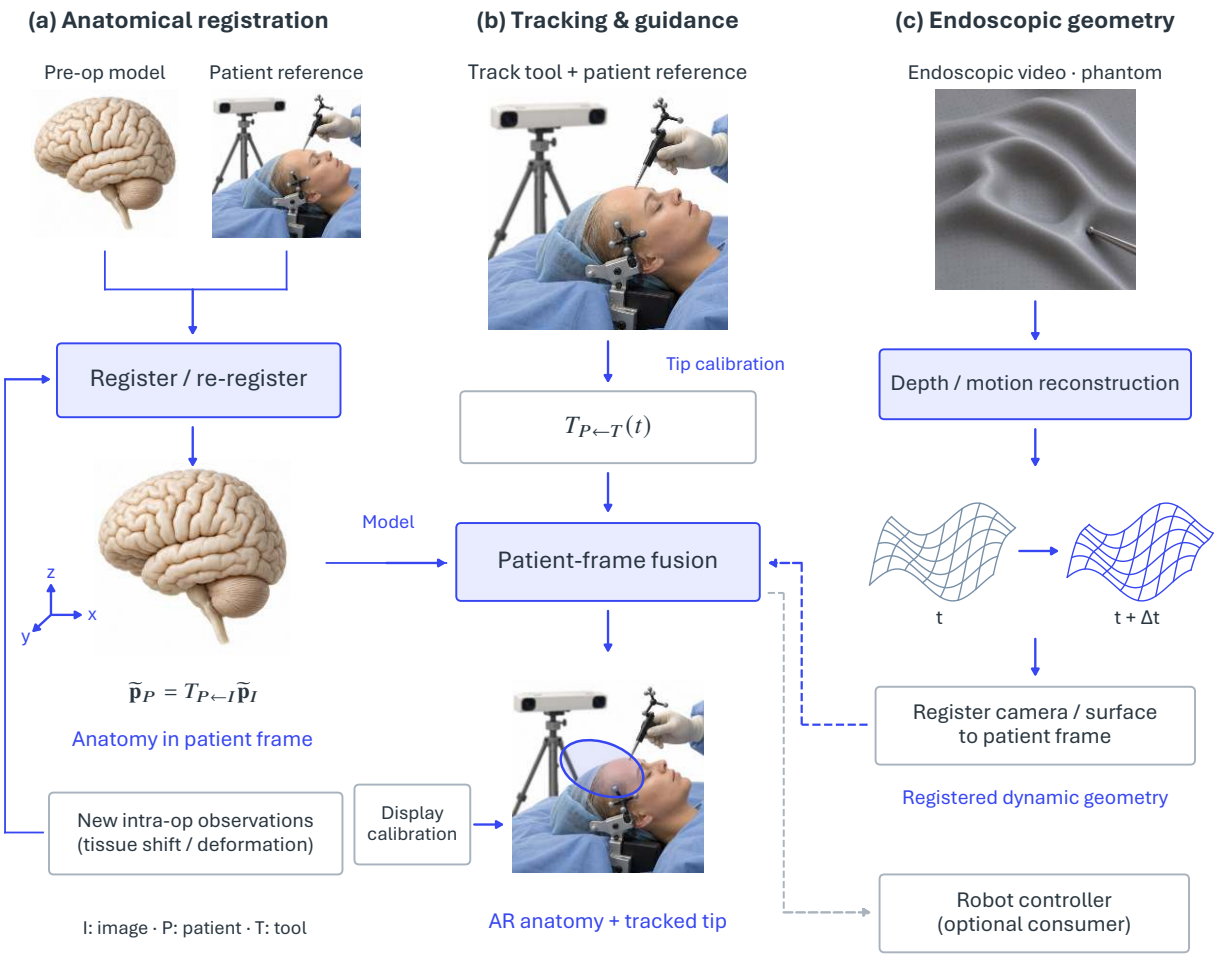


Figure 12.4: Patient-frame guidance combines registered anatomy and a calibrated tracked-tool pose. New intraoperative observations update anatomical registration; display calibration supports AR presentation. A separate endoscopic branch reconstructs dynamic geometry and registers it to the patient before optional fusion. The robot controller consumes the fused state rather than raw endoscopic imagery. Dashed connections denote optional downstream use. Anatomical scenes, overlays, and silicone-phantom reconstruction are schematic; transforms act on homogeneous coordinates.

Table 12.1: Representative benchmarks for spatial intelligence in medicine, spanning the three pivotal tasks.

Pivotal task	Benchmark / dataset	Representative method	Headline metric
Segmentation	MSD [2852]	nnU-Net [2836]	Dice / HD95
Segmentation	BraTS [2853]	Swin-UNETR [2838]	Dice
Multi-organ	AMOS [2854] / BTCV [2855]	UNETR [2837]	Dice
Detection	LUNA16 [2846]	3D CNN detectors (challenge entries) [2846]	FROC sensitivity
Registration	multi-site brain MRI (OASIS, ADNI, ABIDE)	VoxelMorph [2848]	TRE / Dice
Reconstruction	MM-WHS [2880]	whole-heart mesh [2860]	surface distance
Modeling	patient-specific cardiac	virtual heart [2863]	ablation-target agreement
Endoscopy	SCARED [2873]	EndoNeRF [2875]	depth error
Workflow	Cholec80 [2872]	EndoNet [2872]	phase accuracy

but the frontier of autonomous spatial action is exemplified by the Smart Tissue Autonomous Robot, which performs supervised-autonomous soft-tissue suturing using 3D tissue tracking [2878] and has been extended to in-vivo intestinal anastomosis [2879].

12.1.6 Established Tools and Systems

Open-source frameworks. The research stack runs from voxels to a planned model: **3D Slicer**[1] and **MITK**[2] provide segmentation, registration, and planning environments over the lower-level **ITK** and **SimpleITK**[3] engines; **MONAI**[4] supplies 3D-native deep-learning components and **nnU-Net**[5] the standard segmentation pipeline; and the web viewer **OHIF**[6] renders and annotates volumes in the browser.

Navigation and surgical-robotics vendors. The clinical stack carries the model from preoperative registration to constrained in-OR action. Navigation systems from **Brainlab**[7], **Medtronic** (StealthStation)[8], and **Stryker**[9] register and track instruments, and **Augmedics**[10] adds an augmented-reality head-mounted overlay. Surgical robots span teleoperated soft-tissue systems (**Intuitive** da Vinci and Ion[11], **Medtronic** Hugo, **CMR Surgical** Versius[12]) and CT-planned, haptically constrained orthopedic systems (**Stryker** Mako, **Zimmer Biomet** ROSA[13]).

Imaging-AI and clinical platforms. A deployed layer turns spatial findings into care. **Aidoc**[14] triages critical findings on CT, **Viz.ai**[15] alerts stroke teams to large-vessel occlusions, **HeartFlow**[16] computes non-invasive fractional flow reserve from coronary CT, **PathAI**[17] quantifies biomarkers on whole-slide pathology, and **Tempus**[18] fuses imaging with genomic and clinical data.

12.1.7 Challenges

Every capability in the clinical loop must be accurate, robust to deformation, and trustworthy enough to act on a patient. The open problems follow from those demands: generalizing across scanners and sites, obtaining labels under privacy constraints, registering deforming tissue during surgery, earning regulatory and clinical trust, and meeting real-time budgets while fusing imaging with non-imaging evidence.

Generalization across Scanners and Sites. Models trained at one institution degrade at another, because scanner vendor, reconstruction kernel, and acquisition protocol induce a covariate shift that curated benchmarks underrepresent. Domain generalization and adaptation are therefore prerequisites for deployment, and leaderboard results routinely overstate performance at a new site.

Data Scarcity, Annotation Cost, and Privacy. Volumetric expert annotation is slow and expensive, and patient data is locked in privacy-restricted silos. Self-supervised pretraining reduces the labeling burden [2881], and federated learning trains shared models without moving data across institutions [2882], but label efficiency and privacy-preserving collaboration remain limiting.

[1] https://www.slicer.org/
[2] https://www.mitk.org/
[3] https://itk.org/
[4] https://monai.io/
[5] https://github.com/MIC-DKFZ/nnUNet
[6] https://ohif.org/
[7] https://www.brainlab.com/
[8] https://www.medtronic.com/
[9] https://www.stryker.com/
[10] https://augmedics.com/
[11] https://www.intuitive.com/
[12] https://cmrsurgical.com/
[13] https://www.zimmerbiomet.com/
[14] https://www.aidoc.com/
[15] https://www.viz.ai/
[16] https://www.heartflow.com/
[17] https://www.pathai.com/
[18] https://www.tempus.com/

Non-Rigid Intraoperative Registration. The hardest open spatial problem is keeping the model aligned with the patient as tissue moves. Brain shift after craniotomy and soft-tissue deformation during surgery invalidate rigid preoperative registration, and while learning-based deformable registration [2848] addresses the static case, real-time non-rigid intraoperative alignment under sparse, partial observation is not solved.

Validation, Regulation, and Trust. A model that changes a clinical decision needs prospective validation and regulatory clearance before use, and at the bedside it needs calibrated uncertainty and explanations that support appropriate rather than blind reliance. Retrospective benchmark accuracy establishes none of these.

Real-Time Constraints and Multimodal Fusion. Intraoperative guidance must run within a tight latency budget while the scene deforms, a constraint that few surgical-video algorithms yet meet without specialized hardware. At the same time, the richest clinical picture comes from fusing imaging with electronic health records and genomics [2883], so spatial perception must ultimately integrate with non-spatial evidence.

12.2 Natural Science

12.2.1 Introduction

A physical system is a field over space; a molecule or a material is a three-dimensional arrangement of atoms together with an electronic field over that arrangement; a protein, a cell, or a tissue is a three-dimensional structure whose function follows from its form. In every case the scientific quantity of interest, whether an energy, a reaction rate, a biological function, or a material property, is a function of spatial structure under strict symmetry constraints, so that understanding nature is to a large degree a problem of recovering and reasoning about that structure.

The geometric representations of Chapter 4 and the reconstruction and generation of Chapter 8 reappear here specialized to data that obey physical law: to fields constrained by differential equations, to atomic systems constrained by Euclidean and permutation symmetry, and to biomolecules whose geometry determines their function. A single capability triad recurs inside every domain and gives the section its spine: spatial **representation and reconstruction** recovers a continuous field or a 3D structure from sparse, indirect, and noisy measurements; **dynamics, simulation, and law discovery** learns how that structure evolves and what governs it; and **design, discovery, and control** either generates new structure or inverts observations for hidden causes. The third leg reads as discovery and control in physics and as generative design in chemistry and life science.

The section is organized by scientific domain (physics, chemistry, and life science); the earth and environmental sciences are treated at geographic scale in the GeoAI of §11.2, the clinical, patient-scale counterpart of the life-science material in §12.1, and the generic digital twin in §11.1. Figure 12.5 previews the section as a cross-scale map in which the three domains share the same vertical triad, and Table 12.2 collects a representative benchmark for each capability.

12.2.2 Physics

Spatial intelligence in physics constructs spatial structure and discovers the laws that govern its motion, and it earns its place where first-principles solvers are too costly or the governing equations are unknown: fields are reconstructed from limited observations, their dynamics are modeled and their laws discovered, and the systems are optimized and controlled. Figure 12.6 previews this progression from a sparsely observed field to a learned dynamical rollout and a differentiable-control loop.

12.2.2.1 Spatial Representation and Reconstruction

Many key quantities in physical systems are available only through sparse and noisy observations, and representation and reconstruction recover continuous spatial fields from such data, the input to law discovery and control. Embedding physical priors such as governing partial differential equations and symmetry constraints keeps these reconstructions physically consistent in low-data regimes.

A classical task is physical field reconstruction, which aims to recover continuous physical fields such as temperature, pressure, and velocity from sparse sensor readings or boundary measurements; the central challenge lies in ensuring the physical plausibility of the solution under the ill-posed conditions caused by insufficient observations. **Physics-informed neural networks** are a classical approach for this task [2680], embedding partial differential equations such as the Navier–Stokes equations as soft constraints in the loss function and computing physical residuals via automatic differentiation, which enables accurate reconstruction of fluid and thermal fields in data-scarce settings; the same networks serve as real-time surrogates in digital twins (§11.1). **Neural fields** implicitly represent continuous physical fields by using coordinate-based networks that map spatial locations to physical quantities. Sitzmann *et al*. [2884] proposed SIREN, which employs periodic activation functions to capture high-frequency spatial details with high

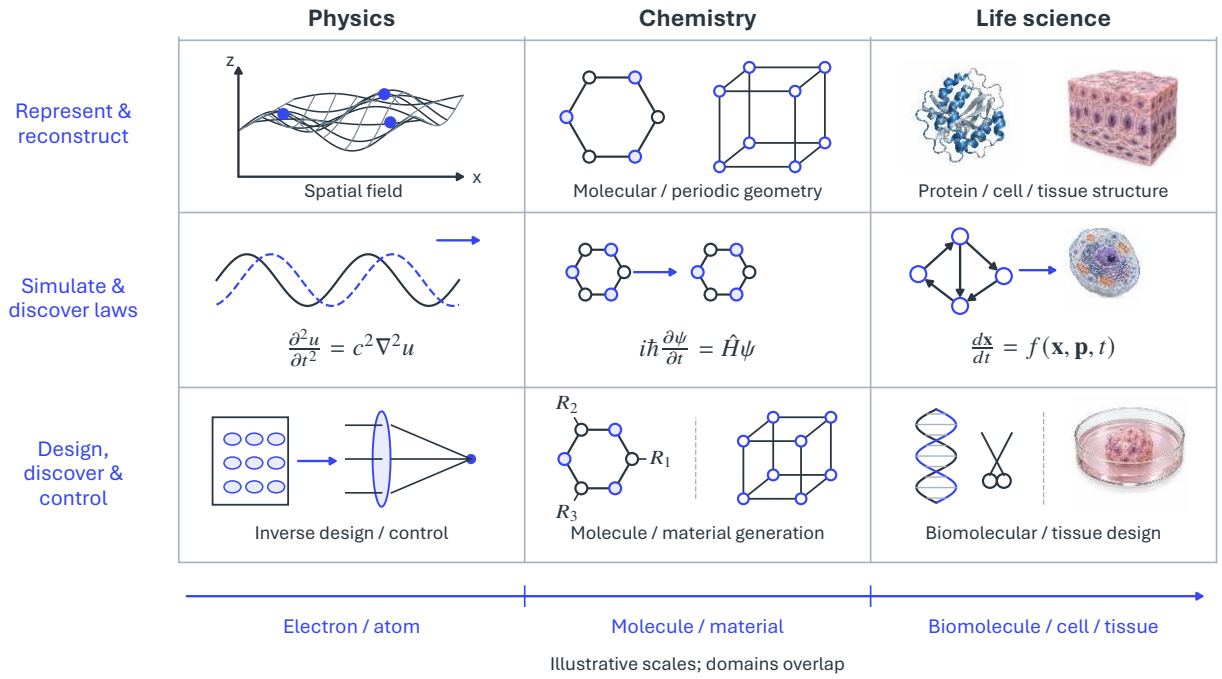


Figure 12.5: Spatial intelligence across physics, chemistry, and life science (columns), each repeating the same capability triad (rows). The lower electron-to-tissue strip indicates illustrative scales rather than exclusive domain boundaries. Fields, molecular and periodic structures, and biomolecular and tissue structures illustrate representation; the equations exemplify wave, quantum, and biological dynamics. Bottom-row design examples are schematic; molecule/material and biomolecular/tissue examples are alternatives rather than literal conversion pipelines. Biological imagery is synthetic.

fidelity and has been applied to solving Poisson equations and reconstructing wave fields. The general machinery of coordinate-based neural representation is developed in Chapter 8.

Inverse scattering aims to infer the internal structure or material parameters of an object from measured wave-scattering data, and it remains a canonical ill-posed problem in electromagnetics and acoustics [2885]. Wei and Chen proposed deep-learning schemes [2886] for full-wave nonlinear inverse scattering that directly invert dielectric-permittivity profiles from scattered fields, thereby moving beyond linearized Born-type approximations in practical reconstructions. SwitchNet [2887] introduced a unified neural architecture that learns both the forward map from scatterers to scattered fields and the inverse map from measurements to scatterer profiles, using a switching layer that exploits low-rank structure to reduce model complexity and facilitate training. Physics-guided losses that inject multiple-scattering near-field priors into the objective further improve robustness.

12.2.2.2 Dynamics, Simulation, and Law Discovery

Law discovery aims to learn the generative structures or explicit laws that govern the evolution of physical systems from observational data, emphasizing the automatic extraction of conservation laws, symmetries, and interpretable mechanisms beyond accurate prediction. Hamiltonian Neural Networks [2888] directly learn the Hamiltonian of a system, thereby enforcing energy conservation and symplectic structure at the model level and providing a principled foundation for interpretable dynamical modeling. Lagrangian Neural Networks [2889] parameterize arbitrary Lagrangians with neural networks, learning energy conserving dynamics without canonical coordinates, as validated on the double pendulum and a relativistic particle. Cranmer *et al*. [2890] combine graph networks with symbolic regression on their internal representations, recovering known force laws and Hamiltonians, and applying the approach to cosmology. The SINDy

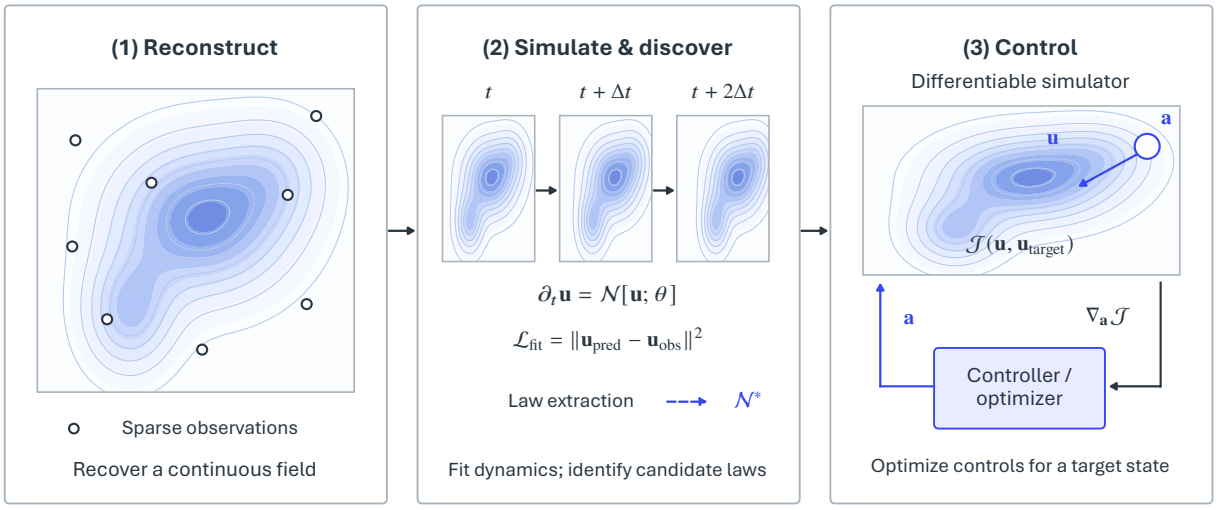


Figure 12.6: The physics triad: (1) reconstruct a field from sparse sensors, (2) fit dynamics and roll out the field, with a separate law-extraction step, and (3) optimize controls through a differentiable simulator. $\mathcal{N}[\mathbf{u};\theta]$ is the learned operator and $\mathcal{N}^*$ a candidate extracted law. The state $\mathbf{u}$ and control $\mathbf{a}$ are distinct; the control objective $\mathcal{J}$ supplies gradients to the optimizer, whereas $\mathcal{L}_{\text{fit}}$ fits observed dynamics. Field contours and temporal shifts are illustrative, not experimental results.

algorithm [2891] employs sparse regression to identify governing differential equations directly from time-series data, enabling data-driven discovery of interpretable physical laws.

Efficient physical solvers focus on predicting the spatio-temporal evolution of physical systems under known governing equations or interaction rules, with the goal of achieving fast and scalable computation while preserving physical consistency beyond the limits of traditional numerical methods. The Fourier Neural Operator [2892] learns mappings between function spaces rather than individual discretized instances, enabling trained models to predict the evolution of partial differential equations at arbitrary spatial resolutions without relying on a fixed computational grid. DeepONet [2893], grounded in the universal approximation theorem for operators, learns nonlinear mappings from input functions to output functions and thereby approximates the solution operators that govern physical systems. Interaction Networks [1201] introduced the relational reasoning architecture that learns physical interactions between objects, laying the foundation for graph-based physical simulation. The Graph Network Simulator [2894] employs graph neural networks to model spatial interactions among particles or objects, achieving high-fidelity simulations of fluids, rigid bodies, and deformable materials, and MeshGraphNets [2895] extend graph-based models to unstructured meshes, allowing efficient simulation of continuum mechanics on irregular geometries. For the classical challenge of turbulence modeling, Mohan *et al*. [2896] proposed spatio-temporal deep-learning models with physics-informed diagnostics that improve the prediction accuracy of three-dimensional turbulent flows, and Lusch *et al*. [2897] developed a deep-learning framework for universal linear embeddings of nonlinear dynamics, enabling low-dimensional representations and long-term prediction of nonlinear fluid systems. Neural representations of the many-body wavefunction are treated with electronic structure in the chemistry subsection.

12.2.2.3 Control and Optimization

Optimization and control in physical systems seek optimal trajectories, shapes, or control policies under governing physical laws and environmental constraints. Differentiable physics formulates physical simulators as differentiable operators, enabling gradient-based optimization over control inputs, design variables, and physical parameters, and providing a unified framework for optimal control, system identification, and

design. Neural Ordinary Differential Equations [2898] provide a differentiable formulation of continuous-time dynamical systems by parameterizing the vector field with a neural network and enabling gradient computation through adjoint sensitivity analysis, serving as a flexible building block for learning and control in continuous time. ChainQueen [2899] demonstrates differentiable simulation of soft and deformable bodies based on the moving least squares material point method, enabling gradient-based optimization in high-degree-of-freedom systems with contact and supporting applications in soft-robot control and co-design. DiffTaichi [2678] generalizes this paradigm through a differentiable programming framework for high-performance physical simulation, allowing efficient construction of differentiable solvers and gradient-driven optimization for fluid control, shape optimization, and material-parameter estimation; the same tools calibrate digital twins (§11.1). In practice, a mature software substrate of differentiable and physics-informed libraries, such as **DeepXDE**[19] and **NVIDIA Modulus**[20] for operator and PINN training, and **JAX-MD**[21] and **DeePMD-kit**[22] for differentiable and learned-potential molecular dynamics, turns this triad into runnable physics.

12.2.3 Chemistry

Chemistry is intrinsically spatial: a molecule or a material is a three-dimensional arrangement of atoms together with an electronic field over it, and its behavior, whether energy, reactivity, or a bulk property, is a function of that geometry under strict symmetry. Physical observables must be invariant or equivariant to the Euclidean group of rotations, translations, and reflections and to permutations of identical atoms; for crystalline solids they must also respect periodic lattice symmetry, and for electrons they must obey antisymmetry under exchange. Spatial intelligence in chemistry is therefore largely the art of building these symmetries into models that represent atomic and electronic structure, simulate its dynamics, and generate new structure under chemical feasibility. Figure 12.7 previews this triad from an equivariant representation of a 3D structure to a learned-potential simulation and a 3D generative design.

12.2.3.1 Spatial Representation and Reconstruction

The central representational problem of chemistry is to map a 3D atomic geometry to energies, forces, and properties while respecting symmetry, and successive representations have captured progressively more of that geometry. **Geometry-grounded potentials.** The Behler–Parrinello neural network potential founded the area by decomposing the total energy into atomic contributions computed from rotation-, translation-, and permutation-invariant symmetry functions of each atom's local environment [2900]. SchNet [2901] replaced hand-designed descriptors with continuous-filter convolutions over interatomic distances, giving a smooth invariant representation for energy and force prediction, while DimeNet [2902] added directional message passing that encodes bond angles through spherical harmonics, moving beyond pairwise distances to angular structure, and PaiNN [2903] extended message passing to rotationally equivariant vector features for tensorial properties and spectra. The Deep Potential model [2904] preserves all natural symmetries at linear cost, and the ANI potentials [2905] demonstrated transferable organic-chemistry accuracy at force-field cost. **Equivariant interatomic potentials.** The decisive recent step was to carry geometric tensors, not merely invariant scalars, through the network. NequIP [1380] introduced E(3)-equivariant convolutions over geometric tensors and achieved striking data efficiency, often requiring orders of magnitude less training data; Allegro [2906] made the representation strictly local and thereby scalable to 100 million atoms, and MACE [1381] used higher body-order equivariant messages to reach high accuracy in only two message-passing layers. A further shift is from per-system fits to **universal potentials** that span the periodic table: M3GNet [2907] couples a graph network with explicit three-body interactions trained on materials

[19] https://github.com/lululxvi/deepxde
[20] https://developer.nvidia.com/modulus
[21] https://github.com/jax-md/jax-md
[22] https://github.com/deepmodeling/deepmd-kit

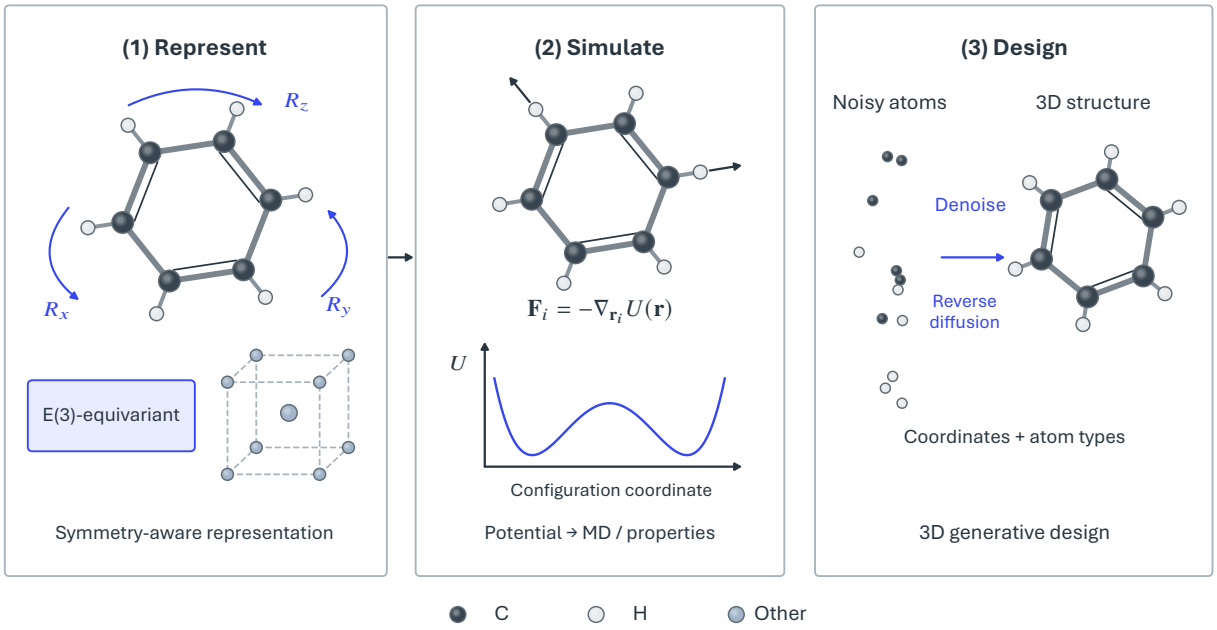


Figure 12.7: The chemistry triad: (1) symmetry-aware molecular and periodic representations, illustrated by E(3)-equivariant features; (2) a learned invariant interatomic energy U whose coordinate gradients provide forces for molecular dynamics, with related representations supporting property prediction; and (3) complementary 3D generative design by reverse diffusion. Molecular graphs, force directions, and the one-dimensional energy profile are schematic, not a calculated trajectory or energy surface. The denoising example preserves the atom count; generative methods may model coordinates and atom types jointly.

relaxations, and CHGNet [2908] augments the 3D-structure graph with atomic magnetic moments to capture electronic degrees of freedom alongside geometry. These equivariant backbones share machinery with the geometric representations of Chapter 4. **Neural electronic structure.** At the quantum scale the spatial object is the many-electron wavefunction itself. Neural quantum states parameterize that wavefunction with a neural network over spin configurations [2909], and deep variational ansätze such as FermiNet [2910] and PauliNet [2911] operate directly on 3D electron coordinates while enforcing exact antisymmetry, solving the electronic Schrödinger equation to chemical accuracy from nuclear positions alone.

12.2.3.2 Dynamics, Simulation, and Property Prediction

The potentials of the previous thread are also the simulators: once a model maps a 3D configuration to energies and forces, it drives molecular dynamics at a fraction of the cost of first-principles calculation, and the same representations predict quantum properties such as energies, dipoles, and frontier-orbital gaps directly from coordinates. The gradient-domain machine-learning force field made this explicit by exploiting energy conservation and the geometry of conformations, and it established the widely used MD17 trajectory benchmark of energies and forces for small molecules [2912]. Heterogeneous catalysis pushes the same energy-and-force prediction task to a far larger and more diverse chemical space, spanning many combinations of metal surfaces and adsorbates. GemNet-OC [2913] extends directional message passing to this setting, delivering accurate energies and forces across the surfaces and adsorbates of the Open Catalyst datasets. In parallel, structure-to-property prediction is standardized on QM9 [1379], a dataset of roughly 134,000 small organic molecules with density-functional geometries and quantum properties, on which the equivariant networks above are routinely evaluated by the mean absolute error of predicted properties. This ecosystem rests on open toolkits, including the equivariant-network library **e3nn**[23], the

[23] https://e3nn.org/

atomistic-simulation environment **ASE**[24], and the quantum-chemistry package **PySCF**[25], together with universal-potential foundation models such as MACE-MP, and on reference corpora such as the Materials Project [2914], the Open Catalyst dataset [2915], and the MoleculeNet benchmark suite [2916].

12.2.3.3 Generation and Inverse Design

The inverse of property prediction is design: generating new 3D structures that satisfy desired properties under chemical feasibility, by sampling on the appropriate geometric manifold rather than enumerating candidates. G-SchNet [2917] autoregressively places atoms in 3D Euclidean space with each placement respecting the rotational invariance and local symmetry of those already placed, the first strong 3D-native generator. Diffusion models then became the dominant paradigm: the equivariant diffusion model [2918] jointly denoises atom coordinates and types under E(3) equivariance, GeoDiff [2919] generates conformers from the molecular graph through roto-translation-equivariant Markov kernels that induce an invariant distribution, and torsional diffusion [2920] restricts the process to the hypertorus of torsion angles, the true intrinsic conformational degrees of freedom, achieving state-of-the-art conformer quality on GEOM-Drugs [2921] at a fraction of the cost. For crystalline materials, CDVAE [2922] generates stable structures under periodicity and SE(3) invariance and supports inverse design by optimizing target properties in latent space. The line has since moved to diffusion directly on the periodic structure: DiffCSP [2923] jointly denoises lattice and fractional coordinates under periodic equivariance for crystal structure prediction, and MatterGen [2924] scales this into a property-conditioned foundation model for inorganic materials design, with generated candidates validated by experimental synthesis. The general diffusion machinery underlying these models is developed in Chapter 8; what is chemistry-specific is the constraint manifold (valence and bonding feasibility, energy-minimum geometries, lattice periodicity) on which generation must take place.

12.2.4 Life Science

At the molecular, cellular, and tissue scale of basic biology the object of interest is again a three-dimensional structure: a protein fold, a molecular complex, a spatial map of gene expression, or the wiring of a neural circuit, and in each case function follows from spatial form. Spatial intelligence here recovers that structure from indirect measurement, models how biomolecules interact and move, and designs new structure. Figure 12.8 previews the triad from structure recovery through interaction modeling to generative design.

12.2.4.1 Spatial Representation and Reconstruction

The defining achievement of spatial intelligence in biology is the recovery of 3D structure from indirect signals. **Protein structure prediction** maps a one-dimensional sequence to a three-dimensional fold. The original AlphaFold predicted inter-residue distance distributions as a learned potential to fold the chain [2925], and AlphaFold2 [2926] reached near-experimental accuracy through an end-to-end architecture that reasons explicitly over residue frames and pairwise geometry, the exemplar of spatial intelligence in the natural sciences. AlphaFold3 [2927] generalized this to the joint structure of proteins, nucleic acids, and small molecules through a diffusion decoder; RoseTTAFold [2928] let one-dimensional, two-dimensional, and three-dimensional information flow between tracks, and ESMFold [2929] showed that atomic-level structure emerges from a protein language model without explicit alignments, enabling metagenomic-scale folding. **Cryo-electron microscopy** solves the inverse problem of reconstructing a 3D density map from many noisy 2D projections. RELION cast this as Bayesian refinement [2930], CryoSPARC enabled rapid ab-initio reconstruction without a starting model [2931], and cryoDRGN used a coordinate-based neural decoder to reconstruct continuous distributions of conformations and thereby resolve molecular heterogeneity and motion [2932]. **Spatial transcriptomics** places gene expression back into tissue space. The founding

[24] https://wiki.fysik.dtu.dk/ase/
[25] https://pyscf.org/

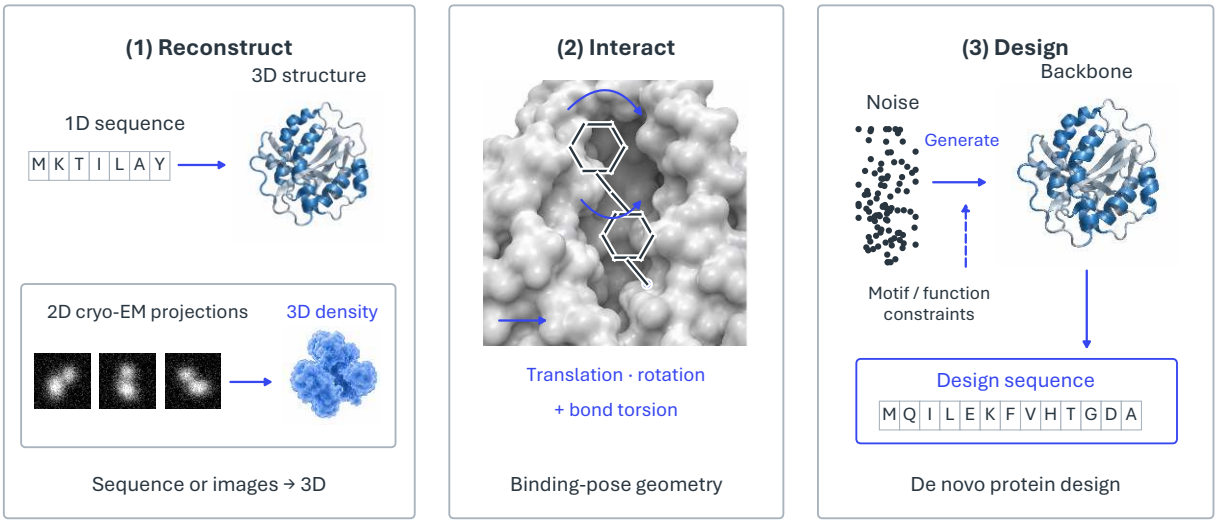


Figure 12.8: The life-science triad: (1) complementary sequence-to-structure prediction and cryo-EM projection-to-density reconstruction, (2) ligand binding-pose geometry with translation, rotation, and bond torsion, and (3) de novo backbone generation followed by sequence design. These are complementary capabilities, not an obligatory serial pipeline. Protein, pocket, and density renderings, noisy projection examples, and sequence fragments are schematic; they do not constitute experimentally validated structures or designs.

barcoded-array assay preserved the positional information of the transcriptome in tissue sections [2933]; Tangram learns a spatial mapping that aligns dissociated single cells onto a spatial reference [2934], and novoSpaRc reconstructs spatial arrangement from expression alone through optimal transport, assuming that expression-similar cells lie physically close [2935]. **Connectomics** reconstructs neural wiring from electron-microscopy volumes, where flood-filling networks iteratively trace the 3D morphology of individual neurites through the volume [2936].

12.2.4.2 Dynamics, Simulation, and Interaction

Beyond static structure, spatial intelligence models how biomolecules bind and interact. Molecular docking predicts the 3D pose in which a ligand binds a protein, and recent methods treat it as a geometric problem on the natural manifold of poses. DiffDock [2937] formulates docking as diffusion over the product space of ligand translation, rotation, and torsion, sampling binding poses in their intrinsic degrees of freedom, while EquiBind [2938] regresses the binding location and pose in a single shot using an SE(3)-equivariant network whose equivariance encodes the rotational and translational symmetry of the binding geometry. Protein–protein and complex geometry is predicted by extending folding models to multiple chains [2939], and AlphaFold3 above provides a generative alternative for protein–ligand and protein–nucleic-acid complexes. The dynamics of biomolecules in turn rest on the learned interatomic potentials of the chemistry subsection, which accelerate the molecular dynamics that sample conformational change.

12.2.4.3 Generation and Design

The generative leg designs new 3D biological structure. RFdiffusion [2940] adapts structure-denoising diffusion to generate novel protein backbones conditioned on functional and geometric constraints such as motif scaffolding, binders, and symmetric assemblies, and ProteinMPNN [2941] solves the inverse-folding problem of designing an amino-acid sequence that folds into a given 3D backbone, the standard partner to backbone generation. Chroma [2942] couples equivariant graph networks with a diffusion process to sample novel all-atom structures and sequences steerable toward desired properties, illuminating regions of

protein space never explored by evolution. As in chemistry, the underlying diffusion machinery is that of Chapter 8; what is biology-specific is that the generated geometry must be foldable and functional, which is why design is paired with structure prediction and, ultimately, wet-lab validation. The field is supported by open infrastructure, including the AlphaFold Protein Structure Database and AlphaFold3 server[26], the ESM Metagenomic Atlas[27], and the open-source folding stack OpenFold[28], together with reference corpora such as the Protein Data Bank, the CASP blind-assessment benchmark [2943], and the PDBBind docking benchmark [2944].

Table 12.2: Representative benchmarks for spatial intelligence across the three natural-science domains.

Domain	Capability	Benchmark / dataset	Representative method	Headline metric
Physics	PDE surrogate	Navier–Stokes / Darcy flow	FNO [2892]	relative L_2 error
Physics	Particle dynamics	Learning-to-Simulate suite (Water / Sand / Goop)	GNS [2894]	rollout MSE
Chemistry	Property prediction	QM9 [1379]	DimeNet [2902]	property MAE
Chemistry	Force field	MD17 [2912]	NequIP [1380]	energy / force MAE
Chemistry	Catalysis	Open Catalyst [2915]	GemNet-OC [2913]	energy / force MAE
Chemistry	Conformer generation	GEOM-Drugs [2921]	torsional diffusion [2920]	coverage / matching RMSD
Life science	Protein folding	CASP [2943]	AlphaFold2 [2926]	GDT-TS
Life science	Docking	PDBBind [2944]	DiffDock [2937]	top-1 RMSD < 2 Å (%)
Life science	Cryo-EM heterogeneity	EMPIAR single-particle datasets	cryoDRGN [2932]	resolution (FSC)
Life science	Protein design	Motif-scaffolding / binder design	RFdiffusion [2940]	in-silico design success rate

12.2.5 Challenges

A learned model of nature has to respect the symmetries and conservation laws of its domain and remain correct outside the data it was fitted to. The six problems below follow from that requirement: how to build symmetry in, how to bridge scales, how to solve ill-posed inverse problems, how to learn from scarce and expensive labels, how to generalize to genuinely new systems, and how to make predictions physically consistent and trustworthy.

Symmetry and Equivariance. Physical observables are constrained by symmetry: Euclidean invariance and equivariance, permutation of identical particles, antisymmetry of electrons, and periodicity of crystals. Whether to build these symmetries into the architecture or to learn them from data is the central design tension, trading the cheaper but less expressive invariant features [2901] against the richer but costlier equivariant tensor features [1380], and it recurs from interatomic potentials to protein structure.

Extreme Multi-Scale Structure. Spatial structure in nature spans many orders of magnitude, from electrons and atoms to molecules and materials, and from biomolecules through cells to tissue. Learned potentials now reach a hundred million atoms [2906], still far below cellular scale; no single representation natively bridges electronic-structure accuracy and mesoscale reachability, and the handoffs between scales, through coarse-graining or learned-potential dynamics, are lossy and remain only partially solved.

Ill-Posed Inverse Problems. Much of natural science is an inverse problem in which a high-dimensional spatial field must be recovered from sparse, indirect, and noisy measurements: field reconstruction from a few sensors, inverse scattering [2886], and the recovery of 3D density from 2D cryo-electron-microscopy projections [2932] all share this under-determined character and demand strong learned priors to yield a unique and physical solution.

[26] https://alphafoldserver.com/
[27] https://esmatlas.com/
[28] https://github.com/aqlaboratory/openfold

Data Scarcity and the Cost of Ground Truth. High-fidelity labels are expensive: accurate energies and forces require costly quantum-chemistry calculations, which is why force-field benchmarks such as MD17 cover only small molecules [2912], and validated 3D biological structures require slow experiments, so the space of trustworthy training data is small relative to the space of possible systems. This drives the field's emphasis on data efficiency, multi-fidelity learning, and physics-guided objectives.

Generalization Beyond the Training Distribution. Models must extrapolate to unseen chemistry, novel protein folds, off-equilibrium and reactive geometries, and configurations far from those they were trained on. Discovery is by definition out of distribution, so leaderboard accuracy says little about it; this is why generative materials design is now judged by whether its candidates can actually be synthesized [2924].

Physical Consistency and Trust. Predictions that violate conservation laws or come without calibrated uncertainty cannot support a scientific conclusion. Building conservation into the model, as Hamiltonian networks do for energy [2888], and quantifying confidence are what let a learned model extend the scientific record rather than interpolate it.

Chapter 13

Spatial Intelligence for Intelligent Assistance



NTELLIGENT Assistance is the assistant-facing application of spatial intelligence: an agent must perceive a user's lived environment, remember its changing state, ground natural instructions in that state, and act in ways that are useful, legible, and safe. Unlike a task-specific robot in a factory, vehicle, or clinic, a personal assistant operates in homes, offices, streets, and wearable or XR interfaces where the scene is private, cluttered, and shaped by individual routines. This chapter therefore treats assistance as a persistent human-centered loop over spatial memory, language grounding, action, and feedback. The same spatial model supports tidying and search, close collaboration, and accessibility support for vulnerable users, but each setting changes how much the assistant must model the person as well as the place.

13.1 Assistant

13.1.1 Introduction

An intelligent assistant is a generalist embodied agent that shares the user's own private, unstructured world and acts on the user's behalf. Unlike the domain agents of the preceding sections, which are optimized for a bounded operational envelope (a road network, a warehouse aisle, an operating theater), the assistant must function inside the open-ended geometry of a home, an office, or a sidewalk, in the continuous physical presence of the person it serves. Its defining spatial challenge is therefore not a new perception, reasoning, generation, or memory primitive: those capabilities are surveyed in Chapter 6 (perception), Chapter 7 (reasoning), Chapter 8 (generation), and Section 7.3.6 (retrieval- and memory-augmented spatial reasoning). The challenge here is rather to build and maintain a persistent, personalized, and queryable spatial-semantic model of the user's environment, to ground human intent onto that model, and to act on it legibly and safely while a person occupies the same space.

Three properties separate the assistant from the embodied agents discussed elsewhere in this part. The environment is the user's own, so the assistant cannot assume an engineered layout and must instead generalize across the long tail of real homes. It is also personalized: the same instruction ("tidy up," "take me to the exit") resolves to different geometric goals for different users, so the spatial model must absorb individual preference. And because the human is always co-present, otherwise routine motion becomes a social act that must be predictable, comfortable, and safe. These three demands (an unengineered space, an individual user, and a person sharing that space) recur across every regime in which the assistant operates.

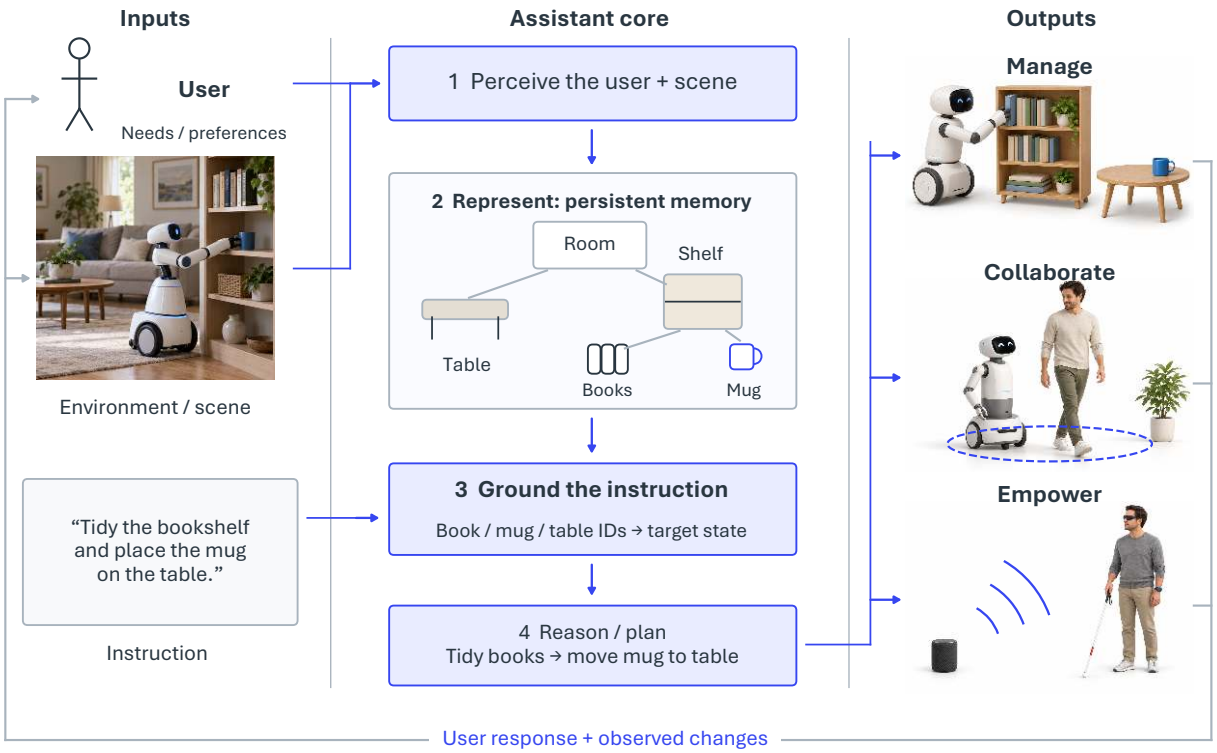


Figure 13.1: The intelligent-assistant loop. User and scene observations populate persistent room–object memory; instructions are grounded to specific objects and target states before planning. Alternative outputs illustrate household management, shared-space collaboration, and audio/haptic assistance. Memory links denote spatial associations; the dashed ellipse marks shared interaction space. User responses and observed changes close the feedback loop.

13.1.2 System Overview

Across these settings, every assistant closes the same loop. It **perceives** the environment and the user, **represents** what it perceives in a persistent spatial-semantic memory, **grounds** a natural-language or gestural instruction onto that memory, **reasons** about a goal-directed plan, **acts** on the physical or informational world, and observes the human's **feedback**, which re-grounds the next interaction. The persistent memory is the part that distinguishes an assistant from a single-shot perception or manipulation system: because the assistant lives in one place over long horizons, it can accumulate a map of where objects belong, where the user tends to be, and how the user prefers tasks to be performed. Long-horizon mobile-manipulation benchmarks such as EMMOE instantiate one slice of this loop (language-conditioned planning and action in open environments) but still leave persistent personalization, memory update, and human feedback as broader assistant-level requirements [2945]. Figure 13.1 schematizes the loop, with the spatial-semantic memory at its center and the human present at every stage.

The assistant is deployed in three escalating regimes of human-proximity, which organize the remainder of this section. In the first, the assistant acts *on* the space while the human is a beneficiary at a distance: it tidies, fetches, and searches. In the second, the assistant acts *alongside* the human as a partner that must move and manipulate in shared space without intruding. In the third, the assistant acts *for* a vulnerable human, standing in for an impaired capability under hard safety, privacy, and accessibility constraints. The three regimes share the loop above but place progressively more weight on modeling the person rather than the place; Table 13.1 collects benchmarks for all three.

13.1.3 Pivotal Task-1: Managing the Physical Environment

The most autonomous regime asks the assistant to maintain the physical state of a human environment with little real-time supervision. A persistent model of the dwelling supports two complementary operations: grounding instructions for physical action, and retrieving remembered evidence to answer queries. New observations update the model, including after an action changes the environment. Figure 13.2 separates these action and recall branches around their shared home memory.

Representing the home. Everything downstream depends on a queryable model of the space, which extends the scene-understanding methods of Chapter 6 into a persistent, semantically indexed memory. The memory architectures themselves are treated in Section 7.3.6; here they are the infrastructure that lets an assistant know not only *where* surfaces are, but *what* and *whose*. The 3D scene graph [341] introduced a hierarchical structure that unifies objects, rooms, and cameras with the underlying geometry, and Hydra [1554] brought such graphs into real-time, loop-closure-optimized robot perception so that the memory stays current as the assistant moves. To make the memory addressable by free-form language, ConceptFusion [2946] builds an open-set, multimodal 3D map that answers text, image, or click queries, and ConceptGraphs [1555] fuses foundation-model outputs across views into a captioned, language-queryable 3D scene graph suitable for planning. This open-vocabulary memory is what lets an assistant act on objects it was never explicitly trained to recognize.

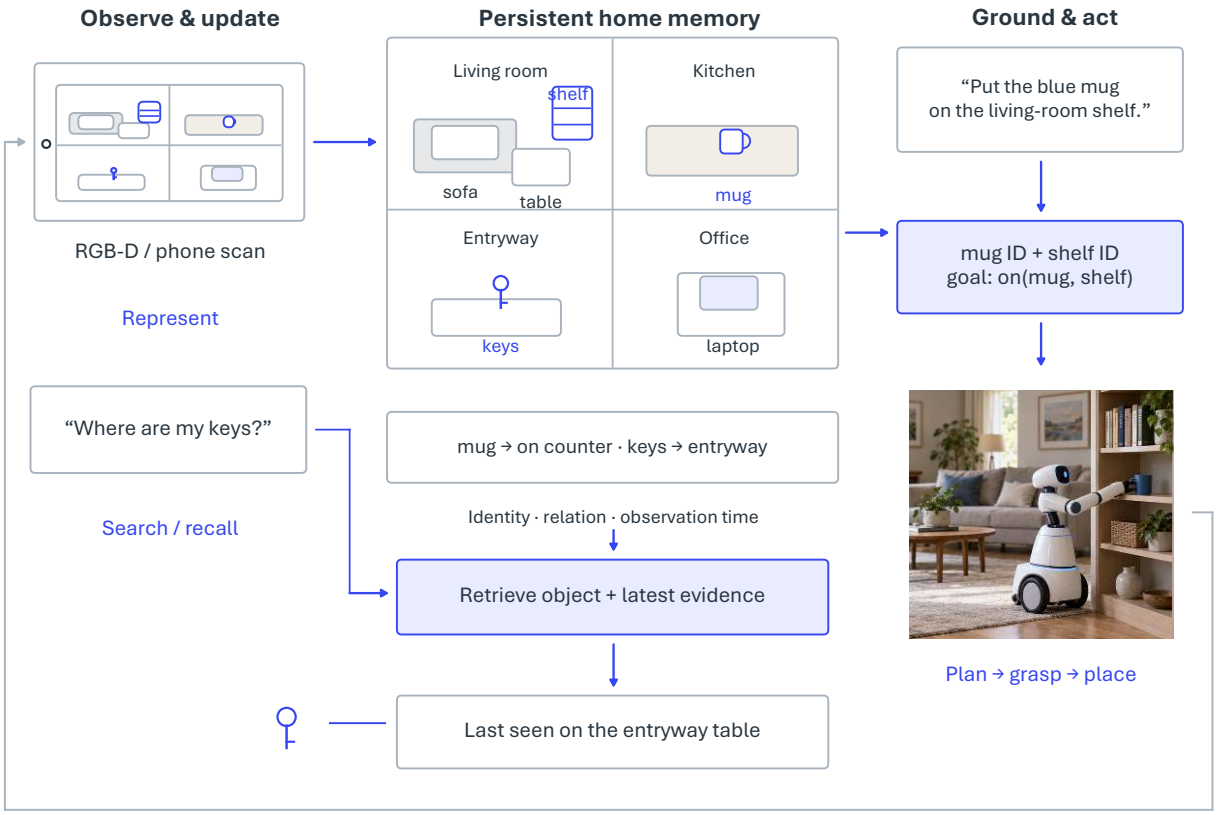


Figure 13.2: Managing the physical environment through a shared home memory. Observations maintain room, object, and relation records; grounded instructions drive manipulation, while queries independently retrieve last-seen evidence. Physical outcomes are re-observed to update memory.

Acting on the space: tidy-up and rearrangement. Restoring order is the canonical management task, framed by Batra *et al.* [2947] as bringing an environment from an arbitrary configuration into a target spatial state. The benchmark substrate includes the Visual Room Rearrangement (RoomR) challenge [2948], which

asks an agent to observe a goal arrangement and then reset one to five displaced objects, and Housekeep [2949], which removes the explicit goal and instead judges placements against human preferences over nearly two thousand objects. Method work has moved from fixed rules to learned commonsense priors. TIDEE [2950] tidies a never-before-seen room from pixels using a visuo-semantic out-of-place detector and an associative memory of where receptacles tend to be, and TidyBot [2951] makes the placement *preference-aware*, using few-shot summarization with a large language model to generalize a particular user's tidying rules from a handful of examples. The shift these systems mark, from a universal notion of tidiness to one conditioned on the resident, is exactly the personalization property that defines the assistant.

Acting on the space: open-vocabulary mobile manipulation. Tidying a real home requires picking and placing arbitrary objects at arbitrary locations, the task formalized as open-vocabulary mobile manipulation by HomeRobot [2952], which pairs a simulation and real benchmark with a software stack for the Hello Robot Stretch. Notable deployment evidence comes from OK-Robot [2953], which combines a phone-scanned dense vision-language map with off-the-shelf navigation and grasping to achieve zero-shot pick-and-drop across ten real homes, and from Mobile ALOHA [2954], which shows that low-cost whole-body teleoperation plus imitation learning can acquire dexterous bimanual household skills. These systems lean directly on the manipulation primitives of Section 9.1, increasingly instantiated as generalist vision-language-action policies (*e.g.*, OpenVLA [99], π_0 [100]) that fold perception, grounding, and control into one model; the assistant contribution is to drive them in unstructured homes from open-vocabulary goals rather than in an engineered cell. Commercial and pre-commercial examples illustrate the same pressure from different directions: mapping vacuum robots such as iRobot's Roomba, which build semantic room maps for targeted cleaning,[1] represent the mature narrow end, while emerging home-humanoid platforms such as 1X's Neo[2] and Figure[3] show how the field is attempting, so far without a general-purpose result, to generalize mobile manipulation into domestic settings.

Searching and recalling the model. The dual of acting on the space is querying it: an assistant is, among other things, a search engine for the physical world that answers "where did I leave my keys?" Object-goal navigation supplies the active half, with goal-oriented semantic exploration (SemExp) [1416] building an episodic semantic map and a policy that exploits object co-occurrence to find a target efficiently rather than searching blindly. Embodied question answering [1404] couples navigation with answering, and its open-vocabulary successor OpenEQA [1563] spans both an episodic-memory setting, in which the assistant must recall what it has already seen, and an active-exploration setting, over more than a hundred real environments. For wearable assistants that record the user's day, the same recall problem becomes natural-language search of long egocentric video, the episodic-memory task defined by Ego4D [681]. This is the application-facing counterpart of the memory-augmented reasoning systems in Section 7.3.6.

13.1.4 Pivotal Task-2: Co-habiting and Collaborating

As the distance between assistant and human shrinks, spatial intelligence must become social intelligence. The assistant must model the human as a co-located agent with intent, viewpoint, and comfort zones, not as a moving obstacle or a bare command source, and act predictably and collaboratively. Running through every part of this task is the reconciliation of reference frames: the assistant's own egocentric view, the human's egocentric view, and a shared allocentric map must be kept aligned.

Socially-aware navigation. Moving through a populated space is not collision avoidance; it is adherence to the unwritten contracts of proxemics. The classical account is the social force model [2955], which casts pedestrian motion as attraction to a goal under repulsion from others, operationalizing Hall's anthropological zones of personal space. Data-driven successors learn the interaction directly: Social LSTM [2249] couples

[1] https://www.irobot.com
[2] https://www.1x.tech/neo
[3] https://www.figure.ai

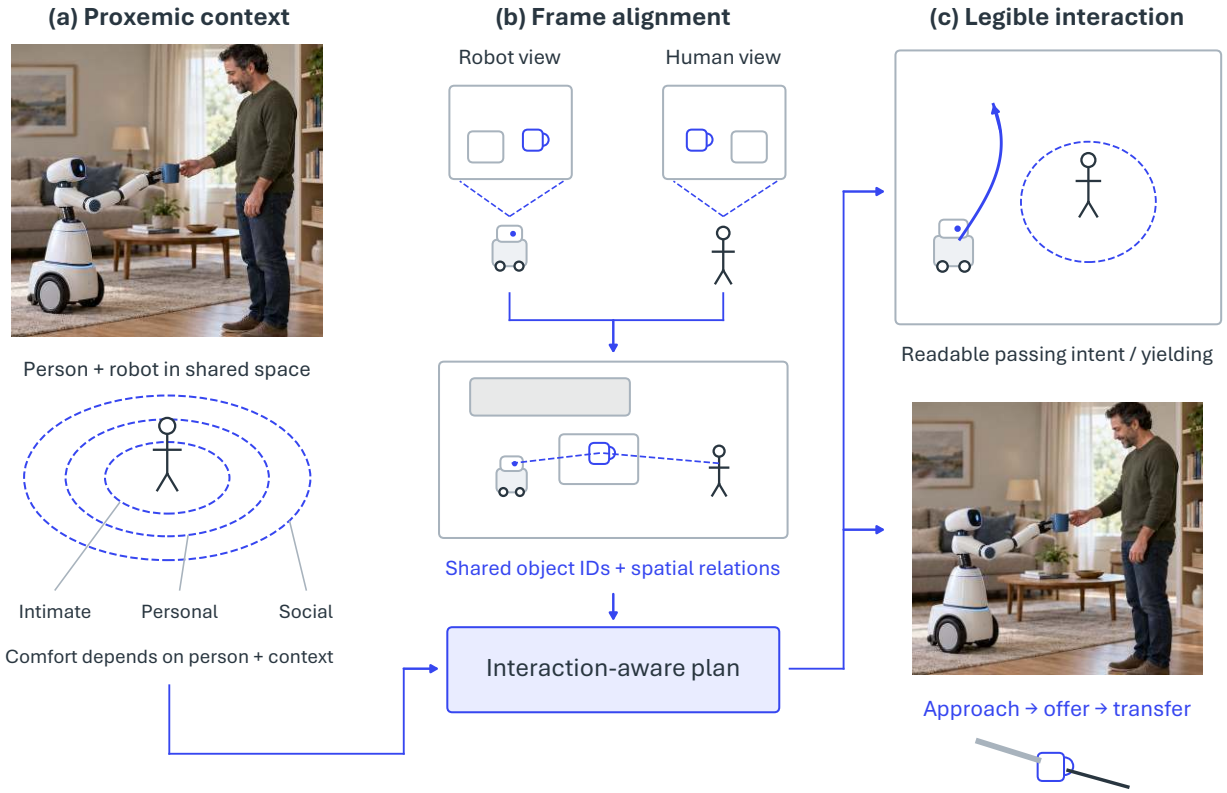


Figure 13.3: Collaborating in shared space: (a) person- and context-dependent comfort zones and (b) aligned robot/human views inform an interaction-aware plan; (c) navigation and handover illustrate alternative behaviors. Dashed rings indicate comfort regions, and dashed sight lines associate views with shared objects.

per-person recurrent predictors through a social-pooling layer to forecast jointly plausible trajectories, and deep reinforcement-learning planners such as the attention-based crowd-robot model [2956] weigh neighbors to navigate dense crowds while avoiding the "freezing robot" failure that purely reactive planners suffer [2957]. Progress is measured on human-demonstration corpora such as SCAND [2958], which records hours of socially compliant teleoperation for imitation and inverse reinforcement learning. The trend is toward anticipatory planning, in which the assistant infers the human's intended path and yields proactively, and toward learnable cost functions that adapt passing distances to individual or cultural preference rather than enforcing one universal margin.

Grounded instruction following and spatial dialogue. A collaborating assistant must resolve spatial language against what it sees, including the deictic references ("put it next to the *other* bowl," "the one over *there*") that pervade human speech. Vision-and-language navigation, introduced with the Room-to-Room benchmark [1415], established the problem of following natural-language route instructions in photorealistic buildings; ALFRED [1307] extended it to long-horizon household tasks with irreversible state changes, and TEACh [1308] cast the task as a two-agent dialogue in which a commander and follower must build shared spatial reference over multiple turns. Resolving a referring expression to a specific object in a cluttered 3D scene is studied directly by ScanRefer [373] and ReferIt3D [374], whose same-class distractors force the model to rely on spatial relations rather than appearance. These benchmarks build on the reasoning capabilities of Chapter 7; the assistant requirement is to maintain a spatial working memory that links current observations to earlier instructions so that deixis can be resolved.

Shared autonomy and collaboration. When full autonomy is brittle, the assistant acts as a third hand under shared control. The defining formulation is shared autonomy via hindsight optimization [2959], in which the robot infers a distribution over the user's spatial goal and assists toward it without committing to a

single confident prediction, formalizing the earlier policy-blending account of arbitration between human and robot intent [2960]. Equally important is making the assistant's own motion readable: legible motion [2639] shapes a reaching trajectory so a human can quickly infer its intended target, which is what allows close-quarters collaboration to feel safe. Physical handovers, where giver and receiver must coordinate in space and time [2961], are the elementary collaborative act on which assistive manipulation is built.

Perspective-taking and egocentric grounding. Many assistants observe the world from a wearable, first-person viewpoint, so the frame-alignment problem becomes explicit. Ego4D [681] and its successors anchor egocentric perception at scale, and dedicated capture platforms such as Project Aria[4] supply the calibrated multi-sensor first-person stream such assistants run on, while spatial-VQA backbones such as SpatialVLM [229] give a vision-language model the quantitative 3D reasoning (distances, sizes, directions) needed to answer spatial questions about the user's scene. The hardware and pose-estimation side of wearable sensing is treated in Section 9.3, and the augmented-reality runtime that renders guidance into the user's view is treated in Section 10.1; here the contribution is the reasoning that converts a first-person observation into a collaborative action in the human's frame. Embodied-AR applications make this alignment problem explicit when directions, warnings, or manipulation cues must be registered into the user's view [2962]. Figure 13.3 illustrates the elements of this regime.

13.1.5 Pivotal Task-3: Empowering Vulnerable Users

The highest-impact regime uses spatial intelligence to compensate for a sensory, cognitive, or motor impairment, acting as a prosthesis for the user's perception or agency. The assistant translates the environment into a queryable, navigable, and actionable representation and delivers it through a channel the user can still use (audio, haptics, or physical action). Because the loop closes around a vulnerable person in their own home, safety, privacy, and personalization are design constraints from the start.

Assistance for the blind and visually impaired. For blind and low-vision users, spatial intelligence turns an unknown and hazardous environment into a queryable one. The oldest strand is sensory substitution, exemplified by the vOICe [2963], which sonifies the spatial structure of an image into sound. The modern strand is conversational visual question answering: VizWiz [2964] introduced a deployed system in which a blind user photographs their surroundings and asks a spoken question, and the VizWiz Grand Challenge [2965] turned the resulting real-world distribution, including blurry and unanswerable images, into the reference benchmark for assistive VQA. Indoor wayfinding is addressed by smartphone navigation aids such as NavCog [2966], which deliver turn-by-turn audio guidance from accurate beacon-based localization, and by three-dimensional spatial audio that places points of interest around the user. Commercial assistive applications include Microsoft's Seeing AI for scene and text narration[5], Be My Eyes for volunteer- and model-mediated visual assistance[6], and wearable systems from OrCam[7], Envision[8], and biped.ai[9] for scene description and obstacle warning. These tools move from passive obstacle warning to active interrogation, in which the user asks "is there a free seat?" or "read the sign on the door" and the assistant retrieves the answer.

Cognitive and mobility assistance. For users with cognitive decline or limited mobility, spatial assistants provide external scaffolding for the activities of daily living. Ambient assisted living systems such as the CASAS smart home [2967] instrument a dwelling to track the resident's location and activity, learning the spatio-temporal patterns whose violation (a fall, a missed meal) signals a need to intervene; wearable accelerometry supplies the canonical fall-detection paradigm [2968].

[4] https://www.projectaria.com
[5] https://www.seeingai.com
[6] https://www.bemyeyes.com
[7] https://www.orcam.com
[8] https://www.letsenvision.com
[9] https://biped.ai

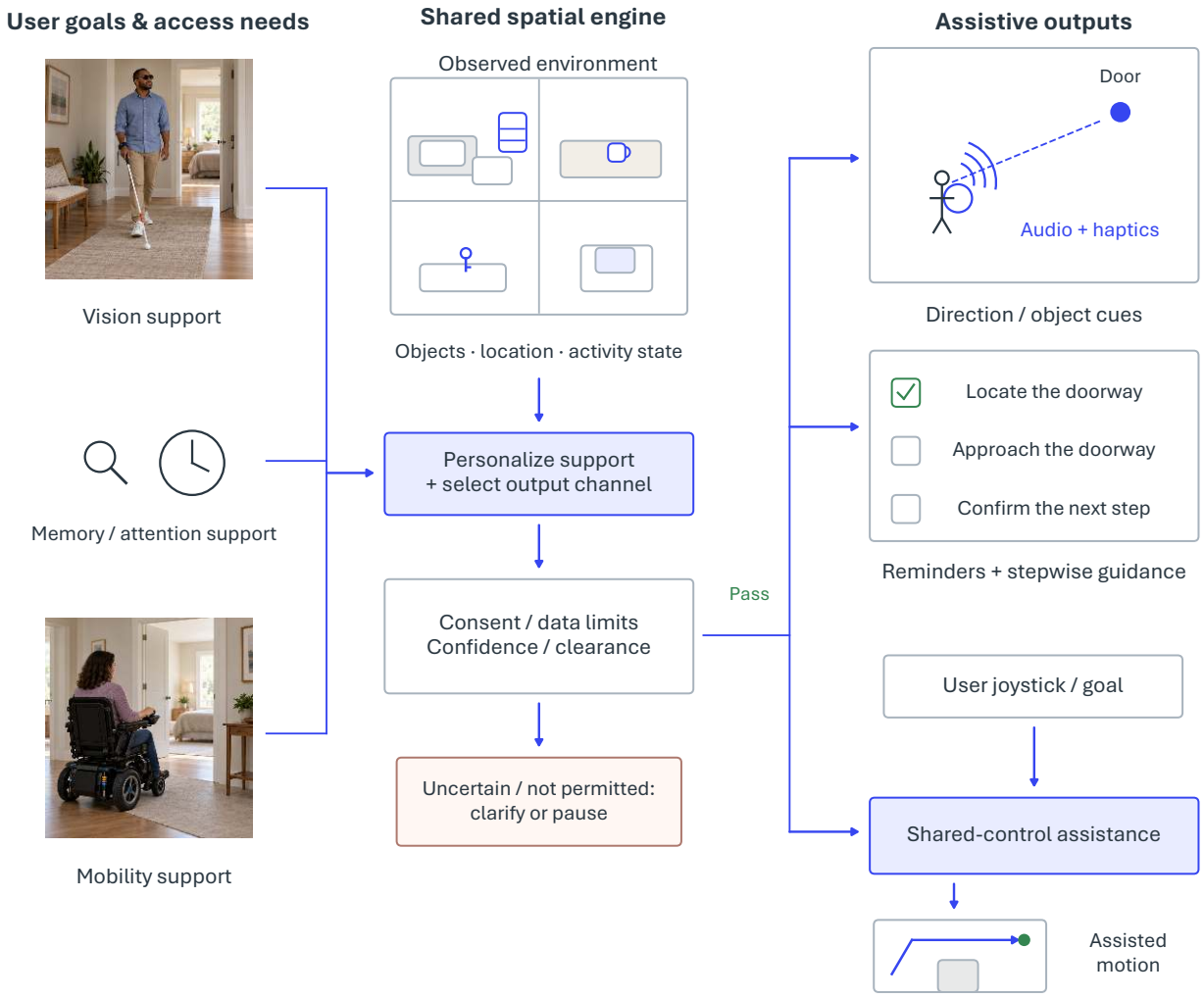


Figure 13.4: Personalized assistance combines observed spatial state with user goals and access needs. Consent, confidence, and clearance checks gate audio/haptic cues, stepwise guidance, and shared control; uncertainty or missing permission prompts clarification or a pause. Shared control also receives the user's command. Scenes are illustrative.

Where monitoring gives way to physical action, the assistant becomes a shared-control effector. Smart wheelchairs [2969] and wheelchair-mounted assistive arms use collaborative control to enforce spatial safety zones, for example forbidding a chair from driving down a staircase, so that the user retains mobility without constant supervision; the goal-arbitration that underlies such control is the same shared-autonomy machinery introduced above, here cast as a decision-theoretic problem [2970]. Assistive feeding [2971] shows how fine spatial reasoning (where to acquire a bite, how to transfer it to the user's mouth) becomes a caregiving act. The platform lineage runs from Robots for Humanity [2972], which let a man with quadriplegia perform self-care through a teleoperated PR2, to the affordable Stretch mobile manipulator [2973], with physics-based simulation of caregiving tasks consolidated in Assistive Gym [2974]. Commercial assistive hardware in this space includes the Kinova JACO arm[10] and the Obi feeding device[11]. This regime overlaps the clinical setting of Section 12.1, but the boundary is clear: there the focus is diagnosis and surgical intervention, whereas here it is daily living in the home. Figure 13.4 summarizes the compensation principle.

[10] https://www.kinovarobotics.com
[11] https://meetobi.com

Table 13.1: Representative benchmarks across the three assistant regimes. Where a benchmark has no standard method, the method column names the benchmark paper's own baselines.

Regime / Pivot	Benchmark	Representative method	Headline metric
Managing (rearrangement)	Visual Room Rearrangement [2948]	TIDEE [2950]	objects restored / episode success
Managing (preference tidy)	Housekeep [2949]	LLM-based planner (paper baseline) [2949]	correct-placement rate
Managing (mobile manip.)	HomeRobot OVMM [2952]	modular and RL baselines [2952]	overall success rate
Managing (search / recall)	EQA [1404] / OpenEQA [1563]	VLM with episodic memory (paper baselines) [1563]	LLM-Match score
Collaborating (navigation)	SCAND [2958]	imitation from demonstrations [2958]	demonstration corpus (no leaderboard)
Collaborating (instruction)	ALFRED [1307] / TEACh [1308]	seq2seq baselines [1307, 1308]	task / goal-condition success
Collaborating (grounding)	ScanRefer [373] / ReferIt3D [374]	ScanRefer baseline [373]	Acc@0.25 / 0.5 IoU
Empowering (BVI VQA)	VizWiz [2965]	VQA baselines [2965]	VQA accuracy
Empowering (assistive ADL)	Assistive Gym [2974]	RL policies (paper baselines) [2974]	task success / reward

13.1.6 Established Tools and Systems

Deployed assistants have so far approached the persistent, personalized spatial model of this section from opposite ends: software agents know a great deal about the user but nothing about the user's space, while home robots occupy the space but act with limited autonomy.

Personal agents. The most widely used assistants now act on a user's behalf in the digital world. OpenAI's ChatGPT agent[12] (July 2025) runs on its own virtual computer, browsing and filling in forms across multi-step tasks and asking permission before consequential actions. Meta's Muse[13] (September 2026), billed as a personal agent for everyday users, likewise runs in an isolated virtual machine and handles errands such as shopping and travel booking, and its request for access to email, calendars, and payments made privacy the main point of public debate around its launch. Voice assistants are being rebuilt on the same models: Amazon's Alexa+[14], generally available in the United States since February 2026, adds generative dialogue to the smart-home control that Alexa already provided, and Apple's Siri[15] is being rebuilt on Google's Gemini models, adding personal context and on-screen awareness. For spatial intelligence, these agents matter mainly as the layer that holds personal context and preference; apart from smart-home control, none of them maintains a model of the physical home.

Camera-grounded assistants. A second line brings the physical world into the conversation through a camera. Gemini Live's camera mode[16], the product form of Google's Project Astra, lets a user point a phone at the surroundings and ask about what it sees, and Meta AI on camera glasses does the same from the wearer's viewpoint (Section 9.3). These systems answer spatial questions about the current view, and the assistive applications of Pivotal Task-3 build on the same capability, but they keep at most a short history of what they have seen, so the long-horizon spatial memory this section calls for remains outside them.

Home robots. Embodied assistants remain the least mature tier. Beyond mapping vacuums, the home humanoids introduced under Pivotal Task-1 are moving toward sale: 1X opened preorders for NEO in October 2025, at $20,000 or a monthly subscription, with first deliveries promised for 2026 and with remote operators teleoperating tasks that its onboard autonomy cannot yet complete, and Figure's third-generation humanoid, Figure 03 (2025), is designed for homes and driven by its Helix vision-language-action model.

[12] https://openai.com/index/introducing-chatgpt-agent/
[13] CNBC, 8 September 2026.
[14] https://www.aboutamazon.com
[15] https://www.apple.com/siri/
[16] https://gemini.google/

The reliance on teleoperation shows how far these products are from the autonomous, open-vocabulary mobile manipulation of Pivotal Task-1, and it raises the in-home privacy question discussed below.

13.1.7 Challenges

The three regimes share a common obstacle: the assistant must be reliable in the one setting that resists engineering, namely a real person's private and unstructured life. The challenges below are ordered from the perceptual frontier (generalization, memory) through the human frontier (personalization, social safety) to the deployment frontier (privacy, compute, trust), and each cuts across all three pivotal tasks above. Governance benchmarks for embodied agents have begun to measure the deployment end directly, testing whether a system stays controllable, recoverable, and auditable under perturbation rather than only whether it completes tasks [2975].

Generalization to unstructured homes. An assistant cannot be retrained per household, yet homes vary without bound in layout, lighting, and contents. Zero-shot results such as OK-Robot [2953] are encouraging but remain far from the reliability a daily aide requires, and closing the gap depends on the open-vocabulary representations and sim-to-real transfer surveyed in Part III rather than on more in-domain data.

Long-horizon spatial memory. Because the assistant persists in one place, it must remember spatial state over days and weeks: where objects were last seen, which routines the user follows, what was already done. Real-time scene-graph memories [1554] and episodic-memory benchmarks [1563] begin to address this, but the general memory mechanisms are treated in Section 7.3.6; the assistant-specific unsolved problem is maintaining a consistent, updatable, and forgetting-resistant model over long deployments in a private home.

Personalization and preference learning. The assistant's geometric goals are user-specific, so it must learn an individual's placement rules, comfort distances, and routines from sparse interaction. Few-shot preference summarization [2951] and learnable social cost functions are early steps, but acquiring rich preferences without burdening the user remains an open problem.

Social compliance and safety. Operating in continuous contact with people turns motion into a social and safety-critical act. Anticipatory, legible navigation and motion [2957, 2639] and shared-control safety zones [2970] mitigate risk, yet guaranteeing comfort and physical safety across the diversity of real human behavior is far from settled.

Privacy of persistent in-home perception. An assistant that continuously perceives a private home raises acute privacy concerns, the more so for assistive cameras worn by users who cannot themselves audit what is captured. The concern becomes sharper when assistance depends on remote operation or cloud annotation, because the raw spatial stream itself may reveal the user's routines and vulnerabilities. Persistent spatial memory must be reconciled with selective perception, on-device processing, and user control over what is retained.

On-device compute and trust. Wearable and home platforms are power- and thermally constrained, forcing the heavy perception and reasoning stacks of Part III onto efficient, often on-device, inference. Compounding this, assistance for vulnerable users is safety-critical, so calibrated reliability and appropriate user trust, neither over- nor under-reliance, are prerequisites for deployment rather than refinements of it.

Chapter 14

Spatial Intelligence for Others

Other application domains of spatial intelligence are too numerous to enumerate, because spatial intelligence is a horizontal capability: any field that must perceive, reconstruct, localize within, or reason about the geometry of the physical world inherits the methods of Part III.

This chapter takes five further domains where that inheritance is already visible in deployed systems. Each stresses a capability the earlier chapters do not: registering a building site against its design, tracking many near-identical agents through occlusion, reconstructing fragile objects faithfully, working without clean light or satellite positioning underwater, and binding people to products across a network of cameras.

14.1 Architecture, Engineering, and Construction

The architecture, engineering, and construction (AEC) sector is built around a single spatial-intelligence problem: continuously registering the messy as-built reality of a site against the structured as-designed model that specifies what it should become. This is the same Building-Information-Modeling (BIM) substrate that anchors the city-scale digital twins of Section 11.1, but here the model is the working blueprint of an individual building under construction rather than a synchronized replica of a finished one, and the central task is detecting and acting on the discrepancy between plan and reality. Figure 14.1 sketches this as-built versus as-designed loop. **Scan-to-BIM** is its perception front end: dense laser or photogrammetric scans of a structure are semantically parsed into walls, floors, columns, and components, building on the indoor point-cloud understanding of Chapter 6 and its benchmarks S3DIS [507] and ScanNet [448]. A specialized sub-problem is vectorizing a scan into an editable floorplan, where synthetic supervision such as Structured3D [1943] and transformer models that predict all room polygons in parallel [2976] now recover clean geometry from cluttered scans. **Progress monitoring** closes the loop in time: by registering an image-based or scanned as-built point cloud against a four-dimensional IFC-BIM model, deviations between planned and actual construction state can be recognized automatically [2977], the spatiotemporal core of the commercial reality-capture platforms now common on job sites. **Construction robotics** acts on the model directly, with mobile fabricators that localize against the design to print, place, or assemble structural elements at full scale [2978]. Finally, **structural inspection** turns the same perception stack toward maintenance, localizing cracks and defects on concrete surfaces from imagery [2979], increasingly from a UAV-borne 3D reconstruction of a bridge or facade. Reality-capture and BIM tooling from Matterport,[1]

[1] https://matterport.com/

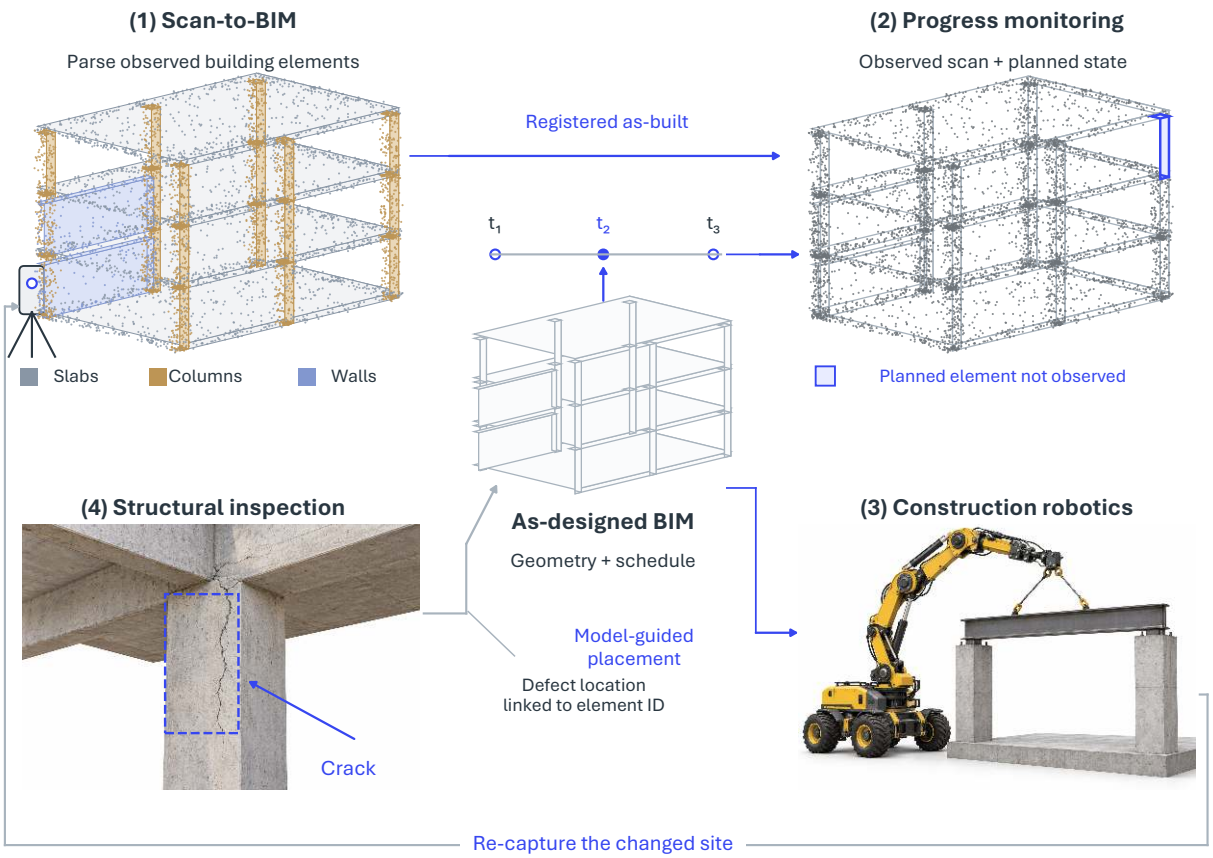


Figure 14.1: AEC tasks around an as-designed BIM: (1) semantic parsing of site scans; (2) comparison with the scheduled design state, highlighting a planned element not observed in the scan; (3) model-guided robotic placement; and (4) crack localization linked to a model element. Physical work prompts re-capture of the site. Scenes and point clouds are illustrative.

Autodesk,[2] and Leica,[3] progress-monitoring services such as OpenSpace[4] and Buildots,[5] and on-site robots from Dusty Robotics[6] and Built Robotics[7] are representative of the deployed layer.

14.2 Sports Analytics and Performance

Sports analytics lifts raw broadcast or arena video into a calibrated metric model of the playing surface, then reasons over the resulting positional field. What sets it apart is fine-grained, multi-agent spatiotemporal tracking under severe and persistent occlusion, where many near-identical players and a small fast ball must be followed and assigned consistent identities at sub-second resolution. Figure 14.2 illustrates three linked capabilities: registered trajectories feed tactical analysis, while calibrated multi-view pose supports biomechanics. **Tracking and registration** form the geometric foundation: multi-object tracking benchmarks such as SoccerNet-Tracking [2980] and the multi-sport SportsMOT [2981] drive player and ball association, while sports-field registration estimates the homography from the broadcast frame to a metric pitch template so that pixel positions become field coordinates [2982]. **3D pose and biomechanics** recover athlete motion

[2] https://www.autodesk.com/products/recap/overview
[3] https://leica-geosystems.com/products/laser-scanners
[4] https://www.openspace.ai/
[5] https://buildots.com/
[6] https://www.dustyrobotics.com/
[7] https://www.builtrobotics.com/

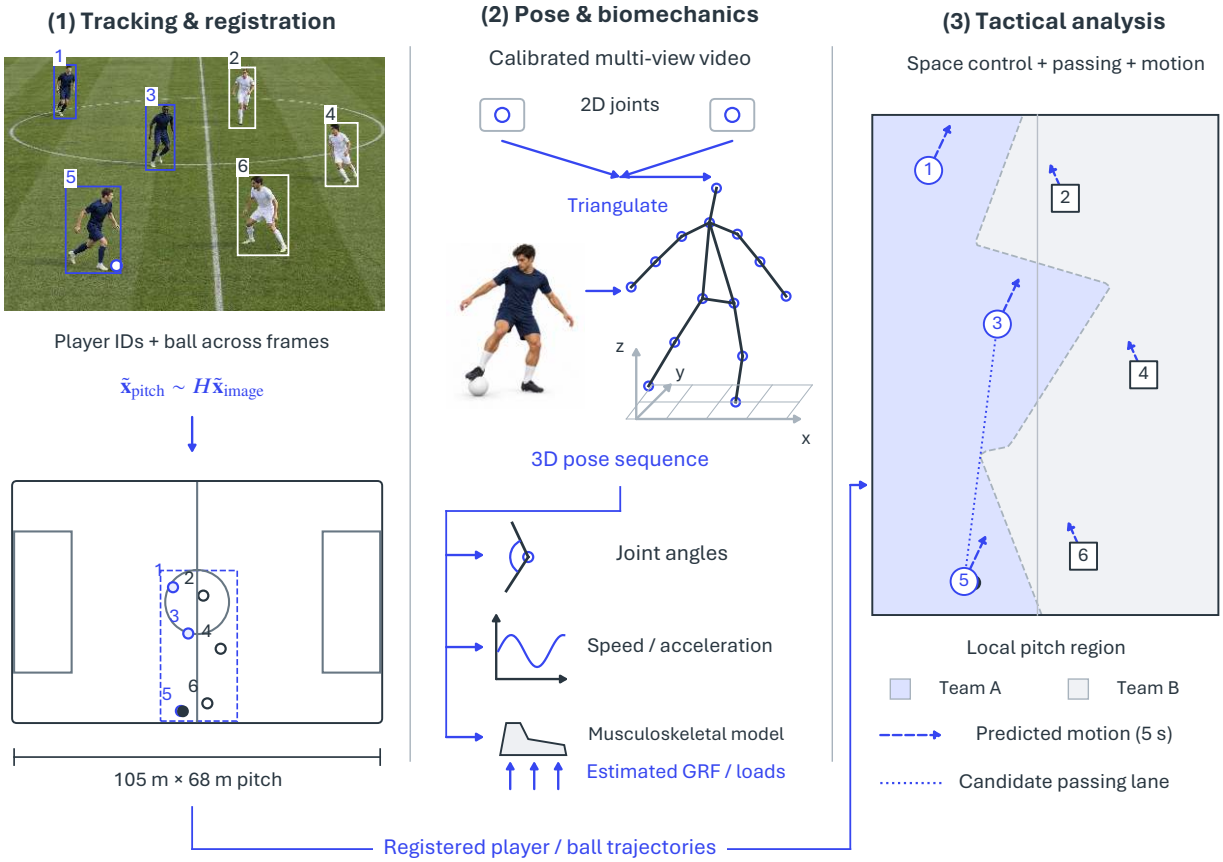


Figure 14.2: Sports analytics: (1) player/ball tracking and planar registration; (2) calibrated multi-view pose, kinematics, and model-estimated loads; and (3) tactical analysis from registered trajectories. The tactical view enlarges the dashed pitch region, preserving player IDs. Shading illustrates team control, dotted lines candidate passes, and dashed arrows predicted motion. Scenes and predictions are schematic.

in three dimensions, with markerless capture systems estimating not only kinematics but joint dynamics from ordinary video, as in OpenCap [2983], typically by triangulating multi-view 2D pose into a 3D skeleton [538]. **Tactical and team-space analysis** then treats the positional field itself as the object of study, valuing the space each player controls and forecasting where players will move; deep models of expected possession value [2984] and multi-agent trajectory generation [2985] build directly on the spatial-reasoning methods of Chapter 7. The deployed layer is mature, spanning optical tracking and officiating from Hawk-Eye,[8] player-tracking data from Second Spectrum (now part of Genius Sports)[9] and Stats Perform,[10] and wearable spatial sensing from KINEXON[11] and Catapult.[12]

14.3 Cultural Heritage and Archaeology

Cultural heritage and archaeology apply spatial intelligence to the past, running from capture through recovery to discovery. What distinguishes it is high-fidelity, metrically accurate, and usually non-contact reconstruction, coupled to a preservation imperative: the artifact may be fragile, fragmentary, or buried, and the digital record must be faithful enough to stand in for an object that cannot be handled or that no

[8] https://www.hawkeyeinnovations.com/
[9] https://www.geniussports.com/
[10] https://www.statsperform.com/
[11] https://kinexon-sports.com/
[12] https://www.catapult.com/

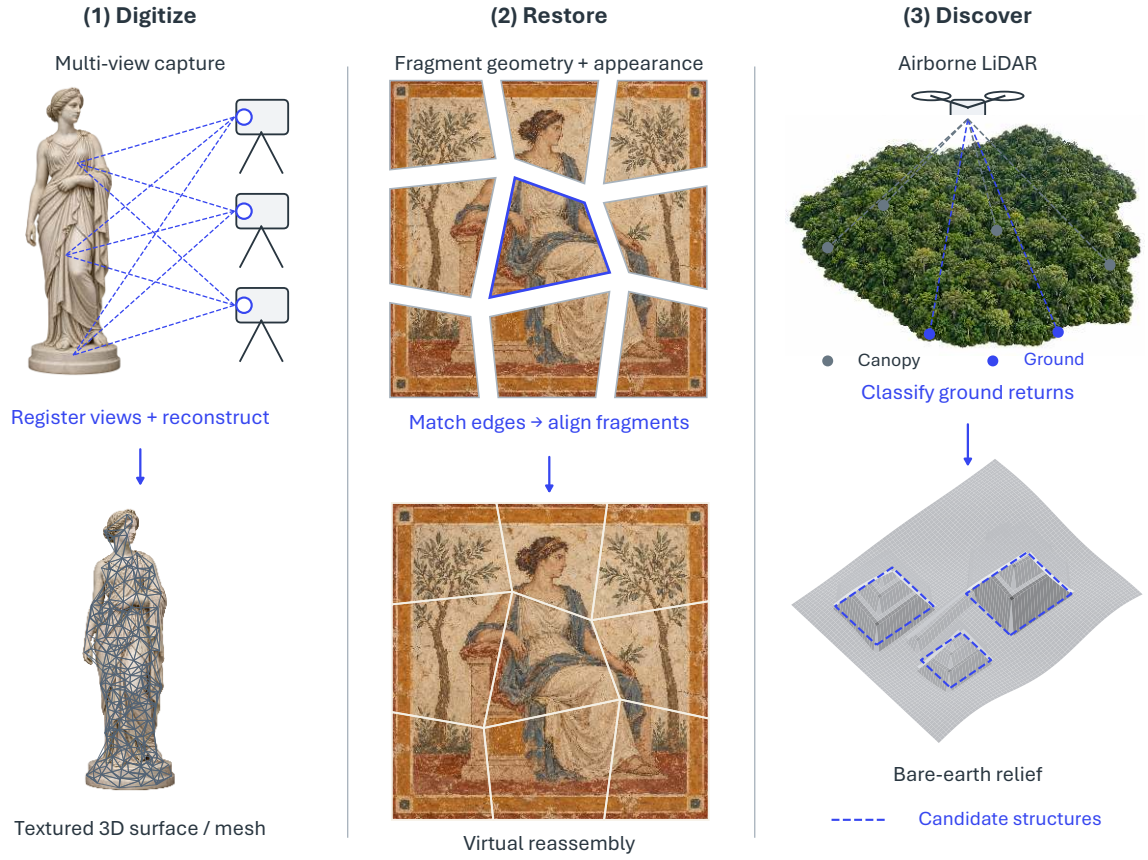


Figure 14.3: Three cultural-heritage tasks: (1) multi-view capture and surface reconstruction; (2) geometric and appearance-based matching of fragments for virtual reassembly; and (3) classification of airborne-LiDAR ground returns to reveal candidate structures in bare-earth relief. The tasks are independent; scenes, mesh, and terrain are illustrative.

longer exists intact. Figure 14.3 organizes the field into three movements. **Digitization** captures artifacts and monuments in three dimensions, a lineage that runs from the landmark in-field laser scanning of the Digital Michelangelo Project [2986] through structure-from-motion photogrammetry [115] to the neural radiance fields [119] and 3D Gaussian splatting [121] now used for photorealistic heritage capture, all drawing on the reconstruction methods of Chapter 8. **Virtual restoration** uses geometric AI to reassemble what time has broken, posing fragment reassembly as a 3D puzzle over fractured parts, with large benchmarks such as Breaking Bad [2987] and the real fresco-reassembly dataset RePAIR [2988] driving the methods. **Archaeological prospection** discovers sites at landscape scale, most strikingly through airborne LiDAR whose bare-earth models reveal structures hidden under forest canopy, as in the canopy-penetrating surveys of the Maya lowlands at Caracol [2989] and across northern Guatemala [2990]. Organizations such as CyArk[13] and the Factum Foundation,[14] together with photogrammetry tools including Agisoft Metashape[15] and the Open Heritage 3D archive,[16] anchor the practical layer.

[13] https://www.cyark.org/
[14] https://www.factumfoundation.org/
[15] https://www.agisoft.com/
[16] https://openheritage3d.org/

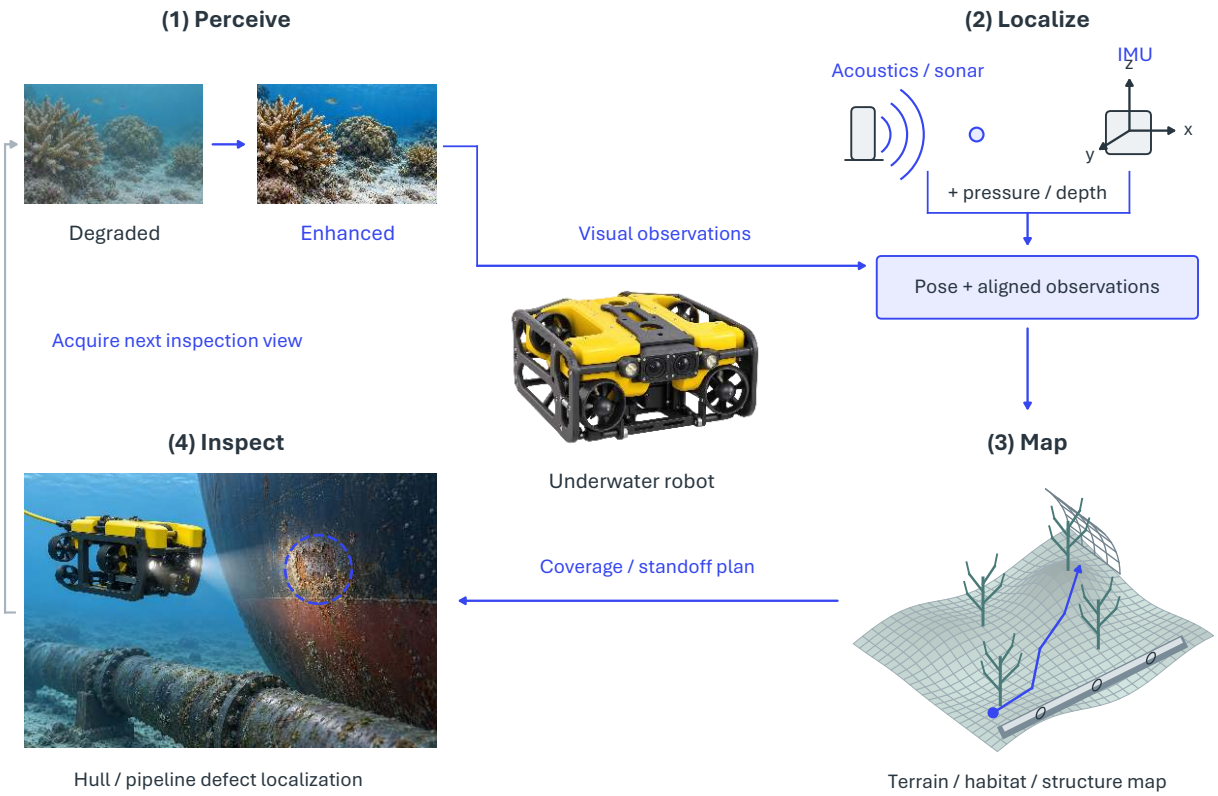


Figure 14.4: Underwater spatial intelligence under light attenuation/scattering and unavailable submerged GPS fixes. Enhanced visual observations combine with acoustic, inertial, and depth inputs for localization and mapping. Terrain and structure maps support coverage and standoff planning for hull/pipeline inspection; new views close the loop. Blue marks the survey track and selected defect. Scenes and enhancement are illustrative.

14.4 Marine and Underwater Systems

Underwater spatial intelligence confronts a setting in which the two assumptions that anchor terrestrial methods both fail at once: light no longer propagates cleanly, since wavelength-dependent attenuation and scattering degrade vision with range, and satellite positioning does not penetrate water, so absolute position must be inferred from acoustics, inertia, or the structure being surveyed. Figure 14.4 depicts the resulting perceive-localize-map-inspect loop. **Navigation and SLAM** solve the GPS-denied localization problem, surveyed comprehensively for autonomous underwater vehicles by Paull *et al.* [2991], with foundational deep-sea visual SLAM demonstrated on the wreck of the Titanic [2992] and modern visual-inertial-pressure localization benchmarked by datasets such as AQUALOC [2993]. **Perception and enhancement** contend with the optical medium itself: physically grounded models remove the water from an image to recover true scene color and range [2994], learned enhancement restores contrast in real time for downstream perception [2995], and benchmarks such as UIEB standardize the evaluation [2996], while large annotated image databases like FathomNet [2997] enable ocean-scale detection. **Mapping and inspection** reconstruct the seafloor, habitats, and submerged structures, with structure-relative perception and coverage planning enabling autonomous inspection of complex ship hulls [2998]. The operational layer spans low-cost platforms such

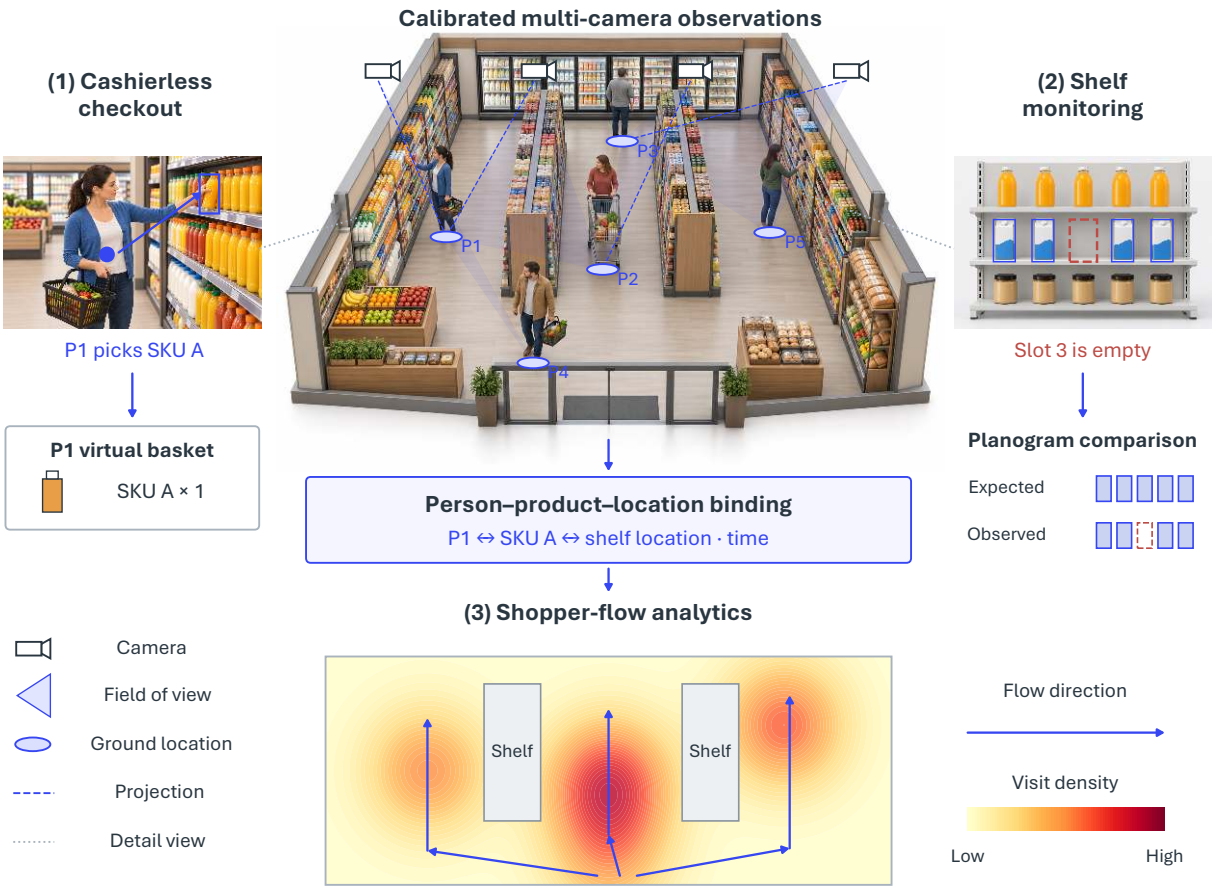


Figure 14.5: Retail and smart-space intelligence across calibrated ceiling cameras: (1) shopper–product association updates a virtual basket; (2) detected shelf contents are compared with a planogram; (3) ground-plane tracks yield shopper-flow and visit-density maps. Dotted lines link scene details; the geometry and density are illustrative.

as the BlueROV2,[17] survey vehicles from Kongsberg[18] and research institutions including WHOI[19] and MBARI.[20]

14.5 Retail and Smart Spaces

Retail and smart-space intelligence reduces to one persistent spatial problem: jointly associating, across a network of calibrated overlapping cameras and over time, which person is interacting with which product at which shelf location. Unlike the warehouse logistics of Section 9.5, where the environment is engineered for the robot, here the space is a crowded store full of untracked people, and its distinguishing challenge is dense multi-object tracking with reliable human-object association through frequent mutual occlusion. Figure 14.5 shows how the major applications share this core. **Multi-camera tracking** provides the spatial backbone, with benchmarks such as WILDTRACK [2999] and MMPTRACK [3000] and methods that project per-camera features onto a shared ground plane to detect and track people across views [3001]; this is the machinery behind cashierless checkout, where a shopper's path and pickups must be reconstructed without a checkout line. **Shelf monitoring** turns perception toward the products, detecting densely packed items

[17] https://bluerobotics.com/store/rov/bluerov2/
[18] https://www.kongsberg.com/discovery/autonomous-and-uncrewed-solutions/auv/hugin/
[19] https://www.whoi.edu/
[20] https://www.mbari.org/

Table 14.1: Representative benchmarks for the five domains of this chapter.

Domain	Benchmark / dataset	Representative method	Headline metric
AEC	S3DIS [507]	point-cloud semantic segmentation	mIoU
AEC	Structured3D [1943]	RoomFormer [2976]	room / corner / angle F1
Sports	SoccerNet-Tracking [2980] / SportsMOT [2981]	multi-object trackers	HOTA
Sports	broadcast field registration	learned-error optimization [2982]	IoU (whole / part)
Heritage	Breaking Bad [2987] / RePAIR [2988]	learned fragment reassembly	rotation / translation error
Marine	UIEB [2996]	Water-Net [2996]	PSNR / SSIM
Marine	AQUALOC [2993]	visual-inertial-pressure SLAM	trajectory error
Retail	WILDTRACK [2999]	MVDet [3001]	MODA
Retail	SKU-110K [3002]	dense detector with EM-merger [3002]	AP

for inventory and planogram compliance, supported by the dense-detection benchmark SKU-110K [3002] and by methods that formulate compliance as matching the observed shelf layout against the ideal one as a sub-graph isomorphism [3003]. **In-store analytics** aggregates the same tracks into shopper-flow heatmaps and layout insights, building on cross-camera re-identification [3004] to stitch one customer's trajectory across non-overlapping views. The deployed layer includes cashierless systems such as Amazon's Just Walk Out[21] and platforms from Trigo,[22] AiFi,[23] and Zippin,[24] alongside shelf-analytics vendors such as Trax.[25]

14.6 Outlook

The five domains above, whose benchmarks are collected in Table 14.1, are a sample, not a census. Spatial intelligence is equally at work in fields this survey does not treat at length: virtual try-on and 3D e-commerce, immersive education and high-fidelity training simulators, security and surveillance, energy and infrastructure inspection, precision agriculture beyond the remote-sensing view of Section 11.2, and many others. What unites them, and what unites them with the headline applications of the preceding chapters, is that each takes the perception, reasoning, and generation capabilities of Part III and specializes them to the geometry, sensing modality, and constraints of a particular slice of the physical world.

[21] https://www.justwalkout.com/
[22] https://www.trigoretail.com/
[23] https://www.aifi.com/
[24] https://www.getzippin.com/
[25] https://traxretail.com/

Part V

Discussions and Futures

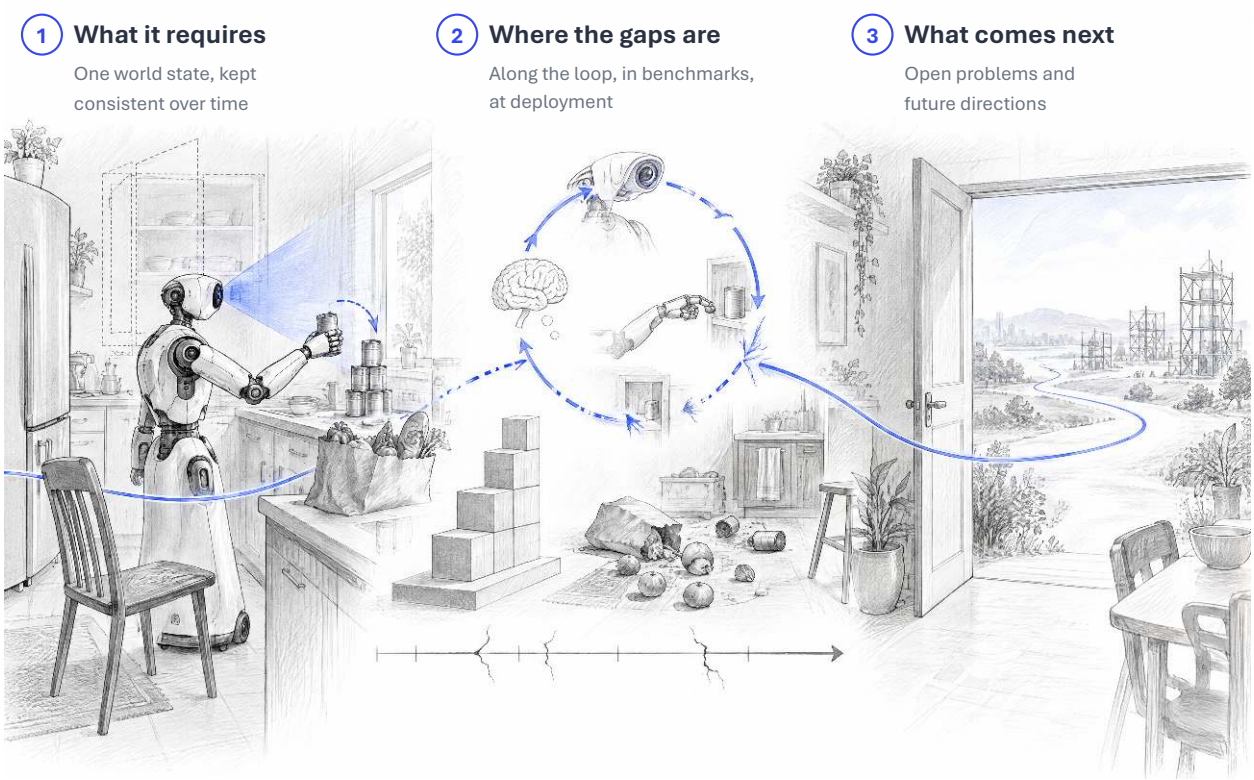


Figure V: Part V in one view. (1) What it requires: a household robot must keep one consistent account of its kitchen while it turns away, people move things, and a stack of cans may topple. (2) Where the gaps are: the loop from perception to action still breaks, benchmark success diverges from deployment, and consistency erodes over long horizons. (3) What comes next: open problems that set the directions for future work.

Parts II to IV described how spatial representations are constructed, how they support perception, reasoning, and generation, and how they are deployed in applications ranging from household robots and surgical systems to city-scale digital twins. However, current systems often perceive or act well at a single moment, while most of the open problems arise when a system must maintain a consistent picture of the world as it moves, revisits places, and acts over extended periods.

A household robot that puts away groceries must remember the location of a cupboard after turning away from it, register a chair that has been pushed into its path, track which bags it has already emptied, and judge whether a stack of cans will remain stable if another can is added. Methods described earlier in this handbook handle each of these steps reasonably well in isolation. The difficulty lies in maintaining a single account of the kitchen that all steps read from and write to, and in keeping that account correct while the robot and the people around it change the scene. In the formalism of Chapter 2, this account is the world state $\mathcal{R}_t^{\mathcal{S}}$, and keeping it correct is the role of the update $\Psi_{\mathcal{S}}$. It is also the problem that Section 1.2 identifies as open for the current era, long-horizon consistency under motion and partial observability.

Chapter 15 asks what makes such a world state usable: what can be inferred from limited observations, what must be represented, how beliefs are kept current, when predictions can be trusted, and how spatial knowledge is acquired through interaction in shared space. We present the evidence on each question and state which answers remain open. Chapter 16 turns to gaps in current methods and benchmarks, the difficulties of deployment, and directions for future work. Chapter 17 concludes by revisiting the definition of spatial intelligence introduced in Chapter 2.

Chapter 15

Foundations: What Spatial Intelligence Requires

Advances in spatial perception, reasoning, and generation have enabled AI systems to recover geometry from images, answer questions about layout, and simulate complex environments, as Parts II to IV documented. Success on these tasks does not establish that a system maintains a reliable model of the physical world. A model can reconstruct a room to millimeter accuracy, answer most questions about a photograph, or continue a video convincingly, and still fail when observations are incomplete, when the environment changes while it is not looking, or when its actions depart from the conditions under which it was trained. The failures reported throughout this handbook cluster at these points, which suggests a shared deficiency rather than a collection of unrelated task-level shortcomings.

This chapter asks what makes an internal model of physical space usable for continued reasoning and action. By usable we mean that the model can be queried for what a task requires, extended when new evidence arrives, and relied upon when its predictions are acted on; we do not presuppose that such a model must be an explicit 3D reconstruction, nor that it must be a video generator. The question decomposes into five, each posed at a different level. What can be inferred from limited observations, and what remains underdetermined (Section 15.1)? Which information must a representation retain, and for which operations (Section 15.2)? How can a world state be maintained and revised over time (Section 15.3)? When can its predictions be trusted (Section 15.4)? How is spatial knowledge acquired and used through interaction, in environments shared with other agents (Section 15.5)? The five are interdependent: action produces new evidence, which re-enters at the first level and constrains all of the others.

15.1 Spatial Observability and Identifiability

Spatial understanding begins with an incomplete view of the world. A system never has direct access to the world state $\mathcal{R}_t^{\mathcal{S}}$ of Chapter 2; it has images, video, depth, or sound, from which that state must be inferred. Images and videos carry rich evidence about appearance and structure, but they do not uniquely determine every geometric, physical, or functional property of a scene. A single image fixes the scale of a scene only through the prior knowledge brought to it (Section 6.2.3), an occluding surface hides whatever lies behind it, and the mass of a box or the state of a lock leaves no trace in a photograph. A usable model must therefore keep track of which of its contents are observed and which are inferred.

Whether a spatial inference fails because the model cannot extract the evidence available, or because the evidence itself is insufficient, is a distinction current evaluation rarely draws. The former is a limitation of

learning or computation, and better models or more data can be expected to reduce it. The latter is a problem of identifiability, which more capable inference alone does not remove. Asked whether a sofa seen in one photograph will fit through a door seen in another, a multimodal model typically produces a confident answer with a coherent explanation, yet without a common scale between the two images the question has no determinate answer, and a model with perfect geometric reasoning should say so. Evaluation that counts every wrong answer as a model failure conceals this class of cases, while evaluation that accepts every ambiguity as irreducible would excuse failures that are not.

Additional modalities constrain different aspects of the underlying state, and their contributions are neither interchangeable nor universally sufficient. A second viewpoint supplies parallax, which fixes relative depth and reveals part of what the first view occluded. A depth sensor supplies distances directly, with an accuracy and a scale behavior that depend on the sensing principle (Section 6.2.1). Motion adds constraints on dynamic geometry; sound carries direction, distance, and room structure (Section 6.5.1); contact and force reveal mass, friction, and compliance, which no camera observes (Section 7.1.2). Assessing a modality therefore requires specifying which ambiguities it resolves, under which assumptions, and which properties remain unidentifiable once it has been added. Part II treated the modalities as alternative representations of space (Chapter 4); here they appear as complementary constraints on one underdetermined state.

Observability is not fixed by the initial input. Moving to another viewpoint, changing the illumination, or pushing an object can reveal previously ambiguous properties, and some properties, such as a drawer's operability, may require physical interaction. Theory of Space and VABench bear directly on this point: models perform better with supplied observations than when they must gather evidence themselves, and allowing a model to move its camera and look again raises its manipulation success substantially [231, 3005]. Spatial intelligence therefore depends partly on recognizing when an additional observation would change what is knowable. How an agent should choose such observations is a question of decision making, taken up in Section 15.5, which properties are identifiable depends on the agent's available actions.

Distinguishing uncertainty from fundamental underdetermination, and stating what evidence would resolve it, are abilities that current systems are not asked to demonstrate. Static question accuracy cannot distinguish correct reasoning under insufficient evidence from incorrect reasoning despite sufficient evidence, and controls as simple as removing the image or shuffling the input remain rare (Section 6.5.2). A benchmark designed around identifiability would include questions that are unanswerable from the given input and reward a model for saying so, pair each ambiguous case with the modality or action that resolves it, and score whether the model asks for it. Given these limits, which obtainable properties should a spatial representation preserve?

15.2 Representation and State Sufficiency



The same physical environment can be represented through images, point clouds, occupancy grids, scene graphs, neural fields, or the latent state of a learned model, and Part II surveyed each of these forms (Chapter 4). They differ less in how much information they hold than in which aspects of the environment they preserve, which they abstract away, and how they organize what remains. Scene graphs typically emphasize entities and their relations, Gaussian representations primarily encode appearance and geometry, and a latent state retains what its training objective rewarded; each can be extended with further structure, but none is informative in isolation from the questions and operations it is meant to support.

That spatial knowledge and spatial reasoning are separable is shown by large models that acquire the former without the latter. A transformer trained only on Othello move sequences forms an internal representation of the board (Section 4.4), and language models encode the geographic positions of cities in their activations without ever being shown a map [3006]. The same models remain well below human performance on perspective taking, mental rotation, and building a model of a place from a sequence of views [3007], and

chain-of-thought prompting, which helps on arithmetic and code, tends to degrade their accuracy on visual spatial benchmarks [1501]. Performance recovers when the model is allowed to draw a coarse cognitive map, render an imagined view, or call a geometric tool before answering (Sections 7.3.5 and 7.3.8). Encoding spatial information and computing with it are therefore distinct requirements, and reasoning performance depends partly on whether the representation in use supports the operations the task demands. This does not show that language is unsuited to spatial reasoning; it suggests that a representation in which distances and viewpoints are described rather than held tends to accumulate error each time they are manipulated.

Spatial tasks differ not only in the information they require but in the computations they demand of it. Judging a relative position needs a reference frame and a relation; metric reasoning needs distances, angles, and a scale; changing viewpoint needs a rigid or projective transformation applied to 3D structure; navigation needs topological connectivity and traversability; manipulation needs local geometry together with contact constraints and material properties. A representation suited to semantic relations can be insufficient for metric transformation, and a detailed geometric reconstruction can still omit the properties on which physical interaction depends, a gap Section 11.1 summarized as geometry is not behavior. The effectiveness of a representation is therefore a matter of its compatibility with the downstream operations. The recurring finding of Part III that hybrid systems are limited by their weakest interface (Section 7.4) is an instance of this: the interface is where one representation's operations are asked to run on another representation's content.

These observations suggest a task-relative criterion of sufficiency. Spatial-state sufficiency depends on the queries, predictions, and actions it must support; a representation is sufficient for a family of tasks if every decision the family requires can be made from it as well as from the world itself. Information irrelevant to one task may be essential to another, so sufficiency cannot be judged from the representation alone. This creates a tension between compact states optimized for an immediate objective and general-purpose states intended to remain useful across tasks that are not yet known. A latent world model can succeed at the tasks it was trained for while modeling the environment's dynamics unfaithfully [3008], which satisfies sufficiency for those tasks and gives no assurance for others; a dense geometric reconstruction, by contrast, serves many queries but remains silent about physical properties that are not visible.

Whether a finite shared state can preserve the distinctions required by an open-ended range of tasks, or whether a spatial system should instead maintain adaptable representations whose detail and structure depend on downstream demands, remains open. Shared spatial latents read by many task heads, task-specific representations maintained in parallel, and interfaces through which modules request information at the precision they need (Section 7.4) represent three answers, and the available evidence does not yet favor one, particularly when future tasks are unknown.

Evaluating a spatial representation therefore requires examining the information it preserves and the operations it supports, rather than its expressiveness in the abstract. Reconstruction quality and recognition accuracy are the usual proxies, and both are insufficient: a photometrically accurate replica can fail a metric tolerance check and encode nothing about which drawers open (Section 11.1), and a correct answer can rest on language priors rather than on a correct belief about the scene. Sufficiency has to be tested across tasks, by asking whether a representation fixed for one purpose supports queries, transformations, and action predictions it was not built for. Whether a state that is sufficient at one moment remains so as the world changes and objects leave the field of view is a separate question, taken up next.

15.3 Persistent World States and Belief Updating



NDERSTANDING a scene at one moment differs from maintaining an understanding of a changing world. An assistant asked where the keys were left must have observed them, retained their location after the user walked away, distinguished that remembered location from the current view, and allowed for the possibility that someone moved them in the meantime. Egocentric benchmarks built around such

queries show that an explicit memory, updated as video arrives and tracking objects relative to the moving user, substantially improves leading multimodal models [3009], whereas most commercial glasses answer from a single frame and keep no map (Section 9.3). The household robot in this part's introduction faces the same requirement at every step. In the formalism of Chapter 2 this is the role of the update $\Psi_{\mathcal{S}}$, the component that Parts II to IV examined least.

Access to a long observation history does not ensure a consistent world state. Video world models steered through long explorations fail to render revisited places consistently even as image quality improves [3010]; multimodal models given several views of a room rarely detect that one view was captured after an object moved [3011]; and over routes spanning several streets, models answer by recognizing landmarks and matching text, with little evidence of path integration [3012]. A model may retain every earlier frame and still fail to recognize a revisited place, preserve an object's identity, or decide which remembered properties have become outdated. Research on spatial memory has concentrated on recall, through longer contexts, better retrieval, and richer memory structures (Section 7.3.6), while these failures concern revision. Persistence therefore involves maintaining and revising beliefs about the world, not merely retaining observations.

A persistent world state must distinguish absence of observation from evidence of absence, maintain uncertainty during occlusion, and reconcile new observations with prior beliefs. When a new observation conflicts with a stored one, the system must decide whether the environment changed, the earlier estimate was wrong, or the two observations refer to different entities, each requiring different state updates. Classical systems partly address this problem: SLAM detects loop closures and corrects the map upon revisiting a known place (Section 6.3.5), object trackers maintain identities through occlusion, and dynamic scene graphs timestamp their edges. Learned systems often struggle to make such revisions reliably over extended interactions. Models retain outdated beliefs after the world changes [231], append-only memories accumulate contradictory entries, and agents drift over long episodes as their context fills (Section 7.3.7). Increasing capacity alone may not suffice, since a memory that only grows keeps each outdated belief beside its replacement. A write policy that decides what enters the state, with what confidence, and when an entry is revised or removed, is a prerequisite that current systems leave largely implicit.

Where such a state should be kept is a question of architecture rather than of representation format. Explicit structures such as maps, scene graphs, and splats (Section 4.2) persist by construction and make selected state variables accessible and correctable, but they must anticipate every type of object, property, and change they will be asked to encode. Learned memories and recurrent latent states can encode properties no designer specified, and whether they track the world is difficult to verify [3008]. Generated frames are the most flexible option and lose whatever leaves the context window. Hybrid designs that store static geometry and dynamic objects separately and retrieve them on revisit [3013], or that keep slowly changing layout and identities in an explicit layer and leave appearance and short-term dynamics to a learned one, combine these advantages at a cost of their own: geometry, object identity, uncertainty, and temporal updates must stay synchronized across layers, and coordinate frames and scales can silently diverge at the interface (Section 7.4). The separation can be taken further. State World Models keep a typed state ledger that evolves under executable rules written by a coding agent and use a frozen video model only to render each resulting event; on counterfactual twin pairs that share an initial frame and action but differ in hidden state, this harness produces the rule-required outcome in 92% of pairs, where the best of eleven video models prompted with the same rules reaches 32% [3014]. The result suggests that what fails in generator-only designs is the maintenance of state the current frame does not show, rather than rendering. The appropriate architecture follows from the requirements of state maintenance, how long each part must persist and whether it must be inspected, rather than from a general preference for explicit or learned form.

World states should remain persistent without becoming rigid and revisable without losing established structure, and evaluation currently measures neither property. Most benchmarks refer to a single moment, and the protocols that would expose persistence failures are not yet standard (Section 7.5): streaming protocols that withhold future frames [3015], revisit protocols that move the camera away and back [3010],

and change protocols that alter the scene behind the agent. A more informative evaluation reads out the state itself, as Theory of Space does at each exploration step [231], and scores separately what the system has observed, what it remembers, and what it predicts, since conflating the three is how a stale belief passes as a current one. Deployed systems could likewise report how far their world state lags behind the world it describes, a quantity that digital twins call the liveness gap (Section 11.1) and that is rarely measured elsewhere. A state that remains consistent over hours can nonetheless describe appearance accurately and physical behavior poorly.

15.4 Predictive Validity and Physical Grounding

A video generator asked to continue a clip of a ball rolling off a table usually produces a smooth, well-lit continuation in which the ball falls, often along the wrong trajectory or with an implausible bounce. Benchmarks of physical plausibility find that the physical accuracy of video generators is poor and does not improve with visual quality (Section 7.5), and Chapter 8 reported the same pattern across its subfields: shapes that change between views, feet that slide over the ground, and motions that violate conservation (Section 8.8). A plausible continuation reproduces the appearance of a physical event without predicting its trajectory, contacts, or consequences. Yet the same class of models learns substantial scene structure, since image generators adapted for perception segment, estimate depth, and ground referring expressions at the level of specialist models [1136]. Structure and dynamics are apparently learned to different degrees from the same data.

World-model correctness has several distinct meanings, and the criteria are often run together. Visual plausibility concerns realistic appearance; geometric consistency concerns coherent spatial structure across views and time; dynamical fidelity concerns how the world evolves under actions; decision reliability concerns whether actions planned in the model succeed in reality. The criteria are related but not interchangeable, and they are ordered by how much they depend on action. Common video metrics measure the first and do not register physically impossible motion (Section 8.3.6); view-synthesis metrics measure the second and cannot tell a solid object from a cut-out that matches the training views (Section 8.2.1); the third has no standard protocol (Section 8.6.4); the fourth is tested, when it is tested at all, by deploying the resulting policy. A model can score well on the first two and fail the last two, and vision-language models used as judges share the spatial blind spots of the models they evaluate (Section 8.8).

Learning how scenes typically evolve from observational data does not guarantee correct predictions under novel interventions. In the action-conditioned definition of Chapter 2, $f^{\mathcal{S}}_{\text{pred}}$ maps a state and an action to the next state, but a generator trained on video sees only the actions that happened to be recorded, mostly everyday activity and camera motion, and sees them correlated with the states and circumstances in which they occurred. A model that has watched thousands of doors being pushed open has learned what usually follows a push, not what determines the outcome, and so has little basis for predicting the locked door, the obstructed door, or the push applied at an unusual angle. The regularities it learns are real and useful for geometry, appearance, and common motion; they constrain little about the consequences of an action outside the data. An interventional test, which fixes the initial state, varies the action, and checks whether predicted outcomes vary as real outcomes do, is the most direct way to distinguish a model of what tends to happen from a model of what an action causes.

Accurate geometry does not imply an actionable physical model. Mass, friction, articulation, and compliance are often visually unobservable yet critical for manipulation (Sections 7.1.2 and 9.1), and a Gaussian-splat replica of a room can be photometrically faithful while encoding nothing about which drawers open or how much force a door requires (Section 11.1). Real-to-sim pipelines estimate such quantities by manipulating objects and observing the response [196], thereby estimating properties needed for prediction, and world-action models that predict compact encodings of future observations rather than pixels avoid spending capacity on appearance that control does not use [3016]. Confidence sets calibrated under one

data-collection policy may become unreliable under policy shifts [3017], and model-based action selection may induce policy shifts that invalidate this calibration.

A tractable form of the question is where, rather than whether, a world model can be trusted. Treated like a measuring instrument, a world model would be calibrated against the environment it claims to simulate, carry a stated range of states and actions within which it is valid, and be used with caution outside it. Establishing that range requires tests under controlled interventions and unfamiliar actions, under environmental change, and against downstream decisions, in place of average visual or predictive accuracy. World models expressed as code, whose rules are explicit and correctable when they conflict with observation, admit such tests more readily than learned ones (Section 8.6.2.3), and the natural sciences already hold generative models to this standard, counting a proposed material only once it has been synthesized (Section 12.2). When an action falls outside a model's established range, the agent must decide how to obtain further evidence, which actions are safe meanwhile, and how to use the outcome to extend the model.

15.5 Situated and Interactive Spatial Intelligence



PATIAL competence is not exhausted by interpreting observations or predicting future states. In an unfamiliar environment an agent must decide where to look, what to investigate, and when to act under uncertainty, and models that perform well on provided evidence can perform poorly when responsible for gathering it. Theory of Space makes the comparison directly: foundation models asked to explore until they knew where objects were did worse when they collected the observations themselves than when the same observations were supplied, and explored less systematically than simple scripted strategies [231]. On VABench, models that judge spatial relations well still fail many manipulation tasks and almost all long-horizon ones, and the gap narrows when they may move the camera and look again [3005]. These results locate part of the weakness in deciding what to do, rather than only in recognizing what is seen.

Section 15.1 asked which properties can be made identifiable by acting; the question here is which action to choose. Information gathering should be guided not only by uncertainty but by the consequences of that uncertainty for the decision at hand. An observation that sharpens a reconstruction may be worth less than one that settles whether a passage is traversable or a grasp is safe, and the value of a view depends on the goals, costs, and risks of the actions that follow it. Exploration policies learned in simulation formalize this for navigation objectives (Section 7.2.5); the counterpart for the richer states of Sections 15.2 and 15.3, where the quantity to resolve may be an object's identity, a person's belief, or a door's state, is largely untrained. Reporting accuracy as a function of an interaction budget, the number of views, moves, or queries a model may spend, would place passive and active evaluation on one scale and reward agents that reach correct decisions with few observations.

Passive data, simulation, and real-world interaction provide complementary forms of experience. Internet video, the largest source, supports perceptual and physical priors and already enables robot planning after modest fine-tuning (Section 8.6); simulation and generated worlds permit controlled exploration without risk (Section 7.2.5); real interaction alone exposes the consequences of the agent's own decisions, which passive video does not contain and on which both the update policy of Section 15.3 and the predictor of Section 15.4 depend. Each source carries its own distortions. The simulation gap is widest where contact, materials, lighting, and sensor noise matter (Sections 9.1 and 11.3), and experience generated inside a world model is useful only within the model's range of validity. Learning inside a world model, with rollbacks to earlier decision points and correction of the model from real transitions [3018], lowers the cost of real interaction without removing the need for it. How to combine the three sources without transferring incorrect assumptions from simulated or observational distributions into real-world action remains unresolved, and existing robot datasets, concentrated on tabletop success, offer little of the failures and recoveries such a combination requires (Section 7.3.2). The consequences of an action, moreover, are not confined to the agent's own observations: in a shared space, acting also changes what other agents observe, believe, and do.

Different agents observe and understand the same environment differently, and a shared physical world does not imply shared knowledge of it. A system must relate its own reference frame to those of others, since a location on the user's left may lie on the robot's right and people switch between egocentric and allocentric descriptions without noticing (Section 7.3.1). It must distinguish the physical state from what a person believes about it, since a user who left the kitchen before the kettle was moved still believes it is on the stove. It must anticipate how its own actions change what others perceive and do. Mental World Modeling maintains joint physical and mental states for this reason and finds both necessary to predict human decisions (Section 8.6.3.2) [2352]. People, on this view, belong in the world state on the same footing as walls and doors, with positions, viewpoints, and beliefs treated as quantities to be estimated.

In human environments, spatial actions are inherently interactive. Robots that model people only as moving obstacles freeze in dense crowds once every path appears blocked, unable to anticipate that people will make room (Section 13.1), and an assistant's motion, speech, and help change what the people around it do next, so a plan that ignores this is planning against a stale state. Persistent spatial memory carries a second consequence: a world state that includes people records where they live, what they own, and how they spend their days, and the design choices of what to store, at what detail, and for how long matter more than the access controls applied afterward (Sections 10.1 and 13.1). Such systems also act on their users' own spatial abilities, which weaken under habitual reliance on navigation aids and strengthen with practice [3019]. Reliable spatial agency must therefore account for other agents' responses and for the consequences of maintaining and using spatial knowledge about them.

Agents that acquire the information they need, act despite incomplete knowledge, and revise their world models from the consequences of interaction would address each of the preceding questions in operational form: action changes what is identifiable, exposes which information the representation lacked, supplies the observations that keep the state current, and tests the predictions the model made. Spatial intelligence, on this view, is a closed loop in which perception, representation, prediction, and action continually constrain and improve one another. A usable model of physical space is one that remains open to new evidence, explicit about its limits, and corrected by what its own actions reveal.

Chapter 16

Research Gaps and Future Directions

The problems in this chapter are widely acknowledged and concern missing methods, inadequate measurements, and systems that work in the laboratory but not yet in open environments. Section 16.1 describes research gaps along the loop of Figure 2.1, Section 16.2 gaps in how progress is measured, Section 16.3 bottlenecks in deployment, and Section 16.4 directions for future work.

Chapter 15 asked why these problems are hard in principle; this chapter asks where current systems fail, how the failures can be measured, and which methods could be tested against them. Four themes are its own: competence at tabletop or room scale does not transfer to buildings and cities; correct spatial judgments do not reliably become completed long-horizon actions; much evaluation does not measure the abilities it is named for; and deployed systems face latency, distribution shift, security, data scarcity, and multi-agent synchronization that research systems rarely confront. Where a failure has a principled explanation in the preceding chapter, we point to it rather than repeat it.

16.1 Research Gaps along the Loop

Most of the following gaps are invisible in single-step evaluations and appear only when a system must integrate many views, revise its memory, act over many steps, or operate beyond room scale.

Models recognize scenes without relating views to one another. Multimodal models can name nearly every object in a room yet cannot reliably say where the objects are; Chapter 6 found them strong on contextual and functional relations and weak on distances, sizes, and angles (Section 6.3.1). Given several views of a room, they rarely notice that one was captured after an object moved [3011], and over routes spanning several streets they answer by recognizing landmarks and matching text, with little evidence of path integration [3012]. These failures admit more than one explanation: the model may hold no account of the scene against which a later view could be checked, or it may hold spatial information that it cannot associate across viewpoints, update, or use in a metric computation. Current benchmarks do not separate these cases (Section 16.2).

SLAM registers each new view against a map and treats disagreement as a signal to correct one or the other (Section 6.3.5). Whether registering views in a common frame before language is involved (Section 5.5), for instance with feed-forward models that regress point maps from image collections (Section 6.2.4), closes the gap for multimodal models is unsettled, and metric estimates must in any case survive unknown or wrong camera calibration, the main open problem even for specialist pipelines (Section 6.2.3).

Memory breaks when it has to be revised. Section 15.3 argued that persistence requires revising beliefs, not only retaining observations. The gap is that few learned systems implement a write policy deciding

what enters the world state, with what confidence, and when an entry is revised or removed. Work on spatial memory has concentrated on recall, through longer contexts, better retrieval, and richer memory structures (Section 7.3.6), whereas the reported failures concern revision: outdated beliefs kept after the world changes [231], append-only memories that accumulate contradictions, and agents that drift as their context fills (Section 7.3.7). Applications make the requirement concrete: a manipulator must keep an object's pose current while its own gripper occludes it (Section 9.1), and a home assistant must track object locations over weeks as the household moves things (Section 13.1). SLAM makes a version of these decisions when it selects keyframes, merges landmarks, and discards map regions that no longer match (Section 6.3.5); we are not aware of a learned counterpart for the richer world states that applications now require.

Pixel prediction is mismatched with the needs of control. Video world models predict pixels, while a controller needs the quantities that decide whether an action succeeds, such as contact, mass, and friction, which are often invisible in images (Section 7.1.2). A pixel loss spends most of a model's capacity on lighting and texture, pixel predictions drift over long horizons (Section 8.3.6), and a model can score well on tasks with incorrect dynamics [3008]. The objective rewards appearance; the use requires outcomes. World-action models that predict compact encodings of future observations match generative ones at much lower cost [3016], following the predictive-embedding principle of the JEPA family (Section 8.6.1), and a further step is to predict physical quantities directly, estimated through pushing, lifting, or probing. Whether such predictions hold under intervention is the question of Section 15.4; the gap here is that these quantities are rarely prediction targets at all.

Correct perception does not carry through to completed actions. Chapter 7 described the gap between spatial judgment and action as instructions interpreted correctly and executed incorrectly (Section 7.3.2). On VABench, models that judge spatial relations well still fail many manipulation tasks and almost all long-horizon ones, and degrade markedly on unfamiliar shapes [3005]; generalist robot policies improve rapidly, yet their spatial representations drift over long action sequences (Section 7.3.2). Small errors compound over many steps, and in contact-rich tasks a single misjudged grasp can end an episode, so the gap is one of execution rather than judgment and grows with the horizon. An agent must keep observing and correcting until the task is complete, as the gains from letting a model look again suggest (Section 15.5); this is the third open tension of Chapter 3, the integration of spatial reasoning into long-horizon embodied agents. Closing it requires training and evaluating the act–observe–correct loop as a whole, with benchmarks that reward recovery as well as first-attempt success.

Room-scale results say little about larger spaces. Most results in this handbook concern single rooms or tabletops. At street and city scale, models revert to landmark recognition even over a single kilometer [3012], geospatial models trained on one region or sensor degrade on another (Section 11.2), aerial and ground robots must align views from very different heights (Section 9.4), and skills learned on one robot body transfer poorly to another (Section 9.1). Scale changes more than the size of the numbers: the dominant representation shifts from local metric geometry to routes and topology and then to georeferenced coordinates, the sensing from egocentric views to aerial imagery, most of the state from seen to remembered or inferred, and the rate of change from seconds to years.

Geographic information systems handle scale by nesting reference frames, placing an object in a room, the room in a building, and the building in a georeferenced map, each level with its own coordinates and precision. Learned spatial models rarely represent such nesting. We expect nested frames to be part of the solution, together with benchmarks that pose the same question at several scales (Section 16.2), and we regard cross-scale transfer as a challenge in its own right.

Each of these gaps appears where one part of the loop depends on another's output, which is why evaluating each part on its own misses them.

16.2 Benchmarking Gaps

The preface asked how the field can measure genuine progress without mistaking benchmark saturation for it. Chapter 15 described the properties a usable spatial model should have; this section asks how to measure them. Most benchmarks compared in Section 7.5 pose questions about a single image or video, and few require the model to act. This format admits shortcuts, which models have repeatedly exploited: models that nearly solved synthetic visual reasoning datasets failed on novel combinations of familiar elements and on real images (Section 7.2.1), and much of a referring-expression score survives when the words of the expression are shuffled (Section 6.5.2).

Measured against the loop of Chapter 2, current evaluation covers some components far better than others. Perception is evaluated frame by frame, through detection, segmentation, depth, and tracking accuracy, and rarely by whether its uncertainty is calibrated or its output stable over time. Reasoning is evaluated by answer accuracy on static questions, rarely by whether the answer rests on a correct belief about the scene. Prediction is evaluated by visual quality, rarely by correctness under actions absent from training. Action is evaluated by task success in simulation, rarely by recovery after failure. The update Ψ_S is rarely evaluated on its own, and neither is the effect a system has on the people who use it.

Score the belief behind the answer. A correct answer may come from language priors, regularities in the answer format, or a correct model of the scene, and accuracy alone cannot tell them apart, nor separate the explanations left open for the first gap of Section 16.1. Benchmarks can instead read out a model's belief about the scene and score it directly, as Theory of Space does at each step of an exploration [231], or probe hidden states, as was done for the Othello model of Section 4.4. A cheaper variant asks several questions about the same scene and checks that the answers are consistent with a single layout. Controls that remove the image or shuffle the input are inexpensive and should be standard.

Gathering evidence is a separate ability. Static benchmarks supply the evidence; deployed systems must collect it. Deciding where to look, when to move, and when to ask is a skill in its own right, and the active–passive gap of Section 15.5 shows that current models lack it. We propose reporting accuracy as a function of an interaction budget, the number of views, moves, or queries a model may use, which places passive and active evaluation on one scale and rewards models that reach correct answers with little effort. In deployment the budget has a direct cost in time and energy.

Spatial failures unfold over time. Most benchmark questions refer to a single moment, while many failures in Section 16.1 emerge only over time. Streaming protocols show the model only the video seen so far, so it cannot use future frames to answer about the past [3015]; revisit protocols move the camera away and back to check whether a place is rendered consistently or changed correctly [3010]; change protocols alter the scene behind the agent and test whether its belief follows. None is yet standard, and long-horizon tasks remain rare even among embodied benchmarks (Section 7.5).

The same question should be asked at several scales. Benchmarks cluster at room scale and are often built on a few indoor scan collections, while surgery, orbital operations, and underwater robotics have benchmarks of their own that rarely share protocols with them (Sections 12.1 and 11.3 and Chapter 14). Posing the same spatial question at the scale of a table, a building, and a city would show where competence stops transferring and whether a model represents nested frames; a model that answers well about a desk and poorly about a building has learned the regularities of tabletop scenes, not a general spatial ability.

Generated worlds should be judged by what an agent can do in them. Generated and reconstructed worlds are still evaluated mainly by appearance, which Chapter 8 showed to be misleading. Video metrics do not register physically impossible motion (Section 8.3.6), rendered-view similarity cannot distinguish a solid object from a cut-out that matches the training views (Section 8.2.1), no standard protocol exists for the physical correctness or long-term stability of world models (Section 8.6.4), and vision-language judges share

the blind spots of the models they evaluate (Section 8.8). A more informative test asks whether an agent that plans or trains in the generated world succeeds in the real one, complemented by the interventional tests of Section 15.4.

Benchmarks should borrow construct validity from psychometrics. Spatial benchmarks adopted tasks from psychometrics, from mental rotation to perspective taking (Chapter 2), but not its standards. A psychometric test reports the distribution of human performance, establishes reliability across repeated testing, and checks construct validity through agreement with other tests of the same ability and divergence from tests of different ones; most AI benchmarks report at most a single human score. A benchmark whose questions can be answered from text alone does not measure spatial ability, and automatically generated benchmarks, now common, need such checks all the more.

16.3 Deployment Bottlenecks



ost spatial systems that operate reliably in deployment do so by restricting the world they must handle, a pattern that recurs across Part IV. Driving systems are approved for a restricted operational design domain and add sensors and fallbacks to stay within it (Section 9.2); warehouse robots work in sites with fixed item sets and layouts (Section 9.5); surgical robots are mostly teleoperated and planetary rovers act within ground-supervised workflows (Sections 12.1 and 11.3); headsets track most reliably in previously mapped rooms, and the first home robots rely on remote operators for tasks they cannot yet perform (Sections 10.1 and 13.1). When the domain is widened, the world state is typically the first component to fail.

Guarantees end at the edge of the model. A constraint-based planner certifies collision avoidance only for the obstacles it knows about, and a driving system is validated only on anticipated scenarios (Sections 9.1 and 9.2). Learning moves these boundaries without notice: a model accurate on average can be confidently wrong in rare situations, its uncertainty estimates can lose validity when its use changes (Section 15.4), and an update that improves one behavior can silently degrade another (Section 9.2). No practical method yet states and verifies a property of a learned spatial system over a realistic domain. Legal responsibility is bounded the same way, since under conditional automation the manufacturer is responsible for driving only within a narrow domain (Section 9.2), and each extension reopens the question.

The large model cannot sit in the inner loop. A headset must respond to head motion within tens of milliseconds, a robot arm must update its state estimate at control rate, a drone must detect small objects in real time on onboard hardware, and a spacecraft must do similar work on radiation-hardened processors far weaker than ground computers (Sections 10.1, 9.1, 9.4, and 11.3). Large foundation models are much slower, even after work that accelerates them by predicting geometry internally and compressing spatial tokens [3020]. A practical architecture lets a small, fast component keep the world state current at sensor rate and lets the large model consult it when time permits, mirroring the fast–slow division that Section 7.3.5 described for tool-augmented reasoning. Whether that state is explicit, learned, or mixed is the question of Section 15.3; the deployment constraint is only that it be maintained at sensor rate and remain consultable.

A map starts going stale as soon as it is made. Furniture is moved, shop displays change, buildings are renovated, crops grow, and tissue shifts during surgery, so a deployed map must be updated continually, which Part IV found difficult in every domain. How a state should be revised is the algorithmic question of Section 15.3; the deployment problem is the cost and latency of keeping it revised and the synchronization of its copies across devices. Digital twins call the discrepancy between model and asset the liveness gap and treat closing it at scale as open (Section 11.1); lifelong SLAM must withstand months of appearance change (Section 6.3.5); shared augmented-reality maps are hard to keep current (Section 10.2); surgical navigation

loses its preoperative registration once the brain shifts (Section 12.1). We suggest that deployed systems report how far their world state lags behind the world it describes, a quantity that is rarely measured.

Real data are scarcest where simulation is least trustworthy. Spacecraft perception is trained mostly on simulated images, and simulators do not reproduce the lighting of orbit, reflections from spacecraft materials, or radiation noise (Section 11.3); touch, force, and sound have the least data of all modalities and are also hard to simulate (Sections 7.3.8 and 9.1); clinical data are scarce because of privacy regulation and the cost of expert annotation (Section 12.1). In the first two cases simulation cannot compensate, since the simulator is least accurate for exactly the effects the missing data would capture. Simulations built from real measurements, data engines that target rare cases, and cross-institution training without sharing raw data all help, but none yet operates at the breadth that general spatial competence requires.

A persistent map is also an attack surface. Geospatial data raise issues of sovereignty and adversarial manipulation (Section 11.2), a compromised digital-twin server exposes the layout and vulnerabilities of the asset it mirrors (Section 11.1), and world-action models can be attacked through their visual inputs, with small perturbations producing large changes in behavior [3021]. These risks grow with the persistence of a map and the number of parties sharing it, turning the privacy concerns of Section 15.5 into security concerns.

16.4 Future Directions



The following directions build on the gaps above and on the questions left open in Chapter 15. Each is a hypothesis rather than a recommendation: where it takes an architectural position, we name the alternative it must beat and an experiment that would decide.

Persistence and rendering can be separated. Interactive video models remember a place by regenerating it, which ties memory to the context window and to the renderer. A factorized design would keep what must persist, such as layout, object identities, and their uncertainty, in an explicit layer, and use a learned layer to predict change and render what the agent needs to see, with Ψ_S promoting confirmed predictions into the explicit layer and revising it when observations contradict it. The alternatives are a latent state with learned memory, as in models that factorize memory inside the generator [3013], and a growing context with retrieval. The hypothesis is that at equal compute the factorized design stays consistent over longer horizons and more changes; it is refuted if a latent-only model matches it on revisit and change protocols while passing the faithfulness tests of Section 15.4. The decisive experiment is an agent that leaves a building, returns hours later, detects what changed, and acts accordingly; for the factorized variant, the open question is the interface that decides when a prediction enters the explicit layer and when an entry is overridden.

The medium of spatial reasoning can be learned per query. Section 15.2 found that spatial answers improve when a model can work with maps, imagined views, or geometric tools. The choice of medium can be learned per query from whether it led to a correct answer and at what cost, so that simple questions are answered in language and a map or simulation is invoked only when needed, with the intermediate steps later distilled into the model (Section 7.3.8). The comparison is with the same model reasoning in text alone and with a policy that always invokes the tool, on map building and perspective taking, with image-removed controls; the hypothesis fails if the learned selector does not beat the always-tool policy on cost at equal accuracy.

The update policy can be trained from experience. Passive data leave the update policy of Section 15.3 largely untrained (Section 15.5). It can be trained directly, by rewarding observations that add information to the map and penalizing beliefs kept after the evidence has changed, with learning inside a world model lowering the cost by trying alternatives in imagination first (Section 7.3.7); benchmarks that read out a model's belief (Section 16.2) supply the training signal as well as the evaluation. The target is an agent that builds as accurate a world state from its own exploration as from a curated set of views, and the scripted strategies that currently outperform foundation models [231] are the baseline to beat.

Verification is feasible where part of the state is explicit. Neural networks are difficult to verify, whereas objects, poses, and programs can be checked against constraints. The Algebraic World Model turns exploration into hypotheses checked by machine-verified proofs [3022], and uncertainty can be verified once its calibration is matched to the policy actually executed [3017]. The alternative is to verify latent states indirectly, through probes and faithfulness tests [3008], which is cheaper and yields statistical rather than logical guarantees. In either case the obstacle is perception, since any guarantee about the state inherits the errors of its estimate from sensor data. A machine-checked property, such as a bound on collision probability, holding for a learned agent over a deployment-scale domain would change the situation of Section 16.3; no current system provides one.

A shared world state needs the machinery of version control. When several agents and people maintain one world state, they need a common reference frame, stable object identities, a record of changes, a procedure for resolving conflicts, and permissions on who may read and write what. Version control solved the analogous problems for shared code, and multi-agent systems that build cognitive maps already record changes as atomic commits and cross-check one another [3023]. Embodied teams add difficulty, since ground and aerial robots must first align very different viewpoints (Section 9.4) and city-scale and planetary digital twins supply the georeferenced frames into which local maps must fit (Sections 11.1 and 11.2); permissions are also where the privacy requirements of Section 15.5 can be enforced. A suitable test is a team of heterogeneous robots and a person who maintain one consistent world state of a building over several days, each able to query how the others currently perceive it.

A theory of spatial representation has a concrete agenda. Section 15.2 proposed sufficiency relative to tasks and actions as the criterion for a representation. Turning it into a theory requires identifiability results stating which world-state variables are recoverable from a given set of modalities and actions, extending the sensor-specific analyses of Chapter 6; sufficiency results stating which variables must be preserved for a task family under every action the agent may take, of which equivariance to rigid motion (Section 7.2.3) is the one case already worked out; faithfulness tests deciding, for any architecture, whether a latent state's dynamics follow the environment's [3008]; and a specification of which parts of a representation must be explicit for inspection and verification. Such results would let the world-centered property required by Chapter 2 be tested for any system, regardless of architecture.

Chapter 17

Conclusion

Poincaré wrote in 1902 that to localize an object is to represent to ourselves the movements needed to reach it, and this handbook opened with that sentence. The work surveyed here supports his view, as the systems that act most reliably are those whose representations are most closely tied to prediction and action, and their main weakness is keeping those representations consistent over time.

This handbook set out to provide a common map of a fragmented field, reading work from SLAM and computer vision to geographic information science and the cognitive science of space through the representations each of them constructs. Part I traced five eras of spatial representation and proposed a definition in which a system and its environment are coupled through a world-centered representation that perception writes, action reads, reasoning and prediction operate on, and the update Ψ_S keeps current. Parts II to IV followed these functions to the state of the art. In the terms of Chapter 2, different communities built different parts of this loop: vision and graphics the representation, robotics the operators that act on it, SLAM an update with geometric guarantees; few systems close the entire loop.

Part V located the open problems mainly at the junctions between these parts. The grocery-handling robot introduced at the beginning of this part fails when its perception, memory, prediction, and action cease to agree about the kitchen, even though each capability works in isolation. Chapter 15 asked what such a shared state must provide to be usable: what can be identified from limited evidence, what must be represented, how beliefs are kept current, when predictions can be trusted, and how knowledge is acquired through interaction in space shared with others. Chapter 16 showed the same state failing to persist in research systems, rarely measured by benchmarks, and kept manageable in deployment only by restricting the world, and stated directions for closing these gaps as hypotheses that can be tested.

The preface posed four questions without claiming final answers, and Part V suggests provisional ones. A system is spatially intelligent, beyond its performance on particular tasks, when it maintains a world state that stays consistent as it moves and acts and that can serve tasks it was not designed for. Geometric, semantic, and commonsense structure integrate best when each part of the state is stored according to how long it must persist and whether it must be inspected, and modalities as different as pixels, point clouds, language, and audio contribute as complementary constraints on one underdetermined state, each resolving ambiguities the others leave open; registering views in a common frame before language is involved appears to be the most promising route to cross-view alignment, though whether it closes the gap remains to be shown. Spatial knowledge is most likely to transfer between environments when it is anchored in nested frames, from objects to rooms to maps. Planning and action emerge from spatial understanding through prediction that is tested under intervention and through a loop in which the agent keeps observing and correcting. Genuine progress shows in how well a system gathers evidence, keeps its beliefs current, and uses them, more than in its answers to static questions.

Chapter 2 called its definition a definition in motion, and Part V suggests three revisions to it. The definition refers to a single system $\mathcal{S}$, whereas deployed systems increasingly share spaces and world states with other agents and with people, so it requires a plural form. It requires the representation to support perception, reasoning, prediction, and action, and deployment adds the requirement that planners, regulators, and the people it describes can inspect it. It admits social settings only as a property of the environment, whereas Chapter 15 argued that people belong in the state itself, with their positions, viewpoints, and beliefs predicted and respected alongside walls and doors.

As the preface anticipated, some of the systems discussed in this handbook will soon be superseded, and some of the positions taken here will need revision as new evidence arrives. We hope that a clear account of the open problems makes that revision easier, and that the next account of spatial intelligence finds more of the loop closed.

Acknowledge

Thanks for the support from xxx.

Bibliography

[1] Chanjun Park, Minsoo Khang, and Dahyun Kim. Model-based data-centric ai: Bridging the divide between academic ideals and industrial pragmatism. *arXiv preprint arXiv:2403.01832*, 2024.

[2] Xiaoyuan Zhang, Chengdong Ma, Yizhe Huang, Weidong Huang, Siyuan Qi, Song-Chun Zhu, Xue Feng, and Yaodong Yang. World models should prioritize the unification of physical and social dynamics. *arXiv preprint arXiv:2510.21219*, 2025.

[3] Xiaoxiao Long, Qingrui Zhao, Kaiwen Zhang, Zihao Zhang, Dingrui Wang, Yumeng Liu, Zhengjie Shu, Yi Lu, Shouzheng Wang, Xinzhe Wei, et al. A survey: Learning embodied intelligence from physical simulators and world models. *arXiv preprint arXiv:2507.00917*, 2025.

[4] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[5] Fei-Fei Li. From words to worlds: Spatial intelligence is ai's next frontier, 2025. URL https://drfeifei.substack.com/p/from-words-to-worlds-spatial-intelligence.

[6] Qishuo Hua, Lyumanshan Ye, Dayuan Fu, Yang Xiao, Xiaojie Cai, Yunze Wu, Jifan Lin, Junfei Wang, and Pengfei Liu. Context engineering 2.0: The context of context engineering. *arXiv preprint arXiv:2510.26493*, 2025.

[7] Fei-Fei Li. With spatial intelligence, AI will understand the real world. TED Talk, 2024. URL https://www.ted.com/talks/fei_fei_li_with_spatial_intelligence_ai_will_understand_the_real_world.

[8] James M. Coughlan and Alan L. Yuille. Manhattan world: Compass direction from a single image by bayesian inference. In *Proc. ICCV*, pages 941–947, 1999.

[9] Xinyu Dou, Cuijuan Liao, Hengqi Wang, Ying Huang, Ying Tu, Xiaomeng Huang, Yiran Peng, Biqing Zhu, Jianguang Tan, Zhu Deng, et al. Estimates of daily ground-level no2 concentrations in china based on random forest model integrated k-means. *Advances in Applied Energy*, 2:100017, 2021.

[10] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. 2016.

[11] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[12] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

[13] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. PaLM: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[14] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[15] Yutaro Yamada, Yihan Bao, Andrew K Lampinen, Jungo Kasai, and Ilker Yildirim. Evaluating spatial understanding of large language models. *arXiv preprint arXiv:2310.14540*, 2023.

[16] Wanyue Zhang, Yibin Huang, Yangbin Xu, JingJing Huang, Helu Zhi, Shuo Ren, Wang Xu, and Jiajun Zhang. Why do mllms struggle with spatial understanding? a systematic analysis from data to architecture. *arXiv preprint arXiv:2509.02359*, 2025.

[17] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

[18] Richard Sutton. The bitter lesson. *Incomplete Ideas (blog)*, 13(1):38, 2019.

[19] David Silver and Richard S. Sutton. Welcome to the era of experience. Preprint of a chapter in *Designing an Intelligence*, George Konidaris (Ed.), MIT Press, 2025.

[20] Yoshua Bengio. The consciousness prior. *arXiv preprint arXiv:1709.08568*, 2017.

[21] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

[22] Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen Wang, Daisy Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8154–8173, 2023.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 770–778, 2016.

[24] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[25] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.

[26] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. In *Forty-first International Conference on Machine Learning*, 2024.

[27] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

[28] Heeseung Yun, Youngjae Yu, Wonsuk Yang, Kangil Lee, and Gunhee Kim. Pano-avqa: Grounded audio-visual question answering on 360deg videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2031–2041, 2021.

[29] Jianguo Zhang, Kun Qian, Zhiwei Liu, Shelby Heinecke, Rui Meng, Ye Liu, Zhou Yu, Huan Wang, Silvio Savarese, and Caiming Xiong. Dialogstudio: Towards richest and most diverse unified dataset collection for conversational ai. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 2299–2315, 2024.

[30] Qiang Li, Oliver Kroemer, Zhe Su, Filipe Fernandes Veiga, Mohsen Kaboli, and Helge Joachim Ritter. A review of tactile information: Perception and action through touch. *IEEE Transactions on Robotics*, 36(6):1619–1634, 2020.

[31] Haoming Liu and Zhenyu Liu. A multimodal dynamic hand gesture recognition based on radar–vision fusion. *IEEE Transactions on Instrumentation and Measurement*, 72:1–15, 2023.

[32] Seonghyun Park, An Gia Vien, and Chul Lee. Cross-modal transformers for infrared and visible image fusion. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(2):770–785, 2023.

[33] HK Salih and AA Al-Ani. Odors detection and recognition based on intelligent e-nose. *J. Electr. Syst.*, 20:956–66, 2024.

[34] Eric J Topol. High-performance medicine: the convergence of human and artificial intelligence. *Nature medicine*, 25(1):44–56, 2019.

[35] Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International journal of robotics research*, 37(4-5):421–436, 2018.

[36] Emran Alotaibi and Nadia Nassif. Artificial intelligence in environmental monitoring: in-depth analysis. *Discover Artificial Intelligence*, 4(1):84, 2024.

[37] SS Muhammad Nizam, Rimaniza Zainal Abidin, Nurhazarifah Che Hashim, Meng Chun Lam, Haslina Arshad, and NAA Majid. A review of multimodal interaction technique in augmented reality environment. *Int. J. Adv. Sci. Eng. Inf. Technol*, 8(4-2):1460, 2018.

[38] Ashutosh Saxena, Min Sun, and Andrew Y Ng. Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840, 2008.

[39] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 815–824, 2023.

[40] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381: 61–88, 2020.

[41] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *proceedings of the CVPR*, pages 19129–19139, 2022.

[42] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[43] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 33: 1877–1901, 2020.

[44] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2425–2433, 2015.

[45] Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Wang. Vision-and-language navigation: A survey of tasks, methods, and future directions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7606–7623, 2022.

[46] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Soundspaces: Audio-visual navigation in 3d environments. In *European conference on computer vision*, pages 17–36, 2020.

[47] Changan Chen, Carl Schissler, Sanchit Garg, Philip Kobernik, Alexander Clegg, Paul Calamia, Dhruv Batra, Philip Robinson, and Kristen Grauman. Soundspaces 2.0: A simulation platform for visual-acoustic learning. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:8896–8911, 2022.

[48] David Howard, Agoston E Eiben, Danielle Frances Kennedy, Jean-Baptiste Mouret, Philip Valencia, and Dave Winkler. Evolving embodied intelligence from materials to machines. *Nature Machine Intelligence*, 1(1):12–19, 2019.

[49] Agrim Gupta, Silvio Savarese, Surya Ganguli, and Li Fei-Fei. Embodied intelligence via learning and evolution. *Nature communications*, 12(1):5721, 2021.

[50] Bhargavi Janga, Gokul Prathin Asamani, Ziheng Sun, and Nicoleta Cristea. A review of practical ai for remote sensing in earth sciences. *Remote Sensing*, 15(16):4112, 2023.

[51] John O'keefe and Lynn Nadel. *The hippocampus as a cognitive map*. 1978.

[52] Lawrence W Barsalou. Grounded cognition. *Annu. Rev. Psychol.*, 59(1):617–645, 2008.

[53] Benjamin Kuipers. Modeling spatial knowledge. *Cognitive science*, 2(2):129–153, 1978.

[54] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. Probabilistic robotics. *Communications of the ACM*, 45(3):52–57, 2002.

[55] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems (NeurIPS)*, 27, 2014.

[56] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 8469–8488, 2023.

[57] Benjamin Kuipers. The spatial semantic hierarchy. *Artificial intelligence*, 119(1-2):191–233, 2000.

[58] Ron Sun. Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial Intelligence*, 75(2):241–295, 1995.

[59] Howard Eichenbaum, Paul Dudchenko, Emma Wood, Matthew Shapiro, and Heikki Tanila. The hippocampus, memory, and place cells: is it spatial memory or a memory space? *Neuron*, 23(2): 209–226, 1999.

[60] Gary L Allen. Human spatial memory. *Mahwah, NJ*, pages 101–124, 2004.

[61] Nils J Nilsson. Shakey the robot. Technical Report Technical Note 323, SRI International, Artificial Intelligence Center, 1984.

[62] R. C. Arkin. *Behavior-Based Robotics*. 1998.

[63] Edward C Tolman. Cognitive maps in rats and men. *Psychological review*, 55(4):189, 1948.

[64] John McCarthy, Marvin L. Minsky, Nathaniel Rochester, and Claude E. Shannon. A proposal for the Dartmouth summer research project on artificial intelligence, August 31, 1955. *AI Magazine*, 27(4): 12–14, 2006.

[65] Patrick Krauss. What is artificial intelligence? In *Artificial Intelligence and Brain Research: Neural Networks, Deep Learning and the Future of Cognition*, pages 107–112. 2024.

[66] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[67] R. E. Fikes and N. J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3-4):189–208, 1971.

[68] Nils J Nilsson. *Principles of artificial intelligence*. 2014.

[69] Judea Pearl. *Heuristics: intelligent search strategies for computer problem solving*. 1983.

[70] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[71] Roger N. Shepard and Jacqueline Metzler. Mental rotation of three-dimensional objects. *Science*, 171 (3972):701–703, 1971.

[72] Terry Winograd. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.

[73] Marvin Minsky. A framework for representing knowledge. Technical Report AI Memo 306, MIT Artificial Intelligence Laboratory, 1974.

[74] J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11): 832–843, 1983.

[75] Howard E Gardner. *Frames of mind: The theory of multiple intelligences*. 2011.

[76] D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic based on regions and connection. In *Proc. KR*, pages 165–176, 1992.

[77] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052), 2005.

[78] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:24824–24837, 2022.

[79] H. P. Moravec and A. Elfes. High resolution maps from wide angle sonar. In *Proceedings of IEEE ICRA*, pages 116–121, 1985.

[80] Randall C. Smith and Peter Cheeseman. On the representation and estimation of spatial uncertainty. *The International Journal of Robotics Research*, 5(4):56–68, 1986.

[81] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, 1986.

[82] Feng Lu and Evangelos Milios. Globally consistent range scan alignment for environment mapping. *Autonomous robots*, 4(4):333–349, 1997.

[83] Frank Dellaert, Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Monte carlo localization for mobile robots. In *IEEE International Conference on Robotics and Automation*, 1999.

[84] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. Fastslam: A factored solution to the simultaneous localization and mapping problem. *Aaai/iaai*, 593598(2):593–598, 2002.

[85] Sebastian Thrun, Mike Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, et al. Stanley: The robot that won the darpa grand challenge. *Journal of Field Robotics*, 23(9), 2006.

[86] Sebastian Thrun and Michael Montemerlo. The graph slam algorithm with applications to large-scale mapping of urban structures. *The International Journal of Robotics Research*, 25(5-6):403–429, 2006.

[87] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2007.

[88] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, 2007.

[89] Armin Hornung, Kai M Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. Octomap: An efficient probabilistic 3d mapping framework based on octrees. *Autonomous robots*, 34(3):189–206, 2013.

[90] Ji Zhang and Sanjiv Singh. Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, 2014.

[91] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 2015.

[92] Wolfgang Hess, Damon Kohler, Holger Rapp, and Daniel Andor. Real-time loop closure in 2d lidar slam. In *IEEE International Conference on Robotics and Automation*, 2016.

[93] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9068–9079, 2018.

[94] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019.

[95] Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:16558–16569, 2021.

[96] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[97] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong T. Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pages 2165–2183, 2023.

[98] Xavier Puig, Eric Undersander, Andrew Szot, Mikael Dallaire Cote, Tsung-Yen Yang, Ruslan Partsey, Ruta Desai, Alexander William Clegg, Michal Hlavac, So Yeon Min, et al. Habitat 3.0: A co-habitat for humans, avatars and robots. *arXiv preprint arXiv:2310.13724*, 2023.

[99] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Paul Foster, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. Openvla: An open-source vision-language-action model. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pages 2679–2713, 2024. URL https://proceedings.mlr.press/v270/kim25c.html.

[100] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. *CoRR*, abs/2410.24164, 2024. doi:10.48550/ARXIV.2410.24164. URL https://doi.org/10.48550/arXiv.2410.24164.

[101] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[102] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133–135, 1981.

[103] Bruce D Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, 1981.

[104] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. 1982.

[105] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998.

[106] Paul J. Besl and Neil D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[107] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: a factorization method. *International journal of computer vision (IJCV)*, 9(2):137–154, 1992.

[108] Michael F Goodchild. Geographical information science. *International journal of geographical information systems*, 6(1):31–45, 1992.

[109] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996.

[110] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece*, 2000.

[111] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[112] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. 2nd edition, 2004.

[113] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM siggraph 2006 papers*, pages 835–846. 2006.

[114] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE international symposium on mixed and augmented reality*, pages 127–136, 2011.

[115] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[116] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018.

[117] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 165–174, 2019.

[118] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4460–4470, 2019.

[119] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[120] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022.

[121] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

[122] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20697–20709, 2024.

[123] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 5294–5306, 2025.

[124] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 248–255, 2009.

[125] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems (NeurIPS)*, 25, 2012.

[126] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2014.

[127] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2015.

[128] Alexey Dosovitskiy et al. Flownet: Learning optical flow with convolutional networks. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2015.

[129] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[130] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 779–788, 2016.

[131] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017.

[132] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 652–660, 2017.

[133] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1851–1858, 2017.

[134] Sen Wang, Ronald Clark, Hongkai Wen, and Niki Trigoni. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017.

[135] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4490–4499, 2018.

[136] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229, 2020.

[137] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, 2020.

[138] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *IEEE/CVF International Conference on Computer Vision, Proceedings of theInternational Conference on Computer Vision (ICCV) 2023, Paris, France, October 1-6, 2023*, pages 3992–4003, 2023.

[139] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. URL https://arxiv.org/abs/2304.07193.

[140] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2024.

[141] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems (NeurIPS)*, 2017.

[142] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[143] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021.

[144] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35: 23716–23736, 2022.

[145] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.

[146] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas J. Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 14455–14465, 2024.

[147] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

[148] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588 (7839):604–609, 2020.

[149] Scott Reed, Konrad Zolna, Emilio Parisotto, and et al. A generalist agent. *Transactions on Machine Learning Research (TMLR)*, 2022.

[150] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023.

[151] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding

predictive architecture. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15619–15629, 2023.

[152] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[153] Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *International Conference on Machine Learning*, 2024.

[154] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024.

[155] Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

[156] NVIDIA. Cosmos world foundation model platform for physical ai, 2025.

[157] Philip J. Ball, Jakob Bauer, Frank Belletti, Bethanie Brownfield, Ariel Ephrat, Shlomi Fruchter, Agrim Gupta, Kristian Holsheimer, Aleksander Holynski, Jiri Hron, Christos Kaplanis, Marjorie Limont, Matt McGill, Yanko Oliveira, Jack Parker-Holder, Frank Perbet, Guy Scully, Jeremy Shar, Stephen Spencer, Omer Tov, Ruben Villegas, Emma Wang, Jessica Yung, Cip Baetu, Jordi Berbel, David Bridson, Jake Bruce, Gavin Buttimore, Sarah Chakera, Bilva Chandra, Paul Collins, Alex Cullum, Bogdan Damoc, Vibha Dasagi, Maxime Gazeau, Charles Gbadamosi, Woohyun Han, Ed Hirst, Ashyana Kachra, Lucie Kerley, Kristian Kjems, Eva Knoepfel, Vika Koriakin, Jessica Lo, Cong Lu, Zeb Mehring, Alex Moufarek, Henna Nandwani, Valeria Oliveira, Fabio Pardo, Jane Park, Andrew Pierson, Ben Poole, Helen Ran, Tim Salimans, Manuel Sanchez, Igor Saprykin, Amy Shen, Sailesh Sidhwani, Duncan Smith, Joe Stanton, Hamish Tomlinson, Dimple Vijaykumar, Luyu Wang, Piers Wingfield, Nat Wong, Keyang Xu, Christopher Yew, Nick Young, Vadim Zubov, Douglas Eck, Dumitru Erhan, Koray Kavukcuoglu, Demis Hassabis, Zoubin Gharamani, Raia Hadsell, Aäron van den Oord, Inbar Mosseri, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 3: A new frontier for world models, 2025.

[158] Fei-Fei Li. Marble: A multimodal world model, 2025. URL https://www.worldlabs.ai/blog/marble-world-model.

[159] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE signal processing magazine*, 34(6):26–38, 2017.

[160] Frank Dellaert and Michael Kaess. Square root sam: Simultaneous localization and mapping via square root information smoothing. *The International Journal of Robotics Research*, 25(12):1181–1203, 2006.

[161] Randall Smith, Matthew Self, and Peter Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. 1990.

[162] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. 2001.

[163] Sebastian Thrun et al. Robotic mapping: A survey. *Exploring artificial intelligence in the new millennium*, 1(1-35):1, 2002.

[164] Gary Bishop, Greg Welch, et al. An introduction to the kalman filter. *Proc of SIGGRAPH, Course*, 8 (27599-23175):41, 2001.

[165] S. M. LaValle. *Planning Algorithms*. 2006.

[166] MWM Gamini Dissanayake, Paul Newman, Steve Clark, Hugh F Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (slam) problem. *IEEE Transactions on robotics and automation*, 17(3):229–241, 2001.

[167] Tim Bailey, Juan Nieto, Jose Guivant, Michael Stevens, and Eduardo Nebot. Consistency of the ekf-slam algorithm. In *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3562–3568, 2006.

[168] Giorgio Grisetti, Rainer Kümmerle, Cyrill Stachniss, and Wolfram Burgard. A tutorial on graph-based slam. *IEEE Intelligent Transportation Systems Magazine*, 2(4):31–43, 2010.

[169] Dmitry Chetverikov, Dmitry Svirko, Dmitry Stepanov, and Pavel Krsek. The trimmed iterative closest point algorithm. In *2002 International Conference on Pattern Recognition*, volume 3, pages 545–548, 2002.

[170] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Transactions on Graphics (ToG)*, 32(6):1–11, 2013.

[171] Zhenlong Li, Huan Ning, Song Gao, Krzysztof Janowicz, Wenwen Li, Samantha T Arundel, Chaowei Yang, Budhendra Bhaduri, Shaowen Wang, A-Xing Zhu, et al. Giscience in the era of artificial intelligence: A research agenda towards autonomous gis. *Annals of GIS*, 31(4):501–536, 2025.

[172] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 28, 2015.

[173] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 3354–3361, 2012.

[174] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[175] Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449, 2017.

[176] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *European conference on computer vision*, pages 685–702, 2020.

[177] Arun CS Kumar, Suchendra M Bhandarkar, and Mukta Prasad. Depthnet: A recurrent neural network architecture for monocular depth prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 283–291, 2018.

[178] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab. Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6243–6252, 2017.

[179] Chengzhou Tang and Ping Tan. Ba-net: Dense bundle adjustment network. *arXiv preprint arXiv:1806.04807*, 2018.

[180] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3828–3838, 2019.

[181] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 2774–2781, 2023.

[182] Alex Kendall, Hayk Martirosyan, Saumitro Dasgupta, Peter Henry, Ryan Kennedy, Abraham Bachrach, and Adam Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 66–75, 2017.

[183] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[184] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems (NeurIPS)*, 33:6840–6851, 2020.

[185] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022.

[186] Lingdong Kong, Wesley Yang, Jianbiao Mei, Youquan Liu, Ao Liang, Dekai Zhu, Dongyue Lu, Wei Yin, Xiaotao Hu, Mingkai Jia, et al. 3d and 4d world modeling: A survey. *arXiv preprint arXiv:2509.07996*, 2025.

[187] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 10318–10327, 2021.

[188] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6498–6508, 2021.

[189] Weikang Bian, Zhaoyang Huang, Xiaoyu Shi, Yijin Li, Fu-Yun Wang, and Hongsheng Li. Gs-dit: Advancing video generation with pseudo 4d gaussian fields through efficient dense 3d point tracking, 2025. URL https://arxiv.org/abs/2501.02690.

[190] Tianqi Liu, Zihao Huang, Zhaoxi Chen, Guangcong Wang, Shoukang Hu, Liao Shen, Huiqiang Sun, Zhiguo Cao, Wei Li, and Ziwei Liu. Free4d: Tuning-free 4d scene generation with spatial-temporal consistency. *arXiv preprint arXiv:2503.20785*, 2025.

[191] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20310–20320, 2024.

[192] Haoyu Zhen, Qiao Sun, Hongxin Zhang, Junyan Li, Siyuan Zhou, Yilun Du, and Chuang Gan. Tesseract: learning 4d embodied world models. *arXiv preprint arXiv:2504.20995*, 2025.

[193] Yezhen Cong, Samar Khanna, Chenlin Meng, Patrick Liu, Erik Rozi, Yutong He, Marshall Burke, David Lobell, and Stefano Ermon. Satmae: Pre-training transformers for temporal and multi-spectral satellite imagery. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35: 197–211, 2022.

[194] Xian Sun, Peijin Wang, Wanxuan Lu, Zicong Zhu, Xiaonan Lu, Qibin He, Junxi Li, Xuee Rong, Zhujun Yang, Hao Chang, et al. Ringmo: A remote sensing foundation model with masked image modeling. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–22, 2022.

[195] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual KITTI 2, 2020. URL https://arxiv.org/abs/2001.10773.

[196] Nicholas Pfaff, Evelyn Fu, Jeremy Binagia, Phillip Isola, and Russ Tedrake. Scalable real2sim: Physics-aware asset generation via robotic pick-and-place setups. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6296–6303, 2025.

[197] Mutian Xu, Chongjie Ye, Haolin Liu, Yushuang Wu, Jiahao Chang, and Xiaoguang Han. Stable-sim2real: Exploring simulation of real-captured 3d data with two-stage depth diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2609–2619, 2025.

[198] Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

[199] Yufei Wang, Zhou Xian, Feng Chen, Tsun-Hsuan Wang, Yian Wang, Katerina Fragkiadaki, Zackory Erickson, David Held, and Chuang Gan. Robogen: Towards unleashing infinite data for automated robot learning via generative simulation. *arXiv preprint arXiv:2311.01455*, 2023.

[200] Delong Chen, Mustafa Shukor, Theo Moutakanni, Willy Chung, Jade Yu, Tejaswi Kasarla, Yejin Bang, Allen Bolourchi, Yann LeCun, and Pascale Fung. Vl-jepa: Joint embedding predictive architecture for vision-language. *arXiv preprint arXiv:2512.10942*, 2025.

[201] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

[202] Yuncong Yang, Jiageng Liu, Zheyuan Zhang, Siyuan Zhou, Reuben Tan, Jianwei Yang, Yilun Du, and Chuang Gan. Mindjourney: Test-time scaling with world models for spatial reasoning. *Advances in Neural Information Processing Systems*, 38:109855–109885, 2026.

[203] Samuel Garcin, Thomas Walker, Steven McDonagh, Tim Pearce, Hakan Bilen, Tianyu He, Kaixin Wang, and Jiang Bian. Beyond pixel histories: World models with persistent 3d state. *arXiv preprint arXiv:2603.03482*, 2026.

[204] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:251–266, 2021.

[205] Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M. Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X. Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. URL https://arxiv.org/abs/2109.08238.

[206] Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, et al. Emu3. 5: Native multimodal models are world learners. *arXiv preprint arXiv:2510.26583*, 2025.

[207] Jie Feng, Jinwei Zeng, Qingyue Long, Hongyi Chen, Jie Zhao, Yanxin Xi, Zhilun Zhou, Yuan Yuan, Shengyuan Wang, Qingbin Zeng, et al. A survey of large language model-powered spatial intelligence across scales: Advances in embodied agents, smart cities, and earth science. *arXiv preprint arXiv:2504.09848*, 2025.

[208] Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, et al. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. *arXiv preprint arXiv:2502.09560*, 2025.

[209] Max J Egenhofer. Reasoning about binary topological relations. In *Symposium on Spatial Databases*, pages 141–160, 1991.

[210] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11): 832–843, 1983.

[211] Frank Dellaert and Michael Kaess. Factor graphs for robot perception. *Foundations and Trends® in Robotics*, 6(1-2):1–139, 2017.

[212] Tong Wu, Guandao Yang, Zhibing Li, Kai Zhang, Ziwei Liu, Leonidas Guibas, Dahua Lin, and Gordon Wetzstein. Gpt-4v (ision) is a human-aligned evaluator for text-to-3d generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22227–22238, 2024.

[213] Barbara Landau and Ray Jackendoff. Whence and whither in spatial language and spatial cognition? *Behavioral and brain sciences*, 16(2):255–265, 1993.

[214] Daniel R. Montello. Scale and multiple psychologies of space. In *Spatial Information Theory: A Theoretical Basis for GIS (COSIT)*, pages 312–321. 1993.

[215] Stephen C. Levinson. *Space in Language and Cognition: Explorations in Cognitive Diversity*. 2003.

[216] David H. Uttal, Nathaniel G. Meadow, Elizabeth Tipton, Linda L. Hand, Alison R. Alden, Christopher Warren, and Nora S. Newcombe. The malleability of spatial skills: a meta-analysis of training studies. *Psychological Bulletin*, 139(2):352–402, 2013.

[217] Kimberly L Stachenfeld, Matthew M Botvinick, and Samuel J Gershman. The hippocampus as a predictive map. *Nature neuroscience*, 20(11):1643–1653, 2017.

[218] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles Qi, Yin Zhou, Zoey Yang, Aurelien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9710–9719, 2021.

[219] Khan Muhammad, Tanveer Hussain, Hayat Ullah, Javier Del Ser, Mahdi Rezaei, Neeraj Kumar, Mohammad Hijji, Paolo Bellavista, and Victor Hugo C De Albuquerque. Vision-based semantic segmentation in scene understanding for autonomous driving: Recent achievements, challenges, and outlooks. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):22694–22715, 2022.

[220] Lele Chen, Chen Cao, Fernando De la Torre, Jason Saragih, Chenliang Xu, and Yaser Sheikh. High-fidelity face tracking for ar/vr via deep lighting adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13059–13069, 2021.

[221] Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip Davidson, Olaf Kähler, Mingsong Dou, Vladimir Tankovich, and Shahram Izadi. Holoportation: Virtual 3d teleportation in real-time. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST)*, pages 741–754, 2016.

[222] Jinxing Hu, Zhihan Lv, Diping Yuan, Bing He, Wenjiang Chen, Xiongfei Ye, Donghao Li, and Ge Yang. A spatiotemporal intelligent framework and experimental platform for urban digital twins. *Virtual Real. Intell. Hardw.*, 5(3):213–231, 2023.

[223] Diogo Carbonera Luvizon, Bogdan Tomoyuki Nassu, and Rodrigo Minetto. A video-based system for vehicle speed measurement in urban roadways. *IEEE Transactions on Intelligent Transportation Systems*, 18(6):1393–1404, 2017.

[224] Yao Mu, Tianxing Chen, Shijia Peng, Zanxin Chen, Zeyu Gao, Yude Zou, Lunkai Lin, Zhiqiang Xie, and Ping Luo. Robotwin: Dual-arm robot benchmark with generative digital twins (early version). In *European Conference on Computer Vision*, pages 264–273, 2024.

[225] Gauthier Picard, Clément Caron, Jean-Loup Farges, Jonathan Guerra, Cédric Pralet, and Stéphanie Roussel. Autonomous agents and multiagent systems challenges in earth observation satellite constellations. *arXiv preprint*, 2021.

[226] John B. Carroll. *Human Cognitive Abilities: A Survey of Factor-Analytic Studies*. 1993.

[227] Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10632–10643, 2025.

[228] Leonard Talmy. How language structures space. In Herbert L. Pick and Linda P. Acredolo, editors, *Spatial Orientation: Theory, Research, and Application*, pages 225–282. 1983.

[229] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14455–14465, 2024.

[230] Tongtong Feng, Xin Wang, Yu-Gang Jiang, and Wenwu Zhu. Embodied ai: From llms to world models. *arXiv preprint arXiv:2509.20021*, 2025.

[231] Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, et al. Theory of space: Can foundation models construct spatial beliefs through active exploration? In *The Fourteenth International Conference on Learning Representations*, 2026.

[232] Edward T. Hall. *The Hidden Dimension*. 1966.

[233] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. 3rd edition, 2011. Broad intro to image processing fundamentals widely used in vision curricula.

[234] Richard Szeliski. *Computer Vision: Algorithms and Applications*. 2nd edition, 2022.

[235] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

[236] Andrej Karpathy et al. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[237] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 11976–11986, 2022.

[238] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4489–4497, 2015.

[239] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[240] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021.

[241] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3202–3211, 2022.

[242] Xiangtai Li, Henghui Ding, Haobo Yuan, Wenwei Zhang, Jiangmiao Pang, Guangliang Cheng, Kai Chen, Ziwei Liu, and Chen Change Loy. Transformer-based visual segmentation: A survey. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2024.

[243] Alexander Kirillov et al. Panoptic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019.

[244] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016.

[245] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. Cris: Clip-driven referring image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11686–11695, 2022.

[246] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Urvos: Unified referring video object segmentation network with a large-scale benchmark. In *European conference on computer vision*, pages 208–223, 2020.

[247] Sergio Foix, Guillem Alenya, and Carme Torras. Lock-in time-of-flight (tof) cameras: A survey. *IEEE Sensors Journal*, 2011.

[248] Jonathan T. Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[249] Robert J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 1980.

[250] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2015.

[251] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986.

[252] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2015.

[253] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 1981.

[254] Thomas Brox, Andres Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. *European Conference on Computer Vision*, 2004.

[255] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020.

[256] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016.

[257] Chris Harris and Mike Stephens. A combined corner and edge detector. *Alvey Vision Conference*, 1988.

[258] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2014.

[259] Ce Zheng, Wenjun Wu, Chen Chen, Tao Yang, Sijie Zhu, Chao Chen, Runze Yang, and Zhenzhong Yu. Deep learning-based human pose estimation: A survey. *arXiv preprint arXiv:2204.07370*, 2022.

[260] Intel Corporation. Intel realsense sdk (2020) — developer documentation, 2020. URL https://www.intelrealsense.com/sdk-2/.

[261] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 29, 2016.

[262] Camille Couprie, Clément Farabet, Laurent Najman, and Yann LeCun. Indoor semantic segmentation using depth information. *arXiv preprint arXiv:1301.3572*, 2013.

[263] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7364–7373, 2019.

[264] Guilherme N DeSouza and Avinash C Kak. Vision for mobile robot navigation: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 24(2):237–267, 2002.

[265] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In *Proceedings 2000 ICRA. Millennium conference. IEEE international conference on robotics and automation. Symposia proceedings (Cat. No. 00CH37065)*, volume 1, pages 348–353, 2000.

[266] Ashutosh Saxena, Justin Driemeyer, Justin Kearns, and Andrew Ng. Robotic grasping of novel objects. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 19, 2006.

[267] Haoyu Zhao, Linghao Zhuang, Xingyue Zhao, Cheng Zeng, Haoran Xu, Yuming Jiang, Jun Cen, Kexiang Wang, Jiayan Guo, Siteng Huang, et al. Towards affordance-aware robotic dexterous grasping with human-like priors. *arXiv preprint arXiv:2508.08896*, 2025.

[268] Özgün Çiçek, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention*, pages 424–432, 2016.

[269] Behnam Behroozpour, Phillip AM Sandborn, Ming C Wu, and Bernhard E Boser. Lidar system architectures and circuits. *IEEE Communications Magazine*, 55(10):135–142, 2017.

[270] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[271] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 5855–5864, 2021.

[272] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 19697–19705, 2023.

[273] Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, Peter Hedman, Ricardo Martin-Brualla, and Jonathan T. Barron. MultiNeRF: A Code Release for Mip-NeRF 360, Ref-NeRF, and RawNeRF, 2022. URL https://github.com/google-research/multinerf.

[274] Thomas Porter and Tom Duff. Compositing digital images. In *Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, pages 253–259, 1984.

[275] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7708–7717, 2019.

[276] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:6087–6101, 2021.

[277] Delio Vicini, Sébastien Speierer, and Wenzel Jakob. Differentiable signed distance function rendering. *ACM Transactions on Graphics (ToG)*, 41(4):1–18, 2022.

[278] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16123–16133, 2022.

[279] John McCormac, Ankur Handa, Andrew Davison, and Stefan Leutenegger. Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In *2017 IEEE International Conference on Robotics and automation (ICRA)*, pages 4628–4635, 2017.

[280] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 922–928, 2015.

[281] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Proceedings of the IEEE international conference on computer vision*, pages 2088–2096, 2017.

[282] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021.

[283] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in neural information processing systems*, 35:23192–23204, 2022.

[284] Rana Hanocka, Amir Hertz, Noa Fish, Raja Giryes, Shachar Fleishman, and Daniel Cohen-Or. Meshcnn: a network with an edge. *ACM Transactions on Graphics (ToG)*, 38(4):1–12, 2019.

[285] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 52–67, 2018.

[286] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. A papier-mâché approach to learning 3d surface generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 216–224, 2018.

[287] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In *Advances in neural information processing systems (NeurIPS)*, 2019.

[288] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[289] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 5501–5510, 2022.

[290] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 19447–19456, 2024.

[291] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20654–20664, 2024.

[292] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. NeRFactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics*, 40(6):1–18, 2021.

[293] Yiran Wang, Min Shi, Jiaqi Li, Zihao Huang, Zhiguo Cao, Jianming Zhang, Ke Xian, and Guosheng Lin. Neural video depth stabilizer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9466–9476, 2023.

[294] Wenbo Hu, Xiangjun Gao, Xiaoyu Li, Sijie Zhao, Xiaodong Cun, Yong Zhang, Long Quan, and Ying Shan. Depthcrafter: Generating consistent long depth sequences for open-world videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2005–2015, 2025.

[295] Sili Chen, Hengkai Guo, Shengnan Zhu, Feihu Zhang, Zilong Huang, Jiashi Feng, and Bingyi Kang. Video depth anything: Consistent depth estimation for super-long videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 22831–22840, 2025.

[296] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 10486–10496, 2025.

[297] Wei Cao, Chang Luo, Biao Zhang, Matthias Nießner, and Jiapeng Tang. Motion2vecsets: 4d latent vector set diffusion for non-rigid shape reconstruction and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 20496–20506, 2024.

[298] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006.

[299] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 5865–5874, 2021.

[300] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5712–5721, 2021.

[301] Bowen Zhang, Sicheng Xu, Chuxin Wang, Jiaolong Yang, Feng Zhao, Dong Chen, and Baining Guo. Gaussian variation field diffusion for high-fidelity video-to-4d synthesis. *arXiv preprint arXiv:2507.23785*, 2025.

[302] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[303] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[304] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67, 2020.

[305] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8), 2019.

[306] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*, 2022.

[307] Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.

[308] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *NAACL-HLT*, pages 4129–4138, 2019.

[309] Belinda Z. Li, Maxwell Nye, and Jacob Andreas. Implicit representations of meaning in neural language models. *arXiv preprint arXiv:2106.00737*, 2021.

[310] Roma Patel and Ellie Pavlick. Mapping language models to grounded conceptual spaces. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=gJcEM8sxHK.

[311] Mostafa Abdou, Artur Kulmizev, Daniel Hershcovich, Stella Frank, Ellie Pavlick, and Anders Søgaard. Can language models encode perceptual structure without grounding? a case study in color. *arXiv preprint arXiv:2109.06129*, 2021.

[312] Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2210.13382.

[313] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

[314] Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 2016.

[315] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viégas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2668–2677, 2018.

[316] Amirata Ghorbani, James Wexler, James Y. Zou, and Been Kim. Towards automatic concept-based explanations. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[317] Shubham Toshniwal, Sam Wiseman, Karen Livescu, and Kevin Gimpel. Learning chess blindfolded: Evaluating language models on state tracking. *arXiv preprint arXiv:2102.13249*, 2021.

[318] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

[319] Thomas McGrath, Andrei Kapishnikov, Nenad Tomašev, Adam Pearce, Demis Hassabis, Been Kim, Ulrich Paquet, and Vladimir Kramnik. Acquisition of chess knowledge in alphazero. *arXiv preprint arXiv:2111.09259*, 2021.

[320] Ewan A Macpherson and John C Middlebrooks. Listener weighting of cues for lateral angle: The duplex theory of sound localization revisited. *The Journal of the Acoustical Society of America*, 111(5): 2219–2236, 2002.

[321] John C Middlebrooks. Sound localization. *Handbook of clinical neurology*, 129:99–116, 2015.

[322] William G Gardner. *3-D audio using loudspeakers*, volume 444. 1998.

[323] Heinrich Kuttruff. *Room acoustics*. 2016.

[324] Jens Blauert. *Spatial hearing: the psychophysics of human sound localization*. 1997.

[325] Frederic L Wightman and Doris J Kistler. Headphone simulation of free-field listening. i: Stimulus synthesis. *The Journal of the Acoustical Society of America*, 85(2):858–867, 1989.

[326] Franz Zotter and Matthias Frank. *Ambisonics: A practical 3D audio theory for recording, studio production, sound reinforcement, and virtual reality*. 2019.

[327] Goksenin Yuksel, Marcel van Gerven, and Kiki van der Heijden. General-purpose audio representation learning for real-world sound scenes. *arXiv preprint arXiv:2506.00934*, 2025.

[328] Ruohan Gao, Changan Chen, Ziad Al-Halah, Carl Schissler, and Kristen Grauman. Visualechoes: Spatial image representation learning through echolocation. In *European Conference on Computer Vision*, pages 658–676, 2020.

[329] Pedro Morgado, Nuno Nvasconcelos, Timothy Langlois, and Oliver Wang. Self-supervised generation of spatial audio for 360 video. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

[330] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *European Conference on Computer Vision*, pages 208–224, 2020.

[331] Pedro Morgado, Yi Li, and Nuno Nvasconcelos. Learning representations from audio-visual spatial alignment. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 33:4733–4744, 2020.

[332] Zhisheng Zheng, Puyuan Peng, Ziyang Ma, Xie Chen, Eunsol Choi, and David Harwath. Bat: Learning to reason about spatial sounds with large language models. *arXiv preprint arXiv:2402.01591*, 2024.

[333] Subrata Biswas, Mohammad Nur Hossain Khan, and Bashima Islam. Owl: Geometry-aware spatial reasoning for audio large language models. *arXiv preprint arXiv:2509.26140*, 2025.

[334] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics (ToG)*, 36(4):1, 2017.

[335] Peter Florence, Lucas Manuelli, and Russ Tedrake. Dense object nets: Learning dense visual object descriptors by and for robotic manipulation. *Conference on Robot Learning*, 2018.

[336] Jeffrey Mahler, Jacky Liang, Shahrouz Niyaz, Michael Laskey, Ross Doan, Xinyu Liu, Jose Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In *Robotics: Science and Systems (RSS)*, 2017.

[337] Kaichun Mo, Leonidas J Guibas, Mustafa Mukadam, Abhinav Gupta, and Shubham Tulsiani. Where2act: From pixels to actions for articulated 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6813–6823, 2021.

[338] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):248:1–248:16, 2015.

[339] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. H. Engel, R. M. Salas-Moreno, R. Sandilands, M. S. Dill, D. Pang, B. Williams, P. Vajda, R. M. Turner, and R. Newcombe. The replica dataset: A digital replica of indoor spaces. In *Proceeding of CVPR*, pages 4165–4176, 2019.

[340] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Nießner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from RGB-D data in indoor environments. In *Proceedings of the International Conference on 3D Vision (3DV)*, pages 667–676, 2017.

[341] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 3d scene graph: A structure for unified semantics, 3d space, and camera. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5664–5673, 2019.

[342] Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3961–3970, 2020.

[343] Shun-Cheng Wu, Johanna Wald, Keisuke Tateno, Nassir Navab, and Federico Tombari. Scenegraphfusion: Incremental 3d scene graph prediction from rgb-d sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7515–7525, 2021.

[344] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 4904–4916, 2021.

[345] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *European conference on computer vision*, pages 104–120, 2020.

[346] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European conference on computer vision*, pages 121–137, 2020.

[347] Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B Tenenbaum, and Jiajun Wu. The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. *arXiv preprint arXiv:1904.12584*, 2019.

[348] Joy Hsu, Jiayuan Mao, Josh Tenenbaum, and Jiajun Wu. What's left? concept grounding with logic-enhanced foundation models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:38798–38814, 2023.

[349] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18123–18133, 2022.

[350] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

[351] Yiwu Zhong, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16793–16803, 2022.

[352] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021.

[353] Yan Zeng, Xinsong Zhang, and Hang Li. Multi-grained vision language pre-training: Aligning texts with visual concepts. *arXiv preprint arXiv:2111.08276*, 2021.

[354] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:9694–9705, 2021.

[355] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35: 32897–32912, 2022.

[356] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for vision and vision-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19175–19186, 2023.

[357] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021.

[358] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 12888–12900, 2022.

[359] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022.

[360] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1780–1790, 2021.

[361] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10965–10975, 2022.

[362] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55, 2024.

[363] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18082–18091, 2022.

[364] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18134–18144, 2022.

[365] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1179–1189, 2023.

[366] Minghua Liu, Ruoxi Shi, Kaiming Kuang, Yinhao Zhu, Xuanlin Li, Shizhong Han, Hong Cai, Fatih Porikli, and Hao Su. Openshape: Scaling up 3d shape representation towards open-world understanding. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36: 44860–44879, 2023.

[367] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8552–8562, 2022.

[368] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22157–22167, 2023.

[369] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19729–19739, 2023.

[370] Nur Muhammad Mahi Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*, 2022.

[371] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20051–20060, 2024.

[372] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[373] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European conference on computer vision*, pages 202–221, 2020.

[374] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *European conference on computer vision*, pages 422–440, 2020.

[375] Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2911–2921, 2023.

[376] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *European Conference on Computer Vision*, pages 417–433, 2022.

[377] Joy Hsu, Jiayuan Mao, and Jiajun Wu. Ns3d: Neuro-symbolic grounding of 3d objects and relations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2614–2623, 2023.

[378] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Sheng Wang, Zhen Li, and Shuguang Cui. Instancerefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1791–1800, 2021.

[379] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 33:1970–1981, 2020.

[380] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J Guibas. Vector neurons: A general framework for so (3)-equivariant networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12200–12209, 2021.

[381] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[382] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[383] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. Audioclip: Extending clip to image, text and audio. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 976–980, 2022.

[384] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello. Wav2clip: Learning robust audio representations from clip. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4563–4567, 2022.

[385] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*, 2021.

[386] Khaled Koutini, Jan Schlüter, Hamid Eghbal-Zadeh, and Gerhard Widmer. Efficient training of audio transformers with patchout. *arXiv preprint arXiv:2110.05069*, 2021.

[387] Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 646–650, 2022.

[388] Yuan Gong, Cheng-I Lai, Yu-An Chung, and James Glass. Ssast: Self-supervised audio spectrogram transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10699–10709, 2022.

[389] Alan Baade, Puyuan Peng, and David Harwath. Mae-ast: Masked autoencoding audio spectrogram transformer. *arXiv preprint arXiv:2203.16691*, 2022.

[390] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:28708–28720, 2022.

[391] Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei. Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*, 2022.

[392] Thomas Whelan, Stefan Leutenegger, Renato F Salas-Moreno, Ben Glocker, and Andrew J Davison. Elasticfusion: Dense slam without a pose graph. In *Robotics: science and systems*, 2015.

[393] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017.

[394] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15598–15607, 2021.

[395] William Shen, Ge Yang, Alan Yu, Jansen Wong, Leslie Pack Kaelbling, and Phillip Isola. Distilled feature fields enable few-shot language-guided manipulation. *arXiv preprint arXiv:2308.07931*, 2023.

[396] Jianglong Ye, Naiyan Wang, and Xiaolong Wang. Featurenerf: Learning generalizable nerfs by distilling foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8962–8973, 2023.

[397] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15180–15190, 2023.

[398] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 609–617, 2017.

[399] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 631–648, 2018.

[400] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proceedings of the European conference on computer vision (ECCV)*, pages 570–586, 2018.

[401] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *arXiv preprint arXiv:1905.05055*, 2019.

[402] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.

[403] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[404] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[405] Ross Girshick. Fast r-cnn. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2015.

[406] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, and Ping Luo. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[407] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.

[408] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[409] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[410] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, Yonghye Kwon, Jiacong Fang, Kalen Michael, Diego Montes, Jebastin Nadar, Piotr Skalski, et al. ultralytics/yolov5: v6. 1-tensorrt, tensorflow edge tpu and openvino export and inference. *Zenodo*, 2022.

[411] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. *ECCV*, pages 21–37, 2016.

[412] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[413] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Yolov8: A new state-of-the-art in object detection, 2023.

[414] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, et al. Yolov10: Real-time end-to-end object detection. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37: 107984–108011, 2024.

[415] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *Proc. ICLR*, 2021.

[416] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3651–3660, 2021.

[417] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

[418] Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16965–16974, 2024.

[419] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.

[420] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Open-vocabulary object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[421] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *European conference on computer vision*, pages 350–368, 2022.

[422] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In *European conference on computer vision*, pages 728–755, 2022.

[423] Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16901–16911, 2024.

[424] Chaoyang Zhu and Long Chen. A survey on open-vocabulary detection and segmentation: Past, present, and future. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):8954–8975, 2024.

[425] Jianzong Wu, Xiangtai Li, Shilin Xu, Haobo Yuan, Henghui Ding, Yibo Yang, Xia Li, Jiangning Zhang, Yunhai Tong, Xudong Jiang, et al. Towards open vocabulary learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7):5092–5113, 2024.

[426] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[427] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[428] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[429] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021.

[430] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[431] Ishan Misra, Rohit Girdhar, and Armand Joulin. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[432] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[433] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, 2022.

[434] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *Proc. ECCV*, volume 13669, pages 1–18, 2022.

[435] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

[436] Garrick Brazil, Abhinav Kumar, Julian Straub, Nikhila Ravi, Justin Johnson, and Georgia Gkioxari. Omni3d: A large benchmark and model for 3d object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 13154–13164, 2023.

[437] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html, 2007.

[438] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[439] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015.

[440] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755, 2014.

[441] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019.

[442] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981, 2020.

[443] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019.

[444] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 2010.

[445] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 11618–11628, 2020.

[446] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2443–2451, 2020.

[447] Shuran Song, Samuel P Lichtenberg, and Jianxiong Xiao. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576, 2015.

[448] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the CVPR*, pages 5828–5839, 2017.

[449] Xinzhu Ma et al. Delving into localization errors for monocular 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[450] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[451] Rolf Adams and Leanne Bischof. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641–647, 1994.

[452] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, 1967.

[453] Luc Vincent and Pierre Soille. Watersheds in digital spaces: An efficient algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991.

[454] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[455] Yuri Boykov and Marie-Pierre Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 105–112, 2001.

[456] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[457] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pages 282–289, 2001.

[458] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder–decoder architecture for image segmentation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[459] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[460] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *International Conference on Learning Representations (ICLR)*, 2015.

[461] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[462] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[463] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder–decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[464] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[465] Kai Chen, Jiangmiao Pang, Jiaqi Wang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[466] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[467] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[468] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[469] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 10012–10022, 2021.

[470] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[471] Huiyu Wang, Yukun Zhu, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Max-deeplab: End-to-end panoptic segmentation with mask transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[472] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[473] Wenwei Zhang, Jiangmiao Pang, Kai Chen, and Chen Change Loy. K-net: Towards unified image segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[474] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[475] Yuxin Fang, Shusheng Yang, Xinggang Wang, Yu Li, Chen Fang, Ying Shan, Bin Feng, and Wenyu Liu. Instances as queries. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[476] Zhiqi Li, Wenhai Wang, Enze Xie, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, Ping Luo, and Tong Lu. Panoptic segformer: Delving deeper into panoptic segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[477] Yuqing Wang, Zhaoliang Xu, Xinlong Wang, Chunhua Shen, Baoshan Cheng, Hao Shen, and Huaxia Xia. Vistr: End-to-end video instance segmentation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[478] Xiangtai Li, Wenwei Zhang, Jiangmiao Pang, Kai Chen, Guangliang Cheng, Yunhai Tong, and Chen Change Loy. Video k-net: A simple, strong, and unified baseline for video segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[479] Dahun Kim, Jun Xie, Huiyu Wang, Siyuan Qiao, Qihang Yu, Hong-Seok Kim, Hartwig Adam, In So Kweon, and Liang-Chieh Chen. Tubeformer-deeplab: Video mask transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[480] Xueyan Zou, Zi-Yi Dou, Jianwei Yang, Zhe Gan, Linjie Li, Chunyuan Li, Xiyang Dai, Harkirat Behl, Jianfeng Wang, Lu Yuan, Nanyun Peng, Lijuan Wang, Yong Jae Lee, and Jianfeng Gao. Generalized decoding for pixel, image, and language. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[481] Xiangtai Li, Haobo Yuan, Wei Li, Henghui Ding, Size Wu, Wenwei Zhang, Yining Li, Kai Chen, and Chen Change Loy. Omg-seg: Is one model good enough for all segmentation? In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2024.

[482] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

[483] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics*, 38(5), 2019.

[484] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[485] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[486] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[487] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2019.

[488] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024.

[489] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[490] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d: Mask transformer for 3d semantic instance segmentation. In *IEEE International Conference on Robotics and Automation*, 2023.

[491] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. In *Advances in Neural Information Processing Systems*, 2023.

[492] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Alberto Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 633–641, 2017.

[493] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1209–1218, 2018.

[494] Girish Varma, Anbumani Subramanian, Anoop M. Namboodiri, Manmohan Chandraker, and C. V. Jawahar. IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019*, pages 1743–1751, 2019.

[495] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *2019 IEEE/CVF International Conference on Computer Vision, Proceedings of theInternational Conference on Computer Vision (ICCV) 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 7373–7382, 2019.

[496] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. ACDC: the adverse conditions dataset with correspondences for semantic driving scene understanding. In *2021 IEEE/CVF International Conference on Computer Vision, Proceedings of theInternational Conference on Computer Vision (ICCV) 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 10745–10755, 2021.

[497] Ju He, Shuo Yang, Shaokang Yang, Adam Kortylewski, Xiaoding Yuan, Jieneng Chen, Shuai Liu, Cheng Yang, Qihang Yu, and Alan L. Yuille. Partimagenet: A large, high-quality dataset of parts. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part VIII*, volume 13668 of *Lecture Notes in Computer Science*, pages 128–145, 2022.

[498] Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1972–1979, 2009.

[499] David R. Martin, Charless C. Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the Eighth International Conference On Computer Vision (ICCV-01), Vancouver, British Columbia, Canada, July 7-14, 2001 - Volume 2*, pages 416–425, 2001.

[500] Alessandro Prest, Christian Leistner, Javier Civera, Cordelia Schmid, and Vittorio Ferrari. Learning object class detectors from weakly annotated video. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 3282–3289, 2012.

[501] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, 2016.

[502] Gabriel J. Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognit. Lett.*, 30(2):88–97, 2009.

[503] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4990–4999, 2017.

[504] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760, 2012.

[505] Jianxiong Xiao, Andrew Owens, and Antonio Torralba. Sun3d: A database of big spaces reconstructed using sfm and object labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1625–1632, 2013.

[506] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jürgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *2019 IEEE/CVF*

International Conference on Computer Vision, Proceedings of theInternational Conference on Computer Vision (ICCV) 2019, Seoul, Korea (South), October 27 - November 2, 2019, pages 9296–9306, 2019.

[507] Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D Semantic Parsing of Large-Scale Indoor Spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1534–1543, 2016.

[508] Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho. Neural sign language translation based on human keypoint estimation. *Applied sciences*, 9(13):2683, 2019.

[509] Peixuan Li, Huaici Zhao, Pengfei Liu, and Feidao Cao. Rtm3d: Real-time monocular 3d detection from object keypoints for autonomous driving. In *European Conference on Computer Vision*, pages 644–660, 2020.

[510] Jessy Lauer, Mu Zhou, Shaokai Ye, William Menegas, Steffen Schneider, Tanmay Nath, Mohammed Mostafizur Rahman, Valentina Di Santo, Daniel Soberanes, Guoping Feng, et al. Multi-animal pose estimation, identification and tracking with deeplabcut. *Nature Methods*, 19(4):496–504, 2022.

[511] Heqin Zhu, Qingsong Yao, Li Xiao, and S Kevin Zhou. You only learn once: Universal anatomical landmark detection. In *International conference on medical image computing and computer-assisted intervention*, pages 85–95, 2021.

[512] Hans Peter Moravec. *Obstacle avoidance and navigation in the real world by a seeing robot rover*. 1980.

[513] Jianbo Shi et al. Good features to track. In *1994 Proceedings of IEEE conference on computer vision and pattern recognition*, pages 593–600, 1994.

[514] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*, pages 404–417, 2006.

[515] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, pages 430–443, 2006.

[516] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *European conference on computer vision*, pages 778–792, 2010.

[517] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571, 2011.

[518] Timothy F. Cootes, Christopher J. Taylor, David H. Cooper, and Jim Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[519] Timothy F Cootes, Gareth J Edwards, and Christopher J Taylor. Active appearance models. In *European conference on computer vision*, pages 484–498, 1998.

[520] Jason M Saragih, Simon Lucey, and Jeffrey F Cohn. Deformable model fitting by regularized landmark mean-shift. *International journal of computer vision*, 2011.

[521] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 532–539, 2013.

[522] Jonathan Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

[523] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christoph Bregler. Efficient object localization using convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 648–656, 2015.

[524] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.

[525] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, pages 3349–3364, 2020.

[526] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[527] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 43(1): 172–186, 2021.

[528] Debapriya Maji, Soyeb Nagori, Manu Mathew, and Deepak Poddar. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2637–2646, 2022.

[529] Tao Jiang, Peng Lu, Li Zhang, Ningsheng Ma, Rui Han, Chengqi Lyu, Yining Li, and Kai Chen. Rtmpose: Real-time multi-person pose estimation based on mmpose. *arXiv preprint arXiv:2303.07399*, 2023.

[530] Peng Lu, Tao Jiang, Yining Li, Xiangtai Li, Kai Chen, and Wenming Yang. Rtmo: Towards high-performance one-stage real-time multi-person pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1491–1500, 2024.

[531] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[532] Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Su Zhaoen, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. Sapiens: Foundation for human vision models. In *European Conference on Computer Vision*, pages 206–228, 2024.

[533] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[534] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.

[535] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[536] Jia Gong, Lin Geng Foo, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. Diffpose: Toward more reliable 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13041–13051, 2023.

[537] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[538] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable Triangulation of Human Pose. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7718–7727, 2019.

[539] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops (CVPRW)*, pages 224–236, 2018.

[540] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in neural information processing systems (NeurIPS)*, 33:14254–14265, 2020.

[541] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4938–4947, 2020.

[542] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[543] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *IEEE International Conference on Computer Vision Workshops*, 2009.

[544] Jiaxin Li and Gim Hee Lee. Usip: Unsupervised stable interest point detection from 3d point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[545] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[546] Supasorn Suwajanakorn, Noah Snavely, Jonathan J Tompson, and Mohammad Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. In *Advances in Neural Information Processing Systems*, 2018.

[547] Lucas Manuelli, Wei Gao, Peter Florence, and Russ Tedrake. kpam: Keypoint affordances for category-level robotic manipulation. In *The International Symposium of Robotics Research*, pages 132–157, 2019.

[548] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pages 3686–3693, 2014.

[549] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5167–5176, 2018.

[550] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(7):1325–1339, 2014.

[551] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7297–7306, 2018.

[552] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10863–10872, 2019.

[553] Martin Koestinger, Paul Wohlhart, Peter M Roth, and Horst Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pages 2144–2151, 2011.

[554] Vuong Le, Jonathan Brandt, Zhe Lin, Lubomir Bourdev, and Thomas S Huang. Interactive facial feature localization. In *European conference on computer vision*, pages 679–692, 2012.

[555] Xavier P Burgos-Artizzu, Pietro Perona, and Piotr Dollár. Robust face landmark estimation under occlusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1513–1520, 2013.

[556] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 397–403, 2013.

[557] Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.

[558] Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2129–2138, 2018.

[559] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017.

[560] Christian Zimmermann and Thomas Brox. Learning to estimate 3d hand pose from single rgb images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4903–4911, 2017.

[561] Christian Zimmermann, Duygu Ceylan, Jimei Yang, Bryan Russell, Max Argus, and Thomas Brox. Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 813–822, 2019.

[562] Gyeongsik Moon, Shoou-I Yu, He Wen, Takaaki Shiratori, and Kyoung Mu Lee. Interhand2. 6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. In *European Conference on Computer Vision*, pages 548–564, 2020.

[563] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Cewu Lu, and Yu-Wing Tai. Cross-domain adaptation for animal pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9498–9507, 2019.

[564] Hang Yu, Yufei Xu, Jing Zhang, Wei Zhao, Ziyu Guan, and Dacheng Tao. Ap-10k: A benchmark for animal pose estimation in the wild. *arXiv preprint arXiv:2108.12617*, 2021.

[565] Xibin Song, Peng Wang, Dingfu Zhou, Rui Zhu, Chenye Guan, Yuchao Dai, Hao Su, Hongdong Li, and Ruigang Yang. Apollocar3d: A large 3d car instance understanding benchmark for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5452–5462, 2019.

[566] Yang You, Yujing Lou, Chengkun Li, Zhoujun Cheng, Liangwei Li, Lizhuang Ma, Cewu Lu, and Weiming Wang. Keypointnet: A large-scale 3d keypoint dataset aggregated from numerous human annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13647–13656, 2020.

[567] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *International Conference on 3D Vision*, 2017.

[568] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[569] Nagwa R Fisal, Ahmed Fathalla, Dina Elmanakhly, and Ahmad Salah. Reported challenges in deep learning-based human pose estimation: A systematic review. *IEEE Access*, 2025.

[570] Yan Guo, Tianhan Gao, Aoshuang Dong, Xinbei Jiang, Zichen Zhu, and Fuxin Wang. A survey of the state of the art in monocular 3d human pose estimation: Methods, benchmarks, and challenges. *Sensors (Basel, Switzerland)*, 25(8):2409, 2025.

[571] Marsha Mariya Kappan, Eduardo Benitez Sandoval, Erik Meijering, and Francisco Cruz. A survey on deep learning for 2d and 3d human pose estimation. *Artificial Intelligence Review*, 59(1):32, 2026.

[572] Pengfei Ma, Ronghua Gao, Qifeng Li, Qinyang Yu, Rong Wang, Chengrong Lai, Weiwei Huang, Peng Hao, Zhaoyang Wang, and Xuwen Li. 2d animal skeletons keypoint detection: Research progress and future trends. *IEEE Access*, 2025.

[573] Le Jiang, Caleb Lee, Divyang Teotia, and Sarah Ostadabbas. Animal pose estimation: A closer look at the state-of-the-art, existing gaps and opportunities. *Computer Vision and Image Understanding*, 222: 103483, 2022.

[574] Kostiantyn Khabarlak and Larysa Koriashkina. Fast facial landmark detection and applications: A survey. *arXiv preprint arXiv:2101.10808*, 2021.

[575] Constantino Álvarez Casado and Miguel Bordallo López. Real-time face alignment: evaluation methods, training strategies and implementation optimization. *Journal of Real-Time Image Processing*, pages 2239–2267, 2021.

[576] Delu Zeng, Minyu Liao, Mohammad Tavakolian, Yulan Guo, Bolei Zhou, Dewen Hu, Matti Pietikäinen, and Li Liu. Deep learning for scene classification: A survey. *arXiv preprint arXiv:2101.10531*, 2021.

[577] John M Henderson and Andrew Hollingworth. High-level scene perception. *Annual review of psychology*, 50(1):243–271, 1999.

[578] Russell Epstein. The cortical basis of visual scene processing. *Visual Cognition*, 12(6):954–978, 2005.

[579] Yiyi Liao, Sarath Kodagoda, Yue Wang, Lei Shi, and Yong Liu. Understand scene categories by objects: A semantic regularized scene classifier using convolutional neural networks. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 2318–2325, 2016.

[580] Jianjun Ni, Kang Shen, Yinan Chen, Weidong Cao, and Simon X Yang. An improved deep network-based scene classification method for self-driving cars. *IEEE Transactions on Instrumentation and Measurement*, 71:1–14, 2022.

[581] Fangfang Zhang, Arianna Salazar-Miranda, Fábio Duarte, Lawrence J. Vale, Gary Hack, Min Chen, Yu Liu, Michael Batty, and Carlo Ratti. Urban visual intelligence: Studying cities with artificial intelligence and street-level imagery. *Annals of the American Association of Geographers*, 114:876 – 897, 2024. URL https://api.semanticscholar.org/CorpusID:269016687.

[582] Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. Learning multi-label scene classification. *Pattern recognition*, 37(9):1757–1771, 2004.

[583] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *2009 IEEE conference on computer vision and pattern recognition*, pages 413–420, 2009.

[584] R. Raja, S. Md. Mansoor Roomi, D. Dharmalakshmi, and S. Rohini. Classification of indoor/outdoor scene. In *2013 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–4, 2013. doi:10.1109/ICCIC.2013.6724252.

[585] Keiller Nogueira, Otávio AB Penatti, and Jefersson A Dos Santos. Towards better exploiting convolutional neural networks for remote sensing scene classification. *Pattern Recognition*, 61:539–556, 2017.

[586] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017.

[587] Jincheng Huang, Zhu Liu, Yao Wang, Yu Chen, and Edward K Wong. Integration of multimodal features for video scene classification based on hmm. In *1999 IEEE third workshop on multimedia signal processing (Cat. No. 99TH8451)*, pages 53–58, 1999.

[588] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. A multi-device dataset for urban acoustic scene classification. *arXiv preprint arXiv:1807.09840*, 2018.

[589] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[590] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, pages 1–2, 2004.

[591] Marco San Biagio, Loris Bazzani, Marco Cristani, and Vittorio Murino. Weighted bag of visual words for object recognition. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2734–2738, 2014. doi:10.1109/ICIP.2014.7025553.

[592] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, volume 2, pages 2169–2178, 2006.

[593] Keiron O'shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

[594] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[595] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[596] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 27, 2014.

[597] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[598] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

[599] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[600] Jiali Liang, Yufan Deng, and Dan Zeng. A deep neural network combined cnn and gcn for remote sensing scene classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:4325–4338, 2020. URL https://api.semanticscholar.org/CorpusID:221119590.

[601] Xiang Li, Congcong Wen, Yuan Hu, and Nan Zhou. Rs-clip: Zero shot remote sensing scene classification via contrastive vision-language supervision. *International Journal of Applied Earth Observation and Geoinformation*, 124:103497, 2023.

[602] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications. *arXiv preprint arXiv:2402.07927*, 2024.

[603] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023.

[604] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

[605] Oindrila Saha, Grant Van Horn, and Subhransu Maji. Improved zero-shot classification by adapting vlms with text descriptions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17542–17552, June 2024.

[606] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023.

[607] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *Proc. CVPR*, pages 3485–3492, 2010.

[608] Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision*, 119:3–22, 2014. URL https://api.semanticscholar.org/CorpusID:10224573.

[609] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40 (6):1452–1464, 2018.

[610] Yuhui Zhang, Alyssa Unell, Xiaohan Wang, Dhruba Ghosh, Yuchang Su, Ludwig Schmidt, and Serena Yeung-Levy. Why are visually-grounded language models bad at image classification? In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 51727–51753, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/5c7024041be305c94d7311cfcc53d93e-Paper-Conference.pdf.

[611] Peiran Xu, Sudong Wang, Yao Zhu, Jianing Li, Gege Qi, and Yunjian Zhang. Spatialbench: Benchmarking multimodal large language models for spatial cognition. *arXiv preprint arXiv:2511.21471*, 2025.

[612] Erik Daxberger, Nina Wenzel, David Griffiths, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, and Peter Grasch. Mm-spatial: Exploring 3d spatial understanding in multimodal llms. *arXiv preprint arXiv:2503.13111*, 2025.

[613] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2020.

[614] Ashutosh Saxena, Sung H Chung, and Andrew Y Ng. Learning depth from single monocular images. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 18, 2005.

[615] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5162–5170, 2015.

[616] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248, 2016.

[617] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2002–2011, 2018.

[618] Jure Zbontar and Yann LeCun. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1592–1599, 2015.

[619] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016.

[620] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5418, 2018.

[621] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2307–2315, 2017.

[622] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[623] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017.

[624] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279, 2017.

[625] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1983–1992, 2018.

[626] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan L Yuille. Towards unified depth and semantic prediction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2800–2809, 2015.

[627] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *arXiv preprint arXiv:1812.11941*, 2018.

[628] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019.

[629] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2020.

[630] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[631] Zhenyu Li, Zehui Chen, Xianming Liu, and Junjun Jiang. Depthformer: Exploiting long-range correlation and local information for accurate monocular depth estimation. *Machine Intelligence Research*, 2023.

[632] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023.

[633] Jamie Watson, Oisin Mac Aodha, Victor Prisacariu, Gabriel Brostow, and Michael Firman. The temporal opportunist: Self-supervised multi-frame monocular depth. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1164–1174, 2021.

[634] Chaoqiang Zhao, Youmin Zhang, Matteo Poggi, Fabio Tosi, Xianda Guo, Zheng Zhu, Guan Huang, Yang Tang, and Stefano Mattoccia. Monovit: Self-supervised monocular depth estimation with a vision transformer. In *2022 international conference on 3D vision (3DV)*, pages 668–678, 2022.

[635] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5729–5739, 2023.

[636] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9492–9502, 2024.

[637] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *European Conference on Computer Vision*, pages 241–258, 2024.

[638] Misha Mahowald. *VLSI analogs of neuronal visual processing: a synthesis of form and function*. PhD thesis, California Institute of Technology Pasadena, 1992.

[639] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. A 128×128 120 db 15 μs latency asynchronous temporal contrast vision sensor. *IEEE journal of solid-state circuits*, pages 566–576, 2008.

[640] Henri Rebecq, Guillermo Gallego, Elias Mueggler, and Davide Scaramuzza. Emvs: Event-based multi-view stereo—3d reconstruction with an event camera in real-time. *International Journal of Computer Vision*, pages 1394–1414, 2018.

[641] Sio-Hoi Ieng, Joao Carneiro, Marc Osswald, and Ryad Benosman. Neuromorphic event-based generalized time-based stereovision. *Frontiers in neuroscience*, 12:442, 2018.

[642] Yi Zhou, Guillermo Gallego, Henri Rebecq, Laurent Kneip, Hongdong Li, and Davide Scaramuzza. Semi-dense 3d reconstruction with a stereo event camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 235–251, 2018.

[643] Zhen Xie, Shengyong Chen, and Garrick Orchard. Event-based stereo depth estimation using belief propagation. *Frontiers in neuroscience*, 11:535, 2017.

[644] Stepan Tulyakov, Francois Fleuret, Martin Kiefel, Peter Gehler, and Michael Hirsch. Learning an event sequence embedding for dense event-based deep stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1527–1537, 2019.

[645] Kaixuan Zhang, Kaiwei Che, Jianguo Zhang, Jie Cheng, Ziyang Zhang, Qinghai Guo, and Luziwei Leng. Discrete time convolution for fast event-based stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8676–8686, 2022.

[646] Mohammad Mostafavi, Kuk-Jin Yoon, and Jonghyun Choi. Event-intensity stereo: Estimating depth by the best of both worlds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4258–4267, 2021.

[647] Suman Ghosh and Guillermo Gallego. Multi-event-camera depth estimation and outlier rejection by refocused events fusion. *Advanced Intelligent Systems*, page 2200221, 2022.

[648] Hoonhee Cho, Jegyeong Cho, and Kuk-Jin Yoon. Learning adaptive dense event stereo from the image domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17797–17807, 2023.

[649] Diego de Oliveira Hitzges, Suman Ghosh, and Guillermo Gallego. Derd-net: Learning depth from event-based ray densities. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[650] Hoonhee Cho, Jae-Young Kang, and Kuk-Jin Yoon. Temporal event stereo via joint learning with stereoscopic flow. In *European Conference on Computer Vision*, pages 294–314, 2024.

[651] Suman Ghosh and Guillermo Gallego. Event-based stereo depth estimation: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[652] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *Proc. IROS*, pages 573–580, 2012.

[653] A. Geiger, P. Lenz, and R. Urtasun. Vision meets robotics: The KITTI dataset. *International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[654] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 international conference on 3D Vision (3DV)*, pages 11–20, 2017.

[655] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[656] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 311–320, 2018.

[657] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018.

[658] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3260–3269, 2017. doi:10.1109/CVPR.2017.272. URL https://www.eth3d.net/data/schoeps2017cvpr.pdf.

[659] Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Korner. Evaluation of cnn-based single-image depth estimation methods. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[660] Peng Wang, Xinyu Huang, Xinjing Cheng, Dingfu Zhou, Qichuan Geng, and Ruigang Yang. The apolloscape open dataset for autonomous driving and its application. *IEEE transactions on pattern analysis and machine intelligence*, 1, 2019.

[661] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, Deva Ramanan, Peter Carr, and James Hays. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks)*, 2021.

[662] Yiyi Liao, Jun Xie, and Andreas Geiger. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 3292–3310, 2022.

[663] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2485–2494, 2020.

[664] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10912–10922, 2021.

[665] Jose L Gómez, Manuel Silva, Antonio Seoane, Agnès Borrás, Mario Noriega, Germán Ros, Jose A Iglesias-Guitian, and Antonio M López. All for one, and one for all: Urbansyn dataset, the third musketeer of synthetic driving scenes. *arXiv preprint arXiv:2312.12176*, 2023.

[666] Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. Tartanair: A dataset to push the limits of visual slam. In *IROS*, 2020.

[667] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z Dai, Andrea F Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R Walter, et al. Diode: A dense indoor and outdoor depth dataset. *arXiv preprint arXiv:1908.00463*, 2019.

[668] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1790–1799, 2020.

[669] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. *arXiv preprint arXiv:2111.08897*, 2021.

[670] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4521–4530, 2019.

[671] Hongchi Xia, Yang Fu, Sifei Liu, and Xiaolong Wang. Rgbd objects in the wild: Scaling real-world 3d object learning from rgb-d videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22378–22389, 2024.

[672] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. UniDepth: universal monocular metric depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2024.

[673] Jiahao Shao, Yuanbo Yang, Hongyu Zhou, Youmin Zhang, Yujun Shen, Vitor Guizilini, Yue Wang, Matteo Poggi, and Yiyi Liao. Learning temporally consistent video depth from video diffusion priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22841–22852, 2025.

[674] Bingxin Ke, Dominik Narnhofer, Shengyu Huang, Lei Ke, Torben Peters, Katerina Fragkiadaki, Anton Obukhov, and Konrad Schindler. Video depth without video models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7233–7243, 2025.

[675] Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[676] Ziyao Zeng, Yangchao Wu, Hyoungseob Park, Daniel Wang, Fengyu Yang, Stefano Soatto, Dong Lao, Byung-Woo Hong, and Alex Wong. Rsa: Resolving scale ambiguities in monocular depth estimators through language descriptions. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:112684–112705, 2024.

[677] Genki Higashiuchi, Tomoyasu Shimada, Xiangbo Kong, Haimin Yan, and Hiroyuki Tomiyama. Monolens: Monocular lightweight efficient network with separable convolutions for self-supervised monocular depth estimation. *Applied Sciences*, page 10393, 2025.

[678] Jiahuan Long and Xin Zhou. Lmdepth: Lightweight mamba-based monocular depth estimation for real-world deployment. *arXiv preprint arXiv:2505.00980*, 2025.

[679] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2022.

[680] Daniel M. Bear, Elias Wang, Damian Mrowca, Felix J. Binder, Hsiao-Yu Fish Tung, R. T. Pramod, Cameron Holdaway, Sirui Tao, Kevin Smith, Fan-Yun Sun, Li Fei-Fei, Nancy Kanwisher, Joshua B. Tenenbaum, Daniel L. K. Yamins, and Judith E. Fan. Physion: Evaluating physical prediction from vision in humans and machines, 2021. URL https://arxiv.org/abs/2106.08261.

[681] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in

3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022.

[682] Amir R. Zamir, Alexander Sax, William Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[683] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir R. Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[684] Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[685] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10579–10596, 2024.

[686] Chongjie Ye, Lingteng Qiu, Xiaodong Gu, Qi Zuo, Yushuang Wu, Zilong Dong, Liefeng Bo, Yuliang Xiu, and Xiaoguang Han. Stablenormal: Reducing diffusion variance for stable and sharp normal. *arXiv preprint arXiv:2406.16864*, 2024.

[687] Yanrui Bin, Wenbo Hu, Haoyuan Wang, Xinya Chen, and Bing Wang. Normalcrafter: Learning temporally consistent normals from video diffusion priors. *arXiv preprint arXiv:2504.11427*, 2025.

[688] Mingwei Li, Hehe Fan, and Yi Yang. Transnormal: Dense visual semantics for diffusion-based transparent object normal estimation. *arXiv preprint arXiv:2602.00839*, 2026.

[689] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics (SIGGRAPH)*, 33(4), 2014.

[690] Qifeng Chen and Vladlen Koltun. A simple model for intrinsic image decomposition with depth cues. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2013.

[691] Duane C Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[692] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[693] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Computer Vision*, 47(1–3):7–42, 2002.

[694] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 519–528, 2006.

[695] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *International Conference on 3D Vision*, 2021.

[696] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7644–7652, 2019.

[697] Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Dsgn: Deep stereo geometry network for 3d object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[698] Xiaoyang Guo et al. Liga-stereo: Learning lidar geometry aware representations for stereo-based 3d detector. In *IEEE/CVF International Conference on Computer Vision*, 2021.

[699] Antonio Criminisi, Ian Reid, and Andrew Zisserman. Single view metrology. *International Journal of Computer Vision*, 40(2):123–148, 2000.

[700] Bruno Caprile and Vincenzo Torre. Using vanishing points for camera calibration. *International Journal of Computer Vision*, 4(2):127–139, 1990.

[701] David Liebowitz and Andrew Zisserman. Metric rectification for perspective images of planes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 482–488, 1998.

[702] Xiaozhi Chen, Kaustav Kundu, Yukun Zhu, Huimin Ma, Sanja Fidler, and Raquel Urtasun. Monocular 3d object detection for autonomous driving. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2147–2156, 2016.

[703] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7074–7082, 2017.

[704] Manei et al. Monodis: Geometry and context aware monocular 3d object detection. In *IEEE/CVF International Conference on Computer Vision Workshops*, 2019.

[705] Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 9287–9296, 2019.

[706] Zechen Liu, Zizhang Wu, Roland Toth, et al. Smoke: Single-stage monocular 3d object detection via keypoint estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020.

[707] Tai Wang et al. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *IEEE/CVF International Conference on Computer Vision Workshops*, 2021.

[708] Liu et al. Monocon: Monocular 3d object detection with monocular contextual constraints. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[709] Wei Yin et al. Metric3d: Towards zero-shot metric 3d prediction from a single image. *arXiv preprint*, 2024.

[710] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *CVPR*, pages 2642–2651, 2019.

[711] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2636–2645, 2020.

[712] Milind Naphade, Ming-Ching Chang, Anuj Sharma, David C. Anastasiu, Vamsi Jagarlamudi, Pranamesh Chakraborty, Tingting Huang, Shuo Wang, Ming-Yu Liu, Rama Chellappa, Jenq-Neng Hwang, and Siwei Lyu. The 2018 nvidia ai city challenge. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR) Workshop*, page 53–60, 2018.

[713] Jakub Sochor, Roman Juránek, Jakub Spanhel, Lukás Marsík, Adam Siroký, Adam Herout, and Pavel Zemcík. Brnocompspeed: Review of traffic camera calibration and comprehensive dataset for monocular speed measurement. *arXiv preprint arXiv:1702.06441*, 2017.

[714] Md Golam Moazzam, Mohammad Reduanul Haque, and Mohammad Shorif Uddin. Image-based vehicle speed estimation. *Journal of computer and communications*, 7(6):1–5, 2019.

[715] Ruimin Ke, Sung Kim, Zhibin Li, and Yinhai Wang. Motion-vector clustering for traffic speed detection from UAV video. In *ISC2*, pages 1–5, 2015.

[716] Sameer Agarwal, Noah Snavely, Steven M Seitz, and Richard Szeliski. Bundle adjustment in the large. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010.

[717] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes Lutz Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision (ECCV)*, 2024.

[718] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam: Large-scale direct monocular slam. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[719] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2017.

[720] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2024.

[721] Eric Brachmann, Jamie Wynn, Shuai Chen, Tommaso Cavallari, Áron Monszpart, Daniyar Turmukhambetov, and Victor Adrian Prisacariu. Scene coordinate reconstruction: Posing of image collections via incremental learning of a relocalizer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[722] Weirong Chen, Le Chen, Rui Wang, and Marc Pollefeys. Leap-vo: Long-term effective any point tracking for visual odometry. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2024.

[723] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pages 2938–2946, 2015.

[724] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2013.

[725] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac - differentiable ransac for camera localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2017.

[726] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019.

[727] Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. *arXiv*, 2412.08376, 2024.

[728] Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding image matching in 3d with MASt3R. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[729] Bardienus Duisterhof, Lojze Zust, Philippe Weinzaepfel, Vincent Leroy, Yohann Cabon, and Jerome Revaud. MASt3R-SfM: a fully-integrated solution for unconstrained structure-from-motion. *arXiv*, 2409.19152, 2024.

[730] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3R: a simple approach for estimating geometry in the presence of motion. In *ICLR*, 2025.

[731] Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10510–10522, 2025.

[732] Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21924–21935, 2025.

[733] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart. The euroc micro aerial vehicle datasets. *Int. J. Robotics Research*, 35(10):1157–1163, 2016.

[734] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

[735] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[736] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multi-map slam. In *Proc. IEEE IROS*, pages 11786–11793, 2021.

[737] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

[738] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4561–4570, 2019.

[739] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1521–1529, 2017.

[740] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proc. CVPR*, pages 3343–3352, 2019.

[741] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16611–16621, 2021.

[742] Yongzhi Su, Mahdi Saleh, Torben Fetzer, Jason Rambach, Nassir Navab, Benjamin Busam, Didier Stricker, and Federico Tombari. Zebrapose: Coarse to fine surface encoding for 6dof object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[743] Rasmus Laurvig Haugaard and Anders Glent Buch. Surfemb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6739–6748, 2022.

[744] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2781–2790, 2022.

[745] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018.

[746] Y. Labbé, J. Carpentier, M. Aubry, and J. Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation, 2020. URL https://arxiv.org/abs/2008.08465.

[747] Yinlin Hu, Sebastien Speierer, Wenzel Jakob, Pascal Fua, and Mathieu Salzmann. Wide-depth-range 6d object pose estimation in space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15870–15879, 2021.

[748] Wei Chen, Xi Jia, Hyung Jin Chang, Jinming Duan, Linlin Shen, and Aleš Leonardis. Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1581–1590, 2021.

[749] Yan Di, Ruida Zhang, Zhiqiang Lou, Fabian Manhardt, Xiangyang Ji, Nassir Navab, and Federico Tombari. Gpv-pose: Category-level object pose estimation via geometry-guided point-wise voting.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6781–6791, 2022.

[750] Haitao Lin, Zichang Liu, Chilam Cheang, Yanwei Fu, Guodong Guo, and Xiangyang Xue. Sar-net: Shape alignment and recovery network for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6707–6717, 2022.

[751] Jianhui Liu, Yukang Chen, Xiaoqing Ye, and Xiaojuan Qi. Prior-free category-level pose estimation with implicit space transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13880–13889, 2023.

[752] Yamei Chen, Yan Di, Guangyao Zhai, Fabian Manhardt, Chenyangguang Zhang, Ruida Zhang, Federico Tombari, Nassir Navab, and Benjamin Busam. Secondpose: Se(3)-consistent dual-stream feature fusion for category-level pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[753] Xiao Lin, Wenfei Yang, Yuan Gao, and Tianzhu Zhang. Instance-adaptive and geometric-aware keypoint learning for category-level 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[754] Jiyao Zhang, Mingdong Wu, and Hao Dong. Genpose: Generative category-level object pose estimation via diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[755] Jiaming Sun, Zihao Wang, Siyu Zhang, Xingyi He, Hongcheng Zhao, Guofeng Zhang, and Xiaowei Zhou. Onepose: One-shot object pose estimation without CAD models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6825–6834, 2022.

[756] Xingyi He, Jiaming Sun, Yuang Wang, Di Huang, Hujun Bao, and Xiaowei Zhou. Onepose++: Keypoint-free one-shot object pose estimation without CAD models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[757] Yuan Liu, Yilin Wen, Sida Peng, Cheng Lin, Xiaoxiao Long, Taku Komura, and Wenping Wang. Gen6d: Generalizable model-free 6-dof object pose estimation from RGB images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 298–315, 2022.

[758] Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. Megapose: 6d pose estimation of novel objects via render & compare. In *Conference on Robot Learning (CoRL)*, 2022.

[759] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17868–17879, 2024.

[760] Van Nguyen Nguyen, Thibault Groueix, Mathieu Salzmann, and Vincent Lepetit. Gigapose: Fast and robust novel object pose estimation via one correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9903–9913, 2024.

[761] Jiehong Lin, Lihua Liu, Dekun Lu, and Kui Jia. Sam-6d: Segment anything model meets zero-shot 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27906–27916, 2024.

[762] Evin Pınar Örnek, Yann Labbé, Bugra Tekin, Lingni Ma, Cem Keskin, Christian Forster, and Tomas Hodan. Foundpose: Unseen object pose estimation with foundation features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[763] Andrea Caraffa, Davide Boscaini, Amir Hamza, and Fabio Poiesi. Freeze: Training-free zero-shot 6d pose estimation with geometric and vision foundation models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[764] Van Nguyen Nguyen, Stephen Tyree, Andrew Guo, Médéric Fourmy, Anas Gouda, Taeyeop Lee, Sungphill Moon, Hyeontae Son, Lukas Ranftl, Jonathan Tremblay, Eric Brachmann, Bertram Drost, Vincent Lepetit, Carsten Rother, Stan Birchfield, Jiri Matas, Yann Labbé, Martin Sundermeyer, and Tomas Hodan. BOP challenge 2024 on model-based and model-free 6d object pose estimation. *arXiv preprint arXiv:2504.02812*, 2024.

[765] Jian Liu, Wei Sun, Hui Yang, Pengchao Deng, Chongpei Liu, Nicu Sebe, Hossein Rahmani, and Ajmal Mian. Diff9d: Diffusion-based domain-generalized category-level 9-dof object pose estimation. *arXiv preprint arXiv:2502.02525*, 2024.

[766] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian Conference on Computer Vision (ACCV)*, pages 548–562, 2012.

[767] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 536–551, 2014.

[768] T. Hodaň, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis. T-less: An rgb-d dataset for 6d pose estimation of texture-less objects. In *Proc. IEEE WACV*, pages 880–888, 2017.

[769] Roman Kaskman, Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Homebreweddb: Rgb-d dataset for 6d pose estimation of 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2019.

[770] Sandeep Sharma, Tae Hyung Park, and Simone D'Amico. Spacecraft pose estimation dataset (speed). *Stanford Digital Repository*, 2019.

[771] Tomáš Hodaň, Frank Michel, Eric Brachmann, Wadim Kehl, Anders Glent Buch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiri Matas, and Carsten Rother. BOP: Benchmark for 6d object pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–34, 2018.

[772] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. BOP challenge 2020 on 6d object localization. In *Proceedings of the European Conference on Computer Vision Workshops (ECCVW)*, 2020.

[773] Martin Sundermeyer, Tomáš Hodaň, Yann Labbé, Gu Wang, Eric Brachmann, Bertram Drost, Carsten Rother, and Jiří Matas. BOP challenge 2022 on detection, segmentation and pose estimation of specific rigid objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2784–2793, 2023.

[774] Taeyeop Lee, Bowen Wen, Minjun Kang, Gyuree Kang, In So Kweon, and Kuk-Jin Yoon. Any6d: Model-free 6d pose estimation of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[775] Zhiwen Fan, Panwang Pan, Peihao Wang, Yifan Jiang, Dejia Xu, Hanwen Jiang, and Zhangyang Wang. Pope: 6-dof promptable pose estimation of any object, in any scene, with one reference. *arXiv preprint arXiv:2305.15727*, 2023.

[776] Mingjie Xu, Mengyang Wu, Yuzhi Zhao, Jason Chun Lok Li, and Weifeng Ou. Llava-spacesgg: Visual instruct tuning for open-vocabulary scene graph generation with enhanced spatial relations. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6362–6372, 2025.

[777] Rongjie Li, Songyang Zhang, Dahua Lin, Kai Chen, and Xuming He. From pixels to graphs: Open-vocabulary scene graph generation with vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 28076–28086, 2024.

[778] Shengqiong Wu, Hao Fei, and Tat-Seng Chua. Universal scene graph generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14158–14168, 2025.

[779] Arnav Vaibhav Malawade, Shih-Yuan Yu, Brandon Hsu, Harsimrat Kaeley, Anurag Karra, and Mohammad Abdullah Al Faruque. roadscene2vec: A tool for extracting and embedding road scene-graphs. *Knowledge-Based Systems*, 242:108245, 2022.

[780] Yuchen Zhou, Guang Tan, Rui Zhong, Yaokun Li, and Chao Gou. Pit: Progressive interaction transformer for pedestrian crossing intention prediction. *IEEE Transactions on Intelligent Transportation Systems*, 24(12):14213–14225, 2023.

[781] Xinxin Liu, Yuchen Zhou, Yongqi Ye, and Chao Gou. Edge feature-enhanced network for collision risk assessment using traffic scene graphs. *IEEE Intelligent Transportation Systems Magazine*, 2024.

[782] Linkai Liu, Xiaoyan Xiao, Yijian Yang, Yuchen Zhou, Zipeng Guo, and Chao Gou. Boosting road event detection with adaptive multi-modal models. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2025.

[783] Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. Robospatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 15768–15780, 2025.

[784] Shiduo Zhang, Zhe Xu, Peiju Liu, Xiaopeng Yu, Yuan Li, Qinghui Gao, Zhaoye Fei, Zhangyue Yin, Zuxuan Wu, Yu-Gang Jiang, et al. Vlabench: A large-scale benchmark for language-conditioned robotics manipulation with long-horizon reasoning tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11142–11152, 2025.

[785] Kaiyu Yang, Olga Russakovsky, and Jia Deng. Spatialsense: An adversarially crowdsourced benchmark for spatial relation recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2051–2060, 2019.

[786] Yu Zhao, Hao Fei, Bobo Li, Meishan Zhang, and Min Zhang. The acm multimedia 2024 viual spatial description grand challenge. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 11404–11406, 2024.

[787] Yu Zhao, Jianguo Wei, ZhiChao Lin, Yueheng Sun, Meishan Zhang, and Min Zhang. Visual spatial description: Controlled spatial-oriented image-to-text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1437–1449, 2022.

[788] Yu Zhao, Hao Fei, Wei Ji, Jianguo Wei, Meishan Zhang, Min Zhang, and Tat-Seng Chua. Generating visual spatial description via holistic 3d scene understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7960–7977, 2023.

[789] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision-language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:135062–135093, 2024.

[790] Yuchen Zhou, Xinxin Liu, Zipeng Guo, Ming Cai, and Chao Gou. Hktsg: A hierarchical knowledge-guided traffic scene graph representation learning framework for intelligent vehicles. *IEEE Transactions on Intelligent Vehicles*, 2024.

[791] Yu-Chih Liu, Kai-Ying Lin, and Yong-Sheng Chen. Bird's-eye view vision system for vehicle surrounding monitoring. In *International Workshop on Robot Vision*, pages 207–218, 2008.

[792] Yinyu Nie, Xiaoguang Han, Shihui Guo, Yujian Zheng, Jian Chang, and Jian Jun Zhang. Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 55–64, 2020.

[793] Wenxiao Cai, Yaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. *arXiv preprint arXiv:2406.13642*, 2024.

[794] Wufei Ma, Luoxin Ye, Celso M de Melo, Alan Yuille, and Jieneng Chen. Spatialllm: A compound 3d-informed design towards spatially-intelligent large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17249–17260, 2025.

[795] Peng Wang, Minh Huy Pham, Zhihao Guo, and Wei Zhou. A spatial relationship aware dataset for robotics. *arXiv preprint arXiv:2506.12525*, 2025.

[796] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

[797] Anja Belz, Adrian Muscat, Pierre Anguill, Mouhamadou Sow, Gaétan Vincent, and Yassine Zinessabah. Spatialvoc2k: A multilingual dataset of images with annotations and features for spatial relations between objects. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 140–145, 2018.

[798] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5419, 2017.

[799] Xiaojun Chang, Pengzhen Ren, Pengfei Xu, Zhihui Li, Xiaojiang Chen, and Alex Hauptmann. A comprehensive survey of scene graphs: Generation and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):1–26, 2021.

[800] Hongsheng Li, Guangming Zhu, Liang Zhang, Youliang Jiang, Yixuan Dang, Haoran Hou, Peiyi Shen, Xia Zhao, Syed Afaq Ali Shah, and Mohammed Bennamoun. Scene graph generation: A comprehensive survey. *Neurocomputing*, 566:127052, 2024.

[801] Yang Miao, Francis Engelmann, Olga Vysotska, Federico Tombari, Marc Pollefeys, and Dániel Béla Baráth. Scenegraphloc: Cross-modal coarse visual localization on 3d scene graphs. In *European Conference on Computer Vision*, pages 127–150, 2024.

[802] Yiwu Zhong, Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. Comprehensive image captioning via scene graph decomposition. In *European conference on computer vision*, pages 211–229, 2020.

[803] Fan Lu, Wei Wu, Kecheng Zheng, Shuailei Ma, Biao Gong, Jiawei Liu, Wei Zhai, Yang Cao, Yujun Shen, and Zheng-Jun Zha. Benchmarking large vision-language models via directed scene graph for comprehensive image captioning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19618–19627, 2025.

[804] Yuchen Zhou, Yue Zhang, Zhanwei Zhao, Kaidong Zhang, and Chao Gou. Toward driving scene understanding: A paradigm and benchmark dataset for ego-centric traffic scene graph representation. *IEEE Journal of Radio Frequency Identification*, 6:962–967, 2022.

[805] Xinxin Liu, Yuchen Zhou, and Chao Gou. Learning from interaction-enhanced scene graph for pedestrian collision risk assessment. *IEEE Transactions on Intelligent Vehicles*, 8(9):4237–4248, 2023.

[806] Giang Nguyen, Mihai Pomarlan, Sascha Jongebloed, Nils Leusmann, Minh Nhat Vu, and Michael Beetz. Generating actionable robot knowledge bases by combining 3d scene graphs with robot ontologies. *arXiv preprint arXiv:2507.11770*, 2025.

[807] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3d scene graphs for language-grounded robot navigation. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

[808] Sourav Garg, Krishan Rana, Mehdi Hosseinzadeh, Lachlan Mares, Niko Sünderhauf, Feras Dayoub, and Ian Reid. Robohop: Segment-based topological map representation for open-world visual navigation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4090–4097, 2024.

[809] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5831–5840, 2018.

[810] Yao Teng, Limin Wang, Zhifeng Li, and Gangshan Wu. Target adaptive context aggregation for video scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13688–13697, 2021.

[811] Yuren Cong, Wentong Liao, Hanno Ackermann, Bodo Rosenhahn, and Michael Ying Yang. Spatial-temporal transformer for dynamic scene graph generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16372–16382, 2021.

[812] Zhuang Li, Yuyang Chai, Terry Yue Zhuo, Lizhen Qu, Gholamreza Haffari, Fei Li, Donghong Ji, and Quan Hung Tran. Factual: A benchmark for faithful and consistent textual scene graph parsing. *arXiv preprint arXiv:2305.17497*, 2023.

[813] Woo Suk Choi, Yu-Jung Heo, Dharani Punithan, and Byoung-Tak Zhang. Scene graph parsing via abstract meaning representation in pre-trained language models. In *Proceedings of the 2nd workshop on deep learning on graphs for natural language processing (DLG4NLP 2022)*, pages 30–35, 2022.

[814] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3585–3594, 2019.

[815] Jinwang Wang, Wen Yang, Heng-Chao Li, Haijian Zhang, and Gui-Song Xia. Learning center probability map for detecting objects in aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(5):4307–4323, 2020.

[816] Jingkang Yang, Yi Zhe Ang, Zujin Guo, Kaiyang Zhou, Wayne Zhang, and Ziwei Liu. Panoptic scene graph generation. In *European conference on computer vision*, pages 178–196, 2022.

[817] Jingkang Yang, Wenxuan Peng, Xiangtai Li, Zujin Guo, Liangyu Chen, Bo Li, Zheng Ma, Kaiyang Zhou, Wayne Zhang, Chen Change Loy, et al. Panoptic video scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18675–18685, 2023.

[818] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11109–11119, 2021.

[819] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3746–3753, 2020.

[820] Lin Li, Guikun Chen, Jun Xiao, Yi Yang, Chunping Wang, and Long Chen. Compositional feature augmentation for unbiased scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21685–21695, 2023.

[821] Yuchen Zhou, Guang Tan, and Chao Gou. Hierarchical home action understanding with implicit and explicit prior knowledge. In *ICASSP*, pages 4015–4019, 2024.

[822] Jinbae Im, JeongYeon Nam, Nokyung Park, Hyungmin Lee, and Seunghyun Park. Egtr: Extracting graph from transformer for scene graph generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24229–24238, 2024.

[823] Guan Wang, Zhimin Li, Qingchao Chen, and Yang Liu. Oed: Towards one-stage end-to-end dynamic scene graph generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 27938–27947, 2024.

[824] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[825] Hengyue Liu, Ning Yan, Masood Mortazavi, and Bir Bhanu. Fully convolutional scene graph generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11546–11556, 2021.

[826] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11169–11183, 2023.

[827] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19486–19496, 2022.

[828] Xiaodan Liang, Lisa Lee, and Eric P Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 848–857, 2017.

[829] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European conference on computer vision*, pages 852–869, 2016.

[830] Zhen Cui, Chunyan Xu, Wenming Zheng, and Jian Yang. Context-dependent diffusion network for visual relationship detection. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1475–1482, 2018.

[831] Ruichi Yu, Ang Li, Vlad I Morariu, and Larry S Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1974–1982, 2017.

[832] Yuchen Zhou, Guang Tan, Mengtang Li, and Chao Gou. Learning from easy to hard pairs: Multi-step reasoning network for human-object interaction detection. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 4368–4377, 2023.

[833] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

[834] Ce Zhang, Simon Stepputtis, Joseph Campbell, Katia Sycara, and Yaqi Xie. Hiker-sgg: Hierarchical knowledge enhanced robust scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28233–28243, 2024.

[835] Zhanwen Chen, Saed Rezayi, and Sheng Li. More knowledge, less bias: Unbiasing scene graph generation with explicit ontological adjustment. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4023–4032, 2023.

[836] Guikun Chen, Jin Li, and Wenguan Wang. Scene graph generation with role-playing large language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:132238–132266, 2024.

[837] Kibum Kim, Kanghoon Yoon, Jaehyeong Jeon, Yeonjun In, Jinyoung Moon, Donghyun Kim, and Chanyoung Park. Llm4sgg: Large language models for weakly supervised scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28306–28316, 2024.

[838] Mohammad Amin Sadeghi and Ali Farhadi. Recognition using visual phrases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[839] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

[840] Yuanzhi Liang, Yalong Bai, Wei Zhang, Xueming Qian, Li Zhu, and Tao Mei. Vrr-vg: Refocusing visually-relevant relationships. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10403–10412, 2019.

[841] Julia Peyre, Ivan Laptev, Cordelia Schmid, and Josef Sivic. Weakly-supervised learning of visual relations. In *Proceedings of the IEEE international conference on computer vision*, pages 5179–5188, 2017.

[842] Yansheng Li, Kun Li, Yongjun Zhang, Linlin Wang, and Dingwen Zhang. Aug: A new dataset and an efficient model for aerial image urban scene graph generation. *arXiv preprint arXiv:2404.07788*, 2024.

[843] Yansheng Li, Linlin Wang, Tingzhu Wang, Xue Yang, Junwei Luo, Qi Wang, Youming Deng, Wenbin Wang, Xian Sun, Haifeng Li, et al. Star: A first-ever dataset and a large-scale benchmark for scene graph generation in large-size satellite imagery. *IEEE Trans. Pattern Anal. Mach. Intell*, 47(3):1832–1849, 2025.

[844] Qiwei Yan, Chubo Deng, Chenglong Liu, Zhongyan Hou, Xiaorui Liu, Yi Jiang, Wanxuan Lu, Fanglong Yao, Xiaoyu Liu, Lingxiang Hao, et al. Recon1m: A large-scale benchmark dataset for relation comprehension in remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 2025.

[845] Junwei Luo, Zhen Pang, Yongjun Zhang, Tingzhu Wang, Linlin Wang, Bo Dang, Jiangwei Lao, Jian Wang, Jingdong Chen, Yihua Tan, et al. Skysensegpt: A fine-grained instruction tuning dataset and model for remote sensing vision-language understanding. *arXiv preprint arXiv:2406.10100*, 2024.

[846] Xindi Shang, Tongwei Ren, Jingfan Guo, Hanwang Zhang, and Tat-Seng Chua. Video visual relation detection. In *ACM International Conference on Multimedia*, Mountain View, CA USA, October 2017.

[847] Xindi Shang, Donglin Di, Junbin Xiao, Yu Cao, Xun Yang, and Tat-Seng Chua. Annotating objects and relations in user-generated videos. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, pages 279–287, 2019.

[848] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10236–10247, 2020.

[849] Trong-Thuan Nguyen, Pha Nguyen, Xin Li, Jackson Cothren, Alper Yilmaz, and Khoa Luu. Cyclo: Cyclic graph transformer approach to multi-object relationship modeling in aerial videos. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:90355–90383, 2024.

[850] Trong-Thuan Nguyen, Pha Nguyen, and Khoa Luu. Hig: Hierarchical interlacement graph approach to scene graph generation in video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18384–18394, 2024.

[851] Ivan Rodin, Antonino Furnari, Kyle Min, Subarna Tripathi, and Giovanni Maria Farinella. Action scene graphs for long-form understanding of egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18622–18632, 2024.

[852] Jingkang Yang, Jun Cen, Wenxuan Peng, Shuai Liu, Fangzhou Hong, Xiangtai Li, Kaiyang Zhou, Qifeng Chen, and Ziwei Liu. 4d panoptic scene graph generation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:69692–69705, 2023.

[853] Ege Özsoy, Evin Pınar Örnek, Ulrich Eck, Tobias Czempiel, Federico Tombari, and Nassir Navab. 4d-or: Semantic scene graphs for or domain modeling. In *International conference on medical image computing and computer-assisted intervention*, pages 475–485, 2022.

[854] Ege Özsoy, Chantal Pellegrini, Tobias Czempiel, Felix Tristram, Kun Yuan, David Bani-Harouni, Ulrich Eck, Benjamin Busam, Matthias Keicher, and Nassir Navab. Mm-or: A large multimodal operating room dataset for semantic understanding of high-intensity surgical environments. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19378–19389, 2025.

[855] Ege Özsoy, Arda Mamur, Felix Tristram, Chantal Pellegrini, Magdalena Wysocki, Benjamin Busam, and Nassir Navab. Egoexor: An ego-exo-centric operating room dataset for surgical activity understanding. *Advances in Neural Information Processing Systems*, 38, 2025.

[856] Dongil Yang, Minjin Kim, Sunghwan Kim, Beong-woo Kwak, Minjun Park, Jinseok Hong, Woontack Woo, and Jinyoung Yeo. Llm meets scene graph: Can large language models understand and generate scene graphs? a benchmark and empirical study. *arXiv preprint arXiv:2505.19510*, 2025.

[857] Shaoqing Lin, Chong Teng, Fei Li, Donghong Ji, Lizhen Qu, and Zhuang Li. Discosg: Towards discourse-level text scene graph parsing through iterative graph refinement. *arXiv preprint arXiv:2506.15583*, 2025.

[858] Weixiang Yang, Qi Li, Wenxi Liu, Yuanlong Yu, Yuexin Ma, Shengfeng He, and Jia Pan. Projecting your view attentively: Monocular road scene layout estimation via cross-view transformation. In *Proc. CVPR*, pages 15536–15545, 2021.

[859] Varsha Hedau, Derek Hoiem, and David A. Forsyth. Recovering the spatial layout of cluttered rooms. In *Proc. ICCV*, pages 1849–1856, 2009.

[860] Carsten Rother. A new approach to vanishing point detection in architectural environments. *Image Vis. Comput.*, 20(9-10):647–655, 2002.

[861] Arun Mallya and Svetlana Lazebnik. Learning informative edge maps for indoor scene layout prediction. In *Proc. ICCV*, pages 936–944, 2015.

[862] Chen-Yu Lee, Vijay Badrinarayanan, Tomasz Malisiewicz, and Andrew Rabinovich. RoomNet: End-to-end room layout estimation. In *Proc. ICCV*, pages 4865–4874, 2017.

[863] Chuhang Zou, Alex Colburn, Qi Shan, and Derek Hoiem. Layoutnet: Reconstructing the 3d room layout from a single RGB image. In *Proc. CVPR*, pages 2051–2059, 2018.

[864] Cheng Sun, Chi-Wei Hsiao, Min Sun, and Hwann-Tzong Chen. Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[865] Cheng Sun, Min Sun, and Hwann-Tzong Chen. HoHoNet: 360 indoor holistic understanding with latent horizontal features. In *Proc. CVPR*, pages 2573–2582, 2021.

[866] Fu-En Wang, Yu-Hsuan Yeh, Min Sun, Wei-Chen Chiu, and Yi-Hsuan Tseng. LED2-Net: Monocular 360 layout estimation via differentiable depth rendering. In *Proc. CVPR*, pages 12956–12965, 2021.

[867] Shang-Ta Yang, Fu-En Wang, Chi-Han Peng, Peter Wonka, Min Sun, and Hung-Kuo Chu. DuLa-Net: A dual-projection network for estimating room layouts from a single RGB panorama. In *Proc. CVPR*, pages 3363–3372, 2019.

[868] Giovanni Pintore, Marco Agus, and Enrico Gobbetti. AtlantaNet: Inferring the 3d indoor layout from a single 360 image beyond the Manhattan world assumption. In *Proc. ECCV*, pages 432–448, 2020.

[869] Weidong Zhang, Wei Zhang, and Yinda Zhang. Geolayout: Geometry driven room layout estimation based on depth maps of planes. In *Proc. ECCV*, volume 12361, pages 632–648, 2020.

[870] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[871] Shichao Yang, Daniel Maturana, and Sebastian A. Scherer. Real-time 3d scene layout from a single image using convolutional neural networks. In *Proc. ICRA*, pages 2183–2189, 2016.

[872] Axel Furlan, Stephen D. Miller, Domenico G. Sorrenti, Li Fei-Fei, and Silvio Savarese. Free your camera: 3d indoor scene understanding from arbitrary camera motion. In *Proc. BMVC*, 2013.

[873] Grace Tsai, Changhai Xu, Jingen Liu, and Benjamin Kuipers. Real-time indoor scene understanding using bayesian filtering with motion cues. In *Proc. ICCV*, pages 121–128, 2011.

[874] Yinda Zhang, Shuran Song, Ping Tan, and Jianxiong Xiao. PanoContext: A whole-room 3d context model for panoramic scene understanding. In *Proc. ECCV*, pages 668–686, 2014.

[875] Iro Armeni, Sasha Sax, Amir R. Zamir, and Silvio Savarese. Joint 2D-3D-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

[876] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 27, 2014.

[877] Ali Diba, Mohsen Fayyaz, Vivek Sharma, Amir Hossein Karami, Mohammad Mahdi Arzani, Rahman Yousefzadeh, and Luc Van Gool. Temporal 3d convnets: New architecture and transfer learning for video classification. *arXiv preprint arXiv:1711.08200*, 2017.

[878] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018.

[879] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017.

[880] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36, 2016.

[881] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 244–253, 2019.

[882] Jiewen Yang, Xingbo Dong, Liujun Liu, Chao Zhang, Jiajun Shen, and Dahai Yu. Recurring the transformer for video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14063–14073, 2022.

[883] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2914–2923, 2017.

[884] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[885] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weakly-supervised temporal action localization in untrimmed videos. In *Proceedings of the european conference on computer vision (ECCV)*, pages 154–171, 2018.

[886] Can Zhang, Meng Cao, Dongming Yang, Jie Chen, and Yuexian Zou. Cola: Weakly-supervised temporal action localization with snippet contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16010–16019, 2021.

[887] Sujoy Paul, Sourya Roy, and Amit K Roy-Chowdhury. W-talc: Weakly-supervised temporal activity localization and classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 563–579, 2018.

[888] Ziying Xia, Jian Cheng, Siyu Liu, Yongxiang Hu, Shiguang Wang, Yijie Zhang, and Liwan Dang. Realigning confidence with temporal saliency information for point-level weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18440–18450, 2024.

[889] Pilhyeon Lee and Hyeran Byun. Learning action completeness from points for weakly-supervised temporal action localization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13648–13657, 2021.

[890] Jie Fu, Junyu Gao, and Changsheng Xu. Compact representation and reliable classification learning for point-level weakly-supervised action localization. *IEEE Transactions on Image Processing*, 31:7363–7377, 2022.

[891] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3628–3636, 2017.

[892] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1049–1058, 2016.

[893] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 988–996, 2017.

[894] Chuming Lin, Chengming Xu, Donghao Luo, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Learning salient boundary feature for anchor-free temporal action localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3320–3329, 2021.

[895] Yuanjun Xiong, Yue Zhao, Limin Wang, Dahua Lin, and Xiaoou Tang. A pursuit of temporal accuracy in general activity detection. *arXiv preprint arXiv:1703.02716*, 2017.

[896] Guoqiang Gong, Liangfeng Zheng, Kun Bai, and Yadong Mu. Scale matters: Temporal scale aggregation network for precise action localization in untrimmed videos. *arXiv preprint arXiv:1908.00707*, 2019.

[897] Yaosen Chen, Bing Guo, Yan Shen, Wei Wang, Weichen Lu, and Xinhua Suo. Boundary graph convolutional network for temporal action detection. *Image and Vision Computing*, 109:104144, 2021.

[898] Jiannan Wu, Peize Sun, Shoufa Chen, Jiewen Yang, Zihao Qi, Lan Ma, and Ping Luo. Towards high-quality temporal action detection with sparse proposals. *arXiv preprint arXiv:2109.08847*, 2021.

[899] Dingfeng Shi, Yujie Zhong, Qiong Cao, Jing Zhang, Lin Ma, Jia Li, and Dacheng Tao. React: Temporal action detection with relational queries. In *European conference on computer vision*, pages 105–121, 2022.

[900] Jihwan Kim, Miso Lee, and Jae-Pil Heo. Self-feedback detr for temporal action detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10286–10296, 2023.

[901] Yuhan Zhu, Guozhen Zhang, Jing Tan, Gangshan Wu, and Limin Wang. Dual detrs for multi-label temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18559–18569, 2024.

[902] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[903] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International conference on computer vision*, pages 2556–2563, 2011.

[904] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6047–6056, 2018.

[905] Yu-Gang Jiang, Jingen Liu, A Roshan Zamir, George Toderici, Ivan Laptev, Mubarak Shah, and Rahul Sukthankar. Thumos challenge: Action recognition with a large number of classes, 2014.

[906] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015.

[907] Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan. Hacs: Human action clips and segments dataset for recognition and temporal localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8668–8678, 2019.

[908] Yi Liu, Limin Wang, Yali Wang, Xiao Ma, and Yu Qiao. Fineaction: A fine-grained video dataset for temporal action localization. *IEEE transactions on image processing*, 31:6937–6950, 2022.

[909] Serena Yeung, Olga Russakovsky, Ning Jin, Mykhaylo Andriluka, Greg Mori, and Li Fei-Fei. Every moment counts: Dense detailed labeling of actions in complex videos. *International Journal of Computer Vision*, 126(2):375–389, 2018.

[910] Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *European conference on computer vision*, pages 510–526, 2016.

[911] Roeland De Geest, Efstratios Gavves, Amir Ghodrati, Zhenyang Li, Cees Snoek, and Tinne Tuytelaars. Online action detection. In *European Conference on Computer Vision*, pages 269–284, 2016.

[912] Hilde Kuehne, Ali Arslan, and Thomas Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 780–787, 2014.

[913] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 729–738, 2013.

[914] Marcus Rohrbach, Anna Rohrbach, Michaela Regneri, Sikandar Amin, Mykhaylo Andriluka, Manfred Pinkal, and Bernt Schiele. Recognizing fine-grained and composite activities using hand-centric features and script data. *International Journal of Computer Vision*, 119(3):346–373, 2016.

[915] Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1207–1216, 2019.

[916] Kellie Corona, Katie Osterdahl, Roderic Collins, and Anthony Hoogs. Meva: A large-scale multiview, multimodal video dataset for activity detection. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1060–1068, 2021.

[917] Saeed Ghorbani, Kimia Mahdaviani, Anne Thaler, Konrad Kording, Douglas James Cook, Gunnar Blohm, and Nikolaus F Troje. Movi: A large multi-purpose human motion and video dataset. *Plos one*, 16(6):e0253157, 2021.

[918] Zhihao Zhao, Reza Ghoddoosian, Isht Dwivedi, Nakul Agarwal, and Behzad Dariush. Pose-aware weakly-supervised action segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 97–107, 2025.

[919] Reza Ghoddoosian, Isht Dwivedi, Nakul Agarwal, and Behzad Dariush. Weakly-supervised action segmentation and unseen error detection in anomalous instructional videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10128–10138, 2023.

[920] Yuchen Zhou, Linkai Liu, and Chao Gou. Learning from observer gaze: Zero-shot attention prediction oriented by human-object interaction recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28390–28400, 2024.

[921] Wentao Bao, Kai Li, Yuxiao Chen, Deep Patel, Martin Renqiang Min, and Yu Kong. Exploiting vlm localizability and semantics for open vocabulary action detection. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 8291–8301, 2025.

[922] Linkai Liu, Yuchen Zhou, Zipeng Guo, Lei Yu, and Chao Gou. Synergistic audio-textual cues: A cross-modal framework for weakly-supervised temporal action localization. *Pattern Recognition*, page 113526, 2026.

[923] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J. Davison. iMAP: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6229–6238, 2021.

[924] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, and Marc Pollefeys. NICE-SLAM: Neural implicit scalable encoding for SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12786–12796, 2022.

[925] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. SplaTAM: Splat, track & map 3D gaussians for dense RGB-D SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21357–21366, 2024.

[926] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. GS-SLAM: Dense visual SLAM with 3D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19595–19604, 2024.

[927] Hidenobu Matsuki, Riku Murai, Paul H. J. Kelly, and Andrew J. Davison. Monocular 3D gaussian splatting SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[928] Ignacio Vizzo, Tiziano Guadagnino, Benedikt Mersch, Louis Wiesmann, Jens Behley, and Cyrill Stachniss. KISS-ICP: In defense of point-to-point ICP—simple, accurate, and robust registration if done the right way. *IEEE Robotics and Automation Letters*, 8(2):1029–1036, 2023.

[929] Tiziano Guadagnino, Benedikt Mersch, Ignacio Vizzo, Simon Schaupp, Jens Behley, Martin Stumberg, Wolfram Burgard, and Luca Carlone. KISS-SLAM: A simple, robust, and accurate 3D LiDAR SLAM system with enhanced generalization capabilities. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

[930] Kunping Huang, Sen Zhang, Jingwen Zhang, and Dacheng Tao. Event-based simultaneous localization and mapping: A comprehensive survey. *International Journal of Computer Vision*, 134(5):245, 2026.

[931] Riku Murai, Eric Dexheimer, and Andrew J. Davison. MASt3R-SLAM: Real-time dense SLAM with 3D reconstruction priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[932] Andrzej Pronobis and Barbara Caputo. COLD: the cosy localization database. *Int. J. Robotics Res.*, 28 (5):588–594, 2009.

[933] Ankur Handa, Thomas Whelan, John McDonald, and Andrew J. Davison. A benchmark for RGB-D visual odometry, 3d reconstruction and SLAM. In *Proc. ICRA*, pages 1524–1531, 2014.

[934] John McCormac, Ankur Handa, Stefan Leutenegger, and Andrew J. Davison. Scenenet RGB-D: can 5m synthetic images beat generic imagenet pre-training on indoor segmentation? In *Proc. ICV*, pages 2697–2706, 2017.

[935] D. Schubert, T. Goll, N. Demmel, V. Usenko, J. Stückler, and D. Cremers. The tum vi benchmark for evaluating visual-inertial odometry. In *Proc. IROS*, pages 1680–1687, 2018.

[936] Emanuele Palazzolo, Jens Behley, Philipp Lottes, Philippe Giguère, and Cyrill Stachniss. Refusion: 3d reconstruction in dynamic environments for RGB-D cameras exploiting residuals. In *Proc. IROS*, pages 7855–7862, 2019.

[937] Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. In *Proc. BMVC*, page 77, 2018.

[938] Xuesong Shi, Dongjiang Li, Pengpeng Zhao, Qinbin Tian, Yuxin Tian, Qiwei Long, Chunhao Zhu, Jingwei Song, Fei Qiao, Le Song, Yangquan Guo, Zhigang Wang, Yimin Zhang, Baoxing Qin, Wei Yang,

Fangshi Wang, Rosa H. M. Chan, and Qi She. Are we ready for service robots? the openloris-scene datasets for lifelong SLAM. In *Proc. ICRA*, pages 3139–3145, 2020.

[939] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023.

[940] ETH3D. ETH3D SLAM Benchmark. https://www.eth3d.net/slam_benchmark, 2018.

[941] Mathias Gehrig, Willem Aarents, Daniel Gehrig, and Davide Scaramuzza. DSEC: A stereo event camera dataset for driving scenarios. *IEEE Robotics and Automation Letters*, 6(3):4947–4954, 2021.

[942] Doncey Albin, Miles Mena, Annika Thomas, Harel Biggie, Xuefei Sun, Dusty Woods, Steve McGuire, and Christoffer Heckman. Cu-multi: A dataset for multi-robot data association. *arXiv preprint arXiv:2505.17576*, 2025.

[943] Xiaowei Li, Kuan Xu, Fen Liu, Ruofei Bai, Shenghai Yuan, and Lihua Xie. Airswarm: Enabling cost-effective multi-uav research with cots drones. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 21487–21494, 2025.

[944] Sundar Vedula, Simon Baker, Peter Rander, Robert Collins, and Takeo Kanade. Three-dimensional scene flow. In *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV), Kerkyra, Greece, September 20-27, 1999*, pages 722–729, 1999.

[945] Frederic Huguet and Frederic Devernay. A variational method for scene flow estimation from stereo sequences. In *2007 IEEE 11th International Conference on Computer Vision (ICCV), Rio de Janeiro, Brazil, October 14-21, 2007*, pages 1–7, 2007.

[946] Andreas Wedel, Thomas Brox, Tobi Vaudrey, Clemens Rabe, Uwe Franke, and Daniel Cremers. Stereoscopic scene flow computation for 3d motion understanding. *International Journal of Computer Vision*, 95(1):29–51, 2011.

[947] Tali Basha, Yael Moses, and Nahum Kiryati. Multi-view scene flow estimation: A view centered variational approach. *International Journal of Computer Vision*, 101(1):6–21, 2013.

[948] Christoph Vogel, Konrad Schindler, and Stefan Roth. 3d scene flow estimation with a piecewise rigid scene model. *International Journal of Computer Vision*, 115(1):1–28, 2015.

[949] Huaizu Jiang, Deqing Sun, Varun Jampani, Zhaoyang Lv, Erik Learned-Miller, and Jan Kautz. SENSE: A shared encoder network for scene-flow estimation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), October 27 - November 2, 2019*, pages 3195–3204, 2019.

[950] Wei-Chiu Ma, Shenlong Wang, Rui Hu, Yuwen Xiong, and Raquel Urtasun. Deep rigid instance scene flow. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, June 15-20, 2019*, pages 3614–3622, 2019.

[951] Zachary Teed and Jia Deng. RAFT-3D: Scene flow using rigid-motion embeddings. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual, June 19-25, 2021*, pages 8375–8384, 2021.

[952] Fabian Brickwedde, Steffen Abraham, and Rudolf Mester. Mono-SF: Multi-view geometry meets single-view depth for monocular scene flow estimation of dynamic objects. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), October 27 - November 2, 2019*, pages 2780–2790, 2019.

[953] Junhwa Hur and Stefan Roth. Self-supervised monocular scene flow estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, June 13-19, 2020*, pages 7396–7405, 2020.

[954] Xueqian Li, Jhony Kaesemodel Pontes, and Simon Lucey. Neural scene flow prior. In *Advances in Neural Information Processing Systems (NeurIPS) 34*, pages 7838–7851, 2021.

[955] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, June 7-12, 2015*, pages 3061–3070, 2015.

[956] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012.

[957] Lukas Mehl, Jenny Schmalfuss, Azin Jahedi, Yaroslava Nalivayko, and Andrés Bruhn. Spring: A high-resolution high-detail dataset and benchmark for scene flow, optical flow and stereo. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2023.

[958] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[959] Chenxu Luo, Xiaodong Yang, and Alan Yuille. Exploring simple 3d multi-object tracking for autonomous driving. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10488–10497, 2021.

[960] Hsu-kuang Chiu, Jie Li, Rareş Ambruş, and Jeannette Bohg. Probabilistic 3d multi-modal, multi-object tracking for autonomous driving. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 14227–14233, 2021.

[961] Shipra Ojha and Sachin Sakhare. Image processing techniques for object tracking in video surveillance-a survey. In *2015 International Conference on Pervasive Computing (ICPC)*, pages 1–6, 2015.

[962] Matt Deitke, Winson Han, Alvaro Herrasti, Aniruddha Kembhavi, Eric Kolve, Roozbeh Mottaghi, Jordi Salvador, Dustin Schwenk, Eli VanderBilt, Matthew Wallingford, et al. Robothor: An open simulation-to-real embodied ai platform. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3164–3174, 2020.

[963] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5374–5383, 2019.

[964] Matthias Muller, Adel Bibi, Silvio Giancola, Salman Alsubaihi, and Bernard Ghanem. Trackingnet: A large-scale dataset and benchmark for object tracking in the wild. In *Proceedings of the European conference on computer vision (ECCV)*, pages 300–317, 2018.

[965] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016.

[966] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020.

[967] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 7942–7951, 2019.

[968] Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. AB3DMOT: A baseline for 3d multi-object tracking and new evaluation metrics. *CoRR*, abs/2008.08063, 2020.

[969] Umar Iqbal, Anton Milan, and Juergen Gall. Posetrack: Joint multi-person pose estimation and tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2011–2020, 2017.

[970] Soumitra Samanta and Bhabatosh Chanda. A data-driven approach for human pose tracking based on spatio-temporal pictorial structure. *arXiv preprint arXiv:1608.00199*, 2016.

[971] Stefanos Zafeiriou, Grigorios G Chrysos, Anastasios Roussos, Evangelos Ververas, Jiankang Deng, and George Trigeorgis. The 3d menpo facial landmark tracking challenge. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 2503–2511, 2017.

[972] Daniel Stadler and Jurgen Beyerer. Improving multiple pedestrian tracking by track management and occlusion handling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10958–10967, 2021.

[973] J Van Leuven, MB Van Leeuwen, and FCA Groen. Real-time vehicle tracking in image sequences. In *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics (Cat. No. 01CH 37188)*, pages 2049–2054, 2001.

[974] Libo Zhang, Junyuan Gao, Zhen Xiao, and Heng Fan. Animaltrack: A benchmark for multi-animal tracking in the wild. *International Journal of Computer Vision*, 131(2):496–513, 2023.

[975] David S Bolme, J Ross Beveridge, Bruce A Draper, and Yui Man Lui. Visual object tracking using adaptive correlation filters. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 2544–2550, 2010.

[976] João F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. High-speed tracking with kernelized correlation filters. *IEEE transactions on pattern analysis and machine intelligence*, 37(3):583–596, 2014.

[977] Ming Tang and Jiayi Feng. Multi-kernel correlation filter for visual tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3038–3046, 2015.

[978] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ECO: efficient convolution operators for tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 6931–6939, 2017.

[979] Hyeonseob Nam and Bohyung Han. Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4293–4302, 2016.

[980] Luca Bertinetto, Jack Valmadre, João F. Henriques, Andrea Vedaldi, and Philip H. S. Torr. Fully-convolutional siamese networks for object tracking. In Gang Hua and Hervé Jégou, editors, *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II*, pages 850–865, 2016.

[981] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 8971–8980, 2018.

[982] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. ATOM: accurate tracking by overlap maximization. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 4660–4669, 2019.

[983] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *2019 IEEE/CVF International Conference on Computer Vision, Proceedings of theInternational Conference on Computer Vision (ICCV) 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 6181–6190, 2019.

[984] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 8126–8135, 2021.

[985] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *2021 IEEE/CVF International Conference on Computer Vision, Proceedings of theInternational Conference on Computer Vision (ICCV) 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 10428–10437, 2021.

[986] Botao Ye, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Joint feature learning and relation modeling for tracking: A one-stream framework. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXII*, pages 341–357, 2022.

[987] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[988] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. Youtube-vos: Sequence-to-sequence video object segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 585–601, 2018.

[989] Ho Kei Cheng and Alexander G. Schwing. Xmem: Long-term video object segmentation with an atkinson-shiffrin memory model. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXVIII*, pages 640–658, 2022.

[990] Zongxin Yang, Yunchao Wei, and Yi Yang. Associating objects with transformers for video object segmentation. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 2491–2502, 2021.

[991] Zongxin Yang and Yi Yang. Decoupling features in hierarchical propagation for video object segmentation. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[992] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468, 2016.

[993] Harold W. Kuhn. The hungarian method for the assignment problem. In *50 Years of Integer Programming 1958-2008 - From the Early Years to the State-of-the-Art*, pages 29–47, 2010.

[994] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing, ICIP 2017, Beijing, China, September 17-20, 2017*, pages 3645–3649, 2017.

[995] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, and Shengjin Wang. Towards real-time multi-object tracking. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XI*, pages 107–122, 2020.

[996] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *Int. J. Comput. Vis.*, 129(11):3069–3087, 2021.

[997] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXII*, pages 1–21, 2022.

[998] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part IV*, pages 474–490, 2020.

[999] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixé, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 8834–8844, 2022.

[1000] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European conference on computer vision*, pages 659–675, 2022.

[1001] Peize Sun, Jinkun Cao, Yi Jiang, Zehuan Yuan, Song Bai, Kris Kitani, and Ping Luo. Dancetrack: Multi-object tracking in uniform appearance and diverse motion. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 20961–20970, 2022.

[1002] Zhongdao Wang, Hengshuang Zhao, Ya-Li Li, Shengjin Wang, Philip Torr, and Luca Bertinetto. Do different tracking tasks require different appearance models? *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:726–738, 2021.

[1003] Bin Yan, Yi Jiang, Peize Sun, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Towards grand unification of object tracking. In *European conference on computer vision*, pages 733–751, 2022.

[1004] Fan Ma, Mike Zheng Shou, Linchao Zhu, Haoqi Fan, Yilei Xu, Yi Yang, and Zhicheng Yan. Unified transformer tracker for object tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 8771–8780, 2022.

[1005] Minghan Li, Shuai Li, Xindong Zhang, and Lei Zhang. Univs: Unified and universal video segmentation with prompts as queries. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3227–3238, 2024.

[1006] Lingyi Hong, Shilin Yan, Renrui Zhang, Wanyun Li, Xinyu Zhou, Pinxue Guo, Kaixun Jiang, Yiting Chen, Jinglun Li, Zhaoyu Chen, and Wenqiang Zhang. Onetracker: Unifying visual object tracking with foundation models and efficient tuning. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 19079–19091, 2024.

[1007] He Wang, Tianyang Xu, Zhangyong Tang, Xiaojun Wu, and Josef Kittler. Uastrack: A unified adaptive selection framework with modality-customization in single object tracking. *CoRR*, abs/2502.18220, 2025.

[1008] Zongwei Wu, Jilai Zheng, Xiangxuan Ren, Florin-Alexandru Vasluianu, Chao Ma, Danda Pani Paudel, Luc Van Gool, and Radu Timofte. Single-model and any-modality for video object tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 19156–19166, 2024.

[1009] Xiao Wang, Xiujun Shu, Zhipeng Zhang, Bo Jiang, Yaowei Wang, Yonghong Tian, and Feng Wu. Towards more flexible and accurate object tracking with natural language: Algorithms and benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13763–13773, June 2021.

[1010] Yunhao Li, Xiaoqiong Liu, Luke Liu, Heng Fan, and Libo Zhang. Lamot: Language-guided multi-object tracking. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6816–6822, 2025.

[1011] Robert Collins, Xuhui Zhou, and Seng Keat Teh. An open source tracking testbed and evaluation web site. In *IEEE international workshop on performance evaluation of tracking and surveillance*, page 35, 2005.

[1012] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2411–2418, 2013.

[1013] Arnold WM Smeulders, Dung M Chu, Rita Cucchiara, Simone Calderara, Afshin Dehghan, and Mubarak Shah. Visual tracking: An experimental survey. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1442–1468, 2013.

[1014] Pengpeng Liang, Erik Blasch, and Haibin Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE transactions on image processing*, 24(12):5630–5644, 2015.

[1015] Giorgio Roffo, Simone Melzi, et al. The visual object tracking vot2016 challenge results. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II*, pages 777–823, 2016.

[1016] U Benchmark. A benchmark and simulator for uav tracking. In *European conference on computer vision*, volume 7, 2016.

[1017] A Li, M Lin, Y Wu, MH Yang, and S Yan. NUS-PRO: A New Visual Tracking Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):335–349, 2016.

[1018] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, Roman P. Pflugfelder, Luka Cehovin Zajc, Tomas Vojir, Gustav Häger, Alan Lukezic, Abdelrahman Eldesokey, Gustavo Fernández, Álvaro García-Martín, Andrej Muhic, Alfredo Petrosino, Alireza Memarmoghadam, Andrea Vedaldi, Antoine Manzanera, Antoine Tran, A. Aydin Alatan, Bogdan Mocanu, Boyu Chen, Chang Huang, Changsheng Xu, Chong Sun, Dalong Du, David Zhang, Dawei Du, Deepak Mishra, Erhan Gundogdu, Erik Velasco-Salido, Fahad Shahbaz Khan, Francesco Battistone, Gorthi R. K. Sai Subrahmanyam, Goutam Bhat, Guan Huang, Guilherme Sousa Bastos, Guna Seetharaman, Hongliang Zhang, Houqiang Li, Huchuan Lu, Isabela Drummond, Jack Valmadre, Jae-chan Jeong, Jaeil Cho, Jae-Yeong Lee, Jana Noskova, Jianke Zhu, Jin Gao, Jingyu Liu, Ji-Wan Kim, João F. Henriques, José M. Martínez, Junfei Zhuang, Junliang Xing, Junyu Gao, Kai Chen, Kannappan Palaniappan, Karel Lebeda, Ke Gao, Kris M. Kitani, Lei Zhang, Lijun Wang, Lingxiao Yang, Longyin Wen, Luca Bertinetto, Mahdieh Poostchi, Martin Danelljan, Matthias Mueller, Mengdan Zhang, Ming-Hsuan Yang, Nianhao Xie, Ning Wang, Ondrej Miksik, Payman Moallem, Pallavi M. Venugopal, Pedro Senna, Philip H. S. Torr, Qiang Wang, Qifeng Yu, Qingming Huang, Rafael Martin Nieto, Richard Bowden, Risheng Liu, Ruxandra Tapu, Simon Hadfield, Siwei Lyu, Stuart Golodetz, Sunglok Choi, Tianzhu Zhang, Titus Zaharia, Vincenzo Santopietro, Wei Zou, Weiming Hu, Wenbing Tao, Wenbo Li, Wengang Zhou, Xianguo Yu, Xiao Bian, Yang Li, Yifan Xing, Yingruo Fan, Zheng Zhu, Zhipeng Zhang, and Zhiqun He. The visual object tracking VOT2017 challenge results. In *2017 IEEE International Conference on Computer Vision Workshops, Proceedings of theInternational Conference on Computer Vision (ICCV) Workshops 2017, Venice, Italy, October 22-29, 2017*, pages 1949–1972, 2017.

[1019] Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, and Vincent Vanhoucke. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5296–5305, 2017.

[1020] Hamed Kiani Galoogahi, Ashton Fagg, Chen Huang, Deva Ramanan, and Simon Lucey. Need for speed: A benchmark for higher frame rate object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1125–1134, 2017.

[1021] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1562–1577, 2021.

[1022] Shuran Song and Jianxiong Xiao. Tracking revisited using rgbd camera: Unified benchmark and baselines. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 233–240, 2013.

[1023] Jingjing Xiao, Rustam Stolkin, Yuqing Gao, and Aleš Leonardis. Robust fusion of color and depth data for rgb-d target tracking using adaptive range-invariant depth models and spatio-temporal consistency constraints. *IEEE transactions on cybernetics*, 48(8):2485–2499, 2017.

[1024] Alan Lukezic, Ugur Kart, Jani Kapyla, Ahmed Durmush, Joni-Kristian Kamarainen, Jiri Matas, and Matej Kristan. Cdtb: A color and depth visual object tracking dataset and benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10013–10022, 2019.

[1025] Song Yan, Jinyu Yang, Jani Käpylä, Feng Zheng, Aleš Leonardis, and Joni-Kristian Kämäräinen. Depthtrack: Unveiling the power of rgbd tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10725–10733, 2021.

[1026] Haojie Zhao, Junsong Chen, Lijun Wang, and Huchuan Lu. Arkittrack: a new diverse dataset for tracking using mobile rgb-d data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5126–5135, 2023.

[1027] Xue-Feng Zhu, Tianyang Xu, Zhangyong Tang, Zucheng Wu, Haodong Liu, Xiao Yang, Xiao-Jun Wu, and Josef Kittler. Rgbd1k: A large-scale dataset and benchmark for rgb-d object tracking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3870–3878, 2023.

[1028] Benjamin Sapp, David Weiss, and Ben Taskar. Parsing human motion with stretchable models. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1281–1288, 2011.

[1029] Anoop Cherian, Julien Mairal, Karteek Alahari, and Cordelia Schmid. Mixing body-part sequences for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2353–2360, 2014.

[1030] H Joo, T Simon, X Li, H Liu, L Tan, L Gui, S Banerjee, T Godisart, B Nabbe, I Matthews, et al. Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):190–204, 2017.

[1031] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008(1):246309, 2008.

[1032] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision*, 129(2):548–578, 2021.

[1033] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[1034] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[1035] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[1036] Zheng Shou, Junting Pan, Jonathan Chan, Kazuyuki Miyazawa, Hassan Mansour, Anthony Vetro, Xavier Giro-i Nieto, and Shih-Fu Chang. Online detection of action start in untrimmed, streaming videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[1037] Mingfei Gao, Mingze Xu, Larry S. Davis, Richard Socher, and Caiming Xiong. Startnet: Online detection of action start in untrimmed videos. *arXiv preprint arXiv:1903.09868*, 2019.

[1038] Minuk Ma, Sunjae Yoon, Junyeong Kim, Youngjoon Lee, Sunghun Kang, and Chang D. Yoo. Vlanet: Video-language alignment network for weakly-supervised video moment retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[1039] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[1040] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[1041] Benedetta Liberatori, Alessandro Conti, Paolo Rota, Yiming Wang, and Elisa Ricci. Test-time zero-shot temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[1042] Jie Lei, Licheng Yu, Tamara L. Berg, and Mohit Bansal. Tvr: A large-scale dataset for video-subtitle moment retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[1043] Jie Lei, Tamara L. Berg, and Mohit Bansal. Qvhighlights: Detecting moments and highlights in videos via natural language queries. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

[1044] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2013.

[1045] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[1046] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Birkan Seybold, David A. Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster R-CNN architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[1047] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.

[1048] Min Xu, Chen Zhao, Dev Sharma Das, Daniel S. Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[1049] Chen-Lin Zhang, Jianxin Wu, and Yin Li. Actionformer: Localizing moments of actions with transformers. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[1050] Dingfeng Shi, Yujie Zhong, Qiong Cao, Lin Ma, Jia Li, and Dacheng Tao. Tridet: Temporal action detection with relative boundary modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[1051] Jing Tan, Jian Tang, Limin Wang, and Gang Wu. Relaxed transformer decoders for direct action proposal generation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[1052] Xiaolong Liu, Qimeng Wang, Yao Hu, Xu Tang, Shiwei Zhang, Song Bai, and Xiang Bai. End-to-end temporal action detection with transformer. *IEEE Transactions on Image Processing*, 2022.

[1053] Xiaolong Liu, Song Bai, and Xiang Bai. An empirical study of end-to-end temporal action detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[1054] Jihwan Kim, Miso Lee, and Jae-Pil Heo. Te-tad: Towards full end-to-end temporal action detection via time-aligned coordinate expression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[1055] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry S. Davis. Man: Moment alignment network for natural language moment retrieval via iterative graph adjustment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[1056] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[1057] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[1058] Jie Lei, Tamara L. Berg, and Mohit Bansal. Moment-detr: End-to-end transformer for moment retrieval and highlight detection. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

[1059] Michaela Regneri, Marcus Rohrbach, Dietrich Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Tacos: Saarbrücken corpus of textually annotated cooking scenes, 2013. Dataset with timestamped natural-language descriptions aligned to cooking videos.

[1060] Luowei Zhou, Chenliang Xu, and Jason J. Corso. Towards automatic learning of procedures from web instructional videos. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2018. Introduces the YouCook2 dataset.

[1061] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, and et al. Rescaling egocentric vision: Epic-kitchens-100. *arXiv preprint arXiv:2006.13256*, 2020.

[1062] ActivityNet Challenge Organizers. Activitynet challenge 2019: Temporal action proposals evaluation protocol. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. Official AR@AN and AUC metrics.

[1063] Ryo Togashi, Norimichi Nishida, Yasuhiro Nakashima, and Noboru Babaguchi. Axiou: An axiomatically justified measure for video moment retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[1064] Ruijie Tao, Zexu Pan, Rohan Kumar Das, Xinyuan Qian, Mike Zheng Shou, and Haizhou Li. Is someone speaking? exploring long-term temporal features for audio-visual active speaker detection. In *Proceedings of the 29th ACM International Conference on Multimedia (ACM MM)*, pages 3927–3935, 2021.

[1065] Archontis Politis, Kazuki Shimada, Parthasaarathy Sudarsanam, Sharath Adavanne, Daniel Krause, Yuichiro Koyama, Naoya Takahashi, Shusuke Takahashi, Yuki Mitsufuji, and Tuomas Virtanen. STARSS23: An audio-visual dataset of spatial recordings of real scenes with spatiotemporal annotations of sound events. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

[1066] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4358–4366, 2018.

[1067] Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Localizing visual sounds the hard way. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16867–16876, 2021.

[1068] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9248–9257, 2019.

[1069] Rui Qian, Di Hu, Heinrich Dinkel, Mengyue Wu, Ning Xu, and Weiyao Lin. Multiple sound sources localization from coarse to fine. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 292–308, 2020.

[1070] Shentong Mo and Yapeng Tian. A closer look at weakly-supervised audio-visual source localization. In *Advances in Neural Information Processing Systems (NeurIPS) 35*, pages 37524–37536, 2022.

[1071] Juan León Alcázar, Fabian Caba, Long Mai, Federico Perazzi, Joon-Young Lee, Pablo Arbeláez, and Bernard Ghanem. Active speakers in context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12465–12474, 2020.

[1072] Okan Köpüklü, Maja Taseska, and Gerhard Rigoll. How to design a three-stage architecture for audio-visual active speaker detection in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1193–1203, 2021.

[1073] Sharath Adavanne, Archontis Politis, Joonas Nikunen, and Tuomas Virtanen. Sound event localization and detection of overlapping sources using convolutional recurrent neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 13(1):34–48, 2018.

[1074] Kazuki Shimada, Yuichiro Koyama, Naoya Takahashi, Shusuke Takahashi, and Yuki Mitsufuji. ACCDOA: Activity-coupled cartesian direction of arrival representation for sound event localization

and detection. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 915–919, 2021.

[1075] Archontis Politis, Sharath Adavanne, and Tuomas Virtanen. Overview and evaluation of sound event localization and detection in DCASE 2019. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:684–698, 2020.

[1076] Charles Knapp and Glifford Carter. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(4):320–327, 1976.

[1077] Joseph H. DiBiase, Harvey F. Silverman, and Michael S. Brandstein. Robust localization in reverberant rooms. In Michael Brandstein and Darren Ward, editors, *Microphone Arrays: Signal Processing Techniques and Applications*, pages 157–180. 2001.

[1078] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGG-Sound: A large-scale audio-visual dataset. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725, 2020.

[1079] Jinxing Zhou, Jianyuan Wang, Jiayi Zhang, Weixuan Sun, Jing Zhang, Stan Birchfield, Dan Guo, Lingpeng Kong, Meng Wang, and Yiran Zhong. Audio–visual segmentation. In *European Conference on Computer Vision*, pages 386–403, 2022.

[1080] Joseph Roth, Sourish Chaudhuri, Ondrej Klimesch, Felix Lluis, Steven Genuit, Yu Tao, Pedro Trindade, and Caroline Zhu. AVA Active Speaker: An audio-visual dataset for active speaker detection. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4492–4496, 2020.

[1081] Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 247–263, 2018.

[1082] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85, 2016.

[1083] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016.

[1084] Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23592–23601, 2023.

[1085] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9579–9589, 2024.

[1086] Meng Luo, Shengqiong Wu, Liqiang Jing, Tianjie Ju, Li Zheng, Jinxiang Lai, Tianlong Wu, Xinya Du, Jian Li, Siyuan Yan, et al. Dr. v: A hierarchical perception-temporal-cognition framework to diagnose video hallucination by fine-grained spatial-temporal grounding. *International Journal of Computer Vision*, 134(6):278, 2026.

[1087] Qi Chen, Changli Wu, Jiayi Ji, Yiwei Ma, and Liujuan Cao. 3d-dres: Detailed 3d referring expression segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 2966–2974, 2026.

[1088] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014.

[1089] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2641–2649, 2015.

[1090] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the association for computational linguistics*, 2:67–78, 2014.

[1091] Yutao Hu, Qixiong Wang, Wenqi Shao, Enze Xie, Zhenguo Li, Jungong Han, and Ping Luo. Beyond one-to-one: Rethinking the referring image segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4067–4077, 2023.

[1092] Chenyun Wu, Zhe Lin, Scott Cohen, Trung Bui, and Subhransu Maji. Phrasecut: Language-based image segmentation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10216–10225, 2020.

[1093] Samuel Schulter, Yumin Suh, Konstantinos M Dafnis, Zhixing Zhang, Shiyu Zhao, Dimitris Metaxas, et al. Omnilabel: A challenging benchmark for language-based object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11953–11962, 2023.

[1094] Shengcao Cao, Zijun Wei, Jason Kuen, Kangning Liu, Lingzhi Zhang, Jiuxiang Gu, HyunJoon Jung, Liang-Yan Gui, and Yu-Xiong Wang. Refer to any segmentation mask group with vision-language prompts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21853–21863, 2025.

[1095] Runtao Liu, Chenxi Liu, Yutong Bai, and Alan L Yuille. Clevr-ref+: Diagnosing visual reasoning with referring expressions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4185–4194, 2019.

[1096] Donggon Jang, Yucheol Cho, Suin Lee, Taehyeon Kim, and Dae-Shik Kim. Mmr: A large-scale benchmark dataset for multi-target and multi-granularity reasoning segmentation. *arXiv preprint arXiv:2503.13881*, 2025.

[1097] Senqiao Yang, Tianyuan Qu, Xin Lai, Zhuotao Tian, Bohao Peng, Shu Liu, and Jiaya Jia. Lisa++: An improved baseline for reasoning segmentation with large language model. *arXiv preprint arXiv:2312.17240*, 2023.

[1098] Yunqiu Xu, Linchao Zhu, and Yi Yang. Mc-bench: A benchmark for multi-context visual grounding in the era of mllms. *arXiv preprint arXiv:2410.12332*, 2024.

[1099] Qihua Dong, Kuo Yang, Lin Ju, Handong Zhao, Yitian Zhang, Yizhou Wang, Huimin Zeng, Jianglin Lu, and Yun Fu. Ref-adv: Exploring mllm visual reasoning in referring expression tasks. *arXiv preprint arXiv:2602.23898*, 2026.

[1100] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. In *Asian conference on computer vision*, pages 123–141, 2018.

[1101] Kirill Gavrilyuk, Amir Ghodrati, Zhenyang Li, and Cees GM Snoek. Actor and action video segmentation from a sentence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5958–5966, 2018.

[1102] Henghui Ding, Chang Liu, Shuting He, Xudong Jiang, and Chen Change Loy. Mevis: A large-scale benchmark for video segmentation with motion expressions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2694–2703, 2023.

[1103] Cilin Yan, Haochen Wang, Shilin Yan, Xiaolong Jiang, Yao Hu, Guoliang Kang, Weidi Xie, and Efstratios Gavves. Visa: Reasoning video object segmentation via large language models. In *European Conference on Computer Vision*, pages 98–115, 2024.

[1104] Zechen Bai, Tong He, Haiyang Mei, Pichao Wang, Ziteng Gao, Joya Chen, Zheng Zhang, and Mike Zheng Shou. One token to seg them all: Language instructed reasoning segmentation in videos. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:6833–6859, 2024.

[1105] Anurag Bagchi, Zhipeng Bao, Yu-Xiong Wang, Pavel Tokmakov, and Martial Hebert. Refereverything: Towards segmenting everything we can speak of in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23221–23231, 2025.

[1106] Hao Fang, Runmin Cong, Xiankai Lu, Xiaofei Zhou, Sam Kwong, and Wei Zhang. Decoupled motion expression video segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13821–13831, 2025.

[1107] Yiming Zhang, ZeMing Gong, and Angel X Chang. Multi3drefer: Grounding text description to multiple 3d objects. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15225–15236, 2023.

[1108] Shuting He, Henghui Ding, Xudong Jiang, and Bihan Wen. Segpoint: Segment any point cloud via large language model. In *European Conference on Computer Vision*, pages 349–367, 2024.

[1109] Changli Wu, Yihang Liu, Jiayi Ji, Yiwei Ma, Haowei Wang, Gen Luo, Henghui Ding, Xiaoshuai Sun, and Rongrong Ji. 3d-gres: Generalized 3d referring expression segmentation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7852–7861, 2024.

[1110] Andrej Karpathy, Armand Joulin, and Li Fei-Fei. Deep fragment embeddings for bidirectional image sentence mapping. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 27, 2014.

[1111] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1115–1124, 2017.

[1112] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1307–1315, 2018.

[1113] Sibei Yang, Guanbin Li, and Yizhou Yu. Dynamic graph attention for referring expression comprehension. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4644–4653, 2019.

[1114] Zhengyuan Yang, Boqing Gong, Liwei Wang, Wenbing Huang, Dong Yu, and Jiebo Luo. A fast and accurate one-stage approach to visual grounding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4683–4693, 2019.

[1115] Jiajun Deng, Zhengyuan Yang, Tianlang Chen, Wengang Zhou, and Houqiang Li. Transvg: End-to-end visual grounding with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1769–1779, 2021.

[1116] Liunian Li, Zi-Yi Dou, Nanyun Peng, and Kai-Wei Chang. Desco: Learning object recognition with rich language descriptions. *Advances in Neural Information Processing Systems*, 36:37511–37526, 2023.

[1117] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

[1118] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023.

[1119] Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, et al. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv preprint arXiv:2404.07973*, 2024.

[1120] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M. Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S. Khan. Glamm: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2024.

[1121] Haobo Yuan, Xiangtai Li, Tao Zhang, Yueyi Sun, Zilong Huang, Shilin Xu, Shunping Ji, Yunhai Tong, Lu Qi, Jiashi Feng, and Ming-Hsuan Yang. Sa2va: Marrying sam2 with llava for dense grounded understanding of images and videos. *arXiv preprint arXiv:2501.04001*, 2025.

[1122] Ethan Reid. Moondream segmentation: From words to masks. *arXiv preprint arXiv:2604.02593*, 2026.

[1123] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4818–4829, 2024.

[1124] Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy Lillicrap, Jean-baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[1125] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[1126] Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[1127] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

[1128] Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, et al. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*, 2025.

[1129] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[1130] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

[1131] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11888–11898, 2023.

[1132] Muzhi Zhu, Yuzhuo Tian, Hao Chen, Chunluan Zhou, Qingpei Guo, Yang Liu, Ming Yang, and Chunhua Shen. Segagent: Exploring pixel understanding capabilities in mllms by imitating human annotator trajectories. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3686–3696, 2025.

[1133] Ziwei Zheng, Michael Yang, Jack Hong, Chenxiao Zhao, Guohai Xu, Le Yang, Chao Shen, and Xing Yu. Deepeyes: Incentivizing "thinking with images" via reinforcement learning. *arXiv preprint arXiv:2505.14362*, 2025.

[1134] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

[1135] Zixin Zhu, Xuelu Feng, Dongdong Chen, Junsong Yuan, Chunming Qiao, and Gang Hua. Exploring pre-trained text-to-video diffusion models for referring video object segmentation. In *European Conference on Computer Vision*, pages 452–469, 2024.

[1136] Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T Barron, et al. Image generators are generalist vision learners. *arXiv preprint arXiv:2604.20329*, 2026.

[1137] Jierun Chen, Fangyun Wei, Jinjing Zhao, Sizhe Song, Bohuai Wu, Zhuoxuan Peng, S-H Gary Chan, and Hongyang Zhang. Revisiting referring expression comprehension evaluation in the era of large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 513–524, 2025.

[1138] Fangyun Wei, Jinjing Zhao, Kun Yan, Hongyang Zhang, and Chang Xu. A large-scale human-centric benchmark for referring expression comprehension in the lmm era. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024.

[1139] Tao Ma, Bing Bai, Haozhe Lin, Heyuan Wang, Yu Wang, Lin Luo, and Lu Fang. When visual grounding meets gigapixel-level large-scale scenes: Benchmark and approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22119–22128, 2024.

[1140] Rang Li, Lei Li, Shuhuai Ren, Hao Tian, Shuhao Gu, Shicheng Li, Zihao Yue, Yudong Wang, Wenhan Ma, Zhe Yang, et al. Groundingme: Exposing the visual grounding gap in mllms through multi-dimensional evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2412–2422, 2026.

[1141] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:20482–20494, 2023.

[1142] R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. 1994.

[1143] K. M. Lynch and F. C. Park. *Modern Robotics: Mechanics, Planning, and Control*. 2017.

[1144] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Oriolo. *Robotics: Modelling, Planning and Control*. 2010.

[1145] Ronald T. Azuma. A survey of augmented reality. *Presence: Teleoperators and Virtual Environments*, 6 (4):355–385, 1997.

[1146] Joan Solà, Jeremie Deray, and Dinesh Atchuthan. A micro lie theory for state estimation in robotics. *arXiv preprint arXiv:1812.01537*, 2018.

[1147] J. Denavit and R. S. Hartenberg. A kinematic notation for lower-pair mechanisms based on matrices. *ASME Journal of Applied Mechanics*, 22(2):215–221, 1955.

[1148] R. P. Paul. *Robot Manipulators: Mathematics, Programming, and Control*. 1981.

[1149] K. Shoemake. Animating rotation with quaternion curves. In *Proc. SIGGRAPH*, pages 245–254, 1985.

[1150] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. *J. Opt. Soc. Am. A*, 4(4):629–642, 1987.

[1151] K. Daniilidis. Hand–eye calibration using dual quaternions. *Int. J. Robotics Research*, 18(3):286–298, 1999.

[1152] Markus Horn, Thomas Wodtko, Michael Buchholz, and Klaus Dietmayer. Online extrinsic calibration based on per-sensor ego-motion using dual quaternions. *IEEE Robotics and Automation Letters*, 6(2): 982–989, 2021. doi:10.1109/LRA.2021.3056352. URL https://arxiv.org/abs/2101.11440.

[1153] R. I. Hartley. In defense of the eight-point algorithm. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 19(6):580–593, 1997.

[1154] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 26(6):756–770, 2004.

[1155] D. E. Whitney. Resolved motion rate control of manipulators and human prostheses. *IEEE Trans. Man-Machine Systems*, 10(2):47–53, 1969.

[1156] Y. Nakamura and H. Hanafusa. Inverse kinematic solutions with singularity robustness for robot manipulator control. *ASME J. Dynamic Systems, Measurement, and Control*, 108(3):163–171, 1986.

[1157] A. Liegeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Trans. Systems, Man, and Cybernetics*, 7(12):868–871, 1977.

[1158] B. Siciliano and J. J. E. Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. In *Proc. Int. Conf. Advanced Robotics*, pages 1211–1216, 1991.

[1159] O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE J. Robotics and Automation*, 3(1):43–53, 1987.

[1160] V. Chvátal. A combinatorial theorem in plane geometry. *J. Combin. Theory, Series B*, 18(1):39–41, 1975.

[1161] J. O'Rourke. *Art Gallery Theorems and Algorithms*. 1987.

[1162] D. T. Lee. Visibility of a simple polygon. *CVGIP*, 22(2):207–221, 1983.

[1163] L. J. Guibas and J. Hershberger. Optimal shortest path queries in a simple polygon. *J. Comput. Syst. Sci.*, 39(2):126–152, 1987.

[1164] K. A. Tarabanis, P. K. Allen, and R. Y. Tsai. A survey of sensor planning in computer vision. *IEEE Trans. Robotics and Automation*, 11(1):86–104, 1995.

[1165] B. Roth. Performance evaluation of manipulators from a kinematic viewpoint. *ASME J. Mechanical Design*, 97(1):102–111, 1975.

[1166] C. Gosselin and J. Angeles. Singularity analysis of closed-loop kinematic chains. *IEEE Trans. Robotics and Automation*, 6(3):281–290, 1990.

[1167] J. P. Merlet. Determination of 6d workspaces of gough-type parallel manipulators and comparison between different geometries. *Int. J. Robotics Research*, 17(3):338–356, 1998.

[1168] Dimiter Zlatanov, Ilian A. Bonev, and Clément M. Gosselin. Constraint singularities of parallel mechanisms. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, volume 1, pages 496–502, 2002. doi:10.1109/ROBOT.2002.1013408. URL https://ieeexplore.ieee.org/document/1013408.

[1169] Yupu Lu, Yuxiang Ma, and Jia Pan. Richmap: A reachability map balancing precision, efficiency, and flexibility for rich robot manipulation tasks, 2026. URL https://arxiv.org/abs/2604.06778. Accepted by WAFR 2026.

[1170] C. Ferrari and J. Canny. Planning optimal grasps. In *Proc. ICRA*, pages 2290–2295, 1992.

[1171] A. T. Miller and P. K. Allen. Graspit!: A versatile simulator for robotic grasping. *IEEE Robotics and Automation Magazine*, 11(4):110–122, 2004.

[1172] Y. Lin and Y. Sun. Task-oriented grasp quality measures for grasp synthesis. *IEEE Trans. Robotics*, 31 (3):837–848, 2015.

[1173] A. Knapitsch, J. Park, Q. Y. Zhou, and V. Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Trans. Graphics (SIGGRAPH)*, 36(4):1–13, 2017.

[1174] H. Aanæs, R. R. Jensen, G. Vogiatzis, E. Tola, and A. B. Dahl. Large-scale data for multiple-view stereopsis. *Int. J. Computer Vision*, 120(2):153–168, 2016.

[1175] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 5173–5182, 2017.

[1176] T. Sattler, W. Maddern, A. Torii, J. Sivic, J. Kosecka, T. Pajdla, M. Pollefeys, and T. Weyand. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 8601–8610, 2018.

[1177] H. Taira, M. Okutomi, T. Sattler, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and A. Irschara. Inloc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 7199–7209, 2018.

[1178] Patrick Wenzel, Rui Wang, Nan Yang, Qing Cheng, Qadeer Khan, Lukas von Stumberg, Niclas Zeller, and Daniel Cremers. 4Seasons: A cross-season dataset for multi-weather SLAM in autonomous driving. In *Pattern Recognition*, volume 12544 of *Lecture Notes in Computer Science*, pages 404–417, 2021. doi:10.1007/978-3-030-71278-5_29. URL https://arxiv.org/abs/2009.06364.

[1179] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 72–79, 2010.

[1180] E. Johns, S. Leutenegger, and A. J. Davison. Pairwise decomposition of image sequences for active multi-view recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 3813–3822, 2016.

[1181] Y. Jiang, S. Moseson, and A. Saxena. Efficient grasping from rgbd images: Learning using a new rectangle representation. In *Proc. ICRA*, pages 3304–3311, 2011.

[1182] A. Depierre, E. Dellandréa, and L. Chen. Jacquard: A large-scale dataset for robotic grasp detection. In *Proc. IROS*, pages 3511–3516, 2018.

[1183] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11444–11453, 2020.

[1184] Berk Calli, Aaron Walsman, Arjun Singh, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols. *IEEE Robotics & Automation Magazine*, 22(3):36–52, 2015.

[1185] B. J. Cohen and I. A. Şucan. Benchmarking motion planning algorithms: An open source infrastructure for the robotics community. In *Proc. IROS*, pages 4081–4086, 2012.

[1186] Bruno Siciliano and Oussama Khatib, editors. *Springer Handbook of Robotics*. 2nd edition, 2016. ISBN 978-3-319-32550-7. doi:10.1007/978-3-319-32552-1.

[1187] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. doi:10.1109/IROS.2012.6386109.

[1188] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. http://pybullet.org, 2016–2021.

[1189] Russell Smith. Open dynamics engine. https://ode.org/, 2021.

[1190] Graphics Lab and Humanoid Robotics Lab. Dynamic animation and robotics toolkit. https://dartsim.github.io/, 2023.

[1191] Jeffrey C Trinkle, J-S Pang, Sandra Sudarsky, and Grace Lo. On dynamic multi-rigid-body contact problems with coulomb friction. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 77(4):267–279, 1997.

[1192] Richard W Cottle, Jong-Shi Pang, and Richard E Stone. *The linear complementarity problem*. 2009.

[1193] YY Lin and J-S Pang. Iterative methods for large convex quadratic programs: A survey. *SIAM Journal on Control and Optimization*, 25(2):383–411, 1987.

[1194] Farid Alizadeh and Donald Goldfarb. Second-order cone programming. *Mathematical programming*, 95(1):3–51, 2003.

[1195] Oliver Groth, Fabian B Fuchs, Ingmar Posner, and Andrea Vedaldi. Shapestacks: Learning vision-based physical intuition for generalised object stacking. In *Proceedings of the european conference on computer vision (eccv)*, pages 702–717, 2018.

[1196] Anton Bakhtin, Laurens van der Maaten, Justin Johnson, Laura Gustafson, and Ross Girshick. Phyre: A new benchmark for physical reasoning. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[1197] Zhiao Huang, Yuanming Hu, Tao Du, Siyuan Zhou, Hao Su, Joshua B Tenenbaum, and Chuang Gan. Plasticinelab: A soft-body manipulation benchmark with differentiable physics. *arXiv preprint arXiv:2104.03311*, 2021.

[1198] Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, and Emmanuel Dupoux. Intphys: A framework and benchmark for visual intuitive physics reasoning. *arXiv preprint arXiv:1803.07616*, 2018.

[1199] K. Yi, C. Gan, Y. Li, P. Kohli, J. Wu, A. Torralba, and J. B. Tenenbaum. Clevrer: Collision events for video representation and reasoning. In *ICLR*, 2020.

[1200] Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B Tenenbaum, and Chuang Gan. Comphy: Compositional physical reasoning of objects and events from videos. *arXiv preprint arXiv:2205.01089*, 2022.

[1201] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 29, 2016.

[1202] Chae H An, Christopher G Atkeson, and John M Hollerbach. Estimation of inertial parameters of rigid body links of manipulators. In *1985 24th IEEE Conference on Decision and Control*, pages 990–995, 1985.

[1203] Nikhil Muralidhar, Jie Bu, Ze Cao, Long He, Naren Ramakrishnan, Danesh Tafti, and Anuj Karpatne. Phynet: Physics guided neural networks for particle drag force prediction in assembly. In *Proceedings of the 2020 SIAM international conference on data mining*, pages 559–567, 2020.

[1204] Jiajun Wu, Ilker Yildirim, Joseph J Lim, Bill Freeman, and Josh Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 28, 2015.

[1205] T. Lozano-Pérez and M. A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10):560–570, 1979.

[1206] T. Lozano-Pérez. Spatial planning: A configuration space approach. *IEEE Transactions on Computers*, C-32(2):108–120, 1983.

[1207] J. Hershberger and S. Suri. An optimal algorithm for euclidean shortest paths in the plane. *SIAM Journal on Computing*, 28(6):2215–2256, 1999.

[1208] J. S. B. Mitchell. Geometric shortest paths and network optimization. In *Handbook of Computational Geometry*, pages 633–701. 2000.

[1209] D. Ferguson and A. Stentz. Using interpolation to improve path planning: The field d* algorithm. *Journal of Field Robotics*, 23(2):79–101, 2006.

[1210] A. Nash, K. Daniel, S. Koenig, and A. Felner. Theta*: Any-angle path planning on grids. In *Proceedings of AAAI*, pages 1177–1183, 2007.

[1211] D. Harabor and A. Grastien. Online graph pruning for pathfinding on grid maps (jump point search). In *Proceedings of AAAI*, pages 1114–1119, 2011.

[1212] S. Fortune. A sweepline algorithm for voronoi diagrams. *Algorithmica*, 2:153–174, 1987.

[1213] H. Choset and J. Burdick. The hierarchical generalized voronoi graph. *International Journal of Robotics Research*, 19(2):96–125, 2000.

[1214] H. Choset. Sensor-based exploration: Incremental construction of the generalized voronoi graph. *International Journal of Robotics Research*, 19(2):126–148, 2000.

[1215] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Distance transforms of sampled functions. *Theory of Computing*, 8(1):415–428, 2012. doi:10.4086/toc.2012.v008a019. URL https://cs.brown.edu/people/pfelzens/papers/dt-final.pdf.

[1216] B. Lau, C. Sprunk, and W. Burgard. Efficient grid-based spatial representations for robot navigation in dynamic environments. *Robotics and Autonomous Systems*, 61(10):1116–1130, 2013.

[1217] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57, 1989.

[1218] H. Choset and P. Pignon. Coverage path planning: The boustrophedon cellular decomposition. In *Proceedings of Field and Service Robotics*, pages 203–209, 1998.

[1219] E. Galceran and M. Carreras. A survey on coverage path planning for robotics. *Robotics and Autonomous Systems*, 61(12):1258–1276, 2013.

[1220] L. E. Kavraki, P. Švestka, J.-C. Latombe, and M. H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12 (4):566–580, 1996.

[1221] S. M. LaValle. Rapidly-exploring random trees: A new tool for path planning. Technical report, Iowa State University, 1998.

[1222] J. J. Kuffner and S. M. LaValle. Rrt-connect: An efficient approach to single-query path planning. In *Proceedings of IEEE ICRA*, pages 995–1001, 2000.

[1223] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *International Journal of Robotics Research*, 30(7):846–894, 2011.

[1224] L. Janson, E. Schmerling, and M. Pavone. Fast marching tree (fmt*): A fast marching sampling-based method for optimal motion planning. *International Journal of Robotics Research*, 34(7):883–921, 2015.

[1225] J. D. Gammell, S. S. Srinivasa, and T. D. Barfoot. Batch informed trees (bit*): Sampling-based optimal planning via the heuristically guided search of implicit random geometric graphs. In *Proceedings of IEEE ICRA*, pages 3067–3074, 2015.

[1226] S. Koenig and M. Likhachev. Lifelong planning a*. *Artificial Intelligence*, 155(1–2):93–146, 2004.

[1227] S. Koenig and M. Likhachev. D* lite. In *Proceedings of AAAI*, pages 476–483, 2002.

[1228] G. Sharon, R. Stern, A. Felner, and N. Sturtevant. Conflict-based search for optimal multi-agent pathfinding. *Artificial Intelligence*, 219–220:40–66, 2015.

[1229] G. Sharon, R. Stern, M. Goldenberg, and A. Felner. The increasing cost tree search for optimal multi-agent pathfinding. *Artificial Intelligence*, 195:470–495, 2013.

[1230] M. Barer, G. Sharon, R. Stern, and A. Felner. Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem (ecbs). In *Proceedings of SoCS*, pages 19–27, 2014.

[1231] J. Li, W. Ruml, and S. Koenig. Eecbs: A bounded-suboptimal search for multi-agent path finding. In *Proceedings of AAAI*, pages 12353–12362, 2021.

[1232] P. Surynek, A. Felner, R. Stern, and E. Boyarski. An efficient sat approach to multi-agent path finding under the sum-of-costs objective. In *Proceedings of ECAI*, pages 810–818, 2016.

[1233] J. Yu and S. M. LaValle. Optimal multirobot path planning on graphs: Complete algorithms and effective heuristics. *IEEE Transactions on Robotics*, 32(5):1163–1177, 2016.

[1234] A. Botea, M. Müller, and J. Schaeffer. Near optimal hierarchical path-finding. *Journal of Game Development*, 1(1):7–28, 2004.

[1235] R. Geisberger, P. Sanders, D. Schultes, and D. Delling. Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In *Proceedings of WEA*, pages 319–333, 2008.

[1236] N. R. Sturtevant. Benchmarks for grid-based pathfinding. *IEEE Transactions on Computational Intelligence and AI in Games*, 4(2):144–148, 2012.

[1237] Movingai mapf benchmarks. https://movingai.com/benchmarks/, 2025. Accessed 2025.

[1238] Multi-agent pathfinding: Definitions, variants, and benchmarks, 2019. URL https://arxiv.org/abs/1906.08291.

[1239] Ompl planner arena benchmarks. https://ompl.kavrakilab.org/benchmarks.html, 2025. Accessed 2025.

[1240] Dimacs shortest path challenge. http://www.dis.uniroma1.it/challenge9/, 2025. Accessed 2025.

[1241] A. Howard and N. Roy. The robotics data set repository (radish), 2003.

[1242] W. Maddern, G. Pascoe, C. Linegar, and P. Newman. 1 year, 1000 km: The oxford robotcar dataset. *International Journal of Robotics Research*, 36(1):3–15, 2017.

[1243] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice. University of michigan north campus long-term (nclt) dataset. *International Journal of Robotics Research*, 35(9):1023–1035, 2016.

[1244] Daniel Olid, José M Fácil, and Javier Civera. Single-view place recognition under seasonal changes. *arXiv preprint arXiv:1808.06516*, 2018.

[1245] Cyrill Stachniss, Giorgio Grisetti, and Wolfram Burgard. Information gain-based exploration using rao-blackwellized particle filters. *Robotics: Science and Systems (RSS)*, 2:65–72, 2006.

[1246] Brandon D. Luders, Marsette Vona, and Jonathan P. How. Chance-constrained rrt for probabilistic robustness to environmental uncertainty. In *Proc. IEEE Conf. on Robotics and Automation (ICRA)*, pages 4496–4503, 2010.

[1247] Sihua Zhang, Di-Hua Zhai, Yuhan Xiong, Juncheng Lin, and Yuanqing Xia. Safety-critical control for robotic systems with uncertain model via control barrier function. *International Journal of Robust and Nonlinear Control*, 33(6):3661–3676, April 2023. doi:10.1002/rnc.6585. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.6585.

[1248] Daiki Kimura, Masaki Ono, Subhajit Chaudhury, Ryosuke Kohita, Akifumi Wachi, Don Joven Agravante, Michiaki Tatsubori, Asim Munawar, and Alexander Gray. Neuro-symbolic reinforcement learning with first-order logic. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3505–3511, 2021. doi:10.18653/v1/2021.emnlp-main.283. URL https://aclanthology.org/2021.emnlp-main.283/.

[1249] T. Lozano-Pérez and L. P. Kaelbling. A constraint-based method for solving sequential manipulation planning problems. In *Proc. IEEE/RSJ IROS*, 2014.

[1250] F. Lagriffoul, D. Dimitrov, A. Saffiotti, and L. Karlsson. Constraint propagation on interval bounds for dealing with geometric backtracking. In *Proc. IEEE/RSJ IROS*, 2012.

[1251] N. T. Dantam, Z. K. Kingston, S. Chaudhuri, and L. E. Kavraki. An incremental constraint-based framework for task and motion planning. *Int. J. Robotics Research*, 37(10), 2018.

[1252] S. Gao, S. Kong, and E. M. Clarke. dreal: An smt solver for nonlinear theories over the reals. In *Proc. CADE*, 2013.

[1253] Yasser Shoukry, Pierluigi Nuzzo, Indranil Saha, Alberto L. Sangiovanni-Vincentelli, Sanjit A. Seshia, George J. Pappas, and Paulo Tabuada. Scalable motion planning using lazy SMT-based solving. In *Proceedings of the 55th IEEE Conference on Decision and Control*, pages 6683–6688, Las Vegas, NV, USA, 2016. doi:10.1109/CDC.2016.7799296. URL https://people.eecs.berkeley.edu/~sseshia/pubdir/cdc16-plan.pdf.

[1254] T. Kolářík, S. Ratschan, and P. Surynek. Multi-agent path finding with continuous time using sat/smt. *arXiv preprint arXiv:2312.08051*, 2023.

[1255] W.-J. Tan, H. Ma, S. Koenig, and R. Stern. Robust multi-agent pathfinding with continuous time. *Proc. ICAPS*, 2024.

[1256] R. Deits and R. Tedrake. Computing large convex regions of obstacle-free space through semidefinite programming (iris). In *Proc. WAFR*, 2014.

[1257] T. Marcucci, J. Umenberger, P. A. Parrilo, and R. Tedrake. Shortest paths in graphs of convex sets. *SIAM J. Optimization*, 2024.

[1258] T. Marcucci, M. Petersen, D. von Wrangel, and R. Tedrake. Motion planning around obstacles with convex optimization. *Science Robotics*, 2023.

[1259] H. Dai, A. Amice, S. Morozov, and R. Tedrake. Certified polyhedral decompositions of collision-free configuration space (c-iris). *International Journal of Robotics Research*, 2024.

[1260] M. Zucker, N. Ratliff, M. Stolle, J. A. Bagnell, C. G. Atkeson, and J. Kuffner. Chomp: Covariant hamiltonian optimization for motion planning. *Int. J. Robotics Research*, 32(9-10), 2013.

[1261] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal. Stomp: Stochastic trajectory optimization for motion planning. In *Proc. IEEE ICRA*, 2011.

[1262] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel. Motion planning with sequential convex optimization and convex collision checking. *Int. J. Robotics Research*, 33(9), 2014.

[1263] M. Mukadam, J. Dong, X. Yan, F. Dellaert, and B. Boots. Continuous-time gaussian process motion planning via probabilistic inference. *Int. J. Robotics Research*, 37(11), 2018.

[1264] R. Diankov. *Automated Construction of Robotic Manipulation Programs*. PhD thesis, Carnegie Mellon University, 2010.

[1265] P. Beeson and B. Ames. Trac-ik: An open-source library for improved solving of generic inverse kinematics. In *Proc. IEEE-RAS Humanoids*, 2015.

[1266] H. Dai, G. Izatt, and R. Tedrake. Global inverse kinematics via mixed-integer convex optimization. *Int. J. Robotics Research*, 38(2-3), 2019.

[1267] M. Toussaint. Logic-geometric programming: An optimization-based approach to combined task and motion planning. *IJCAI*, 2015.

[1268] M. Toussaint and M. Lopes. Multi-bound tree search for logic-geometric programming in cooperative manipulation domains. In *Proc. IEEE ICRA*, 2017.

[1269] Caelan Reed Garrett, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Pddlstream: Integrating symbolic planners and blackbox samplers via optimistic adaptive planning. In *Proceedings of the international conference on automated planning and scheduling*, volume 30, pages 440–448, 2020.

[1270] K. T. Ly, V. Semenov, M. Risiglione, W. Merkt, and I. Havoutis. R-lgp: A reachability-guided logic-geometric programming framework for optimal task and motion planning on mobile manipulators. *arXiv preprint arXiv:2310.02791*, 2023.

[1271] B. Vu, T. Migimatsu, and J. Bohg. Coast: Constraints and streams for task and motion planning. In *Proc. IEEE ICRA*, 2024.

[1272] I. A. Şucan, M. Moll, and L. E. Kavraki. The open motion planning library. *IEEE Robotics & Automation Magazine*, 19(4):72–82, 2012.

[1273] F. Lagriffoul, N. T. Dantam, C. Garrett, A. Akbari, S. Srivastava, and L. E. Kavraki. Platform-independent benchmarks for task and motion planning. *IEEE Robotics and Automation Letters*, 3(4): 3765–3772, 2018.

[1274] J. Collins, M. Gupta, F. Fang, S. S. Srinivasa, and D. Held. Ramp: A benchmark for evaluating robotic assembly manipulation and planning. *IEEE Robotics and Automation Letters*, 8(6):3454–3461, 2023.

[1275] Y. Tian, R. Li, A. Varava, T. Asfour, M. Toussaint, and K. Goldberg. Asap: Automated sequence planning for complex robotic assembly with physical feasibility. *arXiv preprint arXiv:2309.16909*, 2024.

[1276] M. Heo, Y. Lee, D. Lee, and J. J. Lim. Furniturebench: Reproducible real-world benchmark for long-horizon complex manipulation. In *Proc. Robotics: Science and Systems (RSS)*, 2023.

[1277] L. Blackmore, M. Ono, and B. C. Williams. Chance-constrained optimal path planning with obstacles. *IEEE Trans. Robotics*, 27(6), 2011.

[1278] Vasumathi Raman, Mehdi Maasoumy, and Alexandre Donzé. Model predictive control from signal temporal logic specifications: A case study. In *Proceedings of the 4th ACM SIGBED international workshop on design, modeling, and evaluation of cyber-physical systems*, pages 52–55, 2014.

[1279] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada. Control barrier functions: Theory and applications. *arXiv preprint arXiv:1903.11199*, 2019.

[1280] D. McDermott, M. Ghallab, A. Howe, C. Knoblock, A. Ram, M. Veloso, D. Weld, and D. Wilkins. Pddl—the planning domain definition language. Technical report, AIPS-98 Planning Competition Committee, 1998.

[1281] A. L. Blum and M. L. Furst. Fast planning through planning graph analysis. *Artificial Intelligence*, 90 (1–2):281–300, 1997.

[1282] J. Hoffmann and B. Nebel. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14:253–302, 2001.

[1283] L. P. Kaelbling and T. Lozano-Pérez. Hierarchical planning in the now. In *Proc. IEEE ICRA*, 2011.

[1284] C. R. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. P. Kaelbling, and T. Lozano-Pérez. Integrated task and motion planning. *Annual Review of Control, Robotics, and Autonomous Systems*, 4:265–293, 2021.

[1285] Z. Zhao, S. Cheng, Y. Ding, Z. Zhou, S. Zhang, D. Xu, and Y. Zhao. A survey of optimization-based task and motion planning: From classical to learning approaches. *arXiv preprint arXiv:2404.02817*, 2024.

[1286] D. Harel. Statecharts: A visual formalism for complex systems. *Science of Computer Programming*, 8(3): 231–274, 1987.

[1287] N. J. Nilsson. Teleo-reactive programs for agent control. *Journal of Artificial Intelligence Research*, 1: 139–158, 1994.

[1288] David C. Conner. FlexBE – the flexible behavior engine: Collaborative autonomy in ROS 2, 2023. URL https://roscon.ros.org/2023/talks/FlexBE_-_The_Flexible_Behavior_Engine_Collaborative_Autonomy_in_ROS_2.pdf.

[1289] Damian Isla. Handling complexity in the Halo 2 AI. In *Proceedings of the 2005 Game Developers Conference*, San Francisco, CA, USA, March 2005. URL https://www.gamedeveloper.com/programming/gdc-2005-proceeding-handling-complexity-in-the-i-halo-2-i-ai.

[1290] A. Marzinotto, M. Colledanchise, C. Smith, and P. Ögren. Towards a unified behavior trees framework for robot control. In *Proc. IEEE ICRA*, 2014.

[1291] M. Colledanchise and P. Ögren. How behavior trees modularize robustness and safety in hybrid systems. In *Proc. IEEE/RSJ IROS*, 2017.

[1292] M. Colledanchise and P. Ögren. *Behavior Trees in Robotics and AI: An Introduction*. 2018.

[1293] Francisco Martín, Matteo Morelli, Huáscar Espinoza, Francisco J. Rodríguez-Lera, and Vicente Matellán. Optimized execution of PDDL plans using behavior trees. *CoRR*, abs/2101.01964, 2021. URL https://arxiv.org/abs/2101.01964.

[1294] Ruikai Liu, Guangxi Wan, Maowei Jiang, Haojie Chen, and Peng Zeng. Autonomous robot task execution in flexible manufacturing: Integrating PDDL and behavior trees in ARIAC 2023. *Biomimetics*, 9(10):612, 2024. doi:10.3390/biomimetics9100612. URL https://www.mdpi.com/2313-7673/9/10/612.

[1295] R. A. Agis, S. Gottifredi, and A. J. García. An event-driven behavior trees extension to facilitate non-player multi-agent coordination in video games. *Expert Systems With Applications*, 155, 2020.

[1296] C. L. Forgy. Ops5 user's manual. Technical report, Carnegie Mellon University, 1981.

[1297] C. L. Forgy. Rete: A fast algorithm for the many pattern/many object pattern-matching problem. *Communications of the ACM*, 19(2):17–37, 1982.

[1298] G. Riley, F. Culbert, R. T. Savely, and F. Lopez. Clips: The c language integrated production system. Technical report, NASA Johnson Space Center, 1991.

[1299] J. E. Laird, A. Newell, and P. S. Rosenbloom. Soar: An architecture for general intelligence. *Artificial Intelligence*, 33(1):1–64, 1987.

[1300] M. Tenorth and M. Beetz. Knowrob—a knowledge processing infrastructure for cognition-enabled robots. *International Journal of Robotics Research*, 32(5):566–590, 2013.

[1301] M. Beetz, L. Mösenlechner, and M. Tenorth. Cram—a cognitive robot abstract machine for everyday manipulation in human environments. In *Proc. IEEE/RSJ IROS*, 2010.

[1302] Michael Beetz, Gayane Kazhoyan, and David Vernon. The cram cognitive architecture for robot manipulation in everyday activities, 2023. URL https://arxiv.org/abs/2304.14119.

[1303] C. Finucane, G. Jing, and H. Kress-Gazit. Ltlmop: Experimenting with language, temporal logic and robot control. In *Proc. IEEE/RSJ IROS*, 2010.

[1304] planning.domains: PDDL problem/domain repository and api, 2016. Community resource.

[1305] S. Srivastava, C. Li, R. Lingelbach, R. Martín-Martín, F. Xia, L. Fei-Fei, et al. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecologically valid environments. In *Proc. CoRL (PMLR)*, 2022.

[1306] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, Hang Yin, Michael Lingelbach, Minjune Hwang, Ayano Hiranaka, Sujay Garlanka, Arman Aydin, Sharon Lee, Jiankai Sun, Mona Anvari, Manasi Sharma, Dhruva Bansal, Samuel Hunter, Kyu-Young Kim, Alan Lou, Caleb R. Matthews, Ivan Villa-Renteria, Jerry Huayang Tang, Claire Tang, Fei Xia, Yunzhu Li, Silvio Savarese, Hyowon Gweon, C. Karen Liu, Jiajun Wu, and Li Fei-Fei. BEHAVIOR-1K: A human-centered, embodied AI benchmark with 1,000 everyday activities and realistic simulation, 2024. URL https://arxiv.org/abs/2403.09227.

[1307] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749, 2020.

[1308] A. Padmakumar, J. Thomason, A. Shrivastava, D. Hakkani-Tür, et al. Teach: Task-driven embodied agents that chat. In *Proc. AAAI*, 2022.

[1309] Thomas Wisspeintner, Tijn van der Zant, Luca Iocchi, and Stefan Schiffer. Robocup@home: Results in benchmarking domestic service robots. In *Proc. RoboCup Symposium (Lecture Notes in Computer Science #5949)*, pages 390–401, 2010.

[1310] Tomi Janhunen and Michael Sioutis. Allen's interval algebra makes the difference. In *International Conference on Applications of Declarative Programming and Knowledge Management*, pages 89–98, 2019.

[1311] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Mengyuan Yan, and Andy Zeng. Do as i can, not as i say: Grounding language in robotic affordances, 2022. URL https://arxiv.org/abs/2204.01691.

[1312] J. Pearl. Causality: Models, reasoning, and inference (2nd ed.), 2009. URL https://doi.org/10.1017/CBO9780511803161.

[1313] J. Peters, D. Janzing, and B. Schölkopf. Elements of causal inference: Foundations and learning algorithms, 2017. URL https://mitpress.mit.edu/9780262037310/elements-of-causal-inference/.

[1314] A. Sharma and E. Kiciman. Dowhy: A python library for causal inference, 2020. URL https://arxiv.org/abs/2011.04216.

[1315] Patrick Blöbaum, Peter Götz, Kailash Budhathoki, Atalanti A. Mastakouri, and Dominik Janzing. DoWhy-GCM: An extension of DoWhy for causal inference in graphical causal models. *Journal of Machine Learning Research*, 25(147):1–7, 2024. doi:10.48550/arXiv.2206.06821. URL https://jmlr.org/papers/v25/22-1258.html.

[1316] P. Spirtes, C. Glymour, and R. Scheines. Causation, prediction, and search (2nd ed.), 2000. URL https://mitpress.mit.edu/9780262194402/causation-prediction-and-search/.

[1317] D. M. Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2002.

[1318] C. Meek. Graphical models: Selecting causal and statistical models (ph.d. thesis), 1997. URL https://kilthub.cmu.edu/articles/thesis/Graphical_Models_Selecting_Causal_and_Statistical_Models/6613208.

[1319] D. Colombo and M. H. Maathuis. Order-independent constraint-based causal structure learning. *Journal of Machine Learning Research*, 15:3921–3962, 2014.

[1320] M. Kalisch and P. Bühlmann. Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research*, 8:613–636, 2007.

[1321] Yujia Zheng, Biwei Huang, Wei Chen, Joseph Ramsey, Mingming Gong, Ruichu Cai, Shohei Shimizu, Peter Spirtes, and Kun Zhang. Causal-learn: Causal discovery in python. *Journal of Machine Learning Research*, 25(60):1–8, 2024. URL https://www.jmlr.org/papers/v25/23-0970.html.

[1322] Joseph D. Ramsey, Kun Zhang, Madelyn Glymour, Ruben Sanchez Romero, Biwei Huang, Imme Ebert-Uphoff, Savini Samarasinghe, Elizabeth A. Barnes, and Clark Glymour. TETRAD: A toolbox for causal discovery. In *Proceedings of the 8th International Workshop on Climate Informatics*, pages 1–4, Boulder, CO, USA, 2018. URL https://www.atmos.colostate.edu/~iebert/PAPERS/CI2018_paper_35.pdf.

[1323] Joseph D. Ramsey and Bryan Andrews. Py-Tetrad and RPy-Tetrad: A new python interface with R support for tetrad causal search, 2023. URL https://arxiv.org/abs/2308.07346. Causal Analysis Workshop Series (CAWS) 2023.

[1324] S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7:2003–2030, 2006.

[1325] S. Shimizu, T. Inazumi, Y. Sogawa, A. Hyvärinen, Y. Kawahara, T. Washio, P. O. Hoyer, and K. Bollen. Directlingam: A direct method for learning a linear non-gaussian structural equation model. *Journal of Machine Learning Research*, 12:1225–1248, 2011.

[1326] E. Giunchiglia, J. Lee, V. Lifschitz, N. McCain, and H. Turner. Nonmonotonic causal theories. *Artificial Intelligence*, 153(1–2):49–104, 2004.

[1327] V. Lifschitz. Missionaries and cannibals in the causal calculator. In *Proc. KR*, pages 85–96, 2000.

[1328] Enrico Giunchiglia, Joohyung Lee, Vladimir Lifschitz, and Hudson Turner. Causal laws and multi-valued fluents. *arXiv preprint arXiv:2307.10227*, 2023. doi:10.48550/arXiv.2307.10227.

[1329] David McAllester and David Rosenblitt. Systematic nonlinear planning. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 634–639, Anaheim, CA, USA, 1991. URL https://cdn.aaai.org/AAAI/1991/AAAI91-099.pdf.

[1330] J. S. Penberthy and D. S. Weld. Ucpop: A sound, complete, partial order planner for adl. In *Proc. KR*, pages 103–114, 1992.

[1331] H. L. S. Younes and M. L. Littman. Vhpop: Versatile heuristic partial order planner. *Journal of Artificial Intelligence Research*, 20:405–430, 2004.

[1332] R. Dechter, I. Meiri, and J. Pearl. Temporal constraint networks. *Artificial Intelligence*, 49(1–3):61–95, 1991.

[1333] I. Tsamardinos, N. Muscettola, and P. Morris. Fast transformation of temporal plans for efficient execution. In *Proc. AAAI*, pages 254–261, 1998.

[1334] P. Morris, N. Muscettola, and T. Vidal. Dynamic control of plans with temporal uncertainty. In *Proc. IJCAI*, pages 494–499, 2001.

[1335] S. Kong, S. Gao, W. Chen, and E. M. Clarke. dreach: δ-reachability analysis for hybrid systems. In *Proc. CAV*, pages 200–206, 2015.

[1336] J. Mao, X. Yang, X. Zhang, N. D. Goodman, and J. Wu. Clevrer-humans: Describing physical and causal events the human way. In *Proc. NeurIPS Datasets and Benchmarks*, 2023.

[1337] Hsiao-Yu Tung, Mingyu Ding, Zhenfang Chen, Daniel Bear, Chuang Gan, Josh Tenenbaum, Dan Yamins, Judith Fan, and Kevin Smith. Physion++: Evaluating physical scene understanding that requires online inference of different physical properties. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:67048–67068, 2023.

[1338] A. Saha, S. Joty, and S. C. H. Hoi. Craft: Causal reasoning about forces and interactions from videos. In *Findings of ACL*, pages 3102–3117, 2022.

[1339] Z. Zheng, Y. Ding, M. Ibrayev, M. Juluru, A. Kumar, A. Girdhar, D. Shen, A. Shrivastava, R. Feris, D. Parikh, and A. Jain. Contphy: Continuum physical concept learning and reasoning from videos. In *Proc. ICML*, 2024.

[1340] S. Xiao, J. Wang, L. Ji, L. Chen, S. Yang, S. Geng, W. Ji, and T.-S. Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proc. CVPR*, pages 9777–9786, 2021.

[1341] M. Grunde-McLaughlin, R. Krishna, and M. Agrawala. Agqa: A benchmark for compositional spatio-temporal reasoning. In *Proc. CVPR*, pages 11287–11297, 2021.

[1342] M. Grunde-McLaughlin, R. Krishna, and M. Agrawala. Agqa 2.0: An updated benchmark for compositional spatio-temporal reasoning, 2022. URL https://arxiv.org/abs/2204.06105.

[1343] L. Xu, H. Huang, and J. Liu. Sutd-trafficqa: A question answering benchmark and an efficient network for video reasoning over traffic events. In *Proc. CVPR*, pages 9861–9870, 2021.

[1344] Chengshu Li, Fei Xia, Roberto Martín-Martín, Michael Lingelbach, Sanjana Srivastava, Bokui Shen, Kent Vainio, Cem Gokmen, Gokul Dharan, Tanish Jain, Andrey Kurenkov, C. Karen Liu, Hyowon Gweon, Jiajun Wu, Li Fei-Fei, and Silvio Savarese. iGibson 2.0: Object-centric simulation for robot learning of everyday household tasks. In *Proceedings of the 5th Conference on Robot Learning*, 2021. doi:10.48550/arXiv.2108.03272. URL https://arxiv.org/abs/2108.03272.

[1345] Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. ManiSkill2: A unified benchmark for generalizable manipulation skills. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2302.04659.

[1346] S. James, Z. Ma, D. R. Arrojo, A. J. Davison, and E. Johns. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020.

[1347] S. Mohanty, E. Nygren, F. Laurent, M. Schneider, C. Scheller, N. Bhattacharya, J. Watson, A. Egli, C. Eichenberger, et al. Flatland-rl: Multi-agent reinforcement learning on trains, 2020. URL https://arxiv.org/abs/2012.05893.

[1348] L. Geretti, S. Bak, S. Bogomolov, X. Chen, G. Frehse, T. A. Henzinger, et al. ARCH-COMP 2024 category report: Continuous and hybrid systems with nonlinear dynamics. In *Proc. ARCH Workshop (EPiC Series in Computing)*, 2024.

[1349] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[1350] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2048–2057, 2015.

[1351] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004, 2016.

[1352] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, and Tat-Seng Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5659–5667, 2017.

[1353] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021.

[1354] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:9355–9366, 2021.

[1355] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[1356] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[1357] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2990–2999, 2016.

[1358] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2901–2910, 2017.

[1359] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017.

[1360] Drew A. Hudson and Christopher D. Manning. Compositional attention networks for machine reasoning. In *International Conference on Learning Representations (ICLR)*, 2018. URL https://openreview.net/forum?id=S1Euwz-Rb.

[1361] Drew Hudson and Christopher D Manning. Learning by abstraction: The neural state machine. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[1362] Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Neural module networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 39–48, 2016.

[1363] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2989–2998, 2017.

[1364] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

[1365] Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14953–14962, 2023.

[1366] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500, 2023.

[1367] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

[1368] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 28, 2015.

[1369] Taco S. Cohen and Max Welling. Steerable CNNs. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rJQKYt5ll.

[1370] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

[1371] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[1372] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

[1373] Maurice Weiler and Gabriele Cesa. General e (2)-equivariant steerable cnns. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[1374] Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks. *arXiv preprint arXiv:2207.09453*, 2022.

[1375] Ziqiao Meng, Liang Zeng, Zixing Song, Tingyang Xu, Peilin Zhao, and Irwin King. Towards geometric normalization techniques in se(3) equivariant graph neural networks for physical dynamics simulations. In Kate Larson, editor, *Proceedings of the Thirty-Third International Joint Conference*

on Artificial Intelligence, IJCAI-24, pages 5981–5989, 8 2024. doi:10.24963/ijcai.2024/661. URL https://doi.org/10.24963/ijcai.2024/661. Main Track.

[1376] Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 9323–9332, 2021.

[1377] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1912–1920, 2015.

[1378] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1588–1597, 2019.

[1379] Raghunathan Ramakrishnan, Pavlo O. Dral, Matthias Rupp, and O. Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1(1):140022, 2014.

[1380] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13(1):2453, 2022.

[1381] Ilyes Batatia, Dávid P. Kovács, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[1382] Omri Puny, Matan Atzmon, Heli Ben-Hamu, Ishan Misra, Aditya Grover, Edward J. Smith, and Yaron Lipman. Frame averaging for invariant and equivariant network design. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[1383] Stephen Gould, Dylan Campbell, Itzik Ben-Shabat, Chamin Hewa Koneputugodage, and Zhiwei Xu. Exploiting problem structure in deep declarative networks: Two case studies. *arXiv preprint arXiv:2202.12404*, 2022.

[1384] Brandon Amos and J Zico Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 136–145, 2017.

[1385] Po-Wei Wang, Priya Donti, Bryan Wilder, and Zico Kolter. Satnet: Bridging deep learning and logical reasoning using a differentiable satisfiability solver. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 6545–6554, 2019.

[1386] Brandon Amos, Ivan Jimenez, Jacob Sacks, Byron Boots, and J Zico Kolter. Differentiable mpc for end-to-end planning and control. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[1387] Akshay Agrawal, Brandon Amos, Shane Barratt, Stephen Boyd, Steven Diamond, and J Zico Kolter. Differentiable convex optimization layers. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

[1388] Steven Diamond and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[1389] Stephen Gould, Richard Hartley, and Dylan Campbell. Deep declarative networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(8):3988–4004, 2021.

[1390] Dylan Campbell, Liu Liu, and Stephen Gould. Solving the blind perspective-n-point problem end-to-end with robust differentiable geometric optimization. In *European Conference on Computer Vision*, pages 244–261, 2020.

[1391] Hikaru Shindo, Viktor Pfanschilling, Devendra Singh Dhami, and Kristian Kersting. Learning differentiable logic programs for abstract visual reasoning. *Machine Learning*, 113(11):8533–8584, 2024.

[1392] Zico Kolter, David Duvenaud, and Matt Johnson. Chapter 5: Differentiable optimization. https://implicit-layers-tutorial.org/differentiable_optimization/, 2020.

[1393] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. Deep equilibrium models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[1394] Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky T Q Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, et al. Theseus: A library for differentiable nonlinear optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[1395] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[1396] Jianqing Fan, Zhaoran Wang, Yuchen Xie, and Zhuoran Yang. A theoretical analysis of deep q-learning. In *Learning for dynamics and control*, pages 486–489, 2020.

[1397] Juan C Caicedo and Svetlana Lazebnik. Active object localization with deep reinforcement learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2488–2496, 2015.

[1398] Aviv Tamar, Yi Wu, Garrett Thomas, Sergey Levine, and Pieter Abbeel. Value iteration networks. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 29, 2016.

[1399] Lisa Lee, Emilio Parisotto, Devendra Singh Chaplot, Eric Xing, and Ruslan Salakhutdinov. Gated path planning networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 2947–2955, 2018.

[1400] Peter Karkus, David Hsu, and Wee Sun Lee. Qmdp-net: Deep learning for planning under partial observability. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[1401] Michael L Littman, Anthony R Cassandra, and Leslie Pack Kaelbling. Learning policies for partially observable environments: Scaling up. In *Machine Learning Proceedings 1995*, pages 362–370, 1995.

[1402] Junzi Zhang, Jongho Kim, Brendan O'Donoghue, and Stephen Boyd. Sample efficient reinforcement learning with reinforce. In *Proceedings of the AAAI conference on artificial intelligence*, pages 10887–10895, 2021.

[1403] Devendra Singh Chaplot, Emilio Parisotto, and Ruslan Salakhutdinov. Active neural localization. *arXiv preprint arXiv:1801.08214*, 2018.

[1404] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied Question Answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–10, 2018.

[1405] Saurabh Gupta, James Davidson, Sergey Levine, Rahul Sukthankar, and Jitendra Malik. Cognitive mapping and planning for visual navigation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2616–2625, 2017.

[1406] Michael Janner, Karthik Narasimhan, and Regina Barzilay. Representation learning for grounded spatial reasoning. *Transactions of the Association for Computational Linguistics*, 6:49–61, 2018.

[1407] Vinicius Zambaldi, David Raposo, Adam Santoro, Victor Bapst, Yujia Li, Igor Babuschkin, Karl Tuyls, David Reichert, Timothy Lillicrap, Edward Lockhart, et al. Deep reinforcement learning with relational inductive biases. In *International conference on learning representations*, 2019.

[1408] Thomas Degris, Patrick M Pilarski, and Richard S Sutton. Model-free reinforcement learning with continuous action in practice. In *2012 American control conference (ACC)*, pages 2177–2182, 2012.

[1409] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1928–1937, 2016.

[1410] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi. Target-driven visual navigation in indoor scenes using deep reinforcement learning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3357–3364, 2017.

[1411] Jingwei Zhang, Lei Tai, Ming Liu, Joschka Boedecker, and Wolfram Burgard. Neural slam: Learning to explore with external memory. *arXiv preprint arXiv:1706.09520*, 2017.

[1412] Piotr Mirowski, Razvan Pascanu, Fabio Viola, Hubert Soyer, Andrew J Ballard, Andrea Banino, Misha Denil, Ross Goroshin, Laurent Sifre, Koray Kavukcuoglu, et al. Learning to navigate in complex environments. *arXiv preprint arXiv:1611.03673*, 2016.

[1413] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[1414] Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. DD-PPO: Learning near-perfect pointgoal navigators from 2.5 billion frames. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[1415] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-Language Navigation: Interpreting Visually-Grounded Navigation Instructions in Real Environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3674–3683, 2018.

[1416] Devendra Singh Chaplot, Dhiraj Prakashchand Gandhi, Abhinav Gupta, and Russ R. Salakhutdinov. Object Goal Navigation Using Goal-Oriented Semantic Exploration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[1417] Edward Beeching, Jilles Dibangoye, Olivier Simonin, and Christian Wolf. Learning to plan with uncertain topological maps. In *European Conference on Computer Vision*, pages 473–490, 2020.

[1418] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:53728–53741, 2023.

[1419] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[1420] Xiwen Liang, Min Lin, Weiqi Ruan, Rongtao Xu, Yuecheng Liu, Jiaqi Chen, Bingqian Lin, Yuzheng Zhuang, and Xiaodan Liang. Structured preference optimization for vision-language long-horizon task planning. *arXiv preprint arXiv:2502.20742*, 2025.

[1421] Zhijian Huang, Tao Tang, Shaoxiang Chen, Sihao Lin, Zequn Jie, Lin Ma, Guangrun Wang, and Xiaodan Liang. Making large language models better planners with reasoning-decision alignment. In *European Conference on Computer Vision*, pages 73–90, 2024.

[1422] Han Qiu, Peng Gao, Lewei Lu, Xiaoqin Zhang, Ling Shao, and Shijian Lu. Spatial preference rewarding for mllms spatial understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 720–730, 2025.

[1423] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[1424] Huajie Tan, Yuheng Ji, Xiaoshuai Hao, Xiansheng Chen, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Reason-rft: Reinforcement fine-tuning for visual reasoning of vision language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[1425] Chen Gao, Liankai Jin, Xingyu Peng, Jiazhao Zhang, Yue Deng, Annan Li, He Wang, and Si Liu. Octonav: Towards generalist embodied navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 40074–40084, 2026.

[1426] Kexian Tang, Junyao Gao, Yanhong Zeng, Haodong Duan, Yanan Sun, Zhening Xing, Wenran Liu, Kaifeng Lyu, and Kai Chen. Lego-puzzles: How good are mllms at multi-step spatial reasoning? *arXiv preprint arXiv:2503.19990*, 2025.

[1427] Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. Spacer: Reinforcing mllms in video spatial reasoning. *arXiv preprint arXiv:2504.01805*, 2025.

[1428] Emilio Parisotto and Ruslan Salakhutdinov. Neural map: Structured memory for deep reinforcement learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[1429] Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, and Ruslan Salakhutdinov. Learning to explore using active neural slam. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[1430] Fengda Zhu, Xiwen Liang, Yi Zhu, Qizhi Yu, Xiaojun Chang, and Xiaodan Liang. Soon: Scenario oriented object navigation with graph-based exploration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12689–12699, 2021.

[1431] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, pages 6379–6390, 2017.

[1432] Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 2974–2982, 2018.

[1433] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorization for deep multi-agent reinforcement learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 4295–4304, 2018.

[1434] Anthony Goeckner, Yueyuan Sui, Nicolas Martinet, Xinliang Li, and Qi Zhu. Graph neural network-based multi-agent reinforcement learning for resilient distributed coordination of multi-robot systems, 2024. arXiv:2403.13093.

[1435] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder de Witt, Gregory Farquhar, Nantas Nardelli, Tim G. J. Rudner, Chia-Man Hung, Philip H. S. Torr, Jakob Foerster, and Shimon Whiteson. The starcraft multi-agent challenge. In *Proceedings of the International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 2186–2188, 2019.

[1436] Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zając, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. Google research football: A novel reinforcement learning environment. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 4501–4510, 2020.

[1437] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinicius Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *Proceedings of the International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 2085–2087, 2018.

[1438] Jiechuan Jiang, Chen Dun, Tiejun Huang, and Zongqing Lu. Graph convolutional reinforcement learning for multi-agent cooperation, 2018. arXiv:1810.09202.

[1439] Muning Wen, Jakub Grudzien Kuba, Runji Lin, Weinan Zhang, Ying Wen, Jun Wang, and Yaodong Yang. Multi-agent reinforcement learning is a sequence modeling problem. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[1440] Joseph Suarez, Yilun Du, Igor Mordatch, and Phillip Isola. Neural MMO v1.3: A massively multiagent game environment for training and evaluating neural networks. In *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems, AAMAS*, pages 2020–2022, 2020.

[1441] Joseph Suarez, David Bloomin, Kyoung Whan Choe, Hao Xiang Li, Ryan Sullivan, Nishaanth Kanna, Daniel Scott, Rose S. Shuman, Herbie Bradley, Louis Castricato, Phillip Isola, Chenghui Yu, Yuhao Jiang, Qimai Li, Jiaxin Chen, and Xiaolong Zhu. Neural MMO 2.0: A massively multi-task addition to massively multi-agent learning. In *Advances in Neural Information Processing Systems*, pages 50094–50104, 2023.

[1442] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In *International conference on machine learning*, pages 5887–5896, 2019.

[1443] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178):1–51, 2020.

[1444] Jianhao Wang, Zhizhou Ren, Terry Liu, Yang Yu, and Chongjie Zhang. Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062*, 2020.

[1445] Haoyi Zhu, , Honghui Yang, Yating Wang, Jiange Yang, Limin Wang, and Tong He. Spa: 3d spatial-awareness enables effective embodied representation. *arXiv preprint arxiv:2410.08208*, 2024.

[1446] Daniel Bogdoll, Yitian Yang, Tim Joseph, Melih Yazgan, and J Marius Zollner. Muvo: A multimodal generative world model for autonomous driving with geometric representations. In *2025 IEEE Intelligent Vehicles Symposium (IV)*, pages 2243–2250, 2025.

[1447] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T. Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

[1448] Xinhang Liu, Yu-Wing Tai, and Chi-Keung Tang. Agentic 3d scene generation with spatially contextualized vlms. *arXiv preprint arXiv:2505.20129*, 2025.

[1449] Fangjun Li, David C Hogg, and Anthony G Cohn. Advancing spatial reasoning in large language models: An in-depth evaluation and enhancement using the stepgame benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 18500–18507, 2024.

[1450] Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Sharon Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:75392–75421, 2024.

[1451] Disheng Liu, Tuo Liang, Zhe Hu, Jierui Peng, Yiren Lu, Yi Xu, Yun Fu, and Yu Yin. Deconstructing spatial intelligence in vision-language models. 2025.

[1452] Yuan-Hong Liao, Rafid Mahmood, Sanja Fidler, and David Acuna. Reasoning paths with reference objects elicit quantitative spatial reasoning in large vision-language models. *arXiv preprint arXiv:2409.09788*, 2024.

[1453] Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. Compositional chain-of-thought prompting for large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14420–14431, 2024.

[1454] Jianwei Yang, Hao Zhang, Feng Li, Xueyan Zou, Chunyuan Li, and Jianfeng Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv preprint arXiv:2310.11441*, 2023.

[1455] Dingning Liu, Cheng Wang, Peng Gao, Renrui Zhang, Xinzhu Ma, Yuan Meng, and Zhihui Wang. 3daxisprompt: Promoting the 3d grounding and reasoning in gpt-4o. *Neurocomputing*, 637:130072, 2025.

[1456] Haochen Wang, Anlin Zheng, Yucheng Zhao, Tiancai Wang, Zheng Ge, Xiangyu Zhang, and Zhaoxiang Zhang. Reconstructive visual instruction tuning. *arXiv preprint arXiv:2410.09575*, 2024.

[1457] Junfei Wu, Jian Guan, Kaituo Feng, Qiang Liu, Shu Wu, Liang Wang, Wei Wu, and Tieniu Tan. Reinforcing spatial reasoning in vision-language models with interwoven thinking and visual drawing. *arXiv preprint arXiv:2506.09965*, 2025.

[1458] Yifan Shen, Yuanzhe Liu, Jingyuan Zhu, Xu Cao, Xiaofeng Zhang, Yixiao He, Wenming Ye, James Matthew Rehg, and Ismini Lourentzou. Fine-grained preference optimization improves spatial reasoning in vlms. *arXiv preprint arXiv:2506.21656*, 2025.

[1459] Peiyao Wang and Haibin Ling. Svqa-r1: Reinforcing spatial reasoning in mllms via view-consistent reward optimization. *arXiv preprint arXiv:2506.01371*, 2025.

[1460] Weiyun Wang, Yiming Ren, Haowen Luo, Tiantong Li, Chenxiang Yan, Zhe Chen, Wenhai Wang, Qingyun Li, Lewei Lu, Xizhou Zhu, et al. The all-seeing project v2: Towards general relation comprehension of the open world. In *European Conference on Computer Vision*, pages 471–490, 2024.

[1461] Yihong Tang, Ao Qu, Zhaokai Wang, Dingyi Zhuang, Zhaofeng Wu, Wei Ma, Shenhao Wang, Yunhan Zheng, Zhan Zhao, and Jinhua Zhao. Sparkle: Mastering basic spatial capabilities in vision language models elicits generalization to spatial reasoning. *Findings of the Association for Computational Linguistics: EMNLP*, 2025:4083–4103, 2025.

[1462] Haojun Jiang, Yuanze Lin, Dongchen Han, Shiji Song, and Gao Huang. Pseudo-q: Generating pseudo language queries for visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15513–15523, 2022.

[1463] Qiushan Guo, Shalini De Mello, Hongxu Yin, Wonmin Byeon, Ka Chun Cheung, Yizhou Yu, Ping Luo, and Sifei Liu. Regiongpt: Towards region understanding vision language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13796–13806, 2024.

[1464] Xiaoyan Wang, Zeju Li, Yifan Xu, Jiaxing Qi, Zhifei Yang, Ruifei Ma, Xiangde Liu, and Chao Zhang. Spatial 3d-llm: Progressive spatial awareness for advanced 3d vision-language understanding. *arXiv preprint*, 2025.

[1465] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*, 2024.

[1466] Jiayu Tang, Yuchen Zhou, and Chao Gou. Spatialsv: Internalizing interpretable 3d spatial awareness in mllms via task-oriented visual supervision. *arXiv preprint arXiv:2606.19915*, 2026.

[1467] Ji Hyeok Jung, Eun Tae Kim, Seoyeon Kim, Joo Ho Lee, Bumsoo Kim, and Buru Chang. Is 'right'right? enhancing object orientation understanding in multimodal large language models through egocentric instruction tuning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14257–14267, 2025.

[1468] Shehreen Azad, Yash Jain, Rishit Garg, Vibhav Vineet, and Yogesh Rawat. Understanding depth and height perception in large visual-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3611–3620, 2025.

[1469] Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Jairam Vedagiri IYER, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:87310–87356, 2024.

[1470] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 9161–9175, 2023.

[1471] Ilias Stogiannidis, Steven McDonagh, and Sotirios A Tsaftaris. Mind the gap: Benchmarking spatial reasoning in vision-language models. *arXiv preprint arXiv:2503.19707*, 2025.

[1472] Baiqiao Yin, Qineng Wang, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, et al. Spatial mental modeling from limited views. In *Structural Priors for Vision Workshop at ICCV'25*, 2025.

[1473] xAI Team. Grok-1.5v vision preview. https://x.ai/news/grok-1.5v, 2024. Accessed: 2024.

[1474] Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, Xinqiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. *arXiv preprint arXiv:2506.03135*, 2025.

[1475] Guangzhi Wang, Yixiao Ge, Xiaohan Ding, Mohan Kankanhalli, and Ying Shan. What makes for good visual tokenizers for large language models? *arXiv preprint arXiv:2305.12223*, 2023.

[1476] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578, 2024.

[1477] Yuchen Zhou, Jiayu Tang, Shuo Yang, Xiaoyan Xiao, Yuqin Dai, Wenhao Yang, Chao Gou, Xiaobo Xia, and Tat-Seng Chua. Logic unseen: Revealing the logical blindspots of vision-language models. *arXiv preprint arXiv:2508.11317*, 2025.

[1478] Zehan Wang, Sashuai Zhou, Shaoxuan He, Haifeng Huang, Lihe Yang, Ziang Zhang, Xize Cheng, Shengpeng Ji, Tao Jin, Hengshuang Zhao, et al. Spatialclip: Learning 3d-aware image representations from spatially discriminative language. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29656–29666, 2025.

[1479] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023.

[1480] Palaash Agrawal, Haidi Azaman, and Cheston Tan. Stupd: A synthetic dataset for spatial and temporal relation reasoning. *arXiv preprint arXiv:2309.06680*, 2023.

[1481] Phillip Y Lee, Jihyeon Je, Chanho Park, Mikaela Angelina Uy, Leonidas Guibas, and Minhyuk Sung. Perspective-aware reasoning in vision-language models via mental imagery simulation. *arXiv preprint arXiv:2504.17207*, 2025.

[1482] Dapeng Zhang, Jing Sun, Chenghui Hu, Xiaoyan Wu, Zhenlong Yuan, Rui Zhou, Fei Shen, and Qingguo Zhou. Pure vision language action (vla) models: A comprehensive survey. *arXiv preprint arXiv:2509.19012*, 2025.

[1483] Ranjan Sapkota, Yang Cao, Konstantinos I Roumeliotis, and Manoj Karkee. Vision-language-action models: Concepts, progress, applications and challenges. *arXiv preprint arXiv:2505.04769*, 2025.

[1484] Chengmeng Li, Junjie Wen, Yan Peng, Yaxin Peng, Feifei Feng, and Yichen Zhu. Pointvla: Injecting the 3d world into vision-language-action models. *arXiv preprint arXiv:2503.07511*, 2025.

[1485] Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. *arXiv preprint arXiv:2412.10345*, 2024.

[1486] Julen Urain, Niklas Funk, Jan Peters, and Georgia Chalvatzaki. Se (3)-diffusionfields: Learning smooth cost functions for joint grasp and motion optimization through diffusion. *arXiv preprint arXiv:2209.03855*, 2022.

[1487] Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3d diffuser actor: Policy diffusion with 3d scene representations. *arXiv preprint arXiv:2402.10885*, 2024.

[1488] Jiawen Yu, Hairuo Liu, Qiaojun Yu, Jieji Ren, Ce Hao, Haitong Ding, Guangyu Huang, Guofan Huang, Yan Song, Panpan Cai, et al. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation. *arXiv preprint arXiv:2505.22159*, 2025.

[1489] Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. Vip: Towards universal visual reward and representation via value-implicit pre-training. *arXiv preprint arXiv:2210.00030*, 2022.

[1490] An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024.

[1491] Borong Zhang, Yuhao Zhang, Jiaming Ji, Yingshan Lei, Josef Dai, Yuanpei Chen, and Yaodong Yang. Safevla: Towards safety alignment of vision-language-action model via safe reinforcement learning. *arXiv e-prints*, pages arXiv–2503, 2025.

[1492] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023.

[1493] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024.

[1494] Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction for robotics. *arXiv preprint arXiv:2406.10721*, 2024.

[1495] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. In *International Conference on Machine Learning (ICML)*, 2023.

[1496] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

[1497] Jae-Woo Choi, Youngwoo Yoon, Hyobin Ong, Jaehong Kim, and Minsu Jang. Lota-bench: Benchmarking language-oriented task planners for embodied agents. *arXiv preprint arXiv:2402.08178*, 2024.

[1498] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903, 2024.

[1499] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021.

[1500] Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 653–660, 2024.

[1501] Sai Srinivas Kancheti, Aditya Sanjiv Kanade, Vineeth N. Balasubramanian, and Tanuja Ganu. Chain-of-thought degrades visual spatial reasoning capabilities of multimodal llms. *arXiv preprint arXiv:2604.16060*, 2026.

[1502] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:11809–11822, 2023.

[1503] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, et al. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, pages 17682–17690, 2024.

[1504] Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*, 2022.

[1505] Zhenyu Pan, Haozheng Luo, Manling Li, and Han Liu. Chain-of-action: Faithful and multimodal question answering through large language models. *arXiv preprint arXiv:2403.17359*, 2024.

[1506] Hanxu Hu, Hongyuan Lu, Huajian Zhang, Yun-Ze Song, Wai Lam, and Yue Zhang. Chain-of-symbol prompting elicits planning in large langauge models. *arXiv preprint arXiv:2305.10276*, 2023.

[1507] Hongjie Zhang, Hourui Deng, Jie Ou, and Chaosheng Feng. Mitigating spatial hallucination in large language models for path planning via prompt engineering. *Scientific Reports*, 15(1):8881, 2025.

[1508] Wei Chen, Yancheng Long, Mingqiao Liu, Haojie Ding, Yankai Yang, Hongyang Wei, Yi-Fan Zhang, Bin Wen, Fan Yang, Tingting Gao, Han Li, and Long Chen. Spatial chain-of-thought: Bridging understanding and generation models for spatial reasoning generation. *arXiv preprint arXiv:2602.11980*, 2026.

[1509] Joy Hsu, Jiayuan Mao, Joshua B Tenenbaum, Noah D Goodman, and Jiajun Wu. What makes a maze look like a maze? In *International Conference on Learning Representations (ICLR)*, 2025.

[1510] Bingqian Lin, Yunshuang Nie, Ziming Wei, Jiaqi Chen, Shikui Ma, Jianhua Han, Hang Xu, Xiaojun Chang, and Xiaodan Liang. Navcot: Boosting llm-based vision-and-language navigation via learning disentangled reasoning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[1511] Qingxiang Liu, Ting Huang, Zeyu Zhang, and Hao Tang. Nav-r1: Reasoning and navigation in embodied scenes, 2025. URL https://arxiv.org/abs/2509.10884.

[1512] Guanxing Lu, Ziwei Wang, Changliu Liu, Jiwen Lu, and Yansong Tang. Thinkbot: Embodied instruction following with thought chain reasoning, 2023. URL https://arxiv.org/abs/2312.07062.

[1513] Yao Mu, Qinglong Zhang, Mengkang Hu, Wenhai Wang, Mingyu Ding, Jun Jin, Bin Wang, Jifeng Dai, Yu Qiao, and Ping Luo. Embodiedgpt: Vision-language pre-training via embodied chain of thought, 2023. URL https://arxiv.org/abs/2305.15021.

[1514] Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Ming-Yu Liu, Donglai Xiang, Gordon Wetzstein, and Tsung-Yi Lin. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models, 2025. URL https://arxiv.org/abs/2503.22020.

[1515] Linqing Zhong, Yi Liu, Yifei Wei, Ziyu Xiong, Maoqing Yao, Si Liu, and Guanghui Ren. ACoT-VLA: Action chain-of-thought for vision-language-action models. *arXiv preprint arXiv:2601.11404*, 2026.

[1516] Yi Zhang, Qiang Zhang, Xiaozhu Ju, Zhaoyang Liu, Jilei Mao, Jingkai Sun, Jintao Wu, Shixiong Gao, Shihan Cai, Zhiyuan Qin, Linkai Liang, Jiaxu Wang, Yiqun Duan, Jiahang Cao, Renjing Xu, and Jian Tang. Embodiedvsr: Dynamic scene graph-guided chain-of-thought reasoning for visual spatial tasks, 2025. URL https://arxiv.org/abs/2503.11089.

[1517] Binbin Ji, Siddharth Agrawal, Qiance Tang, and Yvonne Wu. Enhancing spatial reasoning in vision-language models via chain-of-thought prompting and reinforcement learning. *arXiv preprint arXiv:2507.13362*, 2025.

[1518] Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic ai for spatial reasoning with a dynamic api. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19446–19455, 2025.

[1519] Jianing Yang, Xuweiyi Chen, Shengyi Qian, Nikhil Madaan, Madhavan Iyengar, David F Fouhey, and Joyce Chai. Llm-grounder: Open-vocabulary 3d visual grounding with large language model as an agent. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7694–7701, 2024.

[1520] Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[1521] Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. VirtualHome: Simulating household activities via programs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8494–8502, 2018.

[1522] Huanyu Zhang, Chengzu Li, Wenshan Wu, Shaoguang Mao, Yifan Zhang, Haochen Tian, Ivan Vulić, Zhang Zhang, Liang Wang, Tieniu Tan, and Furu Wei. Scaling and beyond: Advancing spatial reasoning in mllms requires new recipes, 2025. URL https://arxiv.org/abs/2504.15037.

[1523] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale, 2023. URL https://arxiv.org/abs/2212.06817.

[1524] Seokju Cho, Ryo Hachiuma, Abhishek Badki, Hang Su, Byung-Kwan Lee, Chan Hee Song, Sifei Liu, Subhashree Radhakrishnan, Seungryong Kim, Yu-Chiang Frank Wang, and Min-Hung Chen. SpatialClaw: Rethinking action interface for agentic spatial reasoning. *arXiv preprint arXiv:2606.13673*, 2026.

[1525] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Noah Brown, Tomas Jackson, Linda Luu, Sergey Levine, Karol Hausman, and Brian Ichter. Inner monologue: Embodied reasoning through planning with language models, 2022. URL https://arxiv.org/abs/2207.05608.

[1526] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. Progprompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11523–11530, 2023.

[1527] Zhutian Yang, Caelan Garrett, Dieter Fox, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Guiding long-horizon task and motion planning with vision language models, 2024. URL https://arxiv.org/abs/2410.02193.

[1528] Yongchao Chen, Jacob Arkin, Charles Dawson, Yang Zhang, Nicholas Roy, and Chuchu Fan. Autotamp: Autoregressive task and motion planning with llms as translators and checkers, 2024. URL https://arxiv.org/abs/2306.06531.

[1529] Vineet Bhat, Ali Umut Kaypak, Prashanth Krishnamurthy, Ramesh Karri, and Farshad Khorrami. Grounding llms for robot task planning using closed-loop state feedback, 2024. URL https://arxiv.org/abs/2402.08546.

[1530] Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and Ranjay Krishna. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models, 2024. URL https://arxiv.org/abs/2406.09403.

[1531] Ziyang Xiao, Dongxiang Zhang, Xiongwei Han, Xiaojin Fu, Wing Yin Yu, Tao Zhong, Sai Wu, Yuan Wang, Jianwei Yin, and Gang Chen. Enhancing llm reasoning via vision-augmented prompting. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:28772–28797, 2024.

[1532] Changye Li, Meng Lu, Yi Wu, and Ligeng Zhu. Perceive, interact, reason: Building tool-augmented visual agents for spatial reasoning. *arXiv preprint arXiv:2606.12830*, 2026.

[1533] Yalun Dai, Hao Li, Shulin Tian, Runmao Yao, Yuhao Dong, Fangzhou Hong, Zhaoxi Chen, Fangfu Liu, Baoliang Tian, Dingwen Zhang, Tao Wang, Kim-Hui Yap, and Ziwei Liu. S-Agent: Spatial tool-use elicits reasoning for spatial intelligence. *arXiv preprint arXiv:2606.20515*, 2026.

[1534] Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. Llm+p: Empowering large language models with optimal planning proficiency, 2023. URL https://arxiv.org/abs/2304.11477.

[1535] Zhun Yang, Adam Ishay, and Joohyung Lee. Coupling large language models with logic programming for robust and general reasoning from text, 2023. URL https://arxiv.org/abs/2307.07696.

[1536] Ruibo Liu, Jason Wei, Shixiang Shane Gu, Te-Yen Wu, Soroush Vosoughi, Claire Cui, Denny Zhou, and Andrew M Dai. Mind's eye: Grounded language model reasoning through simulation. *arXiv preprint arXiv:2210.05359*, 2022.

[1537] Zhun Yang, Adam Ishay, and Joohyung Lee. Neurasp: Embracing neural networks into answer set programming, 2023. URL https://arxiv.org/abs/2307.07700.

[1538] Tanawan Premsri and Parisa Kordjamshidi. Neuro-symbolic training for reasoning over spatial language. In *Findings of the Association for Computational Linguistics: NAACL 2025*, page 2395–2414, 2025. doi:10.18653/v1/2025.findings-naacl.128. URL http://dx.doi.org/10.18653/v1/2025.findings-naacl.128.

[1539] Mingcong Lei, Yiming Zhao, Ge Wang, Zhixin Mai, Shuguang Cui, Yatong Han, and Jinke Ren. Stma: A spatio-temporal memory agent for long-horizon embodied task planning. *arXiv preprint arXiv:2502.10177*, 2025.

[1540] Weizheng Wang, Ike Obi, and Byung-Cheol Min. Multi-agent llm actor-critic framework for social robot navigation, 2025. URL https://arxiv.org/abs/2503.09758.

[1541] Ananth Hariharan, Vardhan Dongre, Dilek Hakkani-Tür, and Gokhan Tur. Plan verification for llm-based embodied task completion agents. *arXiv preprint arXiv:2509.02761*, 2025.

[1542] Antoni Rosinol, Arjun Gupta, Marcus Abate, Jingnan Shi, and Luca Carlone. 3d dynamic scene graphs: Actionable spatial perception with places, objects, and humans. *arXiv preprint arXiv:2002.06289*, 2020.

[1543] Zhiyu Li, Shichao Song, Chenyang Xi, Hanyu Wang, Chen Tang, Simin Niu, Ding Chen, Jiawei Yang, Chunyu Li, Qingchen Yu, et al. Memos: A memory os for ai system. *arXiv preprint arXiv:2507.03724*, 2025.

[1544] Abrar Anwar, John Welsh, Joydeep Biswas, Soha Pouya, and Yan Chang. Remembr: Building and reasoning over long-horizon spatio-temporal memory for robot navigation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2838–2845, 2025.

[1545] Xingyu Tan, Xiaoyang Wang, Qing Liu, Xiwei Xu, Xin Yuan, Liming Zhu, and Wenjie Zhang. Memotime: Memory-augmented temporal knowledge graph enhanced large language model reasoning. *arXiv preprint arXiv:2510.13614*, 2025.

[1546] Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, et al. Cambrian-s: Towards spatial supersensing in video. *arXiv preprint arXiv:2511.04670*, 2025.

[1547] Siyi Chen, Mikaela Angelina Uy, Chan Hee Song, Faisal Ladhak, Adithyavairavan Murali, Qing Qu, Stan Birchfield, Valts Blukis, and Jonathan Tremblay. Spacetools: Tool-augmented spatial reasoning via double interactive rl. *arXiv preprint arXiv:2512.04069*, 2025.

[1548] Xiao-Wen Yang, Jie-Jing Shao, Lan-Zhe Guo, Bo-Wen Zhang, Zhi Zhou, Lin-Han Jia, Wang-Zhou Dai, and Yu-Feng Li. Neuro-symbolic artificial intelligence: Towards improving the reasoning abilities of large language models. *arXiv preprint arXiv:2508.13678*, 2025.

[1549] Zihan Wang, Yaohui Zhu, Gim Hee Lee, and Yachun Fan. Navrag: Generating user demand instructions for embodied navigation through retrieval-augmented llm. *arXiv preprint arXiv:2502.11142*, 2025.

[1550] Shi Yu, Chaoyue Tang, Bokai Xu, Junbo Cui, Junhao Ran, Yukun Yan, Zhenghao Liu, Shuo Wang, Xu Han, Zhiyuan Liu, et al. Visrag: Vision-based retrieval-augmented generation on multi-modality documents. *arXiv preprint arXiv:2410.10594*, 2024.

[1551] Fengyi Wu, Yifei Dong, Zhi-Qi Cheng, Yilong Dai, Guangyu Chen, Hang Wang, Qi Dai, and Alexander G Hauptmann. Govig: Goal-conditioned visual navigation instruction generation. *arXiv preprint arXiv:2508.09547*, 2025.

[1552] Dazhou Yu, Riyang Bao, Gengchen Mai, and Liang Zhao. Spatial-rag: Spatial retrieval augmented generation for real-world spatial reasoning questions. *arXiv preprint arXiv:2502.18470*, 2025.

[1553] Bingshuai Liu, Chenyang Lyu, Zijun Min, Zhanyu Wang, Jinsong Su, and Longyue Wang. Retrieval-augmented multi-modal chain-of-thoughts reasoning for large language models. *arXiv preprint arXiv:2312.01714*, 2023.

[1554] Nathan Hughes, Yun Chang, and Luca Carlone. Hydra: A real-time spatial perception system for 3d scene graph construction and optimization. *arXiv preprint arXiv:2201.13360*, 2022.

[1555] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028, 2024.

[1556] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning. *arXiv preprint arXiv:2307.06135*, 2023.

[1557] Quanting Xie, So Yeon Min, Pengliang Ji, Yue Yang, Tianyi Zhang, Kedi Xu, Aarav Bajaj, Ruslan Salakhutdinov, Matthew Johnson-Roberson, and Yonatan Bisk. Embodied-rag: General non-parametric embodied memory for retrieval and generation. *arXiv preprint arXiv:2409.18313*, 2024.

[1558] Mingcong Lei, Honghao Cai, Zezhou Cui, Liangchen Tan, Junkun Hong, Gehan Hu, Shuangyu Zhu, Yimou Wu, Shaohan Jiang, Ge Wang, et al. Robomemory: A brain-inspired multi-memory agentic framework for lifelong learning in physical embodied systems. In *NeurIPS 2025 Workshop on Space in Vision, Language, and Embodied AI*, 2025.

[1559] Yuncong Yang, Han Yang, Jiachen Zhou, Peihao Chen, Hongxin Zhang, Yilun Du, and Chuang Gan. 3d-mem: 3d scene memory for embodied exploration and reasoning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17294–17303, 2025.

[1560] Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 3dllm-mem: Long-term spatial-temporal memory for embodied 3d large language model. *arXiv preprint arXiv:2505.22657*, 2025.

[1561] Zuntao Liu, Yi Du, Taimeng Fu, Shaoshu Su, Cherie Ho, and Chen Wang. Vision-language memory for spatial reasoning. *arXiv preprint arXiv:2511.20644*, 2025.

[1562] Zhongyi Cai, Yi Du, Chen Wang, and Yu Kong. Vision to geometry: 3d spatial memory for sequential embodied mllm reasoning and exploration. *arXiv preprint arXiv:2512.02458*, 2025.

[1563] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16488–16498, 2024.

[1564] Ziyang Gong, Wenhao Li, Oliver Ma, Songyuan Li, Zhaokai Wang, Songyuan Li, Jiayi Ji, Xue Yang, Gen Luo, Junchi Yan, and Rongrong Ji. Space-10: A comprehensive benchmark for multimodal large language models in compositional spatial intelligence, 2025. URL https://arxiv.org/abs/2506.07966.

[1565] Wenqi Wang, Reuben Tan, Pengyue Zhu, Jianwei Yang, Zhengyuan Yang, Lijuan Wang, Andrey Kolobov, Jianfeng Gao, and Boqing Gong. Site: towards spatial intelligence thorough evaluation. *arXiv preprint arXiv:2505.05456*, 2025.

[1566] Xinmiao Huang, Qisong He, Zhenglin Huang, Boxuan Wang, Zhuoyun Li, Guangliang Cheng, Yi Dong, and Xiaowei Huang. Spatial-DISE: A unified benchmark for evaluating spatial reasoning in vision-language models. *arXiv preprint arXiv:2510.13394*, 2025.

[1567] Weichen Zhang, Zile Zhou, Xin Zeng, Xuchen Liu, Jianjie Fang, Chen Gao, Yong Li, Jinqiang Cui, Xinlei Chen, and Xiao-Ping Zhang. Open3D-VQA: A benchmark for comprehensive spatial reasoning with multimodal large language model in open space. *arXiv preprint arXiv:2503.11094*, 2025.

[1568] Yufan Mao, Hanjing Ye, Wenlong Dong, Chengjie Zhang, and Hong Zhang. Meta-memory: Retrieving and integrating semantic-spatial memories for robot spatial reasoning. *arXiv preprint arXiv:2509.20754*, 2025.

[1569] Yuecheng Liu, Dafeng Chi, Shiguang Wu, Zhanguang Zhang, Yaochen Hu, Lingfeng Zhang, Yingxue Zhang, Shuang Wu, Tongtong Cao, Guowei Huang, et al. Spatialcot: Advancing spatial reasoning through coordinate alignment and chain-of-thought for embodied task planning. *arXiv preprint arXiv:2501.10074*, 2025.

[1570] Yifu Yuan, Haiqin Cui, Yaoting Huang, Yibin Chen, Fei Ni, Zibin Dong, Pengyi Li, Yan Zheng, and Jianye Hao. Embodied-r1: Reinforced embodied reasoning for general robotic manipulation. *arXiv preprint arXiv:2508.13998*, 2025.

[1571] Jiaqi Chen, Bingqian Lin, Ran Xu, Zhenhua Chai, Xiaodan Liang, and Kwan-Yee Wong. Mapgpt: Map-guided prompting with adaptive path planning for vision-and-language navigation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9796–9810, 2024.

[1572] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:8634–8652, 2023.

[1573] Zihao Wang, Shaofei Cai, Guanzhou Chen, Anji Liu, Xiaojian Ma, and Yitao Liang. Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents. *arXiv preprint arXiv:2302.01560*, 2023.

[1574] Delong Chen, Theo Moutakanni, Willy Chung, Yejin Bang, Ziwei Ji, Allen Bolourchi, and Pascale Fung. Planning with reasoning using vision language world model. *arXiv preprint arXiv:2509.02722*, 2025.

[1575] Zhao Mandi, Shreeya Jain, and Shuran Song. Roco: Dialectic multi-robot collaboration with large language models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 286–299, 2024.

[1576] Ceyao Zhang, Kaijie Yang, Siyi Hu, Zihao Wang, Guanghe Li, Yihang Sun, Cheng Zhang, Zhaowei Zhang, Anji Liu, Song-Chun Zhu, et al. Proagent: building proactive cooperative agents with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17591–17599, 2024.

[1577] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

[1578] Yi Yang, Zhihong Liu, Siqi Kou, Yiyang Chen, Yanzhe Hu, Jianbo Zhou, Boyuan Zhao, Zhijie Wei, Xiao Xia, Xueqi Li, Pengfei Liu, and Zhijie Deng. World-language-action model for unified world modeling, language reasoning, and action synthesis. *arXiv preprint arXiv:2606.05979*, 2026.

[1579] Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Devin, Alex X Lee, Maria Bauzá, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, et al. Robocat: A self-improving generalist agent for robotic manipulation. *arXiv preprint arXiv:2306.11706*, 2023.

[1580] Zefu Lin, Rongxu Cui, Junjia Xu, Xiaojuan Jin, Wenling Li, Lue Fan, and Zhaoxiang Zhang. World pilot: Steering vision-language-action models with world-action priors. *arXiv preprint arXiv:2606.12403*, 2026.

[1581] Julius Mayer, Mohamad Ballout, Serwan Jassim, Farbod Nosrat Nezami, and Elia Bruni. iVISPAR: An interactive visual-spatial reasoning benchmark for vlms. *arXiv preprint arXiv:2502.03214*, 2025.

[1582] Ming Li, Jike Zhong, Shitian Zhao, Haoquan Zhang, Shaoheng Lin, Yuxiang Lai, Wei Chen, Konstantinos Psounis, and Kaipeng Zhang. Tir-bench: A comprehensive benchmark for agentic thinking-with-images reasoning. *arXiv preprint arXiv:2511.01833*, 2025.

[1583] Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6924–6934, 2025.

[1584] Nianchen Deng, Lixin Gu, Shenglong Ye, Yinan He, Zhe Chen, Songze Li, Haomin Wang, Xingguang Wei, Tianshuo Yang, Min Dou, et al. Internspatial: A comprehensive dataset for spatial reasoning in vision-language models. *arXiv preprint arXiv:2506.18385*, 2025.

[1585] Xingrui Wang, Wufei Ma, Tiezheng Zhang, Celso M de Melo, Jieneng Chen, and Alan Yuille. Spatial457: A diagnostic benchmark for 6d spatial reasoning of large mutimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24669–24679, 2025.

[1586] Chun-Hsiao Yeh, Chenyu Wang, Shengbang Tong, Ta-Ying Cheng, Ruoyu Wang, Tianzhe Chu, Yuexiang Zhai, Yubei Chen, Shenghua Gao, and Yi Ma. Seeing from another perspective: Evaluating multi-view understanding in mllms. *arXiv preprint arXiv:2504.15280*, 2025.

[1587] Mengfei Du, Binhao Wu, Zejun Li, Xuan-Jing Huang, and Zhongyu Wei. Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 346–355, 2024.

[1588] Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, et al. Mmsi-bench: A benchmark for multi-image spatial intelligence. *arXiv preprint arXiv:2505.23764*, 2025.

[1589] Shijie Zhou, Alexander Vilesov, Xuehai He, Ziyu Wan, Shuwang Zhang, Aditya Nagachandra, Di Chang, Dongdong Chen, Xin Eric Wang, and Achuta Kadambi. Vlm4d: Towards spatiotemporal awareness in vision language models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8600–8612, 2025.

[1590] Yun Li, Yiming Zhang, Tao Lin, XiangRui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. Sti-bench: Are mllms ready for precise spatial-temporal world understanding? *arXiv preprint arXiv:2503.23765*, 2025.

[1591] Ziang Zhang, Zehan Wang, Guanghao Zhang, Weilong Dai, Yan Xia, Ziang Yan, Minjie Hong, and Zhou Zhao. Dsi-bench: A benchmark for dynamic spatial intelligence. *arXiv preprint arXiv:2510.18873*, 2025.

[1592] Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. Dragapart: Learning a part-level motion prior for articulated objects. In *ECCV*, 2024.

[1593] Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. Puppet-master: Scaling interactive video generation as a motion prior for part-level dynamics. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2025.

[1594] Aditya Vora, Sauradip Nag, and Hao Zhang. Articulate that object part (atop): 3d part articulation via text and motion personalization. *arXiv preprint arXiv:2502.07278*, 2025.

[1595] Ruijie Lu, Yu Liu, Jiaxiang Tang, Junfeng Ni, Yuxiang Wang, Diwen Wan, Gang Zeng, Yixin Chen, and Siyuan Huang. Dreamart: Generating interactable articulated objects from a single image. *arXiv preprint arXiv:2507.05763*, 2025.

[1596] Shoubin Yu, Yue Zhang, Zun Wang, Jaehong Yoon, Huaxiu Yao, Mingyu Ding, and Mohit Bansal. When and how much to imagine: Adaptive test-time scaling with world models for visual spatial reasoning. *arXiv preprint arXiv:2602.08236*, 2026.

[1597] Meng Cao, Xingyu Li, Xue Liu, Ian Reid, and Xiaodan Liang. Spatialdreamer: Incentivizing spatial reasoning via active mental imagery. *arXiv preprint arXiv:2512.07733*, 2025.

[1598] Siyuan Zhou, Yilun Du, Yuncong Yang, Lei Han, Peihao Chen, Dit-Yan Yeung, and Chuang Gan. Learning 3d persistent embodied world models. *arXiv preprint arXiv:2505.05495*, 2025.

[1599] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 2023.

[1600] Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, et al. Dreamgen: Unlocking generalization in robot learning through video world models. *arXiv preprint arXiv:2505.12705*, 2025.

[1601] Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Yiping Lu, Zhengyuan Yang, Lijuan Wang, Ranjay Krishna, Jiajun Wu, Li Fei-Fei, Yejin Choi, and Manling Li. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[1602] Zhangquan Chen, Manyuan Zhang, Xinlei Yu, Xufang Luo, Mingze Sun, Zihao Pan, Xiang An, Yan Feng, Peng Pei, Xunliang Cai, and Ruqi Huang. Think with 3d: Geometric imagination grounded spatial reasoning from limited views. *arXiv preprint arXiv:2510.18632*, 2025.

[1603] Saurav Jha, M. Jehanzeb Mirza, Wei Lin, Shiqi Yang, and Sarath Chandar. Probing the effectiveness of world models for spatial reasoning through test-time scaling. *arXiv preprint arXiv:2512.05809*, 2025.

[1604] Ningwei Xie, Zizi Tian, Lei Yang, Xiao-Ping Zhang, Meng Guo, and Jie Li. From 2d to 3d cognition: A brief survey of general world models. *arXiv preprint arXiv:2506.20134*, 2025.

[1605] Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. *arXiv preprint arXiv:2506.18903*, 2025.

[1606] Nate Gillman, Charles Herrmann, Michael Freeman, Daksh Aggarwal, Evan Luo, Deqing Sun, and Chen Sun. Force prompting: Video generation models can learn and generalize physics-based control signals, 2025.

[1607] Wanyue Zhang, Wenxiang Wu, Wang Xu, Jiaxin Luo, Helu Zhi, Yibin Huang, Shuo Ren, Zitao Liu, and Jiajun Zhang. World2vlm: Distilling world model imagination into vlms for dynamic spatial reasoning. *arXiv preprint arXiv:2604.26934*, 2026.

[1608] Baining Zhao, Ziyou Wang, Jianjie Fang, Chen Gao, Fanhang Man, Jinqiang Cui, Xin Wang, Xinlei Chen, Yong Li, and Wenwu Zhu. Embodied-r: Collaborative framework for activating embodied spatial reasoning in foundation models via reinforcement learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 11071–11080, 2025.

[1609] Anoop Cherian, Radu Corcodel, Siddarth Jain, and Diego Romeres. LLMPhy: Parameter-Identifiable Physical Reasoning Combining Large Language Models and Physics Engines. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2026.

[1610] Yang Jin, Jun Lv, Shuqiang Jiang, and Cewu Lu. Diffgen: Robot demonstration generation via differentiable physics simulation, differentiable rendering, and vision-language model. *arXiv preprint arXiv:2405.07309*, 2024.

[1611] Weijie Zhou, Manli Tao, Chaoyang Zhao, Haiyun Guo, Honghui Dong, Ming Tang, and Jinqiao Wang. Physvlm: Enabling visual language models to understand robotic physical reachability. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6940–6949, 2025.

[1612] Jingquan Wang, Andrew Negrut, Harry Zhang, Khailanii Slaton, Shu Wang, Radu Serban, Jinlong Wu, and Dan Negrut. Chronollm: Customizing language models for physics-based simulation code generation. *arXiv preprint arXiv:2508.13975*, 2025.

[1613] Jiayi Lin, Jiabo Huang, and Shaogang Gong. Neuro-symbolic spatial reasoning in segmentation. *arXiv preprint arXiv:2510.15841*, 2025.

[1614] Yunzhi Zhang, Zizhang Li, Matt Zhou, Shangzhe Wu, and Jiajun Wu. The scene language: Representing scenes with programs, words, and embeddings. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24625–24634, 2025.

[1615] Chenghao Liu, Zhimu Zhou, Jiachen Zhang, Minghao Zhang, Songfang Huang, and Huiling Duan. Msnav: Zero-shot vision-and-language navigation with dynamic memory and llm spatial reasoning. *arXiv preprint arXiv:2508.16654*, 2025.

[1616] Daixun Li, Yusi Zhang, Mingxiang Cao, Donglai Liu, Weiying Xie, Tianlin Hui, Lunkai Lin, Zhiqiang Xie, and Yunsong Li. Towards long-horizon vision-language-action system: Reasoning, acting and memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6839–6848, 2025.

[1617] Hantao Zhang, Jingyang Liu, and Ed Li. See it. say it. sorted: Agentic system for compositional diagram generation. *arXiv preprint arXiv:2508.15222*, 2025.

[1618] Bui Duc Manh, Soumyaratna Debnath, Zetong Zhang, Shriram Damodaran, Arvind Kumar, Yueyi Zhang, Lu Mi, Erik Cambria, and Lin Wang. Mind meets space: Rethinking agentic spatial intelligence from a neuroscience-inspired perspective. *arXiv preprint arXiv:2509.09154*, 2025.

[1619] Hyunjun Kim and Sooyoung Ryu. Drawingbench: Evaluating spatial reasoning and ui interaction capabilities of large language models through mouse-based drawing tasks. *arXiv preprint arXiv:2512.01174*, 2025.

[1620] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. *arXiv preprint arXiv:2210.07474*, 2022.

[1621] Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. AI2-THOR: An interactive 3d environment for visual AI. *arXiv preprint arXiv:1712.05474*, 2017.

[1622] Florian Bordes, Quentin Garrido, Justine T Kao, Adina Williams, Michael Rabbat, and Emmanuel Dupoux. Intphys 2: Benchmarking intuitive physics understanding in complex synthetic environments. *arXiv preprint arXiv:2506.09849*, 2025.

[1623] Luis S Piloto, Ari Weinstein, Peter Battaglia, and Matthew Botvinick. Intuitive physics learning in a deep-learning model inspired by developmental psychology. *Nature human behaviour*, 6(9):1257–1267, 2022.

[1624] Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models understand physical principles? *arXiv preprint arXiv:2501.09038*, 2025.

[1625] Philipp D. Siedler. Sphyr: Spatial-physical reasoning benchmark on material distribution. *arXiv preprint arXiv:2505.16048*, 2025.

[1626] Nils Hoehing, Mayug Maniparambil, Ellen Rushe, Noel E O'Connor, and Anthony Ventresque. Understanding space is rocket science–only top reasoning models can solve spatial understanding tasks. *arXiv preprint arXiv:2509.02175*, 2025.

[1627] Michael Ogezi and Freda Shi. Spare: Enhancing spatial reasoning in vision-language models with synthetic data. *CoRR*, abs/2504.20648, 2025.

[1628] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pages 148–166, 2024.

[1629] Kexin Tian, Jingrui Mao, Yunlong Zhang, Jiwan Jiang, Yang Zhou, and Zhengzhong Tu. Nuscenes-spatialqa: A spatial understanding and reasoning benchmark for vision-language models in autonomous driving. *arXiv preprint arXiv:2504.03164*, 2025.

[1630] Jieyu Zhang, Le Xue, Linxin Song, Jun Wang, Weikai Huang, Manli Shu, An Yan, Zixian Ma, Juan Carlos Niebles, Silvio Savarese, Caiming Xiong, Zeyuan Chen, Ranjay Krishna, and Ran Xu. Provision: Programmatically scaling vision-centric instruction data for multimodal language models. *CoRR*, abs/2412.07012, 2024.

[1631] Wei Liang, Yibiao Zhao, Yixin Zhu, and Song-Chun Zhu. What is where: Inferring containment relations from videos. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 3418–3424, 2016.

[1632] Xianda Guo, Ruijun Zhang, Yiqun Duan, Yuhang He, Dujun Nie, Wenke Huang, Chenming Zhang, Shuai Liu, Hao Zhao, and Long Chen. Surds: Benchmarking spatial understanding and reasoning in driving scenarios with vision language models. *arXiv preprint arXiv:2411.13112*, 2024.

[1633] Zelin Xu, Yupu Zhang, Saugat Adhikari, Saiful Islam, Tingsong Xiao, Zibo Liu, Shigang Chen, Da Yan, and Zhe Jiang. Earthspatialbench: Benchmarking spatial reasoning capabilities of multimodal llms on earth imagery. *arXiv preprint arXiv:2602.15918*, 2026.

[1634] Jinkai Li, Zhenliang Zhang, Lifeng Fan, and Wei Wang. Purespace: A benchmark for abstract spatial reasoning in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[1635] Arijit Ray, Jiafei Duan, Reuben Tan, Dina Bashkirova, Rose Hendrix, Kiana Ehsani, Aniruddha Kembhavi, Bryan A Plummer, Ranjay Krishna, Kuo-Hao Zeng, et al. Sat: Spatial aptitude training for multimodal language models. *arXiv preprint arXiv:2412.07755*, 2024.

[1636] Yuechen Xie, Xiaoyan Zhang, Yicheng Shan, Zhu Hao, Rui Tang, Rong Wei, Mingli Song, Yuanyu Wan, and Jie Song. Spatialqa: A benchmark for evaluating spatial logical reasoning in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2645–2657, 2026.

[1637] Ruiyuan Lyu, Jingli Lin, Tai Wang, Shuai Yang, Xiaohan Mao, Yilun Chen, Runsen Xu, Haifeng Huang, Chenming Zhu, Dahua Lin, et al. Mmscan: A multi-modal 3d scene dataset with hierarchical grounded language annotations. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:50898–50924, 2024.

[1638] Emilia Szymańska, Mihai Dusmanu, Jan-Willem Buurlage, Mahdi Rad, and Marc Pollefeys. Space3d-bench: Spatial 3d question answering benchmark. In *European Conference on Computer Vision*, pages 68–85, 2024.

[1639] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3193–3203, 2021.

[1640] Weili Guan, Haoyu Zhang, Meng Liu, Qianlong Xiang, Yaowei Wang, and Liqiang Nie. Spaceera++: A unified framework towards 3d spatial reasoning in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[1641] Mohsen Gholami, Ahmad Rezaei, Zhou Weimin, Sitong Mao, Shunbo Zhou, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. *arXiv preprint arXiv:2509.06266*, 2025.

[1642] Haoning Wu, Xiao Huang, Yaohui Chen, Ya Zhang, Yanfeng Wang, and Weidi Xie. Spatialscore: Towards comprehensive evaluation for spatial intelligence. *arXiv preprint arXiv:2505.17012*, 2025.

[1643] Chuang Gan, Jeremy Schwartz, Seth Alter, Damian Mrowca, Martin Schrimpf, James Traer, Julian De Freitas, Jonas Kubilius, Abhishek Bhandwaldar, Nick Haber, et al. Threedworld: A platform for interactive multi-modal physical simulation. *arXiv preprint arXiv:2007.04954*, 2020.

[1644] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. *arXiv preprint arXiv:2010.07954*, 2020.

[1645] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4195–4205, 2023.

[1646] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2672–2680, 2014.

[1647] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5907–5915, 2017.

[1648] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1316–1324, 2018.

[1649] Han Zhang, Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Cross-modal contrastive learning for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18501–18511, 2021.

[1650] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *PMLR*, pages 8821–8831, 2021.

[1651] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang M. Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022.

[1652] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, José Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T. Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

[1653] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 16784–16804, 2022.

[1654] Aditya Ramesh et al. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[1655] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[1656] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

[1657] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023.

[1658] Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations (ICLR)*, 2023.

[1659] Patrick Esser et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024.

[1660] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022.

[1661] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023.

[1662] Kaiwen Luo et al. Latent consistency models: Synthesizing high-fidelity images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

[1663] OpenAI. The new chatgpt images is here, December 2025. URL https://openai.com/index/new-chatgpt-images-is-here/.

[1664] Google. Image generation with gemini, 2024. URL https://gemini.google/overview/image-generation/.

[1665] Google DeepMind. Imagen, 2024. URL https://deepmind.google/models/imagen/.

[1666] Midjourney, Inc. Midjourney v7 alpha, April 2025. URL https://www.midjourney.com/updates/v7-alpha.

[1667] Black Forest Labs. Flux.2 [max] - top-tier quality image generation, December 2025. URL https://bfl.ai/models/flux-2-max.

[1668] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[1669] Maria E. Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, pages 722–729, 2008.

[1670] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R. Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[1671] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:36479–36494, 2022.

[1672] Eslam Mohamed Bakr, Pengzhan Sun, Xiaoqian Shen, Faizan Farooq Khan, Li Erran Li, and Mohamed Elhoseiny. HRS-Bench: Holistic, reliable and scalable benchmark for text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20041–20053, 2023.

[1673] Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Benita Teufel, Marco Bellagente, Minguk Kang, Taesung Park, Jure Leskovec, Jun-Yan Zhu, Li Fei-Fei, Jiajun Wu, Stefano Ermon, and Percy Liang. Holistic evaluation of text-to-image models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1674] Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. TIFA: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 20406–20417, 2023.

[1675] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1676] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1677] Tejas Gokhale, Hamid Palangi, Besmira Nushi, Vibhav Vineet, Eric Horvitz, Ece Kamar, Chitta Baral, and Yezhou Yang. Benchmarking spatial relationships in text-to-image generation. *arXiv preprint arXiv:2212.10015*, 2022.

[1678] Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. ELLA: Equip diffusion models with LLM for enhanced semantic alignment. *arXiv preprint arXiv:2403.05135*, 2024.

[1679] Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, and Deva Ramanan. Evaluating and improving compositional text-to-visual generation. *arXiv preprint arXiv:2406.13743*, 2024.

[1680] Yuwei Niu, Munan Ning, Mengren Zheng, Bin Lin, Peng Jin, Jiaqi Liao, Kunpeng Ning, Bin Zhu, and Li Yuan. WISE: A world knowledge-informed semantic evaluation for text-to-image generation. *arXiv preprint arXiv:2503.07265*, 2025.

[1681] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 6626–6637, 2017.

[1682] Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. *Advances in neural information processing systems*, 31, 2018.

[1683] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019.

[1684] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7514–7528, 2021.

[1685] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

[1686] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1687] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1688] Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[1689] Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. VIEScore: Towards explainable metrics for conditional image synthesis evaluation. *arXiv preprint arXiv:2312.14867*, 2023.

[1690] Yujie Lu, Xianjun Yang, Xiujun Li, Xin Eric Wang, and William Yang Wang. LLMScore: Unveiling the power of large language models in text-to-image synthesis evaluation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1691] Agneet Chatterjee, Gabriela Ben Melech Stan, Estelle Aflalo, Sayak Paul, Dhruba Ghosh, Tejas Gokhale, Ludwig Schmidt, Hannaneh Hajishirzi, Vasudev Lal, Chitta Baral, et al. Getting it right: Improving spatial consistency in text-to-image models. In *European Conference on Computer Vision*, pages 204–222, 2024.

[1692] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. MultiDiffusion: Fusing Diffusion Paths for Controlled Image Generation. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1737–1752, 2023.

[1693] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.

[1694] Chong Mou et al. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023.

[1695] Guangcong Zheng, Xianpan Zhou, Xuewei Li, Zhongang Qi, Ying Shan, and Xi Li. Layoutdiffusion: Controllable diffusion model for layout-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22490–22499, 2023.

[1696] Leigang Qu, Shengqiong Wu, Hao Fei, Liqiang Nie, and Tat-Seng Chua. Layoutllm-t2i: Eliciting layout guidance from llm for text-to-image generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 643–654, 2023.

[1697] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. GLIGEN: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22511–22521, 2023.

[1698] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018.

[1699] Dustin Podell et al. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[1700] Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. BoxDiff: Text-to-image synthesis with training-free box-constrained diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7452–7461, 2023.

[1701] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-based semantic guidance for text-to-image diffusion models. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

[1702] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Reddy Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for

compositional text-to-image synthesis. In *International Conference on Learning Representations (ICLR)*, 2023.

[1703] Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36: 18225–18250, 2023.

[1704] Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal LLMs. *arXiv preprint arXiv:2401.11708*, 2024.

[1705] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11461–11471, 2022.

[1706] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18208–18218, 2022.

[1707] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with Fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2149–2159, 2022.

[1708] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *International Conference on Learning Representations (ICLR)*, 2023.

[1709] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6007–6017, 2023.

[1710] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6038–6047, 2023.

[1711] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. URL https://arxiv.org/abs/2211.09800.

[1712] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1713] Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu Edit: Precise image editing via recognition and generation tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8871–8881, 2024.

[1714] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by Example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18381–18391, 2023.

[1715] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. IP-Adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.

[1716] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your GAN: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023.

[1717] Yujun Shi, Chuhui Xue, Jun Hao Liew, Jiachun Pan, Hanshu Yan, Wenqing Zhang, Vincent Y.F̃. Tan, and Song Bai. DragDiffusion: Harnessing diffusion models for interactive point-based image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8839–8849, 2024.

[1718] Su Wang, Chitwan Saharia, Ceslee Montgomery, Jordi Pont-Tuset, Shai Noy, Stefano Pellegrini, Yasumasa Onoe, Sarah Laszlo, David J Fleet, Radu Soricut, et al. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *2023 ieee/cvf conference on computer vision and pattern recognition (cvpr)*, pages 18359–18369, 2023.

[1719] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? Investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6048–6058, 2023.

[1720] Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y. Zhao. Glaze: Protecting artists from style mimicry by text-to-image models. In *Proceedings of the 32nd USENIX Security Symposium*, pages 2187–2204, 2023.

[1721] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21469–21480, 2025.

[1722] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:31841–31854, 2022.

[1723] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 29, 2016.

[1724] Bo Yang, Hongkai Wen, Sen Wang, Ronald Clark, Andrew Markham, and Niki Trigoni. 3d object reconstruction from a single depth view with adversarial learning. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 679–688, 2017.

[1725] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Josh Tenenbaum, and Bill Freeman. Visual object networks: Image generation with disentangled 3d representations. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

[1726] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning efficient point cloud generation for dense 3d object reconstruction. In *proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[1727] Zitian Huang, Yikuan Yu, Jiawen Xu, Feng Ni, and Xinyi Le. Pf-net: Point fractal network for 3d point cloud completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7662–7670, 2020.

[1728] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C Berg. Transformation-grounded image generation network for novel 3d view synthesis. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 3500–3509, 2017.

[1729] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7588–7597, 2019.

[1730] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5799–5809, 2021.

[1731] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. Gram: Generative radiance manifolds for 3d-aware image generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10673–10683, 2022.

[1732] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11453–11464, 2021.

[1733] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 33:20154–20166, 2020.

[1734] Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2416–2425, 2023.

[1735] J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20875–20886, 2023.

[1736] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4563–4573, 2023.

[1737] Christopher Wewer, Kevin Raj, Eddy Ilg, Bernt Schiele, and Jan Eric Lenssen. latentsplat: Autoencoding variational gaussians for fast generalizable 3d reconstruction. In *European conference on computer vision*, pages 456–473, 2024.

[1738] Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulò, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3dg: Latent 3d gaussian diffusion. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024.

[1739] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10324–10335, 2024.

[1740] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 867–876, 2022.

[1741] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[1742] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CV)*, pages 300–309, 2023.

[1743] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12619–12629, 2023.

[1744] Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6796–6807, 2024.

[1745] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1746] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[1747] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2349–2359, 2023.

[1748] Junyoung Seo, Wooseok Jang, Min-Seop Kwak, Hyeonsu Kim, Jaehoon Ko, Junho Kim, Jin-Hwa Kim, Jiyoung Lee, and Seungryong Kim. Let 2d diffusion model know 3d-consistency for robust text-to-3d generation. *arXiv preprint arXiv:2303.07937*, 2023.

[1749] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023.

[1750] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023.

[1751] Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023.

[1752] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023.

[1753] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:22226–22246, 2023.

[1754] Yukang Lin, Haonan Han, Chaoqun Gong, Zunnan Xu, Yachao Zhang, and Xiu Li. Consistent123: One image to highly consistent 3d asset using case-aware diffusion priors. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6715–6724, 2024.

[1755] Hanzhe Hu, Zhizhuo Zhou, Varun Jampani, and Shubham Tulsiani. Mvd-fusion: Single-view 3d via depth-consistent multi-view generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9698–9707, 2024.

[1756] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023.

[1757] Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

[1758] Chuanxia Zheng and Andrea Vedaldi. Free3d: Consistent novel view synthesis without 3d representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9720–9731, 2024.

[1759] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023.

[1760] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent

multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10072–10083, 2024.

[1761] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*, 2023.

[1762] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.

[1763] Xuying Zhang, Yupeng Zhou, Kai Wang, Yikai Wang, Zhen Li, Shaohui Jiao, Daquan Zhou, Qibin Hou, and Ming-Ming Cheng. Ar-1-to-3: Single image to consistent 3d object via next-view prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26273–26283, 2025.

[1764] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023.

[1765] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:35799–35813, 2023.

[1766] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023.

[1767] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18, 2024.

[1768] Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. Crm: Single image to 3d textured mesh with convolutional reconstruction model. In *European conference on computer vision*, pages 57–74, 2024.

[1769] Hao Wang, Ying Zhou, Haoyu Zhao, Rui Wang, Qiang Hu, Xing Zhang, Qiang Li, and Zhiwei Wang. Moving light adaptive colonoscopy reconstruction via illumination-attenuation-aware 3d gaussian splatting. *arXiv preprint arXiv:2510.18739*, 2025.

[1770] Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, et al. 3dtopia-xl: Scaling high-quality 3d asset generation via primitive diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26576–26586, 2025.

[1771] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions On Graphics (TOG)*, 42(4):1–16, 2023.

[1772] Chenchen Ye, Lizi Liao, Fuli Feng, Wei Ji, and Tat-Seng Chua. Structured and natural responses co-generation for conversational search. In *ACM SIGIR*, 2022.

[1773] Yuanbo Yang, Jiahao Shao, Xinyang Li, Yujun Shen, Andreas Geiger, and Yiyi Liao. Prometheus: 3d-aware latent diffusion models for feed-forward text-to-3d scene generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2857–2869, 2025.

[1774] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024.

[1775] Yiyang Shen, Kun Zhou, He Wang, Yin Yang, and Tianjia Shao. High-fidelity 3d object generation from single image with rgbn-volume gaussian reconstruction model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21558–21569, 2025.

[1776] Tencent Hunyuan3D Team. Hunyuan3d 1.0: A unified framework for text-to-3d and image-to-3d generation, 2024.

[1777] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[1778] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2553–2560, 2022.

[1779] Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 803–814, 2023.

[1780] Qi Zuo, Xiaodong Gu, Yuan Dong, Zhengyi Zhao, Weihao Yuan, Lingteng Qiu, Liefeng Bo, and Zilong Dong. High-fidelity 3d textured shapes generation by sparse encoding and adversarial decoding. In *European Conference on Computer Vision*, pages 52–69, 2024.

[1781] Xiao Cai, Sitong Su, Jingkuan Song, Pengpeng Zeng, Ji Zhang, Qinhong Du, Mengqi Li, Heng Tao Shen, and Lianli Gao. Gt23d-bench: A comprehensive general text-to-3d generation benchmark. *arXiv preprint arXiv:2412.09997*, 2024.

[1782] Yujie Zhang, Bingyang Cui, Qi Yang, Zhu Li, and Yiling Xu. Benchmarking and learning multi-dimensional quality evaluator for text-to-3d generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18563–18574, 2025.

[1783] Haohui Li, Bowen Qu, and Wei Gao. T23d-qa: An open dataset and benchmark for text-driven 3d generation quality assessment. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 13398–13404, 2025.

[1784] Shalini Maiti, Lourdes Agapito, and Filippos Kokkinos. Gen3deval: Using vllms for automatic evaluation of generated 3d objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18552–18562, 2025.

[1785] Zixuan Huang, Mark Boss, Aaryaman Vasishta, James M Rehg, and Varun Jampani. Spar3d: Stable point-aware reconstruction of 3d objects from single images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16860–16870, 2025.

[1786] Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151*, 2024.

[1787] Tencent Hunyuan3D Team. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation, 2025.

[1788] Raphael Bensadoun, Tom Monnier, Yanir Kleiman, Filippos Kokkinos, Yawar Siddiqui, Mahendra Kariya, Omri Harosh, Roman Shapovalov, Benjamin Graham, Emilien Garreau, et al. Meta 3d gen. *arXiv preprint arXiv:2407.02599*, 2024.

[1789] Team Hunyuan3D, Bowen Zhang, Chunchao Guo, Haolin Liu, Hongyu Yan, Huiwen Shi, Jingwei Huang, Junlin Yu, Kunhong Li, Penghao Wang, et al. Hunyuan3d-omni: A unified framework for controllable generation of 3d assets. *arXiv preprint arXiv:2509.21245*, 2025.

[1790] Xinpeng Wang, Chandan Yeshwanth, and Matthias Nießner. Sceneformer: Indoor scene generation with transformers. In *2021 International Conference on 3D Vision (3DV)*, pages 106–115, 2021.

[1791] Jiapeng Tang, Yinyu Nie, Lev Markhasin, Angela Dai, Justus Thies, and Matthias Nießner. Diffuscene: Denoising diffusion models for generative indoor scene synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20507–20518, 2024.

[1792] Fan-Yun Sun, Weiyu Liu, Siyi Gu, Dylan Lim, Goutam Bhat, Federico Tombari, Manling Li, Nick Haber, and Jiajun Wu. Layoutvlm: Differentiable optimization of 3d layout via vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29469–29478, 2025.

[1793] Gege Gao, Weiyang Liu, Anpei Chen, Andreas Geiger, and Bernhard Schölkopf. Graphdreamer: Compositional 3d scene synthesis from scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21295–21304, 2024.

[1794] Guangyao Zhai, Evin Pınar Örnek, Shun-Cheng Wu, Yan Di, Federico Tombari, Nassir Navab, and Benjamin Busam. Commonscenes: Generating commonsense 3d indoor scenes with scene graph diffusion. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:30026–30038, 2023.

[1795] Guangyao Zhai, Evin Pınar Örnek, Dave Zhenyu Chen, Ruotong Liao, Yan Di, Nassir Navab, Federico Tombari, and Benjamin Busam. Echoscene: Indoor scene generation via information echo over scene graph diffusion. In *European Conference on Computer Vision*, pages 167–184, 2024.

[1796] Chenguo Lin and Yadong Mu. Instructscene: Instruction-driven 3d indoor scene synthesis with semantic graph prior. *arXiv preprint arXiv:2402.04717*, 2024.

[1797] Jie Deng, Wenhao Chai, Junsheng Huang, Zhonghan Zhao, Qixuan Huang, Mingyan Gao, Jianshu Guo, Shengyu Hao, Wenhao Hu, Jenq-Neng Hwang, et al. Citycraft: A real crafter for 3d city generation. *arXiv preprint arXiv:2406.04983*, 2024.

[1798] Xiaoyu Zhou, Xingjian Ran, Yajiao Xiong, Jinlin He, Zhiwei Lin, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Gala3d: Towards text-to-3d complex scene generation via layout-guided generative gaussian splatting. *arXiv preprint arXiv:2402.07207*, 2024.

[1799] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4578–4587, 2021.

[1800] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19457–19467, 2024.

[1801] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 370–386, 2024.

[1802] Yuedong Chen, Chuanxia Zheng, Haofei Xu, Bohan Zhuang, Andrea Vedaldi, Tat-Jen Cham, and Jianfei Cai. Mvsplat360: Feed-forward 360 scene synthesis from sparse views. In *Neural Information Processing Systems (NeurIPS)*, 2024.

[1803] Haoran Li, Haolin Shi, Wenli Zhang, Wenjun Wu, Yong Liao, Lin Wang, Lik-hang Lee, and Peng Yuan Zhou. Dreamscene: 3d gaussian-based text-to-3d scene generation via formation pattern sampling. In *European Conference on Computer Vision*, pages 214–230, 2024.

[1804] Katja Schwarz, Norman Müller, and Peter Kontschieder. Generative gaussian splatting: Generating 3d scenes with video diffusion priors. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV) (ICCV)*, 2025.

[1805] Mehdi SM Sajjadi, Henning Meyer, Etienne Pot, Urs Bergmann, Klaus Greff, Noha Radwan, Suhani Vora, Mario Lučić, Daniel Duckworth, Alexey Dosovitskiy, et al. Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6229–6238, 2022.

[1806] Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. In *The Thirteenth International Conference on Learning Representations*, 2025.

[1807] Jason J Yu, Fereshteh Forghani, Konstantinos G Derpanis, and Marcus A Brubaker. Long-term photometric consistent novel view synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7094–7104, 2023.

[1808] Jensen Jinghao Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *arXiv preprint arXiv:2503.14489*, 2025.

[1809] Xin Kong, Daniel Watson, Yannick Strümpler, Michael Niemeyer, and Federico Tombari. Causnvs: Autoregressive multi-view diffusion for flexible 3d novel view synthesis. *arXiv preprint arXiv:2509.06579*, 2025.

[1810] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024.

[1811] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.

[1812] Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024.

[1813] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[1814] Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26057–26068, 2025.

[1815] Daniel Watson, Saurabh Saxena, Lala Li, Andrea Tagliasacchi, and David J Fleet. Controlling space and time with diffusion models. *arXiv preprint arXiv:2407.07860*, 2024.

[1816] Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, and Dacheng Tao. 3D-FRONT: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10933–10942, 2021.

[1817] Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3D-FUTURE: 3d furniture shape with texture. *International Journal of Computer Vision*, 129(12): 3313–3337, 2021.

[1818] Dani Valevski, Yaniv Leviathan, Moab Arar, and Shlomi Fruchter. Diffusion models are real-time game engines. *arXiv preprint arXiv:2408.14837*, 2024.

[1819] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2405.12399.

[1820] World Labs. Rtfm: A real-time frame model. https://www.worldlabs.ai/blog/rtfm, 2025. Accessed: 14 December 2025.

[1821] Jiahui Zhang, Yuelei Li, Anpei Chen, Muyu Xu, Kunhao Liu, Jianyuan Wang, Xiao-Xiao Long, Hanxue Liang, Zexiang Xu, Hao Su, et al. Advances in feed-forward 3d reconstruction and view synthesis: A survey. *arXiv preprint arXiv:2507.14501*, 2025.

[1822] Robert C Bolles, H Harlyn Baker, and David H Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International journal of computer vision (IJCV)*, 1(1): 7–55, 1987.

[1823] Andrew W Fitzgibbon and Andrew Zisserman. Automatic camera recovery for closed or open image sequences. In *European conference on computer vision (ECCV)*, pages 311–326, 1998.

[1824] Shimon Ullman. *The interpretation of visual motion.* 1979.

[1825] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015.

[1826] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pages 501–518, 2016.

[1827] Julian Chibane, Gerard Pons-Moll, et al. Neural unsigned distance fields for implicit function learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:21638–21652, 2020.

[1828] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems (NeurIPS)*, 35:25018–25032, 2022.

[1829] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision (ECCV)*, pages 523–540, 2020.

[1830] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021.

[1831] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 5470–5479, 2022.

[1832] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5752–5761, 2021.

[1833] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 5481–5490, 2022.

[1834] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European conference on computer vision (ECCV)*, pages 333–350, 2022.

[1835] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 5480–5490, 2022.

[1836] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 12892–12901, 2022.

[1837] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 12882–12891, 2022.

[1838] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5438–5448, 2022.

[1839] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5589–5599, 2021.

[1840] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Realfusion: 360deg reconstruction of any object from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 8446–8455, 2023.

[1841] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 12663–12673, 2023.

[1842] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 22819–22829, 2023.

[1843] Jaeyoung Chung, Jeongtaek Oh, and Kyoung Mu Lee. Depth-regularized optimization for 3d gaussian splatting in few-shot images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 811–820, 2024.

[1844] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pages 1–11, 2024.

[1845] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5354–5363, 2024.

[1846] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 38(10):2024–2039, 2015.

[1847] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision (ECCV)*, pages 628–644, 2016.

[1848] Haoqiang Fan, Hao Su, and Leonidas J Guibas. A point set generation network for 3d object reconstruction from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 605–613, 2017.

[1849] Georgia Gkioxari, Jitendra Malik, and Justin Johnson. Mesh r-cnn. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 9785–9795, 2019.

[1850] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (CVPR)*, pages 8092–8101, 2019.

[1851] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 17627–17638, 2023.

[1852] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 5974–5983, 2017.

[1853] Jure Žbontar and Yann LeCun. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research*, 17(65):1–32, 2016.

[1854] Zhenggang Tang, Yuchen Fan, Dilin Wang, Hongyu Xu, Rakesh Ranjan, Alexander Schwing, and Zhicheng Yan. Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

[1855] Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2025.

[1856] Vincent Sitzmann, Semon Rezchikov, Bill Freeman, Josh Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:19313–19325, 2021.

[1857] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Light field neural rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8269–8279, 2022.

[1858] Mohammed Suhail, Carlos Esteves, Leonid Sigal, and Ameesh Makadia. Generalizable patch-based neural rendering. In *European Conference on Computer Vision (ECCV)*, pages 156–174, 2022.

[1859] Hanwen Jiang, Hao Tan, Peng Wang, Haian Jin, Yue Zhao, Sai Bi, Kai Zhang, Fujun Luan, Kalyan Sunkavalli, Qixing Huang, et al. Rayzer: A self-supervised large view synthesis model. In *International Conference on Computer Vision (ICCV)*, October 2025.

[1860] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 14124–14133, 2021.

[1861] Haofei Xu, Anpei Chen, Yuedong Chen, Christos Sakaridis, Yulun Zhang, Marc Pollefeys, Andreas Geiger, and Fisher Yu. Murf: Multi-baseline radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20041–20050, 2024.

[1862] Yuedong Chen, Haofei Xu, Qianyi Wu, Chuanxia Zheng, Tat-Jen Cham, and Jianfei Cai. Explicit correspondence matching for generalizable neural radiance fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2025.

[1863] Yilun Du, Cameron Smith, Ayush Tewari, and Vincent Sitzmann. Learning to render novel views from wide-baseline stereo pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4970–4980, 2023.

[1864] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10208–10217, 2024.

[1865] Stanislaw Szymanowicz, Eldar Insafutdinov, Chuanxia Zheng, Dylan Campbell, João F Henriques, Christian Rupprecht, and Andrea Vedaldi. Flash3d: Feed-forward generalisable 3d scene reconstruction from a single image. In *International Conference on 3D Vision (3DV)*, 2025.

[1866] Brandon Smart, Chuanxia Zheng, Iro Laina, and Victor Adrian Prisacariu. Splatt3r: Zero-shot gaussian splatting from uncalibarated image pairs. *arXiv preprint arXiv:2408.13912*, 2024.

[1867] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19, 2024.

[1868] Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. In *International Conference on Learning Representations (ICLR)*, 2025.

[1869] Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. Lion: Latent point diffusion models for 3d shape generation. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:10021–10039, 2022.

[1870] Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, Bin Fu, Tao Chen, Gang Yu, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *Advances in neural information processing systems (NeurIPS)*, 36:73969–73982, 2023.

[1871] Xuanchi Ren, Jiahui Huang, Xiaohui Zeng, Ken Museth, Sanja Fidler, and Francis Williams. Xcube: Large-scale 3d generative modeling using sparse voxel hierarchies. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4209–4219, 2024.

[1872] Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *arXiv preprint arXiv:2502.06608*, 2025.

[1873] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[1874] Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9420–9429, 2024.

[1875] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. Reconfusion: 3d reconstruction with diffusion priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 21551–21561, 2024.

[1876] Matthew Wallingford, Anand Bhattad, Aditya Kusupati, Vivek Ramanujan, Matt Deitke, Aniruddha Kembhavi, Roozbeh Mottaghi, Wei-Chiu Ma, and Ali Farhadi. From an image to a scene: Learning to imagine the world from a million 360° videos. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:17743–17760, 2024.

[1877] Fangfu Liu, Wenqiang Sun, Hanyang Wang, Yikai Wang, Haowen Sun, Junliang Ye, Jun Zhang, and Yueqi Duan. Reconx: Reconstruct any scene from sparse views with video diffusion model, 2024. URL https://arxiv.org/abs/2408.16767.

[1878] Qihang Zhang, Shuangfei Zhai, Miguel Angel Bautista, Kevin Miao, Alexander Toshev, Joshua Susskind, and Jiatao Gu. World-consistent video diffusion with explicit 3d modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[1879] Stanislaw Szymanowicz, Jason Y Zhang, Pratul Srinivasan, Ruiqi Gao, Arthur Brussee, Aleksander Holynski, Ricardo Martin-Brualla, Jonathan T Barron, and Philipp Henzler. Bolt3d: Generating 3d scenes in seconds. In *International Conference on Computer Vision (ICCV)*, October 2025.

[1880] Rui Li, Biao Zhang, Zhenyu Li, Federico Tombari, and Peter Wonka. Lari: Layered ray intersections for single-view 3d geometric reasoning. *arXiv preprint arXiv:2504.18424*, 2025.

[1881] Hanwen Jiang, Zexiang Xu, Desai Xie, Ziwen Chen, Haian Jin, Fujun Luan, Zhixin Shu, Kai Zhang, Sai Bi, Xin Sun, et al. Megasynth: Scaling up 3d scene reconstruction with synthesized data. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16441–16452, 2025.

[1882] Desai Xie, Sai Bi, Zhixin Shu, Kai Zhang, Zexiang Xu, Yi Zhou, Sören Pirk, Arie Kaufman, Xin Sun, and Hao Tan. Lrm-zero: Training large reconstruction models with synthesized data. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:53285–53316, 2024.

[1883] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics*, 2019.

[1884] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[1885] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[1886] Zhe Zhang, Rui Peng, Yuxi Hu, and Ronggang Wang. Geomvsnet: Learning multi-view stereo with geometry perception. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21508–21518, 2023.

[1887] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

[1888] Zhaoxi Chen, Guangcong Wang, and Ziwei Liu. Text2light: Zero-shot text-driven hdr panorama generation. *ACM Transactions on Graphics*, 41(6):1–16, 2022.

[1889] Christopher May and Daniel Aliaga. Cubegan: omnidirectional image synthesis using generative adversarial networks. In *Computer Graphics Forum*, 2023.

[1890] Weicai Ye, Chenhao Ji, Zheng Chen, Junyao Gao, Xiaoshui Huang, Song-Hai Zhang, Wanli Ouyang, Tong He, Cairong Zhao, and Guofeng Zhang. Diffpano: Scalable and consistent text to panorama generation with spherical epipolar-aware diffusion. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[1891] Xiancheng Sun, Mai Xu, Shengxi Li, Senmao Ma, Xin Deng, Lai Jiang, and Gang Shen. Spherical manifold guided diffusion model for panoramic image generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2025.

[1892] Chaoyang Wang, Xiangtai Li, Lu Qi, Xiaofan Lin, Jinbin Bai, Qianyu Zhou, and Yunhai Tong. Conditional panoramic image generation via masked autoregressive modeling. *arXiv:2505.16862*, 2025.

[1893] Jionghao Wang, Ziyu Chen, Jun Ling, Rong Xie, and Li Song. 360-degree panorama generation from few unregistered nfov images. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6811–6821, 2023.

[1894] Tianhao Wu, Chuanxia Zheng, and Tat-Jen Cham. Panodiffusion: 360-degree panorama outpainting via diffusion. In *International Conference on Learning Representations*, 2024.

[1895] Hai Wang, Xiaoyu Xiang, Yuchen Fan, and Jing-Hao Xue. Customizing 360-degree panoramas through text-to-image diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4933–4943, 2024.

[1896] Cheng Zhang, Qianyi Wu, Camilo Cruz Gambardella, Xiaoshui Huang, Dinh Phung, Wanli Ouyang, and Jianfei Cai. Taming stable diffusion for text to 360 panorama image generation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2024.

[1897] Naofumi Akimoto, Yuhi Matsuo, and Yoshimitsu Aoki. Diverse plausible 360-degree image outpainting for efficient 3dcg background creation. In *CVPR*, 2022.

[1898] Zhuqiang Lu, Kun Hu, Chaoyue Wang, Lei Bai, and Zhiyong Wang. Autoregressive omni-aware outpainting for open-vocabulary 360-degree image generation. In *AAAI*, 2024.

[1899] Atsuya Nakata and Takao Yamanaka. 2s-odis: Two-stage omni-directional image synthesis by geometric distortion correction. *arXiv preprint arXiv:2409.09969*, 2024.

[1900] Dian Zheng, Cheng Zhang, Xiao-Ming Wu, Cao Li, Chengfei Lv, Jian-Fang Hu, and Wei-Shi Zheng. Panorama generation from nfov image done right. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21610–21619, 2025.

[1901] Nikolai Kalischek, Michael Oechsle, Fabian Manhardt, Philipp Henzler, Konrad Schindler, and Federico Tombari. Cubediff: Repurposing diffusion-based image models for panorama generation. In *International Conference on Learning Representations*, 2025.

[1902] Yukun Huang, Yanning Zhou, Jianan Wang, Kaiyi Huang, and Xihui Liu. Dreamcube: Rgb-d panorama generation via multi-plane synchronization. In *International Conference on Computer Vision*, 2025.

[1903] Wancheng Feng, Chen An, Zhenliang He, Meina Kan, Shiguang Shan, and Lukun Wang. Jopano: Unified panorama generation via joint modeling. *arXiv preprint arXiv:2512.06885*, 2025.

[1904] Jialu Li and Mohit Bansal. Panogen: Text-conditioned panoramic environment generation for vision-and-language navigation. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1905] Qinsheng Zhang, Jiaming Song, Xun Huang, Yongxin Chen, and Ming-Yu Liu. Diffcollage: Parallel generation of large content with diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023.

[1906] Aoming Liu, Zhong Li, Zhang Chen, Nannan Li, Yi Xu, and Bryan A Plummer. Panofree: Tuning-free holistic multi-view image generation with cross-view self-guidance. In *European Conference on Computer Vision*, pages 146–164, 2024.

[1907] Minho Park, Taewoong Kang, Jooyeol Yun, Sungwon Hwang, and Jaegul Choo. Spherediff: Tuning-free omnidirectional panoramic image and video generation via spherical latent representation. *arXiv:2504.14396*, 2025.

[1908] Shitao Tang, Fuayng Zhang, Jiacheng Chen, Peng Wang, and Furukawa Yasutaka. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[1909] Hakan Çapuk, Andrew Bond, Muhammed Burak Kızıl, Emir Göçen, Erkut Erdem, and Aykut Erdem. Tandit: Tangent-plane diffusion transformer for high-quality 360-degree panorama generation. *arXiv preprint arXiv:2506.21681*, 2025.

[1910] Qian Wang, Weiqi Li, Chong Mou, Xinhua Cheng, and Jian Zhang. 360dvd: Controllable panorama video generation with 360-degree video diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6913–6923, 2024.

[1911] Ke Xing, Hanwen Liang, Dejia Xu, Yuyang Yin, Konstantinos N Plataniotis, Yao Zhao, and Yunchao Wei. Tip4gen: Text to immersive panorama 4d scene generation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 9267–9276, 2025.

[1912] Yifei Xia, Shuchen Weng, Siqi Yang, Jingqi Liu, Chengxuan Zhu, Minggui Teng, Zijian Jia, Han Jiang, and Boxin Shi. Panowan: Lifting diffusion video generation models to 360 {\deg} with latitude/longitude-aware mechanisms. *arXiv preprint arXiv:2505.22016*, 2025.

[1913] Muyang Zhang, Yuzhi Chen, Rongtao Xu, Changwei Wang, JinMing Yang, Weiliang Meng, Jianwei Guo, Huihuang Zhao, and Xiaopeng Zhang. Panodit: Panoramic videos generation with diffusion transformer. In *AAAI*, pages 10040–10048, 2025.

[1914] Zeyu Dong, Yuyang Yin, Yuqi Li, Eric Li, Hao-Xiang Guo, and Yikai Wang. Panolora: Bridging perspective and panoramic video generation with lora adaptation. *arXiv preprint arXiv:2509.11092*, 2025.

[1915] Jinxiu Liu, Shaoheng Lin, Yinxiao Li, and Ming-Hsuan Yang. Dynamicscaler: Seamless and scalable video generation for panoramic scenes. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6144–6153, 2025.

[1916] Renjie Li, Panwang Pan, Bangbang Yang, Dejia Xu, Shijie Zhou, Xuanyang Zhang, Zeming Li, Achuta Kadambi, Zhangyang Wang, Zhengzhong Tu, et al. 4k4dgen: Panoramic 4d generation at 4k resolution. *arXiv preprint arXiv:2406.13527*, 2024.

[1917] Zixun Fang, Kai Zhu, Zhiheng Liu, Yu Liu, Wei Zhai, Yang Cao, and Zheng-Jun Zha. Panoramic video generation with pretrained diffusion models. *arXiv preprint arXiv:2506.23513*, 2025.

[1918] Haiyang Zhou, Wangbo Yu, Jiawen Guan, Xinhua Cheng, Yonghong Tian, and Li Yuan. Holotime: Taming video diffusion models for panoramic 4d scene generation. *arXiv preprint arXiv:2504.21650*, 2025.

[1919] Taiming Lu, Tianmin Shu, Junfei Xiao, Luoxin Ye, Jiahao Wang, Cheng Peng, Chen Wei, Daniel Khashabi, Rama Chellappa, Alan Yuille, et al. Genex: Generating an explorable world. *arXiv preprint arXiv:2412.09624*, 2024.

[1920] Cheng Zhang, Hanwen Liang, Donny Y. Chen, Qianyi Wu, Konstantinos N. Plataniotis, Camilo Cruz Gambardella, and Jianfei Cai. Panflow: Decoupled motion control for panoramic video generation. In *AAAI*, 2026.

[1921] Weiqi Li, Shijie Zhao, Chong Mou, Xuhan Sheng, Zhenyu Zhang, Qian Wang, Junlin Li, Li Zhang, and Jian Zhang. Omnidrag: Enabling motion control for omnidirectional image-to-video generation. *International Journal of Computer Vision*, 134(1):44, 2026.

[1922] Yuyang Yin, HaoXiang Guo, Fangfu Liu, Mengyu Wang, Hanwen Liang, Eric Li, Yikai Wang, Xiaojie Jin, Yao Zhao, and Yunchao Wei. Panoworld-x: Generating explorable panoramic worlds via sphere-aware video diffusion. *arXiv preprint arXiv:2509.24997*, 2025.

[1923] Jingwei Ma, Erika Lu, Roni Paiss, Shiran Zada, Aleksander Holynski, Tali Dekel, Brian Curless, Michael Rubinstein, and Forrester Cole. Vidpanos: Generative panoramic videos from casual panning videos. In *SIGGRAPH Asia 2024 Conference Papers*, December 2024.

[1924] Jing Tan, Shuai Yang, Tong Wu, Jingwen He, Yuwei Guo, Ziwei Liu, and Dahua Lin. Imagine360: Immersive 360 video generation from perspective anchor. *arXiv:2412.03552*, 2024.

[1925] Rundong Luo, Matthew Wallingford, Ali Fahardi, Noah Snavely, and Wei-Chiu Ma. Beyond the frame: Generating 360° panoramic videos from perspective videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[1926] Huajian Huang, Yingshu Chen, Tianjian Zhang, and Sai-Kit Yeung. 360roam: Real-time indoor roaming using geometry-aware 360° radiance fields. *arXiv preprint arXiv:2208.02705*, 2022.

[1927] Shreyas Kulkarni, Peng Yin, and Sebastian Scherer. 360fusionnerf: Panoramic neural radiance fields with joint guidance. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7202–7209, 2023.

[1928] Changwoon Choi, Sang Min Kim, and Young Min Kim. Balanced spherical grid for egocentric view synthesis. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 16590–16599, 2023.

[1929] Kai Gu, Thomas Maugey, Sebastian Knorr, and Christine Guillemot. Omni-nerf: neural radiance field from 360 image captures. In *2022 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2022.

[1930] Jiayang Bai, Letian Huang, Jie Guo, Wen Gong, Yuanqi Li, and Yanwen Guo. 360-gs: Layout-guided panoramic gaussian splatting for indoor roaming. *arXiv preprint arXiv:2402.00763*, 2024.

[1931] Suyoung Lee, Jaeyoung Chung, Jaeyoo Huh, and Kyoung Mu Lee. Odgs: 3d scene reconstruction from omnidirectional images with 3d gaussian splattings. In *Adv. Neural Inform. Process. Syst.*, 2024.

[1932] Letian Huang, Jiayang Bai, Jie Guo, Yuanqi Li, and Yanwen Guo. On the error analysis of 3d gaussian splatting and an optimal projection strategy. In *European Conference on Computer Vision (ECCV)*, 2024.

[1933] Longwei Li, Huajian Huang, Sai-Kit Yeung, and Hui Cheng. Omnigs: Omnidirectional gaussian splatting for fast radiance field reconstruction using omnidirectional images. *arXiv preprint arXiv:2404.03202*, 2024.

[1934] Changha Shin, Woong Oh Cho, and Seon Joo Kim. Seam360gs: Seamless 360° gaussian splatting from real-world omnidirectional images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[1935] Shuai Yang, Jing Tan, Mengchen Zhang, Tong Wu, Yixuan Li, Gordon Wetzstein, Ziwei Liu, and Dahua Lin. Scene4u: Layered panoramic reconstruction for hyper-immersive scene generation. *arXiv preprint arXiv:2408.13252*, 2024.

[1936] Zheng Chen, Yan-Pei Cao, Yuan-Chen Guo, Chen Wang, Ying Shan, and Song-Hai Zhang. Panogrf: generalizable spherical radiance fields for wide-baseline panoramas. In *Adv. Neural Inform. Process. Syst.*, volume 36, pages 6961–6985, 2023.

[1937] Zheng Chen, Zhelun Shen, Chenming Wu, Weicai Ye, Haocheng Feng, Errui Ding, and Song-Hai Zhang. Panosplatt3r: Unposed wide-baseline panoramic 3d gaussian splatting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025.

[1938] Cheng Zhang, Haofei Xu, Qianyi Wu, Camilo Cruz Gambardella, Dinh Phung, and Jianfei Cai. Pansplat: 4k panorama synthesis with feed-forward gaussian splatting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11437–11447, 2025.

[1939] Shuai Yang, Jing Tan, Mengchen Zhang, Tong Wu, Yixuan Li, Gordon Wetzstein, Ziwei Liu, and Dahua Lin. Layerpano3d: Layered 3d panorama for hyper-immersive scene generation. *arXiv preprint arXiv:2408.13252*, 2024.

[1940] Shijie Zhou, Zhiwen Fan, Dejia Xu, Haoran Chang, Pradyumna Chari, Tejas Bharadwaj, Suya You, Zhangyang Wang, and Achuta Kadambi. Dreamscene360: Unconstrained text-to-3d scene generation with panoramic gaussian splatting. In *Eur. Conf. Comput. Vis.*, pages 324–342, 2024.

[1941] Zhongqi Yang, Wenhang Ge, Yuqi Li, Jiaqi Chen, Haoyuan Li, Mengyin An, Fei Kang, Hua Xue, Baixin Xu, Yuyang Yin, et al. Matrix-3d: Omnidirectional explorable 3d world generation. *arXiv preprint arXiv:2508.08086*, 2025.

[1942] HunyuanWorld Team, Zhenwei Wang, Yuhao Liu, Junta Wu, Zixiao Gu, Haoyuan Wang, Xuhui Zuo, Tianyu Huang, Wenhuan Li, Sheng Zhang, et al. Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. *arXiv:2507.21809*, 2025.

[1943] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *European Conference on Computer Vision*, pages 519–535, 2020.

[1944] Jianxiong Xiao, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Recognizing scene viewpoint using panoramic place representation. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2695–2702, 2012.

[1945] Alexander Veicht, Paul-Edouard Sarlin, Philipp Lindenberger, and Marc Pollefeys. Geocalib: Learning single-image calibration with geometric optimization. In *European Conference on Computer Vision*, pages 1–20, 2024.

[1946] Hao Chen, Yuqi Hou, Chenyuan Qu, Irene Testini, Xiaohan Hong, and Jianbo Jiao. 360+x: A panoptic multi-modal scene understanding dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20261–20272, 2024.

[1947] Tewodros Habtegebrial, Christiano Gava, Marcel Rogge, Didier Stricker, and Varun Jampani. Somsi: Spherical novel view synthesis with soft occlusion multi-sphere images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15725–15734, 2022.

[1948] Huajian Huang, Changkun Liu, Yipeng Zhu, Hui Cheng, Tristan Braud, and Sai-Kit Yeung. 360loc: A dataset and benchmark for omnidirectional visual localization with cross-device queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22314–22324, 2024.

[1949] Zhaohui Wu, Yuanhan Zhang, Yuming Jiang, Ziqi Huang, Chenyang Si, Yinan Si, Qingyang Jin, Xinyuan Chen, Yaohui Wang, Yaohui Wang, et al. Q-Align: Teaching LMMs for Visual Scoring via Discrete Text-Defined Levels. In *International Conference on Machine Learning (ICML)*, 2024.

[1950] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Michalski. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

[1951] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. VBench: Comprehensive Benchmark Suite for Video Generative Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[1952] Chen-Yang Zhu, Xin-Yao Liu, Kai Xu, and Ren-Jiao Yi. A survey on 3d editing based on nerf and 3dgs. *Frontiers of Computer Science*, 20(4):2004701, 2026.

[1953] Chong Bao, Yinda Zhang, Bangbang Yang, Tianxing Fan, Zesong Yang, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Sine: Semantic-driven image-based nerf editing with prior-guided editing field. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20919–20929, 2023.

[1954] Umar Khalid, Hasan Iqbal, Nazmul Karim, Muhammad Tayyab, Jing Hua, and Chen Chen. Latentedit: Text driven local editing of 3d scenes. In *European Conference on Computer Vision*, pages 364–380, 2024.

[1955] Aryan Mikaeili, Or Perel, Mehdi Safaee, Daniel Cohen-Or, and Ali Mahdavi-Amiri. Sked: Sketch-guided text-based 3d editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14607–14619, 2023.

[1956] Xingchen Zhou, Ying He, F Richard Yu, Jianqiang Li, and You Li. Repaint-nerf: Nerf editting via semantic masks and diffusion models. *arXiv preprint arXiv:2306.05668*, 2023.

[1957] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19740–19750, 2023.

[1958] Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. Editing conditional radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5773–5783, 2021.

[1959] Nazmul Karim, Hasan Iqbal, Umar Khalid, Chen Chen, and Jing Hua. Free-editor: zero-shot text-driven 3d scene editing. In *European Conference on Computer Vision*, pages 436–453, 2024.

[1960] Kai He, Chin-Hsuan Wu, and Igor Gilitschenski. Ctrl-d: Controllable dynamic 3d scene editing with personalized 2d diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26630–26640, 2025.

[1961] Cyrus Vachha and Ayaan Haque. Instruct-gs2gs: Editing 3d gaussian splats with instructions, 2024.

[1962] Yuxuan Wang, Xuanyu Yi, Zike Wu, Na Zhao, Long Chen, and Hanwang Zhang. View-consistent 3d editing with gaussian splatting. In *European conference on computer vision*, pages 404–420, 2024.

[1963] Jamie Wynn, Zawar Qureshi, Jakub Powierza, Jamie Watson, and Mohamed Sayed. Morpheus: Text-driven 3d gaussian splat shape and color stylization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7825–7836, 2025.

[1964] Minghao Chen, Iro Laina, and Andrea Vedaldi. Dge: Direct gaussian 3d editing by consistent multi-view editing. In *European Conference on Computer Vision*, pages 74–92, 2024.

[1965] Jing Wu, Jia-Wang Bian, Xinghui Li, Guangrun Wang, Ian Reid, Philip Torr, and Victor Adrian Prisacariu. Gaussctrl: Multi-view consistent text-driven 3d gaussian splatting editing. In *European Conference on Computer Vision*, pages 55–71, 2024.

[1966] Ruihao Xia, Yang Tang, and Pan Zhou. Towards scalable and consistent 3d editing. *arXiv preprint arXiv:2510.02994*, 2025.

[1967] Eyal Gomel and Lior Wolf. Diffusion-based attention warping for consistent 3d scene editing. *arXiv preprint arXiv:2412.07984*, 2024.

[1968] Ziyang Yan, Lei Li, Yihua Shao, Siyu Chen, Zongkai Wu, Jenq-Neng Hwang, Hao Zhao, and Fabio Remondino. 3dsceneeditor: Controllable 3d scene editing with gaussian splatting. *arXiv preprint arXiv:2412.01583*, 2024.

[1969] Minghao Wen, Shengjie Wu, Kangkan Wang, and Dong Liang. Intergsedit: Interactive 3d gaussian splatting editing with 3d geometry-consistent attention prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26136–26145, 2025.

[1970] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21476–21485, 2024.

[1971] Xu Hu, Yuxi Wang, Lue Fan, Junsong Fan, Junran Peng, Zhen Lei, Qing Li, and Zhaoxiang Zhang. Semantic anything in 3d gaussians. *CoRR*, 2024.

[1972] Jiazhong Cen, Jiemin Fang, Chen Yang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Segment any 3d gaussians. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 1971–1979, 2025.

[1973] Junliang Ye, Zhengyi Wang, Ruowen Zhao, Shenghao Xie, and Jun Zhu. Shapellm-omni: A native multimodal llm for 3d generation and understanding. *arXiv preprint arXiv:2506.01853*, 2025.

[1974] Sining Lu, Guan Chen, Nam Anh Dinh, Itai Lang, Ari Holtzman, and Rana Hanocka. Ll3m: Large language 3d modelers. *arXiv preprint arXiv:2508.08228*, 2025.

[1975] Ziqi Ma, Hongqiao Chen, Yisong Yue, and Georgia Gkioxari. Feedforward 3d editing via text-steerable image-to-3d. *arXiv preprint arXiv:2512.13678*, 2025.

[1976] Ian Huang, Guandao Yang, and Leonidas Guibas. Blenderalchemy: Editing 3d graphics with vision-language models. In *European Conference on Computer Vision*, pages 297–314, 2024.

[1977] Ziya Erkoç, Can Gümeli, Chaoyang Wang, Matthias Nießner, Angela Dai, Peter Wonka, Hsin-Ying Lee, and Peiye Zhuang. Preditor3d: Fast and precise 3d shape editing. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 640–649, 2025.

[1978] Lin Li, Zehuan Huang, Haoran Feng, Gengxiong Zhuang, Rui Chen, Chunchao Guo, and Lu Sheng. Voxhammer: Training-free precise and coherent 3d editing in native 3d space. *arXiv preprint arXiv:2508.19247*, 2025.

[1979] Vivek Madhavaram, Shivangana Rawat, Chaitanya Devaguptapu, Charu Sharma, and Manohar Kaul. Towards a training free approach for 3d scene editing. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2890–2899, 2025.

[1980] Peng Li, Suizhi Ma, Jialiang Chen, Yuan Liu, Congyi Zhang, Wei Xue, Wenhan Luo, Alla Sheffer, Wenping Wang, and Yike Guo. Cmd: Controllable multiview diffusion for 3d editing and progressive generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–10, 2025.

[1981] Stephanie Fu, Netanel Y Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: learning new dimensions of human visual similarity using synthetic data. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 50742–50768, 2023.

[1982] Will Gao, Dilin Wang, Yuchen Fan, Aljaz Bozic, Tuur Stuyck, Zhengqin Li, Zhao Dong, Rakesh Ranjan, and Nikolaos Sarafianos. 3d mesh editing using masked lrms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7154–7165, 2025.

[1983] Zhenyu Shu, Junlong Yu, Kai Chao, Shiqing Xin, and Ligang Liu. Gaussedit: Adaptive 3d scene editing with text and image prompts. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

[1984] Xiao Han, Runze Tian, Yifei Tong, Fenggen Yu, Dingyao Liu, and Yan Zhang. Arap-gs: Drag-driven as-rigid-as-possible 3d gaussian splatting editing with diffusion prior. *arXiv preprint arXiv:2504.12788*, 2025.

[1985] Jian Gao, Chun Gu, Youtian Lin, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao. Relightable 3d gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[1986] Jongsu Park, Shokhikha Amalana Murdivien, Kyungwan Choi, Duhwan Mun, and Jumyung Um. Generative 3d appearance design: A survey of generation, segmentation and editing by artificial intelligence. *Journal of Computational Design and Engineering*, 13(1):1–23, 2026.

[1987] Amir Barda, Matheus Gadelha, Vladimir G Kim, Noam Aigerman, Amit H Bermano, and Thibault Groueix. Instant3dit: Multiview inpainting for fast editing of 3d objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16273–16282, 2025.

[1988] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

[1989] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

[1990] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[1991] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Wing Yan, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. Technical report, OpenAI, 2024. URL https://openai.com/index/video-generation-models-as-world-simulators. Preprint; describes Sora, a text-to-video model.

[1992] Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70m: Captioning 70m videos with multiple cross-modality teachers, 2024. URL https://arxiv.org/abs/2402.19479.

[1993] Xuan Ju, Yiming Gao, Zhaoyang Zhang, Ziyang Yuan, Xintao Wang, Ailing Zeng, Yu Xiong, Qiang Xu, and Ying Shan. Miradata: A large-scale video dataset with long durations and structured captions. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37,

pages 48955–48970, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/57f6683e550eb067936c9e9f0bcb8e31-Paper-Datasets_and_Benchmarks_Track.pdf.

[1994] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2830–2839, 2017.

[1995] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal gan. *International Journal of Computer Vision*, 128(10):2586–2606, 2020.

[1996] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1526–1535, 2018.

[1997] Yu Tian, Jian Ren, Menglei Chai, Kyle Olszewski, Xi Peng, Dimitris N Metaxas, and Sergey Tulyakov. A good image generator is what you need for high-resolution video synthesis. *arXiv preprint arXiv:2104.15069*, 2021.

[1998] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv preprint arXiv:1907.06571*, 2019.

[1999] Sihyun Yu, Jihoon Tack, Sangwoo Mo, Hyunsu Kim, Junho Kim, Jung-Woo Ha, and Jinwoo Shin. Generating videos with dynamics-aware implicit generative adversarial networks. *arXiv preprint arXiv:2202.10571*, 2022.

[2000] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3626–3636, 2022.

[2001] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. Nüwa: Visual synthesis pre-training for neural visual world creation. In *European conference on computer vision*, pages 720–736, 2022.

[2002] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. In *European Conference on Computer Vision*, pages 102–118, 2022.

[2003] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

[2004] Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22963–22974, 2025.

[2005] Sand. ai, Hansi Teng, Hongyu Jia, Lei Sun, Lingzhi Li, Maolin Li, Mingqiu Tang, Shuai Han, Tianning Zhang, W. Q. Zhang, Weifeng Luo, Xiaoyang Kang, Yuchen Sun, Yue Cao, Yunpeng Huang, Yutong Lin, Yuxin Fang, Zewei Tao, Zheng Zhang, Zhongshu Wang, Zixun Liu, Dai Shi, Guoli Su, Hanwen Sun, Hong Pan, Jie Wang, Jiexin Sheng, Min Cui, Min Hu, Ming Yan, Shucheng Yin, Siran Zhang, Tingting Liu, Xianping Yin, Xiaoyu Yang, Xin Song, Xuan Hu, Yankai Zhang, and Yuqiao Li. Magi-1: Autoregressive video generation at scale, 2025. URL https://arxiv.org/abs/2505.13211.

[2006] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:8633–8646, 2022.

[2007] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In

Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 22563–22575, 2023.

[2008] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022.

[2009] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[2010] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

[2011] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023.

[2012] Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7310–7320, 2024.

[2013] Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, et al. Lavie: High-quality video generation with cascaded latent diffusion models. *International Journal of Computer Vision*, 133(5):3059–3078, 2025.

[2014] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7623–7633, 2023.

[2015] Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *International Conference on Learning Representations*, 2024.

[2016] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

[2017] Xin Ma, Yaohui Wang, Gengyun Jia, Xinyuan Chen, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent diffusion transformer for video generation. *arXiv preprint arXiv:2401.03048*, 2024.

[2018] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

[2019] Guoqing Ma, Haoyang Huang, Kun Yan, Liangyu Chen, Nan Duan, Shengming Yin, Changyi Wan, Ranchen Ming, Xiaoniu Song, Xing Chen, Yu Zhou, Deshan Sun, Deyu Zhou, Jian Zhou, Kaijun Tan, Kang An, Mei Chen, Wei Ji, Qiling Wu, Wen Sun, Xin Han, Yanan Wei, Zheng Ge, Aojie Li, Bin Wang, Bizhu Huang, Bo Wang, Brian Li, Changxing Miao, Chen Xu, Chenfei Wu, Chenguang Yu, Dapeng Shi, Dingyuan Hu, Enle Liu, Gang Yu, Ge Yang, Guanzhe Huang, Gulin Yan, Haiyang Feng, Hao Nie, Haonan Jia, Hanpeng Hu, Hanqi Chen, Haolong Yan, Heng Wang, Hongcheng Guo, Huilin Xiong, Huixin Xiong, Jiahao Gong, Jianchang Wu, Jiaoren Wu, Jie Wu, Jie Yang, Jiashuai Liu, Jiashuo Li, Jingyang Zhang, Junjing Guo, Junzhe Lin, Kaixiang Li, Lei Liu, Lei Xia, Liang Zhao, Liguo Tan, Liwen Huang, Liying Shi, Ming Li, Mingliang Li, Muhua Cheng, Na Wang, Qiaohui Chen, Qinglin He, Qiuyan Liang, Quan Sun, Ran Sun, Rui Wang, Shaoliang Pang, Shiliang Yang, Sitong Liu, Siqi Liu, Shuli Gao, Tiancheng Cao, Tianyu Wang, Weipeng Ming, Wenqing He, Xu Zhao,

Xuelin Zhang, Xianfang Zeng, Xiaojia Liu, Xuan Yang, Yaqi Dai, Yanbo Yu, Yang Li, Yineng Deng, Yingming Wang, Yilei Wang, Yuanwei Lu, Yu Chen, Yu Luo, Yuchu Luo, Yuhe Yin, Yuheng Feng, Yuxiang Yang, Zecheng Tang, Zekai Zhang, Zidong Yang, Binxing Jiao, Jiansheng Chen, Jing Li, Shuchang Zhou, Xiangyu Zhang, Xinhao Zhang, Yibo Zhu, Heung-Yeung Shum, and Daxin Jiang. Step-video-t2v technical report: The practice, challenges, and future of video foundation model, 2025. URL https://arxiv.org/abs/2502.10248.

[2020] Google DeepMind. Veo: Google's deepmind text-to-video generation model. https://deepmind.google/models/veo/, 2024. Latest version: Veo 3 (2025) with synchronized audio.

[2021] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

[2022] Genmo Team. Mochi 1: A new state-of-the-art in open-source video generation. https://github.com/genmoai/models, 2024.

[2023] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion, 2024.

[2024] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1728–1738, 2021.

[2025] Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, Fei Yang, Pengfei Wan, and Di Zhang. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[2026] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5288–5296, 2016.

[2027] Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Wang, and Ying Shan. Evalcrafter: Benchmarking and evaluating large video generation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22139–22149, 2024.

[2028] Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2v-compbench: A comprehensive benchmark for compositional text-to-video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[2029] Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023.

[2030] Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024.

[2031] Weiming Ren, Huan Yang, Ge Zhang, Cong Wei, Xinrun Du, Wenhao Huang, and Wenhu Chen. Consisti2v: Enhancing visual consistency for image-to-video generation. *arXiv preprint arXiv:2402.04324*, 2024.

[2032] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pages 399–417, 2024.

[2033] Xuan Ju, Ailing Zeng, Chenchen Zhao, Jianan Wang, Lei Zhang, and Qiang Xu. Humansd: A native skeleton-guided diffusion model for human image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15988–15998, 2023.

[2034] Xuan Ju, Ailing Zeng, Jianan Wang, Qiang Xu, and Lei Zhang. Human-art: A versatile human-centric dataset bridging natural and artificial scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 618–629, June 2023.

[2035] MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. https://github.com/open-mmlab/mmpose, 2020.

[2036] Zhongcong Xu, Jianfeng Zhang, Jun Hao Liew, Hanshu Yan, Jia-Wei Liu, Chenxu Zhang, Jiashi Feng, and Mike Zheng Shou. Magicanimate: Temporally consistent human image animation using diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1481–1490, 2024.

[2037] Li Hu. Animate anyone: Consistent and controllable image-to-video synthesis for character animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8153–8163, 2024.

[2038] Yue Ma, Yingqing He, Xiaodong Cun, Xintao Wang, Siran Chen, Xiu Li, and Qifeng Chen. Follow your pose: Pose-guided text-to-video generation using pose-free videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 4117–4125, 2024.

[2039] Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion image-to-video synthesis via stable diffusion. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22623–22633, 2023.

[2040] Hongxiang Li, Yaowei Li, Yuhang Yang, Junjie Cao, Zhihong Zhu, Xuxin Cheng, and Long Chen. Dispose: Disentangling pose guidance for controllable human image animation. *arXiv preprint arXiv:2412.09349*, 2024.

[2041] Shenhao Zhu, Junming Leo Chen, Zuozhuo Dai, Zilong Dong, Yinghui Xu, Xun Cao, Yao Yao, Hao Zhu, and Siyu Zhu. Champ: Controllable and consistent human image animation with 3d parametric guidance. In *European Conference on Computer Vision*, pages 145–162, 2024.

[2042] Di Chang, Yichun Shi, Quankai Gao, Jessica Fu, Hongyi Xu, Guoxian Song, Qing Yan, Yizhe Zhu, Xiao Yang, and Mohammad Soleymani. Magicpose: Realistic human poses and facial expressions retargeting with identity-aware diffusion. *arXiv preprint arXiv:2311.12052*, 2023.

[2043] Min Zhao, Rongzhen Wang, Fan Bao, Chongxuan Li, and Jun Zhu. Controlvideo: conditional control for one-shot text-driven video editing and beyond. *Science China Information Sciences*, 68(3):132107, 2025.

[2044] Jinbo Xing, Menghan Xia, Yuxin Liu, Yuechen Zhang, Yong Zhang, Yingqing He, Hanyuan Liu, Haoxin Chen, Xiaodong Cun, Xintao Wang, et al. Make-your-video: Customized video generation using textual and structural guidance. *IEEE Transactions on Visualization and Computer Graphics*, 31(2): 1526–1541, 2024.

[2045] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *CoRR*, 2023.

[2046] Bohao Peng, Jian Wang, Yuechen Zhang, Wenbo Li, Ming-Chang Yang, and Jiaya Jia. Controlnext: Powerful and efficient control for image and video generation. *arXiv preprint arXiv:2408.06070*, 2024.

[2047] Yatian Pang, Bin Zhu, Bin Lin, Mingzhe Zheng, Francis EH Tay, Ser-Nam Lim, Harry Yang, and Li Yuan. Dreamdance: Animating human images by enriching 3d geometry cues from 2d poses. *arXiv preprint arXiv:2412.00397*, 2024.

[2048] Yuwei Guo, Ceyuan Yang, Anyi Rao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Sparsectrl: Adding sparse controls to text-to-video diffusion models. In *European Conference on Computer Vision*, pages 330–348, 2024.

[2049] Chen Zhiyuan, Cao Jiajiong, Chen Zhiquan, Li Yuming, and Ma Chenguang. Echomimic: Lifelike audio-driven portrait animations through editable landmark conditioning, 2024.

[2050] Zunnan Xu, Zhentao Yu, Zixiang Zhou, Jun Zhou, Xiaoyu Jin, Fa-Ting Hong, Xiaozhong Ji, Junwei Zhu, Chengfei Cai, Shiyu Tang, et al. Hunyuanportrait: Implicit condition control for enhanced portrait animation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15909–15919, 2025.

[2051] Jianzhu Guo, Dingyun Zhang, Xiaoqiang Liu, Zhizhou Zhong, Yuan Zhang, Pengfei Wan, and Di Zhang. Liveportrait: Efficient portrait animation with stitching and retargeting control. *arXiv preprint arXiv:2407.03168*, 2024.

[2052] Hongxiang Zhao, Xingchen Liu, Mutian Xu, Yiming Hao, Weikai Chen, and Xiaoguang Han. Taste-rob: Advancing video generation of task-oriented hand-object interaction for generalizable robotic manipulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 27683–27693, 2025.

[2053] Bin Lin, Yanzhen Yu, Jianhao Ye, Ruitao Lv, Yuguang Yang, Ruoye Xie, Pan Yu, and Hongbin Zhou. Takin-ada: Emotion controllable audio-driven animation with canonical and landmark loss optimization. *arXiv preprint arXiv:2410.14283*, 2024.

[2054] Yue Ma, Hongyu Liu, Hongfa Wang, Heng Pan, Yingqing He, Junkun Yuan, Ailing Zeng, Chengfei Cai, Heung-Yeung Shum, Wei Liu, et al. Follow-your-emoji: Fine-controllable and expressive freestyle portrait animation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–12, 2024.

[2055] Jinbo Xing, Hanyuan Liu, Menghan Xia, Yong Zhang, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Tooncrafter: Generative cartoon interpolation. *ACM Transactions on Graphics (TOG)*, 43(6):1–11, 2024.

[2056] Lifan Jiang, Shuang Chen, Boxi Wu, Xiaotong Guan, and Jiahui Zhang. Vidsketch: Hand-drawn sketch-driven video generation with diffusion control. *arXiv preprint arXiv:2502.01101*, 2025.

[2057] Yihao Meng, Hao Ouyang, Hanlin Wang, Qiuyu Wang, Wen Wang, Ka Leong Cheng, Zhiheng Liu, Yujun Shen, and Huamin Qu. Anidoc: Animation creation made easier. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18187–18197, 2025.

[2058] Dagmar Loftsdottir and Matthew Guzdial. Sketchbetween: Video-to-video synthesis for sprite animation via sketches. In *Proceedings of the 17th International Conference on the Foundations of Digital Games*, pages 1–7, 2022.

[2059] Yuxue Yang, Lue Fan, Zuzen Lin, Feng Wang, and Zhaoxiang Zhang. Layeranimate: Layer-specific control for animation. *arXiv e-prints*, pages arXiv–2501, 2025.

[2060] Yue Ma, Xiaodong Cun, Sen Liang, Jinbo Xing, Yingqing He, Chenyang Qi, Siran Chen, and Qifeng Chen. Magicstick: Controllable video editing via control handle transformations. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 9385–9395, 2025.

[2061] Ruiyuan Gao, Kai Chen, Enze Xie, Lanqing Hong, Zhenguo Li, Dit-Yan Yeung, and Qiang Xu. Magicdrive: Street view generation with diverse 3d geometry control. *arXiv preprint arXiv:2310.02601*, 2023.

[2062] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-drive world models for autonomous driving. In *European conference on computer vision*, pages 55–72, 2024.

[2063] Wei Wu, Xi Guo, Weixuan Tang, Tingxuan Huang, Chiyu Wang, Dongyue Chen, and Chenjing Ding. Drivescape: Towards high-resolution controllable multi-view driving video generation. *arXiv preprint arXiv:2409.05463*, 2024.

[2064] Jianbiao Mei, Tao Hu, Xuemeng Yang, Licheng Wen, Yu Yang, Tiantian Wei, Yukai Ma, Min Dou, Botian Shi, and Yong Liu. Dreamforge: Motion-aware autoregressive video generation for multi-view driving scenes. *arXiv preprint arXiv:2409.04003*, 2024.

[2065] Junpeng Jiang, Gangyi Hong, Lijun Zhou, Enhui Ma, Hengtong Hu, Xia Zhou, Jie Xiang, Fan Liu, Kaicheng Yu, Haiyang Sun, et al. Dive: Dit-based video generation with enhanced control. *arXiv preprint arXiv:2409.01595*, 2024.

[2066] Jiawei Wang, Yuchen Zhang, Jiaxin Zou, Yan Zeng, Guoqiang Wei, Liping Yuan, and Hang Li. Boximator: Generating rich and controllable motions for video synthesis. *arXiv preprint arXiv:2402.01566*, 2024.

[2067] Ge Ya Luo, ZhiHao Luo, Anthony Gosselin, Alexia Jolicoeur-Martineau, and Christopher Pal. Ctrl-v: Higher fidelity autonomous vehicle video generation with bounding-box controlled object motion. *Transactions on Machine Learning Research*, 2025.

[2068] Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.

[2069] Wanquan Feng, Tianhao Qi, Jiawei Liu, Mingzhen Sun, Pengqi Tu, Tianxiang Ma, Fei Dai, Songtao Zhao, Siyu Zhou, and Qian He. I2vcontrol: Disentangled and unified video motion synthesis control. *arXiv preprint arXiv:2411.17765*, 2024.

[2070] Muyao Niu, Xiaodong Cun, Xintao Wang, Yong Zhang, Ying Shan, and Yinqiang Zheng. Mofa-video: Controllable image animation via generative motion field adaptions in frozen image-to-video diffusion model. In *European Conference on Computer Vision*, pages 111–128, 2024.

[2071] Ryan Burgert, Yuancheng Xu, Wenqi Xian, Oliver Pilarski, Pascal Clausen, Mingming He, Li Ma, Yitong Deng, Lingxiao Li, Mohsen Mu, et al. Go-with-the-flow: Motion-controllable video diffusion models using real-time warped noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[2072] Jinbo Xing, Long Mai, Cusuh Ham, Jiahui Huang, Aniruddha Mahapatra, Chi-Wing Fu, Tien-Tsin Wong, and Feng Liu. Motioncanvas: Cinematic shot design with controllable image-to-video generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–11, 2025.

[2073] Quanhao Li, Zhen Xing, Rui Wang, Hui Zhang, Qi Dai, and Zuxuan Wu. Magicmotion: Controllable video generation with dense-to-sparse trajectory guidance. *arXiv preprint arXiv:2503.16421*, 2025.

[2074] Zhenghao Zhang, Junchao Liao, Menghao Li, Zuozhuo Dai, Bingxue Qiu, Siyu Zhu, Long Qin, and Weizhi Wang. Tora: Trajectory-oriented diffusion transformer for video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2063–2073, 2025.

[2075] Haonan Qiu, Zhaoxi Chen, Zhouxia Wang, Yingqing He, Menghan Xia, and Ziwei Liu. Freetraj: Tuning-free trajectory control in video diffusion models. *arXiv preprint arXiv:2406.16863*, 2024.

[2076] Yash Jain, Anshul Nasery, Vibhav Vineet, and Harkirat Behl. Peekaboo: Interactive video generation via masked-diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8079–8088, 2024.

[2077] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*, 2023.

[2078] Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. Draganything: Motion control for anything using entity representation. In *European Conference on Computer Vision*, pages 331–348, 2024.

[2079] Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. Cameractrl ii: Dynamic scene exploration via camera-controlled video diffusion models. *arXiv preprint arXiv:2503.10592*, 2025.

[2080] Shiyuan Yang, Liang Hou, Haibin Huang, Chongyang Ma, Pengfei Wan, Di Zhang, Xiaodong Chen, and Jing Liao. Direct-a-video: Customized video generation with user-directed camera movement and object motion. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12, 2024.

[2081] Dejia Xu, Yifan Jiang, Chen Huang, Liangchen Song, Thorsten Gernoth, Liangliang Cao, Zhangyang Wang, and Hao Tang. Cavia: Camera-controllable multi-view video diffusion with view-integrated attention. *arXiv preprint arXiv:2410.10774*, 2024.

[2082] Xiao Fu, Xian Liu, Xintao Wang, Sida Peng, Menghan Xia, Xiaoyu Shi, Ziyang Yuan, Pengfei Wan, Di Zhang, and Dahua Lin. 3dtrajmaster: Mastering 3d trajectory for multi-entity motion in video generation. *arXiv preprint arXiv:2412.07759*, 2024.

[2083] Tuna Han Salih Meral, Hidir Yesiltepe, Connor Dunlop, and Pinar Yanardag. Motionflow: Attention-driven motion transfer in video diffusion models. *arXiv preprint arXiv:2412.05275*, 2024.

[2084] Qinghe Wang, Yawen Luo, Xiaoyu Shi, Xu Jia, Huchuan Lu, Tianfan Xue, Xintao Wang, Pengfei Wan, Di Zhang, and Kun Gai. Cinemaster: A 3d-aware and controllable framework for cinematic text-to-video generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–10, 2025.

[2085] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, and Di Zhang. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*, 2025.

[2086] Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models. *arXiv preprint arXiv:2503.05638*, 2025.

[2087] Zekai Gu, Rui Yan, Jiahao Lu, Peng Li, Zhiyang Dou, Chenyang Si, Zhen Dong, Qifeng Liu, Cheng Lin, Ziwei Liu, Wenping Wang, and Yuan Liu. Diffusion as shader: 3d-aware video diffusion for versatile video generation control, 2025. URL https://arxiv.org/abs/2501.03847.

[2088] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jia-Wei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. In *European Conference on Computer Vision*, pages 273–290, 2024.

[2089] Pengyang Ling, Jiazi Bu, Pan Zhang, Xiaoyi Dong, Yuhang Zang, Tong Wu, Huaian Chen, Jiaqi Wang, and Yi Jin. Motionclone: Training-free motion cloning for controllable video generation. *arXiv preprint arXiv:2406.05338*, 2024.

[2090] Hyeonho Jeong, Geon Yeong Park, and Jong Chul Ye. Vmc: Video motion customization using temporal attention adaption for text-to-video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9212–9221, 2024.

[2091] Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Yusuf Aytar, Michael Rubinstein, Chen Sun, et al. Motion prompting: Controlling video generation with motion trajectories. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1–12, 2025.

[2092] Xinyao Liao, Xianfang Zeng, Liao Wang, Gang Yu, Guosheng Lin, and Chi Zhang. Motionagent: Fine-grained controllable video generation via motion field agent. *arXiv preprint arXiv:2502.03207*, 2025.

[2093] Yen-Siang Wu, Chi-Pin Huang, Fu-En Yang, and Yu-Chiang Frank Wang. Motionmatcher: Motion customization of text-to-video diffusion models via motion feature matching. *arXiv preprint arXiv:2502.13234*, 2025.

[2094] Hyeonho Jeong, Jinho Chang, Geon Yeong Park, and Jong Chul Ye. Dreammotion: Space-time self-similarity score distillation for zero-shot video editing. *CoRR*, 2024.

[2095] Yixuan Ren, Yang Zhou, Jimei Yang, Jing Shi, Difan Liu, Feng Liu, Mingi Kwon, and Abhinav Shrivastava. Customize-a-video: One-shot motion customization of text-to-video diffusion models. In *European Conference on Computer Vision*, pages 332–349, 2024.

[2096] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[2097] Shuai Yang, Yifan Zhou, Ziwei Liu, , and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *ACM SIGGRAPH Asia Conference Proceedings*, 2023.

[2098] Gongye Liu, Menghan Xia, Yong Zhang, Haoxin Chen, Jinbo Xing, Xintao Wang, Yujiu Yang, and Ying Shan. Stylecrafter: Enhancing stylized text-to-video generation with style adapter. *arXiv preprint arXiv:2312.00330*, 2023.

[2099] Shuai Yang, Yifan Zhou, Ziwei Liu, , and Chen Change Loy. Fresco: Spatial-temporal correspondence for zero-shot video translation. In *CVPR*, 2024.

[2100] Zixuan Ye, Huijuan Huang, Xintao Wang, Pengfei Wan, Di Zhang, and Wenhan Luo. Stylemaster: Stylize your video with artistic generation and translation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2630–2640, 2025.

[2101] Quanjian Song, Mingbao Lin, Wengyi Zhan, Shuicheng Yan, Liujuan Cao, and Rongrong Ji. Univst: A unified framework for training-free localized video style transfer. *arXiv preprint arXiv:2410.20084*, 2024.

[2102] Sangwon Jang, Taekyung Ki, Jaehyeong Jo, Jaehong Yoon, Soo Ye Kim, Zhe Lin, and Sung Ju Hwang. Frame guidance: Training-free guidance for frame-level control in video diffusion models, 2025. URL https://arxiv.org/abs/2506.07177.

[2103] Dianbing Xi, Jiepeng Wang, Yuanzhi Liang, Xi Qiu, Yuchi Huo, Rui Wang, Chi Zhang, and Xuelong Li. Omnivdiff: Omni controllable video diffusion for generation and understanding, 2025. URL https://arxiv.org/abs/2504.10825.

[2104] Yao Teng, Enze Xie, Yue Wu, Haoyu Han, Zhenguo Li, and Xihui Liu. Drag-a-video: Non-rigid video editing with point-based interaction. *arXiv preprint arXiv:2312.02936*, 2023.

[2105] Hyeonho Jeong, Chun-Hao Paul Huang, Jong Chul Ye, Niloy Mitra, and Duygu Ceylan. Track4gen: Teaching video diffusion models to track points improves video generation, 2025. URL https://arxiv.org/abs/2412.06016.

[2106] Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Xinze Chen, Guan Huang, Xiaoyi Bao, and Xingang Wang. Drivedreamer-2: Llm-enhanced world models for diverse driving video generation, 2024. URL https://arxiv.org/abs/2403.06845.

[2107] Ruiyuan Gao, Kai Chen, Bo Xiao, Lanqing Hong, Zhenguo Li, and Qiang Xu. Magicdrive-v2: High-resolution long video generation for autonomous driving with adaptive control, 2025. URL https://arxiv.org/abs/2411.13807.

[2108] Yuqing Wen, Yucheng Zhao, Yingfei Liu, Fan Jia, Yanhui Wang, Chong Luo, Chi Zhang, Tiancai Wang, Xiaoyan Sun, and Xiangyu Zhang. Panacea: Panoramic and controllable video generation for autonomous driving, 2023. URL https://arxiv.org/abs/2311.16813.

[2109] Lin Liu, Quande Liu, Shengju Qian, Yuan Zhou, Wengang Zhou, Houqiang Li, Lingxi Xie, and Qi Tian. Text-animator: Controllable visual text video generation, 2024. URL https://arxiv.org/abs/2406.17777.

[2110] Linrui Tian, Qi Wang, Bang Zhang, and Liefeng Bo. Emo: Emote portrait alive – generating expressive portrait videos with audio2video diffusion model under weak conditions, 2024. URL https://arxiv.org/abs/2402.17485.

[2111] Sicheng Xu, Guojun Chen, Yu-Xiao Guo, Jiaolong Yang, Chong Li, Zhenyu Zang, Yizhong Zhang, Xin Tong, and Baining Guo. Vasa-1: Lifelike audio-driven talking faces generated in real time, 2024. URL https://arxiv.org/abs/2404.10667.

[2112] Yifeng Ma, Shiwei Zhang, Jiayu Wang, Xiang Wang, Yingya Zhang, and Zhidong Deng. Dreamtalk: When expressive talking head generation meets diffusion probabilistic models. *arXiv preprint arXiv:2312.09767*, 2023.

[2113] Huawei Wei, Zejun Yang, and Zhisheng Wang. Aniportrait: Audio-driven synthesis of photorealistic portrait animations, 2024.

[2114] Junxian Ma, Shiwen Wang, Jian Yang, Junyi Hu, Jian Liang, Guosheng Lin, Jingbo chen, Kai Li, and Yu Meng. Sayanything: Audio-driven lip synchronization with conditional video diffusion, 2025. URL https://arxiv.org/abs/2502.11515.

[2115] Yue Zhang, Zhizhou Zhong, Minhao Liu, Zhaokang Chen, Bin Wu, Yubin Zeng, Chao Zhan, Yingjie He, Junxin Huang, and Wenjiang Zhou. Musetalk: Real-time high-fidelity video dubbing via spatio-temporal sampling, 2025. URL https://arxiv.org/abs/2410.10122.

[2116] Hejia Chen, Haoxian Zhang, Shoulong Zhang, Xiaoqiang Liu, Sisi Zhuang, Yuan Zhang, Pengfei Wan, Di Zhang, and Shuai Li. Cafe-talk: Generating 3d talking face animation with multimodal coarse- and fine-grained control, 2025. URL https://arxiv.org/abs/2503.14517.

[2117] Cong Wang, Kuan Tian, Jun Zhang, Yonghang Guan, Feng Luo, Fei Shen, Zhiwei Jiang, Qing Gu, Xiao Han, and Wei Yang. V-express: Conditional dropout for progressive training of portrait video generation, 2024. URL https://arxiv.org/abs/2406.02511.

[2118] Shurong Yang, Huadong Li, Juhao Wu, Minhao Jing, Linze Li, Renhe Ji, Jiajun Liang, and Haoqiang Fan. Megactor: Harness the power of raw video for vivid portrait animation, 2024. URL https://arxiv.org/abs/2405.20851.

[2119] Minglu Zhao, Wenmin Wang, Tongbao Chen, Rui Zhang, and Ruochen Li. Ta2v: Text-audio guided video generation. *IEEE Transactions on Multimedia*, pages 1–15, 2024. doi:10.1109/TMM.2024.3362149.

[2120] Luca Savant Aira, Antonio Montanaro, Emanuele Aiello, Diego Valsesia, and Enrico Magli. Motioncraft: Physics-based zero-shot video generation, 2024. URL https://arxiv.org/abs/2405.13557.

[2121] Moayed Haji-Ali, Willi Menapace, Aliaksandr Siarohin, Ivan Skorokhodov, Alper Canberk, Kwot Sin Lee, Vicente Ordonez, and Sergey Tulyakov. Av-link: Temporally-aligned diffusion features for cross-modal audio-video generation, 2025. URL https://arxiv.org/abs/2412.15191.

[2122] Lin Zhang, Shentong Mo, Yijing Zhang, and Pedro Morgado. Audio-synchronized visual animation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[2123] Hongwei Yi, Tian Ye, Shitong Shao, Xuancheng Yang, Jiantong Zhao, Hanzhong Guo, Terrance Wang, Qingyu Yin, Zeke Xie, Lei Zhu, Wei Li, Michael Lingelbach, and Daquan Zhou. Magicinfinite: Generating infinite talking videos with your words and voice, 2025. URL https://arxiv.org/abs/2503.05978.

[2124] Anoop Cherian, Moitreya Chatterjee, and Narendra Ahuja. Sound2sight: Generating visual dynamics from sound and context, 2020. URL https://arxiv.org/abs/2007.12130.

[2125] Dan Kondratyuk, Lijun Yu, Xiuye Gu, José Lezama, Jonathan Huang, Grant Schindler, Rachel Hornung, Vighnesh Birodkar, Jimmy Yan, Ming-Chang Chiu, Krishna Somandepalli, Hassan Akbari, Yair Alon, Yong Cheng, Josh Dillon, Agrim Gupta, Meera Hahn, Anja Hauth, David Hendon, Alonso Martinez, David Minnen, Mikhail Sirotenko, Kihyuk Sohn, Xuan Yang, Hartwig Adam, Ming-Hsuan

Yang, Irfan Essa, Huisheng Wang, David A. Ross, Bryan Seybold, and Lu Jiang. Videopoet: A large language model for zero-shot video generation, 2024. URL https://arxiv.org/abs/2312.14125.

[2126] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability, 2023. URL https://arxiv.org/abs/2306.02018.

[2127] Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, and Chao Liang. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models, 2025. URL https://arxiv.org/abs/2502.01061.

[2128] Xuan Ju, Weicai Ye, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, and Qiang Xu. Fulldit: Multi-task video generative foundation model with full attention, 2025. URL https://arxiv.org/abs/2503.19907.

[2129] Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3661–3670, 2021.

[2130] Liangbin Xie, Xintao Wang, Honglun Zhang, Chao Dong, and Ying Shan. Vfhq: A high-quality dataset and benchmark for video face super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 656–665, 2022.

[2131] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Direct inversion: Boosting diffusion-based editing with 3 lines of code, 2023. URL https://arxiv.org/abs/2310.01506.

[2132] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15954–15964, October 2023.

[2133] Duygu Ceylan, Chun-Hao Paul Huang, and Niloy J. Mitra. Pix2video: Video editing using image diffusion, 2023. URL https://arxiv.org/abs/2303.12688.

[2134] Wen Wang, Yan Jiang, Kangyang Xie, Zide Liu, Hao Chen, Yue Cao, Xinlong Wang, and Chunhua Shen. Zero-shot video editing using off-the-shelf image diffusion models, 2024. URL https://arxiv.org/abs/2303.17599.

[2135] Wenqi Ouyang, Yi Dong, Lei Yang, Jianlou Si, and Xingang Pan. I2vedit: First-frame-guided video editing via image-to-video diffusion models, 2024. URL https://arxiv.org/abs/2405.16537.

[2136] Ozgur Kara, Bariscan Kurtkaya, Hidir Yesiltepe, James M. Rehg, and Pinar Yanardag. Rave: Randomized noise shuffling for fast and consistent video editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[2137] Youyuan Zhang, Xuan Ju, and James J. Clark. Fastvideoedit: Leveraging consistency models for efficient text-to-video editing, 2024. URL https://arxiv.org/abs/2403.06269.

[2138] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors, 2023. URL https://arxiv.org/abs/2302.01329.

[2139] Jiaxin Cheng, Tianjun Xiao, and Tong He. Consistent video-to-video transfer using synthetic dataset, 2023. URL https://arxiv.org/abs/2311.00213.

[2140] Shaoteng Liu, Tianyu Wang, Jui-Hsien Wang, Qing Liu, Zhifei Zhang, Joon-Young Lee, Yijun Li, Bei Yu, Zhe Lin, Soo Ye Kim, and Jiaya Jia. Generative video propagation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17712–17722, June 2025.

[2141] Yuxuan Bian, Zhaoyang Zhang, Xuan Ju, Mingdeng Cao, Liangbin Xie, Ying Shan, and Qiang Xu. Videopainter: Any-length video inpainting and editing with plug-and-play context control.

In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, SIGGRAPH Conference Papers '25, New York, NY, USA, 2025. ISBN 9798400715402. doi:10.1145/3721238.3730673. URL https://doi.org/10.1145/3721238.3730673.

[2142] Zixuan Ye, Xuanhua He, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, Qifeng Chen, and Wenhan Luo. Unic: Unified in-context video editing, 2025. URL https://arxiv.org/abs/2506.04216.

[2143] Xuan Ju, Tianyu Wang, Yuqian Zhou, He Zhang, Qing Liu, Nanxuan Zhao, Zhifei Zhang, Yijun Li, Yuanhao Cai, Shaoteng Liu, Daniil Pakhomov, Zhe Lin, Soo Ye Kim, and Qiang Xu. Editverse: Unifying image and video editing and generation with in-context learning, 2025. URL https://arxiv.org/abs/2509.20360.

[2144] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing, 2023. URL https://arxiv.org/abs/2303.09535.

[2145] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *arXiv:2303.04761*, 2023.

[2146] Chaehun Shin, Heeseung Kim, Che Hyun Lee, Sang gil Lee, and Sungroh Yoon. Edit-a-video: Single video editing with object-aware consistency, 2023. URL https://arxiv.org/abs/2303.07945.

[2147] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arxiv:2307.10373*, 2023.

[2148] Jianhong Bai, Tianyu He, Yuchi Wang, Junliang Guo, Haoji Hu, Zuozhu Liu, and Jiang Bian. Uniedit: A unified tuning-free framework for video motion and appearance editing. *arXiv preprint arXiv:2402.13185*, 2024.

[2149] Max Ku, Cong Wei, Weiming Ren, Harry Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks, 2024. URL https://arxiv.org/abs/2403.14468.

[2150] Jun Hao Liew, Hanshu Yan, Jianfeng Zhang, Zhongcong Xu, and Jiashi Feng. Magicedit: High-fidelity and temporally coherent video editing. In *arXiv*, 2023.

[2151] Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jinbin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei Huang, Yuanxi Sun, Rui He, Feng Hu, Junhua Hu, Hai Huang, Hanyu Zhu, Xu Cheng, Jie Tang, Mike Zheng Shou, Kurt Keutzer, and Forrest Iandola. Cvpr 2023 text guided video editing competition. *arXiv preprint arXiv:2310.16003*, 2023.

[2152] Minghan Li, Chenxi Xie, Yichen Wu, Lei Zhang, and Mengyu Wang. FiVE: A fine-grained video editing benchmark for evaluating emerging diffusion and rectified flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

[2153] Yinan Chen, Jiangning Zhang, Teng Hu, Yuxiang Zeng, Zhucun Xue, Qingdong He, Chengjie Wang, Yong Liu, Xiaobin Hu, and Shuicheng Yan. IVEBench: Modern benchmark suite for instruction-guided video editing assessment. In *International Conference on Learning Representations (ICLR)*, 2026.

[2154] Yan Wu, Jiahao Wang, Yan Zhang, Siwei Zhang, Otmar Hilliges, Fisher Yu, and Siyu Tang. SAGA: stochastic whole-body grasping with contact. In *Proc. ECCV*, volume 13666 of *Lecture Notes in Computer Science*, pages 257–274, 2022.

[2155] Omid Taheri, Vasileios Choutas, Michael J. Black, and Dimitrios Tzionas. GOAL: generating 4d whole-body motion for hand-object grasping. In *Proc. CVPR*, pages 13253–13263, 2022.

[2156] Ping Yu, Yang Zhao, Chunyuan Li, Junsong Yuan, and Changyou Chen. Structure-aware human-action generation. In *Proc. ECCV*, volume 12375, pages 18–34, 2020.

[2157] Bruno Degardin, João Neves, Vasco Lopes, João Brito, Ehsan Yaghoubi, and Hugo Proença. Generative adversarial graph convolutional networks for human action synthesis. In *Proc. WACV*, pages 2753–2762, 2022.

[2158] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proc. ACM MM*, pages 2021–2029, 2020.

[2159] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer VAE. In *Proc. ICCV*, pages 10965–10975, 2021.

[2160] Qiujing Lu, Yipeng Zhang, Mingjian Lu, and Vwani Roychowdhury. Action-conditioned on-demand motion generation. In *Proc. ACM MM*, pages 2249–2257, 2022.

[2161] Thomas Lucas, Fabien Baradel, Philippe Weinzaepfel, and Grégory Rogez. Posegpt: Quantization-based 3d human motion generation and forecasting. In *Proc. ECCV*, volume 13666, pages 417–435, 2022.

[2162] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit Haim Bermano. Human motion diffusion model. In *Proc. ICLR*, 2023.

[2163] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proc. CVPR*, pages 18000–18010, 2023.

[2164] Taeryung Lee, Gyeongsik Moon, and Kyoung Mu Lee. Multiact: Long-term 3d human motion generation from multiple action labels. In *Proc. AAAI*, pages 1231–1239, 2023.

[2165] Hyemin Ahn, Timothy Ha, Yunho Choi, Hwiyeon Yoo, and Songhwai Oh. Text2action: Generative adversarial synthesis from language to action. In *Proc. ICRA*, pages 1–5, 2018.

[2166] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *Proc. 3DV*, pages 719–728, 2019.

[2167] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *Proc. Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 1376–1386, 2021.

[2168] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: generating diverse human motions from textual descriptions. In *Proc. ECCV 2022*, volume 13682 of *Lecture Notes in Computer Science*, pages 480–497, 2022.

[2169] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. TEACH: temporal action composition for 3d humans. In *Proc. 3DV*, pages 414–423, 2022.

[2170] Guy Tevet, Brian Gordon, Amir Hertz, Amit H. Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to CLIP space. In *Proc. ECCV*, volume 13682, pages 358–374, 2022.

[2171] Fangzhou Hong, Mingyuan Zhang, Liang Pan, Zhongang Cai, Lei Yang, and Ziwei Liu. Avatarclip: zero-shot text-driven generation and animation of 3d avatars. *ACM Trans. Graph.*, 41(4):161:1–161:19, 2022.

[2172] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. TM2T: stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *Proc. ECCV*, volume 13695, pages 580–597, 2022.

[2173] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2M-GPT: generating human motion from textual descriptions with discrete representations. In *Proc. CVPR*, pages 14730–14740, 2023.

[2174] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. FLAME: free-form language-based motion synthesis & editing. In *Proc. AAAI*, pages 8255–8263, 2023.

[2175] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *Proc. CVPR*, pages 9760–9770, 2023.

[2176] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *Proc. CVPR*, pages 1900–1910, 2024.

[2177] Taoran Tang, Jia Jia, and Hanyang Mao. Dance with melody: An lstm-autoencoder approach to music-oriented dance synthesis. In *Proc. ACM MM*, pages 1598–1606, 2018.

[2178] Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. AI choreographer: Music conditioned 3d dance generation with AIST++. In *Proc. Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 13381–13392, 2021.

[2179] Zixuan Wang, Jia Jia, Haozhe Wu, Junliang Xing, Jinghe Cai, Fanbo Meng, Guowen Chen, and Yanfeng Wang. Groupdancer: Music to multi-people dance synthesis with style collaboration. In *Proc. ACM MM*, pages 1138–1146, 2022.

[2180] Nhat Le, Trong-Thang Pham, Tuong Do, Erman Tjiputra, Quang D. Tran, and Anh Nguyen. Music-driven group choreography. In *Proc. CVPR*, pages 8673–8682, 2023.

[2181] Jinwoo Kim, Heeseok Oh, Seongjean Kim, Hoseok Tong, and Sanghoon Lee. A brand new dance partner: Music-conditioned pluralistic dancing controlled by multiple dance genres. In *Proc. CVPR*, pages 3480–3490, 2022.

[2182] Guillermo Valle Pérez, Gustav Eje Henter, Jonas Beskow, Andre Holzapfel, Pierre-Yves Oudeyer, and Simon Alexanderson. Transflower: probabilistic autoregressive dance generation with multimodal attention. *ACM Trans. Graph.*, 40(6):195:1–195:14, 2021.

[2183] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic GPT with choreographic memory. In *Proc. CVPR*, pages 11040–11049, 2022.

[2184] Jonathan Tseng, Rodrigo Castellon, and C. Karen Liu. EDGE: editable dance generation from music. In *Proc. CVPR*, pages 448–458, 2023.

[2185] Kang Chen, Zhipeng Tan, Jin Lei, Song-Hai Zhang, Yuan-Chen Guo, Weidong Zhang, and Shi-Min Hu. Choreomaster: choreography-oriented music-driven dance synthesis. *ACM Trans. Graph.*, 40(4): 145:1–145:13, 2021.

[2186] Jibin Gao, Junfu Pu, Honglun Zhang, Ying Shan, and Wei-Shi Zheng. Pc-dance: Posture-controllable music-driven dance synthesis. In *Proc. ACM MM*, pages 1261–1269, 2022.

[2187] Ho Yin Au, Jie Chen, Junkun Jiang, and Yike Guo. Choreograph: Music-conditioned automatic dance choreography over a style and tempo consistent dynamic graph. In *Proc. ACM MM*, pages 3917–3925, 2022.

[2188] Ruozi Huang, Huang Hu, Wei Wu, Kei Sawada, Mi Zhang, and Daxin Jiang. Dance revolution: Long-term dance generation with music via curriculum learning. In *Proc. ICLR*, 2021.

[2189] Jiangxin Sun, Chunyu Wang, Huang Hu, Hanjiang Lai, Zhi Jin, and Jian-Fang Hu. You never stop dancing: Non-freezing dance generation via bank-constrained manifold projection. In *Proc. NeurIPS*, 2022.

[2190] Andreas Aristidou, Anastasios Yiannakidis, Kfir Aberman, Daniel Cohen-Or, Ariel Shamir, and Yiorgos Chrysanthou. Rhythm is a dancer: Music-driven motion synthesis with global structure. *IEEE Trans. Vis. Comput. Graph.*, 29(8):3519–3534, 2023.

[2191] Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. Learning individual styles of conversational gesture. In *Proc. CVPR*, pages 3497–3506, 2019.

[2192] Taras Kucherenko, Dai Hasegawa, Gustav Eje Henter, Naoshi Kaneko, and Hedvig Kjellström. Analyzing input and output representations for speech-driven gesture generation. In *Proc. IVA*, pages 97–104, 2019.

[2193] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Lingjie Liu, Hans-Peter Seidel, Gerard Pons-Moll, Mohamed Elgharib, and Christian Theobalt. Learning speech-driven 3d conversational gestures from video. In *Proc. IVA*, pages 101–108, 2021.

[2194] Lingting Zhu, Xian Liu, Xuanyu Liu, Rui Qian, Ziwei Liu, and Lequan Yu. Taming diffusion models for audio-driven co-speech gesture generation. In *Proc. CVPR*, pages 10544–10553, 2023.

[2195] Simon Alexanderson, Rajmund Nagy, Jonas Beskow, and Gustav Eje Henter. Listen, denoise, action! audio-driven motion synthesis with diffusion models. *ACM Trans. Graph.*, 42(4):44:1–44:20, 2023.

[2196] Tenglong Ao, Zeyi Zhang, and Libin Liu. Gesturediffuclip: Gesture diffusion model with CLIP latents. *ACM Trans. Graph.*, 42(4):42:1–42:18, 2023. URL https://doi.org/10.1145/3592097.

[2197] Hongwei Yi, Hualin Liang, Yifei Liu, Qiong Cao, Yandong Wen, Timo Bolkart, Dacheng Tao, and Michael J. Black. Generating holistic 3d human motion from speech. In *Proc. CVPR*, pages 469–480, 2023.

[2198] Chaitanya Ahuja, Dong Won Lee, Yukiko I. Nakano, and Louis-Philippe Morency. Style transfer for co-speech gesture animation: A multi-speaker conditional-mixture approach. In *Proc. ECCV*, volume 12363 of *Lecture Notes in Computer Science*, pages 248–265, 2020.

[2199] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *Proc. ECCV*, volume 12346 of *Lecture Notes in Computer Science*, pages 387–404, 2020.

[2200] Jingbo Wang, Sijie Yan, Bo Dai, and Dahua Lin. Scene-aware generative network for human motion synthesis. In *Proc. CVPR*, pages 12206–12215, 2021.

[2201] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael J. Black. Stochastic scene-aware motion prediction. In *Proc. Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 11354–11364, 2021.

[2202] Wei Mao, Miaomiao Liu, Richard I. Hartley, and Mathieu Salzmann. Contact-aware human motion forecasting. In *Proc. NeurIPS*, 2022.

[2203] Jingbo Wang, Yu Rong, Jingyuan Liu, Sijie Yan, Dahua Lin, and Bo Dai. Towards diverse and natural scene-aware 3d human motion synthesis. In *Proc. CVPR*, pages 20428–20437, 2022.

[2204] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. In *Proc. CVPR*, pages 9401–9411, 2021.

[2205] João Pedro Araújo, Jiaman Li, Karthik Vetrivel, Rishi Agarwal, Jiajun Wu, Deepak Gopinath, Alexander Clegg, and C. Karen Liu. CIRCLE: capture in rich contextual environments. In *Proc. CVPR*, pages 21211–21221, 2023.

[2206] Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. *Big Data*, 4(4):236–252, 2016.

[2207] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proc. CVPR*, pages 5142–5151, 2022.

[2208] Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black. BABEL: bodies, action and behavior with english labels. In *Proc. CVPR*, pages 722–731, 2021.

[2209] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5442–5451, 2019.

[2210] Yanli Ji, Feixiang Xu, Yang Yang, Fumin Shen, Heng Tao Shen, and Wei-Shi Zheng. A large-scale RGB-D database for arbitrary-view human action recognition. In *Proc. ACM MM*, pages 1510–1518, 2018.

[2211] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. NTU RGB+D 120: A large-scale benchmark for 3d human activity understanding. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(10):2684–2701, 2020.

[2212] Ylva Ferstl and Rachel McDonnell. Investigating the use of recurrent motion modelling for speech gesture generation. In *Proc. IVA*, pages 93–98, 2018.

[2213] Haiyang Liu, Zihao Zhu, Naoya Iwamoto, Yichen Peng, Zhengqing Li, You Zhou, Elif Bozkurt, and Bo Zheng. BEAT: A large-scale semantic and emotional multi-modal dataset for conversational gestures synthesis. In *Proc. ECCV*, volume 13667, pages 612–630, 2022.

[2214] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3d human pose ambiguities with 3d scene constraints. In *Proc. ICCV*, pages 2282–2292, 2019.

[2215] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. GRAB: A dataset of whole-body human grasping of objects. In *Proc. ECCV*, volume 12349 of *Lecture Notes in Computer Science*, pages 581–600, 2020.

[2216] Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, Wei Liang, and Siyuan Huang. HUMANISE: language-conditioned human motion generation in 3d scenes. In *Proc. NeurIPS*, 2022.

[2217] Zhongang Cai, Daxuan Ren, Ailing Zeng, Zhengyu Lin, Tao Yu, Wenjia Wang, Xiangyu Fan, Yang Gao, Yifan Yu, Liang Pan, Fangzhou Hong, Mingyuan Zhang, Chen Change Loy, Lei Yang, and Ziwei Liu. Humman: Multi-modal 4d human dataset for versatile sensing and modeling. In *Proc. ECCV*, volume 13667 of *Lecture Notes in Computer Science*, pages 557–577, 2022.

[2218] Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. Dancing to music. In *Proc. NeurIPS*, pages 3581–3591, 2019.

[2219] Buyu Li, Yongchi Zhao, Zhelun Shi, and Lu Sheng. Danceformer: Music conditioned 3d dance generation with parametric motion transformer. In *Proc. AAAI*, pages 1272–1279, 2022.

[2220] Youngwoo Yoon, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In *Proc. ICRA*, pages 4303–4309, 2019.

[2221] Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Speech gesture generation from the trimodal context of text, audio, and speaker identity. *ACM Trans. Graph.*, 39(6):222:1–222:16, 2020.

[2222] Tenglong Ao, Qingzhe Gao, Yuke Lou, Baoquan Chen, and Libin Liu. Rhythmic gesticulator: Rhythm-aware co-speech gesture synthesis with hierarchical neural embeddings. *ACM Trans. Graph.*, 41(6): 209:1–209:19, 2022. URL https://doi.org/10.1145/3550454.3555435.

[2223] Saeed Ghorbani, Ylva Ferstl, Daniel Holden, Nikolaus F. Troje, and Marc-André Carbonneau. Zeroeggs: Zero-shot example-based gesture generation from speech. *Comput. Graph. Forum*, 42(1): 206–216, 2023.

[2224] Christian Mandery, Ömer Terlemez, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. The KIT whole-body human motion database. In *Proc. ICAR*, pages 329–336, 2015.

[2225] Manolis Savva, Angel X. Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. Pigraphs: learning interaction snapshots from observations. *ACM Trans. Graph.*, 35(4):139:1–139:12, 2016.

[2226] Áron Monszpart, Paul Guerrero, Duygu Ceylan, Ersin Yumer, and Niloy J. Mitra. imapper: interaction-guided scene mapping from monocular videos. *ACM Trans. Graph.*, 38(4):92:1–92:15, 2019.

[2227] Vladimir Guzov, Aymen Mir, Torsten Sattler, and Gerard Pons-Moll. Human poseitioning system (HPS): 3d human pose estimation and self-localization in large scenes from body-mounted sensors. In *Proc. CVPR*, pages 4318–4329, 2021.

[2228] Xiaohan Zhang, Bharat Lal Bhatnagar, Sebastian Starke, Vladimir Guzov, and Gerard Pons-Moll. COUCH: towards controllable human-chair interactions. In *Proc. ECCV*, volume 13665 of *Lecture Notes in Computer Science*, pages 518–535, 2022.

[2229] Anand Gopalakrishnan, Ankur Arjun Mali, Dan Kifer, C. Lee Giles, and Alexander G. Ororbia II. A neural temporal model for human motion prediction. In *Proc. CVPR*, pages 12116–12125, 2019.

[2230] Julian Tanke, Chintan Zaveri, and Juergen Gall. Intention-based long-term human motion anticipation. In *Proc. 3DV*, pages 596–605, 2021.

[2231] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(12):2878–2890, 2013.

[2232] Sicheng Yang, Zhiyong Wu, Minglei Li, Zhensong Zhang, Lei Hao, Weihong Bao, and Haolin Zhuang. Qpgesture: Quantization-based and phase-guided motion matching for natural speech-driven gesture generation. In *Proc. CVPR*, pages 2321–2330, 2023.

[2233] Alejandro Hernandez Ruiz, Jürgen Gall, and Francesc Moreno. Human motion prediction via spatio-temporal inpainting. In *Proc. Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 7133–7142, 2019.

[2234] Yan Zhang and Siyu Tang. The wanderings of odysseus in 3d scenes. In *Proc. CVPR*, pages 20449–20459, 2022.

[2235] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Proc. ECCV*, volume 12354 of *Lecture Notes in Computer Science*, pages 346–364, 2020.

[2236] Zhangsihao Yang, Mingyuan Zhou, Mengyi Shan, Bingbing Wen, Ziwei Xuan, Mitch Hill, Junjie Bai, Guo-Jun Qi, and Yalin Wang. Omnimotiongpt: Animal motion generation with limited data. In *Proc. CVPR*, pages 1249–1259, 2024.

[2237] Zeyu Zhang, Yiran Wang, Biao Wu, Shuo Chen, Zhiyuan Zhang, Shiya Huang, Wenbo Zhang, Meng Fang, Ling Chen, and Yang Zhao. Motion avatar: Generate human and animal avatars with arbitrary motion. In *Proc. BMVC*, 2024.

[2238] Sigal Raab, Inbal Leibovitch, Guy Tevet, Moab Arar, Amit Haim Bermano, and Daniel Cohen-Or. Single motion diffusion. In *Proc. ICLR*, 2024.

[2239] Xuan Wang, Kai Ruan, Xing Zhang, and Gaoang Wang. Animo: Species-aware model for text-driven animal motion generation. In *Proc. CVPR*, pages 1929–1939, 2025.

[2240] Xuan Wang, Kai Ruan, Liyang Qian, Zhizhi Guo, Chang Su, and Gaoang Wang. X-mogen: Unified motion generation across humans and animals. *CoRR*, abs/2508.05162, 2025.

[2241] Sinead Kearney, Wenbin Li, Martin Parsons, Kwang In Kim, and Darren Cosker. Rgbd-dog: Predicting canine pose from RGBD sensors. In *Proc. CVPR*, pages 8333–8342, 2020.

[2242] Moira Shooter, Charles Malleson, and Adrian Hilton. Benchmarking monocular 3d dog pose estimation using in-the-wild motion capture data. *CoRR*, abs/2406.14412, 2024.

[2243] Yang Li, Hikari Takehara, Takafumi Taketomi, Bo Zheng, and Matthias Nießner. 4dcomplete: Non-rigid motion estimation beyond the observable surface. In *Proc. Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 12686–12696, 2021.

[2244] Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia Schmid. VectorNet: Encoding HD maps and agent dynamics from vectorized representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[2245] Ming Liang, Bin Yang, Rui Hu, Yun Chen, Renjie Liao, Song Feng, and Raquel Urtasun. Learning lane graph representations for motion forecasting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[2246] Nigamaa Nayakanti, Rami Al-Rfou, Aurick Zhou, Kratarth Goel, Khaled S Refaat, and Benjamin Sapp. Wayformer: Motion forecasting via simple and efficient attention networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[2247] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[2248] Kunpeng Zhang, Xiaoliang Feng, Lan Wu, and Zhengbing He. Trajectory prediction for autonomous driving using spatial-temporal graph attention transformer. *IEEE Trans. Intell. Transp. Syst.*, 23(11): 22343–22353, 2022.

[2249] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2250] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social GAN: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[2251] Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[2252] Junru Gu, Chen Sun, and Hang Zhao. DenseTNT: End-to-end trajectory prediction from dense goal sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[2253] Chiyu Max Jiang, Andre Cornman, Cheolho Park, Benjamin Sapp, Yin Zhou, and Dragomir Anguelov. MotionDiffuser: Controllable multi-agent motion prediction using diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[2254] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. ChauffeurNet: Learning to drive by imitating the best and synthesizing the worst. In *Robotics: Science and Systems (RSS)*, 2019.

[2255] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 8748–8757, 2019.

[2256] Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics (TOG)*, 39(4), 2020.

[2257] Setareh Cohan, Guy Tevet, Daniele Reda, Xue Bin Peng, and Michiel van de Panne. Flexible motion in-betweening with diffusion models. In *ACM SIGGRAPH Conference Papers*, 2024.

[2258] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. Human motion diffusion as a generative prior. In *International Conference on Learning Representations (ICLR)*, 2024.

[2259] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[2260] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. OmniControl: Control any joint at any time for human motion generation. In *International Conference on Learning Representations (ICLR)*, 2024.

[2261] Korrawe Karunratanakul, Konpat Preechakul, Emre Aksan, Thabo Beeler, Supasorn Suwajanakorn, and Siyu Tang. Optimizing diffusion noise can serve as universal motion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[2262] Nikos Athanasiou, Alpar Cseke, Markos Diomataris, Michael J Black, and Gül Varol. MotionFix: Text-driven 3d human motion editing. In *SIGGRAPH Asia Conference Papers*, 2024.

[2263] Purvi Goel, Kuan-Chieh Wang, C Karen Liu, and Kayvon Fatahalian. Iterative motion editing with natural language. In *ACM SIGGRAPH Conference Papers*, 2024.

[2264] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:27953–27965, 2022.

[2265] Haoyu Zhao, Chen Yang, Hao Wang, Xingyue Zhao, and Wei Shen. Sg-gs: Photo-realistic animatable human avatars with semantically-guided gaussian splatting. *arXiv e-prints*, pages arXiv–2408, 2024.

[2266] Haoyu Zhao, Hao Wang, Chen Yang, and Wei Shen. Chase: 3d-consistent human avatars with sparse inputs via gaussian splatting and contrastive learning. *arXiv e-prints*, pages arXiv–2408, 2024.

[2267] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. Text-to-4d dynamic scene generation. *arXiv preprint arXiv:2301.11280*, 2023.

[2268] Yufeng Zheng, Xueting Li, Koki Nagano, Sifei Liu, Otmar Hilliges, and Shalini De Mello. A unified approach for text-and image-guided 4d scene generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7300–7309, 2024.

[2269] Yanqin Jiang, Li Zhang, Jin Gao, Weimin Hu, and Yao Yao. Consistent4d: Consistent 360 {\deg} dynamic object generation from monocular video. *arXiv preprint arXiv:2311.02848*, 2023.

[2270] Huan Ling, Seung Wook Kim, Antonio Torralba, Sanja Fidler, and Karsten Kreis. Align your gaussians: Text-to-4d with dynamic 3d gaussians and composed diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8576–8588, 2024.

[2271] Chen Guo, Tianjian Jiang, Xu Chen, Jie Song, and Otmar Hilliges. Vid2avatar: 3d avatar reconstruction from videos in the wild via self-supervised scene decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12858–12868, 2023.

[2272] Haoyu Zhao, Xingyue Zhao, Lingting Zhu, Weixi Zheng, and Yongchao Xu. Hfgs: 4d gaussian splatting with emphasis on spatial and temporal high-frequency components for endoscopic scene reconstruction. *arXiv preprint arXiv:2405.17872*, 2024.

[2273] Qiaowei Miao, Jinsheng Quan, Kehan Li, and Yawei Luo. Pla4d: Pixel-level alignments for text-to-4d gaussian splatting. *arXiv preprint arXiv:2405.19957*, 2024.

[2274] Ruiyuan Gao, Kai Chen, Zhihao Li, Lanqing Hong, Zhenguo Li, and Qiang Xu. Magicdrive3d: Controllable 3d generation for any-view rendering in street scenes. *arXiv preprint arXiv:2405.14475*, 2024.

[2275] Haoyu Zhao, Hao Wang, Xingyue Zhao, Hao Fei, Hongqiu Wang, Chengjiang Long, and Hua Zou. Efficient physics simulation for 3d scenes via mllm-guided gaussian splatting. *arXiv preprint arXiv:2411.12789*, 2024.

[2276] Ce Chen, Shaoli Huang, Xuelin Chen, Guangyi Chen, Xiaoguang Han, Kun Zhang, and Mingming Gong. Ct4d: Consistent text-to-4d generation with animatable meshes. *arXiv preprint arXiv:2408.08342*, 2024.

[2277] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20331–20341, 2024.

[2278] Dejia Xu, Hanwen Liang, Neel P Bhatt, Hezhen Hu, Hanxue Liang, Konstantinos N Plataniotis, and Zhangyang Wang. Comp4d: Llm-guided compositional 4d scene generation. *arXiv preprint arXiv:2403.16993*, 2024.

[2279] Sisi Dai, Xinxin Su, Boyan Wan, Ruizhen Hu, and Kai Xu. Textmesh4d: High-quality text-to-4d mesh generation. *arXiv preprint arXiv:2506.24121*, 2025.

[2280] Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7996–8006, 2024.

[2281] Heng Yu, Chaoyang Wang, Peiye Zhuang, Willi Menapace, Aliaksandr Siarohin, Junli Cao, László Jeni, Sergey Tulyakov, and Hsin-Ying Lee. 4real: Towards photorealistic 4d scene generation via video diffusion models. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37: 45256–45280, 2024.

[2282] Yukang Cao, Liang Pan, Kai Han, Kwan-Yee K Wong, and Ziwei Liu. Avatargo: Zero-shot 4d human-object interaction generation and animation. *arXiv preprint arXiv:2410.07164*, 2024.

[2283] Hanwen Liang, Yuyang Yin, Dejia Xu, Hanxue Liang, Zhangyang Wang, Konstantinos N Plataniotis, Yao Zhao, and Yunchao Wei. Diffusion4d: Fast spatial-temporal consistent 4d generation via video diffusion models. *arXiv preprint arXiv:2405.16645*, 2024.

[2284] Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. Sv4d: Dynamic 3d content generation with multi-frame and multi-view consistency. *arXiv preprint arXiv:2407.17470*, 2024.

[2285] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369, 2010.

[2286] Zeyu Yang, Zijie Pan, Chun Gu, and Li Zhang. Diffusion^2: Dynamic 3d content generation via score composition of video and multi-view diffusion models. *arXiv preprint arXiv:2404.02148*, 2024.

[2287] Qi Sun, Zhiyang Guo, Ziyu Wan, Jing Nathan Yan, Shengming Yin, Wengang Zhou, Jing Liao, and Houqiang Li. Eg4d: Explicit generation of 4d object without score distillation. *arXiv preprint arXiv:2405.18132*, 2024.

[2288] Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

[2289] Yuyang Zhao, Zhiwen Yan, Enze Xie, Lanqing Hong, Zhenguo Li, and Gim Hee Lee. Animate124: Animating one image to 4d dynamic scene. *arXiv preprint arXiv:2311.14603*, 2023.

[2290] Yuyang Zhao, Chung-Ching Lin, Kevin Lin, Zhiwen Yan, Linjie Li, Zhengyuan Yang, Jianfeng Wang, Gim Hee Lee, and Lijuan Wang. Genxd: Generating any 3d and 4d scenes. *arXiv preprint arXiv:2411.02319*, 2024.

[2291] Ruizhi Shao, Youxin Pang, Zerong Zheng, Jingxiang Sun, and Yebin Liu. 360-degree human video generation with 4d diffusion transformer. *ACM Transactions on Graphics (TOG)*, 43(6):1–13, 2024.

[2292] Hui En Pang, Shuai Liu, Zhongang Cai, Lei Yang, Tianwei Zhang, and Ziwei Liu. Disco4d: Disentangled 4d human generation and animation from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26331–26344, 2025.

[2293] Zhiqi Li, Yiming Chen, and Peidong Liu. Dreammesh4d: Video-to-4d generation with sparse-controlled gaussian-mesh hybrid representation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:21377–21400, 2024.

[2294] Jianqi Chen, Biao Zhang, Xiangjun Tang, and Peter Wonka. V2m4: 4d mesh animation reconstruction from a single monocular video. *arXiv preprint arXiv:2503.09631*, 2025.

[2295] Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. 4diffusion: Multi-view video diffusion model for 4d generation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:15272–15295, 2024.

[2296] Yuyang Yin, Dejia Xu, Zhangyang Wang, Yao Zhao, and Yunchao Wei. 4dgen: Grounded 4d content generation with spatial-temporal consistency. *arXiv preprint arXiv:2312.17225*, 2023.

[2297] Zijie Pan, Zeyu Yang, Xiatian Zhu, and Li Zhang. Efficient4d: Fast dynamic 3d object generation from a single-view video. *arXiv preprint arXiv:2401.08742*, 2024.

[2298] Zijie Wu, Chaohui Yu, Yanqin Jiang, Chenjie Cao, Fan Wang, and Xiang Bai. Sc4d: Sparse-controlled video-to-4d generation and motion transfer. In *European Conference on Computer Vision*, pages 361–379, 2024.

[2299] Jiawei Ren, Cheng Xie, Ashkan Mirzaei, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung Wook Kim, Huan Ling, et al. L4gm: Large 4d gaussian reconstruction model. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 37:56828–56858, 2024.

[2300] Ziya Erkoç, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Hyperdiffusion: Generating implicit neural fields with weight-space diffusion. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 14300–14310, 2023.

[2301] Naima Otberdout, Claudio Ferrari, Mohamed Daoudi, Stefano Berretti, and Alberto Del Bimbo. Sparse to dense dynamic 3d facial expression generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20385–20394, 2022.

[2302] Chaoyue Song, Jianfeng Zhang, Xiu Li, Fan Yang, Yiwen Chen, Zhongcong Xu, Jun Hao Liew, Xiaoyang Guo, Fayao Liu, Jiashi Feng, et al. Magicarticulate: Make your 3d models articulation-ready. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15998–16007, 2025.

[2303] Isabella Liu, Zhan Xu, Wang Yifan, Hao Tan, Zexiang Xu, Xiaolong Wang, Hao Su, and Zifan Shi. Riganything: Template-free autoregressive rigging for diverse 3d assets. *ACM Transactions on Graphics (TOG)*, 44(4):1–12, 2025.

[2304] Chaoyue Song, Xiu Li, Fan Yang, Zhongcong Xu, Jiacheng Wei, Fayao Liu, Jiashi Feng, Guosheng Lin, and Jianfeng Zhang. Puppeteer: Rig and animate your 3d models. *arXiv preprint arXiv:2508.10898*, 2025.

[2305] Zijie Wu, Chaohui Yu, Fan Wang, and Xiang Bai. Animateanymesh: A feed-forward 4d foundation model for text-driven universal mesh animation. *arXiv preprint arXiv:2506.09982*, 2025.

[2306] Yutao Feng, Yintong Shang, Xiang Feng, Lei Lan, Shandian Zhe, Tianjia Shao, Hongzhi Wu, Kun Zhou, Chenfanfu Jiang, and Yin Yang. Elastogen: 4d generative elastodynamics. *arXiv preprint arXiv:2405.15056*, 2024.

[2307] Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. In *European Conference on Computer Vision*, pages 53–72, 2024.

[2308] Qitong Yang, Mingtao Feng, Zijie Wu, Shijie Sun, Weisheng Dong, Yaonan Wang, and Ajmal Mian. Beyond skeletons: Integrative latent mapping for coherent 4d sequence generation. *arXiv e-prints*, pages arXiv–2403, 2024.

[2309] Zenghao Chai, Chen Tang, Yongkang Wong, and Mohan Kankanhalli. Star: Skeleton-aware text-based 4d avatar generation with in-network motion retargeting. *arXiv preprint arXiv:2406.04629*, 2024.

[2310] Zhoujie Fu, Jiacheng Wei, Wenhao Shen, Chaoyue Song, Xiaofeng Yang, Fayao Liu, Xulei Yang, and Guosheng Lin. Sync4d: Video guided controllable dynamics for physics-based 4d generation. *arXiv preprint arXiv:2405.16849*, 2024.

[2311] Jiajing Lin, Zhenzhong Wang, Yongjie Hou, Yuzhou Tang, and Min Jiang. Phy124: Fast physics-driven 4d content generation from a single image. *arXiv preprint arXiv:2409.07179*, 2024.

[2312] Shengqu Cai, Duygu Ceylan, Matheus Gadelha, Chun-Hao Paul Huang, Tuanfeng Yang Wang, and Gordon Wetzstein. Generative rendering: Controllable 4d-guided video generation with 2d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7611–7620, 2024.

[2313] Guo-Wei Yang, Dong-Yu Chen, and Tai-Jiang Mu. Sketch-2-4d: Sketch driven dynamic 3d scene generation. *Graphical Models*, 136:101231, 2024.

[2314] Pascale Fung, Yoram Bachrach, Asli Celikyilmaz, Kamalika Chaudhuri, Delong Chen, Willy Chung, Emmanuel Dupoux, Hongyu Gong, Hervé Jégou, Alessandro Lazaric, et al. Embodied ai agents: Modeling the world. *arXiv preprint arXiv:2506.22355*, 2025.

[2315] Xintao Wang. Taxonomy of world models, 2025. URL https://xinntao.notion.site/videoworldmodel-01.

[2316] Xinqing Li, Xin He, Le Zhang, Min Wu, Xiaoli Li, and Yun Liu. A comprehensive survey on world models for embodied ai. *arXiv preprint arXiv:2510.16732*, 2025.

[2317] Jingtao Ding, Yunke Zhang, Yu Shang, Yuheng Zhang, Zefang Zong, Jie Feng, Yuan Yuan, Hongyuan Su, Nian Li, Nicholas Sukiennik, et al. Understanding world or predicting future? a comprehensive survey of world models. *ACM Computing Surveys*, 58(3):1–38, 2025.

[2318] Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. In *International Conference on Learning Representations*, 2025.

[2319] Randall Balestriero and Yann LeCun. Lejepa: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025.

[2320] Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworldmodel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

[2321] Hai Huang, Yann LeCun, and Randall Balestriero. Llm-jepa: Large language models meet joint embedding predictive architectures. In *International Conference on Learning Representations*, 2026.

[2322] Federico Baldassarre, Marc Szafraniec, Basile Terver, Vasil Khalidov, Francisco Massa, Yann LeCun, Patrick Labatut, Maximilian Seitzer, and Piotr Bojanowski. Back to the features: Dino as a foundation for video world models. *arXiv preprint arXiv:2507.19468*, 2025.

[2323] Chen Min, Dawei Zhao, Liang Xiao, Yiming Nie, and Bin Dai. Uniworld: Autonomous driving pre-training via world models. *arXiv preprint arXiv:2308.07234*, 2023.

[2324] Haoyi Zhu, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Chunhua Shen, Jiangmiao Pang, and Tong He. Aether: Geometric-aware unified world modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8535–8546, 2025.

[2325] Jessy Lin, Yuqing Du, Olivia Watkins, Danijar Hafner, Pieter Abbeel, Dan Klein, and Anca Dragan. Learning to model the world with language. *arXiv preprint arXiv:2308.01399*, 2023.

[2326] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio López, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Conference on Robot Learning (CoRL)*, pages 1–16, 2017.

[2327] NVIDIA. Nvidia isaac sdk and isaac sim (2021) — developer documentation, 2021. URL https://docs.nvidia.com/isaac/.

[2328] OpenAI. Sora 2 is here, 2025. URL https://openai.com/index/sora-2/.

[2329] Google. Veo 3, 2025. URL https://aistudio.google.com/models/veo-3.

[2330] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

[2331] Genesis Authors. Genesis: A generative and universal physics engine for robotics and beyond, December 2024. URL https://github.com/Genesis-Embodied-AI/Genesis.

[2332] Xuemeng Yang, Licheng Wen, Tiantian Wei, Yukai Ma, Jianbiao Mei, Xin Li, Wenjie Lei, Daocheng Fu, Pinlong Cai, Min Dou, et al. Drivearena: A closed-loop generative simulation platform for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26933–26943, 2025.

[2333] Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *European conference on computer vision*, pages 55–72, 2024.

[2334] Lening Wang, Wenzhao Zheng, Yilong Ren, Han Jiang, Zhiyong Cui, Haiyang Yu, and Jiwen Lu. Occsora: 4d occupancy generation models as world simulators for autonomous driving. *arXiv preprint arXiv:2405.20337*, 2024.

[2335] Lunjun Zhang, Yuwen Xiong, Ze Yang, Sergio Casas, Rui Hu, and Raquel Urtasun. Copilot4d: Learning unsupervised world models for autonomous driving via discrete diffusion. *arXiv preprint arXiv:2311.01017*, 2023.

[2336] World Labs. Atlas: A world model for spatial intelligence. https://www.worldlabs.ai/blog/atlas, 2026. Accessed: 29 October 2026.

[2337] Nicola Dainese, Matteo Merler, Minttu Alakuijala, and Pekka Marttinen. Generating code world models with large language models guided by monte carlo tree search. In *Advances in Neural Information Processing Systems*, 2024.

[2338] Hao Tang, Darren Key, and Kevin Ellis. Worldcoder, a model-based llm agent: Building world models by writing code and interacting with the environment. In *Advances in Neural Information Processing Systems*, 2024.

[2339] Ziniu Hu, Ahmet Iscen, Aashi Jain, Thomas Kipf, Yisong Yue, David A Ross, Cordelia Schmid, and Alireza Fathi. Scenecraft: An llm agent for synthesizing 3d scenes as blender code. In *International Conference on Machine Learning*, 2024.

[2340] Lirui Wang, Yiyang Ling, Zhecheng Yuan, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, and Xiaolong Wang. Gensim: Generating robotic simulation tasks via large language models. *arXiv preprint arXiv:2310.01361*, 2023.

[2341] Yi Zhang, Yunshuang Wang, Zeyu Zhang, and Hao Tang. Code2worlds: Empowering coding llms for 4d world generation. *arXiv preprint arXiv:2602.11757*, 2026.

[2342] Haoyu Zhang, Siming Li, Zeyu Zhang, Yi Zhang, Tat-Seng Chua, and Hao Tang. Code2worlds-2: Physically plausible agent harness for embodied world simulation. *arXiv preprint arXiv:2610.17651*, 2026.

[2343] Hongyu Wang, Jingquan Wang, Ashvin Anilkumar, Bocheng Zou, Radu Serban, and Dan Negrut. Chronoagentic: A code-based multi-agent world simulator for physically grounded simulation construction. *arXiv preprint arXiv:2605.14398*, 2026.

[2344] Janet Wilde Astington and Jennifer M Jenkins. Theory of mind development and social understanding. *Cognition & Emotion*, 9(2-3):151–165, 1995.

[2345] Nancy J Nersessian. In the theoretician's laboratory: Thought experimenting as mental modeling. In *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association*, pages 291–301, 1993.

[2346] Nancy Staggers and Anthony F. Norcio. Mental models: concepts for human-computer interaction research. *International Journal of Man-machine studies*, 38(4):587–605, 1993.

[2347] Anand S Rao, Michael P Georgeff, et al. Bdi agents: From theory to practice. In *Icmas*, volume 95, pages 312–319, 1995.

[2348] Michael Georgeff, Barney Pell, Martha Pollack, Milind Tambe, and Michael Wooldridge. The belief-desire-intention model of agency. In *International workshop on agent theories, architectures, and languages*, pages 1–10, 1998.

[2349] Matthew Le, Y-Lan Boureau, and Maximilian Nickel. Revisiting the evaluation of theory of mind through question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5872–5877, 2019.

[2350] Kanishk Gandhi, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah Goodman. Understanding social reasoning in language models with language models. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, volume 36, 2023.

[2351] Chuanyang Jin, Yutong Wu, Jing Cao, Jiannan Xiang, Yen-Ling Kuo, Zhiting Hu, Tomer Ullman, Antonio Torralba, Joshua Tenenbaum, and Tianmin Shu. Mmtom-qa: Multimodal theory of mind question answering. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16077–16102, 2024.

[2352] Hao Fei and Yiran Zhao. Mental world modeling. *arXiv preprint arXiv:2607.27201*, 2026.

[2353] Yoav IH Parish and Pascal Müller. Procedural modeling of cities. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 301–308, 2001.

[2354] Aristid Lindenmayer. Mathematical models for cellular interactions in development i. filaments with one-sided inputs. *Journal of theoretical biology*, 18(3):280–299, 1968.

[2355] George Stiny. Introduction to shape and shape grammars. *Environment and planning B: planning and design*, 7(3):343–351, 1980.

[2356] Qi Bing, Chaoyi Zhang, and Weidong Cai. Learning to synthesize graphics programs for geometric artworks. In *International Conference on Pattern Recognition*, pages 259–274, 2025.

[2357] Kevin Ellis, Daniel Ritchie, Armando Solar-Lezama, and Josh Tenenbaum. Learning to infer graphics programs from hand-drawn images. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

[2358] Yunnong Chen, Shixian Ding, YingYing Zhang, Wenkai Chen, Jinzhou Du, Lingyun Sun, and Liuqing Chen. Designcoder: Hierarchy-aware and self-correcting ui code generation with large language models. *arXiv preprint arXiv:2506.13663*, 2025.

[2359] Zhihao Zhu, Zhan Xue, and Zejian Yuan. Automatic graphics program generation using attention-based hierarchical decoder. In *Asian Conference on Computer Vision*, pages 181–196, 2018.

[2360] R Kenny Jones, Theresa Barton, Xianghao Xu, Kai Wang, Ellen Jiang, Paul Guerrero, Niloy J Mitra, and Daniel Ritchie. Shapeassembly: Learning to generate programs for 3d shape structure synthesis. *ACM Transactions on Graphics (TOG)*, 39(6):1–20, 2020.

[2361] Rio Aguina-Kang, Maxim Gumin, Do Heon Han, Stewart Morris, Seung Jean Yoo, Aditya Ganeshan, R Kenny Jones, Qiuhong Anna Wei, Kailiang Fu, and Daniel Ritchie. Open-universe indoor scene generation using llm program synthesis and uncurated object databases. *arXiv preprint arXiv:2403.09675*, 2024.

[2362] Joy Hsu, Emily Jin, Jiajun Wu, and Niloy J Mitra. From programs to poses: Factored real-world scene generation via learned program libraries. *arXiv preprint arXiv:2510.10292*, 2025.

[2363] Yue Yang, Fan-Yun Sun, Luca Weihs, Eli VanderBilt, Alvaro Herrasti, Winson Han, Jiajun Wu, Nick Haber, Ranjay Krishna, Lingjie Liu, et al. Holodeck: Language guided generation of 3d embodied ai environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16227–16237, 2024.

[2364] Kaichun Mo, Shilin Zhu, Angel X. Chang, Li Yi, Subarna Tripathi, Leonidas J. Guibas, and Hao Su. PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 909–918, 2019.

[2365] Jiayuan Mao, Joshua B Tenenbaum, and Jiajun Wu. Neuro-symbolic concepts. *arXiv preprint arXiv:2505.06191*, 2025.

[2366] Jiaming Guo, Rui Zhang, Shaohui Peng, Qi Yi, Xing Hu, Ruizhi Chen, Zidong Du, Ling Li, Qi Guo, Yunji Chen, et al. Efficient symbolic policy learning with differentiable symbolic expression. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 36:36278–36304, 2023.

[2367] Matt Deitke, Eli VanderBilt, Alvaro Herrasti, Luca Weihs, Jordi Salvador, Kiana Ehsani, Winson Han, Eric Kolve, Ali Farhadi, Aniruddha Kembhavi, and Roozbeh Mottaghi. ProcTHOR: Large-Scale Embodied AI Using Procedural Generation. In *NeurIPS*, 2022. Outstanding Paper Award.

[2368] Rong Wang, Kun Sun, and Jonas Kuhn. A pipeline of neural-symbolic integration to enhance spatial reasoning in large language models. *arXiv e-prints*, pages arXiv–2411, 2024.

[2369] Mara Levy, Siddhant Haldar, Lerrel Pinto, and Abhinav Shirivastava. P3-po: Prescriptive point priors for visuo-spatial generalization of robot policies, 2024. URL https://arxiv.org/abs/2412.06784.

[2370] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021.

[2371] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion, 2023. URL https://arxiv.org/abs/2306.03881.

[2372] Haoyu Zhao, Cheng Zeng, Linghao Zhuang, Yaxi Zhao, Shengke Xue, Hao Wang, Xingyue Zhao, Zhongyu Li, Kehan Li, Siteng Huang, et al. High-fidelity simulated data generation for real-world zero-shot robotic manipulation learning with gaussian splatting. *arXiv preprint arXiv:2510.10637*, 2025.

[2373] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence, 2023. URL https://arxiv.org/abs/2305.15347.

[2374] Yuanchen Ju, Kaizhe Hu, Guowei Zhang, Gu Zhang, Mingrun Jiang, and Huazhe Xu. Robo-abc: Affordance generalization beyond categories via semantic correspondence for robot manipulation, 2024. URL https://arxiv.org/abs/2401.07487.

[2375] Homanga Bharadhwaj, Roozbeh Mottaghi, Abhinav Gupta, and Shubham Tulsiani. Track2act: Predicting point tracks from internet videos enables generalizable robot manipulation. In *European Conference on Computer Vision*, pages 306–324, 2024.

[2376] Xiaolin Fang, Bo-Ruei Huang, Jiayuan Mao, Jasmine Shone, Joshua B. Tenenbaum, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Keypoint abstraction using large models for object-relative imitation learning, 2024. URL https://arxiv.org/abs/2410.23254.

[2377] Shijie Wu, Yihang Zhu, Yunao Huang, Kaizhen Zhu, Jiayuan Gu, Jingyi Yu, Ye Shi, and Jingya Wang. Afforddp: Generalizable diffusion policy with transferable affordance, 2025. URL https://arxiv.org/abs/2412.03142.

[2378] Siddhant Haldar and Lerrel Pinto. Point policy: Unifying observations and actions with key points for robot manipulation, 2025. URL https://arxiv.org/abs/2502.20391.

[2379] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

[2380] Haoran Geng, Helin Xu, Chengyang Zhao, Chao Xu, Li Yi, Siyuan Huang, and He Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7081–7091, 2023.

[2381] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *arXiv preprint arXiv:2108.03298*, 2021.

[2382] Yan Duan, Marcin Andrychowicz, Bradly C. Stadie, Jonathan Ho, Jonas Schneider, Ilya Sutskever, Pieter Abbeel, and Wojciech Zaremba. One-shot imitation learning, 2017. URL https://arxiv.org/abs/1703.07326.

[2383] Thanh-Toan Do, Anh Nguyen, and Ian Reid. Affordancenet: An end-to-end deep learning approach for object affordance detection. In *International Conference on Robotics and Automation (ICRA)*, 2018.

[2384] Zengyi Qin, Kuan Fang, Yuke Zhu, Li Fei-Fei, and Silvio Savarese. Keto: Learning keypoint representations for tool manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7278–7285, 2020.

[2385] Fangchen Liu, Kuan Fang, Pieter Abbeel, and Sergey Levine. Moka: Open-vocabulary robotic manipulation through mark-based visual prompting. *arXiv preprint arXiv:2403.03174*, 2024.

[2386] Martin Sundermeyer, Arsalan Mousavian, Rudolph Triebel, and Dieter Fox. Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13438–13444, 2021.

[2387] Zhenyu Jiang, Yifeng Zhu, Maxwell Svetlik, Kuan Fang, and Yuke Zhu. Synergies between affordance and geometry: 6-dof grasp detection via implicit representations. *Robotics: science and systems*, 2021.

[2388] Wentao Yuan, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. M2t2: Multi-task masked transformer for object-centric pick and place. In *Conference on Robot Learning (CoRL)*, 2023.

[2389] Abhay Deshpande, Yuquan Deng, Arijit Ray, Jordi Salvador, Winson Han, Jiafei Duan, Kuo-Hao Zeng, Yuke Zhu, Ranjay Krishna, and Rose Hendrix. Graspmolmo: Generalizable task-oriented grasping via large-scale synthetic data generation. *arXiv preprint arXiv:2505.13441*, 2025.

[2390] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, et al. Transporter networks: Rearranging the visual world for robotic manipulation. *Conference on Robot Learning*, 2020.

[2391] Weiyu Liu, Chris Paxton, Tucker Hermans, and Dieter Fox. Structformer: Learning spatial structure for language-guided semantic rearrangement of novel objects. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6322–6329, 2022.

[2392] Yuchi Zhao, Miroslav Bogdanovic, Chengyuan Luo, Steven Tohme, Kourosh Darvish, Alán Aspuru-Guzik, Florian Shkurti, and Animesh Garg. Anyplace: Learning generalized object placement for robot manipulation. *arXiv preprint arXiv:2502.04531*, 2025.

[2393] Alexander Stoytchev. Learning the affordances of tools using a behavior-grounded approach. In *Towards Affordance-Based Robot Control: International Seminar, Dagstuhl Castle, Germany, June 5-9, 2006. Revised Papers*, pages 140–158, 2008.

[2394] Vadim Tikhanoff, Ugo Pattacini, Lorenzo Natale, and Giorgio Metta. Exploring affordances and tool use on the icub. In *2013 13th IEEE-RAS international conference on humanoid robots (humanoids)*, pages 130–137, 2013.

[2395] Tongzhou Mu, Zhan Ling, Fanbo Xiang, Derek C Yang, Xuan Li, Stone Tao, Jia Huang, Zhiao and-JZhiwei, and Hao Su. Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations. In *Proceedings of the Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021.

[2396] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11097–11107, 2020. Key reference for the PartNet-Mobility dataset.

[2397] Ishika Singh, Ankit Goyal, Stan Birchfield, Dieter Fox, Animesh Garg, and Valts Blukis. Og-vla: 3d-aware vision language action model via orthographic image generation. *arXiv preprint arXiv:2506.01196*, 2025.

[2398] Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

[2399] Qi Sun, Pengfei Hong, Tej Deep Pala, Vernon Toh, U Tan, Deepanway Ghosal, Soujanya Poria, et al. Emma-x: An embodied multimodal action model with grounded chain of thought and look-ahead spatial reasoning. *arXiv preprint arXiv:2412.11974*, 2024.

[2400] Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

[2401] Haoquan Fang, Jiafei Duan, Donovan Clay, Sam Wang, Shuo Liu, Weikai Huang, Xiang Fan, Wei-Chuan Tsai, Shirui Chen, Yi Ru Wang, et al. Molmoact2: Action reasoning models for real-world deployment. *arXiv preprint arXiv:2605.02881*, 2026.

[2402] Lin Shao et al. Robo-abc: Affine-invariant spatial reasoning for robot manipulation. *arXiv preprint arXiv:2401.10321*, 2024.

[2403] Muyang Yan, Miras Mengdibayev, Ardon Floros, Weihang Guo, Lydia E. Kavraki, and Zachary Kingston. Using VLM reasoning to constrain task and motion planning, 2025. Under Review, arXiv:2510.25548.

[2404] Lucas Chen, Shrutheesh R. Iyer, and Zachary Kingston. Differentiable particle optimization for fast sequential manipulation, 2025. Under Review, arXiv:2510.07674.

[2405] C. Daniel Freeman, Erik Frey, Anton Raichuk, Sertan Girgin, Igor Mordatch, and Olivier Bachem. Brax–a differentiable physics engine for large scale rigid body simulation. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021.

[2406] C. P. Quintero et al. Stochastic implicit neural signed distance functions for safe motion planning under sensing uncertainty. *IEEE Robotics and Automation Letters*, 2024.

[2407] Wonsuhk Jung, Utkarsh A. Mishra, et al. Joint model-based model-free diffusion for planning with constraints, 2025. arXiv:2501.xxxxx.

[2408] Jianlan Luo et al. Fmb: a functional manipulation benchmark for generalizable robotic learning. *arXiv preprint arXiv:2301.12345*, 2023.

[2409] Ralf Römer, Alexander von Rohr, and Angela P. Schoellig. Diffusion predictive control with constraints, 2025. URL https://arxiv.org/abs/2412.09342.

[2410] Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8:58443–58469, 2020.

[2411] Bernd Carsten Stahl, Job Timmermans, and Brent Daniel Mittelstadt. The ethics of computing: A survey of the computing-oriented literature. *ACM Comput. Surv.*, 48(4):55:1–55:38, 2016.

[2412] Ronald C Fisher. Governmental expenditure implications of autonomous vehicles. *National Tax Journal*, 73(1):235–258, 2020.

[2413] Hong Wang, Bing Lu, Jun Li, Teng Liu, Yang Xing, Chen Lv, Dongpu Cao, Jingxuan Li, Jinwei Zhang, and Ehsan Hashemi. Risk assessment and mitigation in local path planning for autonomous vehicles with LSTM based predictive model. *IEEE Trans Autom. Sci. Eng.*, 19(4):2738–2749, 2022.

[2414] Dekang Zhu, Qianyi Bu, Zhongpan Zhu, Yujie Zhang, and Zhipeng Wang. Advancing autonomy through lifelong learning: a survey of autonomous intelligent systems. *Frontiers Neurorobotics*, 18, 2024.

[2415] Di Feng, Ali Harakeh, Steven L. Waslander, and Klaus Dietmayer. A review and comparative study on probabilistic object detection in autonomous driving. *IEEE Trans. Intell. Transp. Syst.*, 23(8):9961–9980, 2022.

[2416] Philip Koopman and Frank Fratrik. How many operational design domains, objects, and events? In Huáscar Espinoza, Seán Ó hÉigeartaigh, Xiaowei Huang, José Hernández-Orallo, and Mauricio Castillo-Effen, editors, *Proc. Workshop on Artificial Intelligence Safety AAAI*, volume 2301, 2019.

[2417] Yuxiao Zhang, Alexander Carballo, Hanting Yang, and Kazuya Takeda. Perception and sensing for autonomous vehicles under adverse weather conditions: A survey. *ISPRS Journal of Photogrammetry and Remote Sensing*, 196:146–177, 2023.

[2418] Noor Ul Ain Tahir, Zuping Zhang, Muhammad Asim, Junhong Chen, and Mohammed Ahmed El-Affendi. Object detection in autonomous vehicles under adverse weather: A review of traditional and deep learning approaches. *Algorithms*, 17(3):103, 2024.

[2419] Mazin Hnewa and Hayder Radha. Object detection under rainy conditions for autonomous vehicles: A review of state-of-the-art and emerging techniques. *IEEE Signal Process. Mag.*, 38(1):53–67, 2021.

[2420] M. Hassaballah, Mourad A. Kenk, Khan Muhammad, and Shervin Minaee. Vehicle detection and tracking in adverse weather using a deep learning framework. *IEEE Trans. Intell. Transp. Syst.*, 22(7): 4230–4242, 2021.

[2421] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, Jie Zhou, and Jifeng Dai. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proc. CVPR*, pages 17830–17839, 2023.

[2422] Yangguang Li, Bin Huang, Zeren Chen, Yufeng Cui, Feng Liang, Mingzhu Shen, Fenggang Liu, Enze Xie, Lu Sheng, Wanli Ouyang, and Jing Shao. Fast-bev: A fast and strong bird's-eye view perception baseline. *IEEE Trans. Pattern Anal. Mach. Intell.*, 46(12):8665–8679, 2024.

[2423] Haisong Liu, Yao Teng, Tao Lu, Haiguang Wang, and Limin Wang. Sparsebev: High-performance sparse 3d object detection from multi-camera videos. In *Proc. ICCV*, pages 18534–18544, 2023.

[2424] Wei Zhou, Julie Stephany Berrio, Stewart Worrall, and Eduardo M. Nebot. Automated evaluation of semantic segmentation robustness for autonomous driving. *IEEE Trans. Intell. Transp. Syst.*, 21(5): 1951–1963, 2020.

[2425] Federico Nesti, Giulio Rossolini, Saasha Nair, Alessandro Biondi, and Giorgio C. Buttazzo. Evaluating the robustness of semantic segmentation for autonomous driving against real-world adversarial patch attacks. In *Proc. WACV*, pages 2826–2835, 2022.

[2426] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proc. CVPR*, pages 9223–9232, 2023.

[2427] Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proc. ICCV*, pages 21672–21683, 2023.

[2428] Binhui Xie, Shuang Li, Mingjia Li, Chi Harold Liu, Gao Huang, and Guoren Wang. Sepico: Semantic-guided pixel contrast for domain adaptive semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(7):9004–9021, 2023.

[2429] Abolfazl Abdollahi, Biswajeet Pradhan, Gaurav Sharma, Khairul Nizam Abdul Maulud, and Abdullah M. Alamri. Improving road semantic segmentation using generative adversarial network. *IEEE Access*, 9:64381–64392, 2021.

[2430] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. Citypersons: A diverse dataset for pedestrian detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3221, 2017.

[2431] Junyi Ma, Xieyuanli Chen, Jingyi Xu, and Guangming Xiong. SeqOT: A spatial-temporal transformer network for place recognition using sequential lidar data. *IEEE Transactions on Industrial Electronics*, 70(8):8225–8234, 2022.

[2432] Alex X Lee, Anusha Nagabandi, Pieter Abbeel, and Sergey Levine. Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[2433] Maria Huegle, Gabriel Kalweit, Branka Mirchevska, Moritz Werling, and Joschka Boedecker. Dynamic input for deep reinforcement learning in autonomous driving. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7566–7573, 2019.

[2434] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[2435] Dat Vu, Bao Ngo, and Hung Phan. HybridNets: End-to-end perception network. *arXiv preprint arXiv:2203.09035*, 2022.

[2436] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12878–12895, 2023.

[2437] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17853–17862, 2023.

[2438] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7):e0130140, 2015.

[2439] Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuPlan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021.

[2440] Shyamal Patel, Hyung Park, Paolo Bonato, Leighton Chan, and Mary Rodgers. A review of wearable sensors and systems with application in rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 9:21, 2012.

[2441] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. *ACM Transactions on Graphics*, 37(6):1–15, 2018.

[2442] Timo von Marcard, Bodo Rosenhahn, Michael J. Black, and Gerard Pons-Moll. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. *Computer Graphics Forum*, 36(2):349–360, 2017.

[2443] Manfredo Atzori, Arjan Gijsberts, Claudio Castellini, Barbara Caputo, Anne-Gabrielle Mittaz Hager, Simone Elsig, Giorgio Giatsidis, Franco Bassetto, and Henning Müller. Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific Data*, 1:140053, 2014.

[2444] Xinyu Yi, Yuxiao Zhou, and Feng Xu. Transpose: Real-time 3d human translation and pose estimation with six inertial sensors. *ACM Transactions on Graphics*, 40(4):1–13, 2021.

[2445] Matthew Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2017.

[2446] Yiyue Luo, Yunzhu Li, Michael Foshey, Wan Shou, Pratyusha Sharma, Tomás Palacios, Antonio Torralba, and Wojciech Matusik. Learning human–environment interactions using conformal tactile textiles. *Nature Electronics*, 4:193–201, 2021.

[2447] Manfredo Atzori, Matteo Cognolato, and Henning Müller. Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Frontiers in Neurorobotics*, 10:9, 2016.

[2448] Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2023.

[2449] Claudio Pacchierotti, Stephen Sinclair, Massimiliano Solazzi, Antonio Frisoli, Vincent Hayward, and Domenico Prattichizzo. Wearable haptic systems for the fingertip and the hand: Taxonomy, review, and perspectives. *IEEE Transactions on Haptics*, 10(4):580–600, 2017.

[2450] Aaron J. Young and Daniel P. Ferris. State of the art and future directions for lower limb robotic exoskeletons. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(2):171–182, 2017.

[2451] Sebastian O. H. Madgwick, Andrew J. L. Harrison, and Ravi Vaidyanathan. Estimation of imu and marg orientation using a gradient descent algorithm. In *IEEE International Conference on Rehabilitation Robotics (ICORR)*, pages 1–7, 2011.

[2452] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Soshi Shimada, Vladislav Golyanik, Christian Theobalt, and Feng Xu. Physical inertial poser (pip): Physics-aware real-time human motion tracking from sparse inertial sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13167–13178, 2022.

[2453] Rayan Armani, Changlin Qian, Jiaxi Jiang, and Christian Holz. Ultra inertial poser: Scalable motion capture and tracking from sparse inertial sensors and ultra-wideband ranging. In *ACM SIGGRAPH 2024 Conference Papers*, 2024.

[2454] Xinyu Yi, Yuxiao Zhou, Marc Habermann, Vladislav Golyanik, Shaohua Pan, Christian Theobalt, and Feng Xu. Egolocate: Real-time motion capture, localization, and mapping with sparse body-mounted sensors. *ACM Transactions on Graphics (SIGGRAPH)*, 42(4), 2023.

[2455] Francisco Javier Ordóñez and Daniel Roggen. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1):115, 2016.

[2456] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[2457] Harish Haresamudram, Irfan Essa, and Thomas Plötz. Contrastive predictive coding for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(2):1–26, 2021.

[2458] Huatao Xu, Pengfei Zhou, Rui Tan, Mo Li, and Guobin Shen. Limu-bert: Unleashing the potential of unlabeled data for imu sensing applications. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2021.

[2459] Seungwhan Moon, Andrea Madotto, Zhaojiang Lin, Aparajita Saraf, Amy Bearman, and Babak Damavandi. Imu2clip: Language-grounded motion sensor translation with multimodal contrastive learning. In *Findings of the Association for Computational Linguistics: EMNLP*, 2023.

[2460] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

[2461] Patrick Kaifosh, Thomas R. Reardon, et al. A generic non-invasive neuromotor interface for human-computer interaction. *Nature*, 645:702–711, 2025.

[2462] Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *Proceedings of the 16th International Symposium on Wearable Computers (ISWC)*, pages 108–109, 2012.

[2463] Ricardo Chavarriaga, Hesam Sagha, Alberto Calatroni, Sundara Tejaswi Digumarti, Gerhard Tröster, José del R. Millán, and Daniel Roggen. The opportunity challenge: A benchmark database for on-body sensor-based activity recognition. *Pattern Recognition Letters*, 34(15):2033–2042, 2013.

[2464] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Scaling egocentric vision: The EPIC-KITCHENS dataset. *CoRR*, abs/1804.02748, 2018.

[2465] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2022.

[2466] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[2467] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[2468] Viswanath Sivakumar et al. emg2qwerty: A large dataset with baselines for touch typing using surface electromyography. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024.

[2469] Jakob Engel, Kiran Somasundaram, Michael Goesele, Albert Sun, Alexander Gamino, Andrew Turner, Arjang Talattof, Arnie Yuan, Bilal Souti, Brighid Meredith, et al. Project aria: A new tool for egocentric multi-modal ai research. *arXiv preprint arXiv:2308.13561*, 2023.

[2470] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[2471] Lingni Ma, Yuting Ye, Fangzhou Hong, Vladimir Guzov, Yifeng Jiang, Rowan Postyeni, Luis Pesqueira, Alexander Gamino, Vijay Baiyya, Hyo Jin Kim, et al. Nymeria: A massive collection of egocentric multi-modal human motion in the wild. In *European Conference on Computer Vision*, 2024.

[2472] Per K Enge. The global positioning system: Signals, measurements, and performance. *International Journal of Wireless Information Networks*, 1(2):83–105, 1994.

[2473] Alfred Leick, Lev Rapoport, and Dmitry Tatarnikov. *GPS satellite surveying*. 2015.

[2474] Elliott D Kaplan and Christopher Hegarty. *Understanding GPS/GNSS: principles and applications*. 2017.

[2475] N Ivanov and V Salischev. The glonass system–an overview. *The Journal of Navigation*, 45(2):175–182, 1992.

[2476] J Benedicto, S Dinwiddy, G Gatti, R Lucas, and M Lugert. Galileo: Satellite system design. In *European Space Agency*. 2000.

[2477] Yuanxi Yang, Weiguang Gao, Shuren Guo, Yue Mao, and Yufei Yang. Introduction to beidou-3 navigation satellite system. *Navigation*, 66(1):7–18, 2019.

[2478] Yuanxi Yang, Yue Mao, and Bijiao Sun. Basic performance and future developments of beidou global navigation satellite system. *Satellite Navigation*, 1(1):1, 2020.

[2479] Zishen Li, Liang Wang, Ningbo Wang, Ran Li, and Ang Liu. Real-time gnss precise point positioning with smartphones for vehicle navigation. *Satellite Navigation*, 3(1):19, 2022.

[2480] Farzaneh Zangenehnejad and Yang Gao. Gnss smartphones positioning: Advances, challenges, opportunities, and future perspectives. *Satellite navigation*, 2(1):24, 2021.

[2481] Khondokar Fida Hasan, Yanming Feng, and Yu-Chu Tian. Gnss time synchronization in vehicular ad-hoc networks: Benefits and feasibility. *IEEE Transactions on Intelligent Transportation Systems*, 19(12): 3915–3924, 2018.

[2482] J-P Malingreau. Global vegetation dynamics: satellite observations over asia. *International Journal of Remote Sensing*, 7(9):1121–1146, 1986.

[2483] DV Dobrynin, VV Rozhnov, AA Saveliev, OV Sukhova, and AA Yachmennikova. Integration of satellite tracking data and satellite images for detailed characteristics of wildlife habitats. *Izvestiya, Atmospheric and Oceanic Physics*, 53(9):1060–1071, 2017.

[2484] Vagner G Ferreira, Zheng Gong, Xiufeng He, Yonglei Zhang, and Samuel A Andam-Akorful. Estimating total discharge in the yangtze river basin using satellite-based observations. *Remote Sensing*, 5(7):3415–3430, 2013.

[2485] Compton J Tucker and John RG Townshend. Strategies for monitoring tropical deforestation using satellite data. *International Journal of Remote Sensing*, 21(6-7):1461–1471, 2000.

[2486] Matt Finer, Sidney Novoa, Mikaela J Weisse, Rachael Petersen, Joseph Mascaro, Tamia Souto, Forest Stearns, and Raúl García Martinez. Combating deforestation: From satellite to intervention. *Science*, 360(6395):1303–1305, 2018.

[2487] HF Del Valle, NO Elissalde, D Antonio Gagliardini, and Jorge Milovich. Status of desertification in the patagonian region: Assessment and mapping from satellite imagery. *Arid Land Research and Management*, 12(2):95–121, 1998.

[2488] Alfredo D Collado, Emilio Chuvieco, and Ana Camarasa. Satellite remote sensing analysis to monitor desertification processes in the crop-rangeland boundary of argentina. *Journal of Arid Environments*, 52(1):121–133, 2002.

[2489] Woody Turner, Carlo Rondinini, Nathalie Pettorelli, Brice Mora, Allison K Leidner, Zoltan Szantoi, Graeme Buchanan, Stefan Dech, John Dwyer, Martin Herold, et al. Free and open-access satellite data are key to biodiversity conservation. *Biological Conservation*, 182:173–176, 2015.

[2490] Jun Yang, Peng Gong, Rong Fu, Minghua Zhang, Jingming Chen, Shunlin Liang, Bing Xu, Jiancheng Shi, and Robert Dickinson. The role of satellite remote sensing in climate change studies. *Nature climate change*, 3(10):875–883, 2013.

[2491] Yoram J Kaufman, Didier Tanré, and Olivier Boucher. A satellite view of aerosols in the climate system. *Nature*, 419(6903):215–223, 2002.

[2492] Nathalie Pettorelli, Kamran Safi, and Woody Turner. Satellite remote sensing, biodiversity research and conservation of the future, 2014.

[2493] Sarah A Boyle, Christina M Kennedy, Julio Torres, Karen Colman, Pastor E Pérez-Estigarribia, and Noé U De La Sancha. High-resolution satellite imagery is an important yet underutilized resource in conservation biology. *PLoS One*, 9(1):e86908, 2014.

[2494] David L Verbyla. *Satellite remote sensing of natural resources*. 2022.

[2495] Nathalie Pettorelli. *Satellite remote sensing and the management of natural resources*. 2019.

[2496] Manuel Perez-Ruiz and Shrini K Upadhyaya. Gnss in precision agricultural operations. *New approach of indoor and outdoor localization systems*, page 3e26, 2012.

[2497] Jing Guo, Xingxing Li, Zhenhong Li, Leyin Hu, Guijun Yang, Chunjiang Zhao, David Fairbairn, David Watson, and Maorong Ge. Multi-gnss precise point positioning for precision agriculture. *Precision agriculture*, 19(5):895–911, 2018.

[2498] SJ Maas. Using satellite data to improve model estimates of crop yield. *Agronomy Journal*, 80(4): 655–662, 1988.

[2499] Cs Ferencz, P Bognár, János Lichtenberger, Daniel Hamar, Gy Tarcsai, Gábor Timár, Gábor Molnár, SZ Pásztor, Peter Steinbach, Balázs Székely, et al. Crop yield estimation by satellite remote sensing. *International Journal of Remote Sensing*, 25(20):4113–4149, 2004.

[2500] Chong Fang, Kaishan Song, Hans W Paerl, Pierre-Andre Jacinthe, Zhidan Wen, Ge Liu, Hui Tao, Xiaofeng Xu, Tiit Kutser, Zongming Wang, et al. Global divergent trends of algal blooms detected by satellite during 1982–2018. *Global Change Biology*, 28(7):2327–2340, 2022.

[2501] Sudhanshu Pandey, Ritesh Gautam, Sander Houweling, Hugo Denier Van Der Gon, Pankaj Sadavarte, Tobias Borsdorff, Otto Hasekamp, Jochen Landgraf, Paul Tol, Tim Van Kempen, et al. Satellite observations reveal extreme methane leakage from a natural gas well blowout. *Proceedings of the National Academy of Sciences*, 116(52):26376–26381, 2019.

[2502] LiangLiang Zhao and MinLing Zhu. Ms-yolov7: Yolov7 based on multi-scale for object detection on uav aerial photography. *Drones*, 2023.

[2503] Mingwen Shao, Chao Wang, Wangmeng Zuo, and Deyu Meng. Efficient pyramidal gan for versatile missing data reconstruction in remote sensing images. *TGRS*, 60:1–14, 2022.

[2504] Ioannis Pitas and Ioannis Mademlis. Autonomous uav cinematography. In *ACM Multimedia*, 2022.

[2505] Yuanjie Dang, Chong Huang, Peng Chen, Ronghua Liang, Xin Yang, and Kwang-Ting Cheng. Path-analysis-based reinforcement learning algorithm for imitation filming. *TMM*, 2022.

[2506] Zhedong Zheng, Yunchao Wei, and Yi Yang. University-1652: A multi-view multi-source benchmark for drone-based geo-localization. In *ACM Multimedia*, 2020.

[2507] Tingyu Wang, Zhedong Zheng, Chenggang Yan, Jiyong Zhang, Yaoqi Sun, Bolun Zheng, and Yi Yang. Each part matters: Local patterns facilitate cross-view geo-localization. *TCSVT*, 2022.

[2508] Jinliang Lin, Zhedong Zheng, Zhun Zhong, Zhiming Luo, Shaozi Li, Yi Yang, and Nicu Sebe. Joint representation learning and keypoint detection for cross-view geo-localization. *IEEE Transactions on Image Processing*, 31:3780–3792, 2022.

[2509] Payton Goodrich, Omar Betancourt, Ana Claudia Arias, and Tarek Zohdi. Placement and drone flight path mapping of agricultural soil sensors using machine learning. *Computers and Electronics in Agriculture*, 2023.

[2510] Redmond Ramin Shamshiri, Ibrahim A Hameed, Siva K Balasundram, Desa Ahmad, Cornelia Weltzien, and Muhammad Yamin. Fundamental research on unmanned aerial vehicles to support precision agriculture in oil palm plantations. *Agricultural Robots-Fundamentals and Application*, 2018.

[2511] Liu Liu and Hongdong Li. Lending orientation to neural networks for cross-view geo-localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2019.

[2512] Yujiao Shi, Liu Liu, Xin Yu, and Hongdong Li. Spatial-aware feature aggregation for image based cross-view geo-localization. In *NeurIPS*, 2019.

[2513] Yujiao Shi, Xin Yu, Liu Liu, Tong Zhang, and Hongdong Li. Optimal feature transport for cross-view image geo-localization. In *AAAI*, 2020.

[2514] Yujiao Shi, Xin Yu, Dylan Campbell, and Hongdong Li. Where am i looking at? joint location and orientation estimation by cross-view matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2020.

[2515] Yujiao Shi and Hongdong Li. Beyond cross-view image retrieval: Highly accurate vehicle localization using satellite image. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2022.

[2516] Yujiao Shi, Xin Yu, Liu Liu, Dylan Campbell, Piotr Koniusz, and Hongdong Li. Accurate 3-dof camera geo-localization via ground-to-satellite image matching. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2022.

[2517] Yujiao Shi, Xin Yu, Shan Wang, and Hongdong Li. Cvlnet: Cross-view feature correspondence learning for video-based camera localization. In *ACCV*, 2022.

[2518] Yujiao Shi, Fei Wu, Akhil Perincherry, Ankit Vora, and Hongdong Li. Boosting 3-dof ground-to-satellite camera localization accuracy via geometry-guided cross-view transformer. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2023.

[2519] Zhenbo Song, Xianghui Ze, Jianfeng Lu, and Yujiao Shi. Learning dense flow field for highly-accurate cross-view camera localization. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[2520] Francesco Betti Sorbelli, Federico Corò, Lorenzo Palazzetti, Cristina M Pinotti, and Giulio Rigoni. How the wind can be leveraged for saving energy in a truck-drone delivery system. *TITS*, 2023.

[2521] Didula Dissanayaka, Thumeera R Wanasinghe, Oscar De Silva, Awantha Jayasiri, and George KI Mann. Review of navigation methods for uav-based parcel delivery. *TASE*, 2023.

[2522] Kristina Lemmer. Propulsion for cubesats. *Acta astronautica*, 134:231–243, 2017.

[2523] Jesús Gonzalo, Deibi López, Diego Domínguez, Adrián García, and Alberto Escapa. On the capabilities and limitations of high altitude pseudo-satellites. *Progress in Aerospace Sciences*, 98:37–56, 2018.

[2524] John D Cunningham, Hal J Bloom, and Craig S Nelson. National polar-orbiting operational environmental satellite system: meeting future user needs for earth system observations. In *Weather and Environmental Satellites*, volume 5549, pages 1–10, 2004.

[2525] Alexander Köthe. Flight mechanics and flight control for a multibody aircraft. *Technische Universitaet Berlin: Berlin, Germany*, 2019.

[2526] Matt Fladeland, Susan Schoenung, and Randal Albertson. Demonstrating next generation high-altitude, long endurance aircraft for earth science. Technical report, NASA Ames Research Center, 2019.

[2527] Tingyu Wang, Zhedong Zheng, Yaoqi Sun, Chenggang Yan, Yi Yang, and Tat-Seng Chua. Multiple-environment self-adaptive network for aerial-view geo-localization. *Pattern Recognition*, 152:110363, 2024.

[2528] Tingyu Wang, Zhedong Zheng, Zunjie Zhu, Yaoqi Sun, Yi Yang, and Chenggang Yan. Learning cross-view geo-localization embeddings via dynamic weighted decorrelation regularization. *TGRS*, 2024.

[2529] Zhedong Zheng and Yi Yang. Adaptive boosting for domain adaptation: Toward robust predictions in scene segmentation. *IEEE Transactions on Image Processing*, 31:5371–5382, 2022.

[2530] Zhihao Liu, Yuanyuan Shang, Timing Li, Guanlin Chen, Yu Wang, Qinghua Hu, and Pengfei Zhu. Robust multi-drone multi-target tracking to resolve target occlusion: A benchmark. *TMM*, 2023.

[2531] Tianjiao Li, Jun Liu, Wei Zhang, Yun Ni, Wenqian Wang, and Zhiheng Li. Uav-human: A large benchmark for human behavior understanding with unmanned aerial vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2021.

[2532] Pu Jin, Lichao Mou, Gui-Song Xia, and Xiao Xiang Zhu. Anomaly detection in aerial videos with transformers. *TGRS*, 2022.

[2533] Zhedong Zheng and Yi Yang. Unsupervised scene adaptation with memory regularization in vivo. *IJCAI*, 2019.

[2534] Chao Wang, Zhedong Zheng, Ruijie Quan, Yifan Sun, and Yi Yang. Context-aware pretraining for efficient blind image decomposition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2023.

[2535] Hao Ju, Shaofei Huang, Si Liu, and Zhedong Zheng. Video2bev: Transforming drone videos to bevs for video-based geo-localization. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, 2025.

[2536] Haotian Zhang, Gaoang Wang, Zhichao Lei, and Jenq-Neng Hwang. Eye in the sky: Drone-based object tracking and 3d localization. In *ACM Multimedia*, 2019.

[2537] Longyin Wen, Dawei Du, Pengfei Zhu, Qinghua Hu, Qilong Wang, Liefeng Bo, and Siwei Lyu. Detection, tracking, and counting meets drones in crowds: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2021.

[2538] Jianwu Fang, Lei-Lei Li, Kuan Yang, Zhedong Zheng, Jianru Xue, and Tat-Seng Chua. Cognitive accident prediction in driving scenes: A multimodality benchmark. *arXiv:2212.09381*, 2022.

[2539] Yujiao Shi, Dylan John Campbell, Xin Yu, and Hongdong Li. Geometry-guided street-view panorama synthesis from satellite imagery. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2022.

[2540] Meng Chu, Zhedong Zheng, Wei Ji, Tingyu Wang, and Tat-Seng Chua. Towards natural language-guided drones: Geotext-1652 benchmark with spatial relation matching. In *European Conference on Computer Vision*, pages 213–231, 2024.

[2541] Ou Zheng, Mohamed Abdel-Aty, Lishengsa Yue, Amr Abdelraouf, Zijin Wang, and Nada Mahmoud. Citysim: A drone-based vehicle trajectory dataset for safety oriented research and digital twins. *arXiv:2208.11036*, 2022.

[2542] Scott Workman and Nathan Jacobs. On the location dependence of convolutional neural network features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 70–78, 2015.

[2543] Scott Workman, Richard Souvenir, and Nathan Jacobs. Wide-area image geolocalization with aerial reference imagery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3961–3969, 2015.

[2544] Tsung-Yi Lin, Yin Cui, Serge Belongie, and James Hays. Learning deep representations for ground-to-aerial geolocalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5007–5015, 2015.

[2545] Liu Liu, Hongdong Li, and Yuchao Dai. Stochastic attraction-repulsion embedding for large scale image localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2570–2579, 2019.

[2546] Fabian Deuser, Konrad Habel, and Norbert Oswald. Sample4geo: Hard negative sampling for cross-view geo-localisation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16847–16856, 2023.

[2547] Nam N Vo and James Hays. Localizing and orienting street views using overhead imagery. In *European conference on computer vision*, pages 494–509, 2016.

[2548] Menghua Zhai, Zachary Bessinger, Scott Workman, and Nathan Jacobs. Predicting ground-level scene layout from aerial imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 867–875, 2017.

[2549] Sixing Hu, Mengdan Feng, Rang MH Nguyen, and Gim Hee Lee. Cvm-net: Cross-view matching network for image-based ground-to-aerial geo-localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7258–7267, 2018.

[2550] Hongji Yang, Xiufan Lu, and Yingying Zhu. Cross-view geo-localization with layer-to-layer transformer. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:29009–29020, 2021.

[2551] Xiaohan Zhang, Xingyu Li, Waqas Sultani, Yi Zhou, and Safwan Wshah. Cross-view geo-localization via learning disentangled geometric layout correspondence. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3480–3488, 2023.

[2552] Sijie Zhu, Mubarak Shah, and Chen Chen. Transgeo: Transformer is all you need for cross-view image geo-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1162–1171, 2022.

[2553] Runzhe Zhu, Ling Yin, Mingze Yang, Fei Wu, Yuncheng Yang, and Wenbo Hu. Sues-200: A multi-height multi-scene cross-view image benchmark across drone and satellite. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4825–4839, 2023.

[2554] Yicong Tian, Chen Chen, and Mubarak Shah. Cross-view image matching for geo-localization in urban environments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3616, 2017.

[2555] Gui-Song Xia, Xiang Bai, Jian Ding, Zhen Zhu, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Dota: A large-scale dataset for object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3974–3983, 2018.

[2556] Jian Ding, Nan Xue, Gui-Song Xia, Xiang Bai, Wen Yang, Michael Ying Yang, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, et al. Object detection in aerial images: A large-scale benchmark and challenges. *IEEE transactions on pattern analysis and machine intelligence*, 44(11):7778–7796, 2021.

[2557] Dawei Du, Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Lin, Qinghua Hu, Tao Peng, Jiayu Zheng, Xinyao Wang, Yue Zhang, et al. Visdrone-det2019: The vision meets drone object detection in image challenge results. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pages 0–0, 2019.

[2558] Cédric Vermeulen, Philippe Lejeune, Jonathan Lisein, Prosper Sawadogo, and Philippe Bouché. Unmanned aerial survey of elephants. *PloS one*, 8(2):e54700, 2013.

[2559] Yousaf Albaluchi, Biying Fu, Naser Damer, Raghavendra Ramachandra, and Kiran Raja. Uav-based person re-identification: A survey of uav datasets, approaches, and challenges. *Computer Vision and Image Understanding*, 251:104261, 2025.

[2560] Xian Sun, Peijin Wang, Zhiyuan Yan, Feng Xu, Ruiping Wang, Wenhui Diao, Jin Chen, Jihao Li, Yingchao Feng, Tao Xu, et al. Fair1m: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 184: 116–130, 2022.

[2561] Zikun Liu, Hongzhen Wang, Lubin Weng, and Yiping Yang. Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds. *IEEE Geoscience and Remote Sensing Letters*, 13(8):1074–1078, 2016.

[2562] Gong Cheng, Jiabao Wang, Ke Li, Xingxing Xie, Chunbo Lang, Yanqing Yao, and Junwei Han. Anchor-free oriented proposal generator for object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022.

[2563] Yazhou Yao, Tao Chen, Hanbo Bi, Xinhao Cai, Gensheng Pei, Guoye Yang, Zhiyuan Yan, Xian Sun, Xing Xu, and Hai Zhang. Automated object recognition in high-resolution optical remote sensing imagery. *National Science Review*, 10(6):nwad122, 2023.

[2564] Xue Yang, Jirui Yang, Junchi Yan, Yue Zhang, Tengfei Zhang, Zhi Guo, Xian Sun, and Kun Fu. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8232–8241, 2019.

[2565] Xue Yang, Junchi Yan, Ziming Feng, and Tao He. R3det: Refined single-stage detector with feature refinement for rotating object. In *Proceedings of the AAAI conference on artificial intelligence*, pages 3163–3171, 2021.

[2566] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, and Qikai Lu. Learning roi transformer for oriented object detection in aerial images. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2849–2858, 2019.

[2567] Xingxing Xie, Gong Cheng, Jiabao Wang, Xiwen Yao, and Junwei Han. Oriented r-cnn for object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3520–3529, 2021.

[2568] Liping Hou, Ke Lu, Jian Xue, and Yuqiu Li. Shape-adaptive selection and measurement for oriented object detection. In *Proceedings of the AAAI conference on artificial intelligence*, pages 923–932, 2022.

[2569] Jiaming Han, Jian Ding, Nan Xue, and Gui-Song Xia. Redet: A rotation-equivariant detector for aerial object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2786–2795, 2021.

[2570] Qi Ming, Zhiqiang Zhou, Lingjuan Miao, Hongwei Zhang, and Linhao Li. Dynamic anchor learning for arbitrary-oriented object detection. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2355–2363, 2021.

[2571] Kun Fu, Zhonghan Chang, Yue Zhang, and Xian Sun. Point-based estimator for arbitrary-oriented object detection in aerial images. *IEEE Transactions on Geoscience and Remote Sensing*, 59(5):4370–4387, 2020.

[2572] Wentong Li, Yijie Chen, Kaixuan Hu, and Jianke Zhu. Oriented reppoints for aerial object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1829–1838, 2022.

[2573] Yongchao Xu, Mingtao Fu, Qimeng Wang, Yukang Wang, Kai Chen, Gui-Song Xia, and Xiang Bai. Gliding vertex on the horizontal bounding box for multi-oriented object detection. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 43(4):1452–1459, 2020.

[2574] Xue Yang, Junchi Yan, Qi Ming, Wentao Wang, Xiaopeng Zhang, and Qi Tian. Rethinking rotated object detection with gaussian wasserstein distance loss. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 11830–11841, 2021.

[2575] Xue Yang, Xiaojiang Yang, Jirui Yang, Qi Ming, Wentao Wang, Qi Tian, and Junchi Yan. Learning high-precision bounding box for rotated object detection via kullback-leibler divergence. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 34:18381–18394, 2021.

[2576] Gong Cheng, Yanqing Yao, Shengyang Li, Ke Li, Xingxing Xie, Jiabao Wang, Xiwen Yao, and Junwei Han. Dual-aligned oriented detector. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–11, 2022.

[2577] Wen Qian, Xue Yang, Silong Peng, Junchi Yan, and Yue Guo. Learning modulated loss for rotated object detection. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2458–2466, 2021.

[2578] Wei Zhao, Wenping Ma, Licheng Jiao, Puhua Chen, Shuyuan Yang, and Biao Hou. Multi-scale image block-level f-cnn for remote sensing images object detection. *IEEE Access*, 7:43607–43621, 2019.

[2579] Pourya Shamsolmoali, Masoumeh Zareapoor, Jocelyn Chanussot, Huiyu Zhou, and Jie Yang. Rotation equivariant feature image pyramid network for object detection in optical remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021.

[2580] Yukang Chen, Peizhen Zhang, Zeming Li, Yanwei Li, Xiangyu Zhang, Gaofeng Meng, Shiming Xiang, Jian Sun, and Jiaya Jia. Stitcher: Feedback-driven data provider for object detection. *arXiv preprint arXiv:2004.12432*, 2(7):12, 2020.

[2581] Yuanlin Zhang, Yuan Yuan, Yachuang Feng, and Xiaoqiang Lu. Hierarchical and robust convolutional neural network for very high-resolution remote sensing object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 57(8):5535–5548, 2019.

[2582] Zhuo Zheng, Yanfei Zhong, Ailong Ma, Xiaobing Han, Ji Zhao, Yanfei Liu, and Liangpei Zhang. Hynet: Hyper-scale object detection network framework for multiple spatial resolution remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166:1–14, 2020.

[2583] Guanqun Wang, Yin Zhuang, He Chen, Xiang Liu, Tong Zhang, Lianlin Li, Shan Dong, and Qianbo Sang. Fsod-net: Full-scale object detection from optical remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–18, 2022. doi:10.1109/TGRS.2021.3064599.

[2584] Liping Hou, Ke Lu, and Jian Xue. Refined one-stage oriented object detection method for remote sensing images. *IEEE Transactions on Image Processing*, 31:1545–1558, 2022.

[2585] Wenhua Zhang, Licheng Jiao, Yuxuan Li, Zhongjian Huang, and Haoran Wang. Laplacian feature pyramid network for object detection in vhr optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021.

[2586] Haoyuan Guo, Xi Yang, Nannan Wang, Bin Song, and Xinbo Gao. A rotational libra r-cnn method for ship detection. *IEEE Transactions on Geoscience and Remote Sensing*, 58(8):5772–5781, 2020.

[2587] Wei Guo, Wen Yang, Haijian Zhang, and Guang Hua. Geospatial object detection in high resolution satellite images based on multi-scale convolutional neural network. *Remote Sensing*, 10(1):131, 2018.

[2588] Jie-Bo Hou, Xiaobin Zhu, and Xu-Cheng Yin. Self-adaptive aspect ratio anchor for oriented object detection in remote sensing images. *Remote Sensing*, 13(7):1318, 2021.

[2589] Heqian Qiu, Hongliang Li, Qingbo Wu, Fanman Meng, King Ngi Ngan, and Hengcan Shi. A2rmnet: Adaptively aspect ratio multi-scale network for object detection in remote sensing images. *Remote Sensing*, 11(13):1594, 2019.

[2590] Yifan Pu, Yiru Wang, Zhuofan Xia, Yizeng Han, Yulin Wang, Weihao Gan, Zidong Wang, Shiji Song, and Gao Huang. Adaptive rotated convolution for rotated object detection. In *Proceedings of theInternational Conference on Computer Vision (ICCV)*, pages 6589–6600, 2023.

[2591] Yuxuan Li, Qibin Hou, Zhaohui Zheng, Ming-Ming Cheng, Jian Yang, and Xiang Li. Large selective kernel network for remote sensing object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 16794–16805, October 2023.

[2592] Xiaohan Ding, Xiangyu Zhang, Jungong Han, and Guiguang Ding. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 11963–11975, 2022.

[2593] Shiwei Liu, Tianlong Chen, Xiaohan Chen, Xuxi Chen, Qiao Xiao, Boqian Wu, Mykola Pechenizkiy, Decebal Mocanu, and Zhangyang Wang. More convnets in the 2020s: Scaling up kernels beyond 51x51 using sparsity. *arXiv preprint arXiv:2207.03620*, 2022.

[2594] Meng-Hao Guo, Cheng-Ze Lu, Qibin Hou, Zhengning Liu, Ming-Ming Cheng, and Shi-Min Hu. Segnext: Rethinking convolutional attention design for semantic segmentation. *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 35:1140–1156, 2022.

[2595] Yingpeng Dai, Chenglin Li, Xiaohang Su, Hongxian Liu, and Jiehao Li. Multi-scale depthwise separable convolution for semantic segmentation in street–road scenes. *Remote Sensing*, 15(10):2649, 2023.

[2596] Ke Zhang, Inuwa Mamuda Bello, Yu Su, Jingyu Wang, and Ibrahim Maryam. Multiscale depthwise separable convolution based network for high-resolution image segmentation. *International Journal of Remote Sensing*, 43(18):6624–6643, 2022.

[2597] Xinhao Cai, Qiuxia Lai, Yuwei Wang, Wenguan Wang, Zeren Sun, and Yazhou Yao. Poly kernel inception network for remote sensing detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 27706–27716, 2024.

[2598] Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. *ECCV*, pages 370–386, 2018.

[2599] Syed Waqas Zamir, Aditya Arora, Akshita Gupta, Salman Khan, Guolei Sun, Fahad Shahbaz Khan, Fan Zhu, Ling Shao, Gui-Song Xia, and Xiang Bai. isaid: A large-scale dataset for instance segmentation in aerial images. *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR) Workshop*, 2019.

[2600] Guang Yang, Wenqi Li, Jian Zhang, Wei Wang, and Jun Liu. Lai-yolov5s: A lightweight aerial image object detection algorithm. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–12, 2023.

[2601] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616*, 2024.

[2602] Xiangyu Wang, Donglin Yang, Ziqin Wang, Hohin Kwan, Jinyu Chen, Wenjun Wu, Hongsheng Li, Yue Liao, and Si Liu. Towards realistic uav vision-language navigation: Platform, benchmark, and methodology. *arXiv preprint arXiv:2410.07087*, 2024.

[2603] Georgios Albanis, Nikolaos Zioulis, Anastasios Dimou, Dimitrios Zarpalas, and Petros Daras. Dronepose: photorealistic uav-assistant dataset synthesis for 3d pose estimation via a smooth silhouette loss. In *European Conference on Computer Vision*, pages 663–681, 2020.

[2604] Yue Fan, Winson Chen, Tongzhou Jiang, Chun Zhou, Yi Zhang, and Xin Eric Wang. Aerial vision-and-dialog navigation. *arXiv preprint arXiv:2205.12219*, 2022.

[2605] Shubo Liu, Hongsheng Zhang, Yuankai Qi, Peng Wang, Yanning Zhang, and Qi Wu. Aerialvln: Vision-and-language navigation for uavs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15384–15394, 2023.

[2606] Jungdae Lee, Taiki Miyanishi, Shuhei Kurita, Koya Sakamoto, Daichi Azuma, Yutaka Matsuo, and Nakamasa Inoue. Citynav: Language-goal aerial navigation dataset with geographic information. *arXiv preprint arXiv:2406.14240*, 2024.

[2607] Yunpeng Gao, Chenhui Li, Zhongrui You, Junli Liu, Zhen Li, Pengan Chen, Qizhi Chen, Zhonghan Tang, Liansheng Wang, Penghui Yang, et al. Openfly: A versatile toolchain and large-scale benchmark for aerial vision-language navigation. *arXiv e-prints*, pages arXiv–2502, 2025.

[2608] Xiangbo Gao, Yuheng Wu, Fengze Yang, Xuewen Luo, Keshu Wu, Xinghao Chen, Yuping Wang, Chenxi Liu, Yang Zhou, and Zhengzhong Tu. Airv2x: Unified air-ground vehicle-to-everything collaboration. *arXiv preprint arXiv:2506.19283*, 2025.

[2609] Antonio Loquercio, Ana I Maqueda, Carlos R Del-Blanco, and Davide Scaramuzza. Dronet: Learning to fly by driving. *IEEE Robotics and Automation Letters*, 3(2):1088–1095, 2018.

[2610] Katie Kang, Suneel Belkhale, Gregory Kahn, Pieter Abbeel, and Sergey Levine. Generalization through simulation: Integrating simulated and real data into deep reinforcement learning for vision-based autonomous flight. In *2019 international conference on robotics and automation (ICRA)*, pages 6008–6014, 2019.

[2611] Alessandro Giusti, Jérôme Guzzi, Dan C Cireşan, Fang-Lin He, Juan P Rodríguez, Flavio Fontana, Matthias Faessler, Christian Forster, Jürgen Schmidhuber, Gianni Di Caro, et al. A machine learning approach to visual perception of forest trails for mobile robots. *IEEE Robotics and Automation Letters*, 1 (2):661–667, 2015.

[2612] Nikolai Smolyanskiy, Alexey Kamenev, Jeffrey Smith, and Stan Birchfield. Toward low-flying autonomous mav trail navigation using deep neural networks for environmental awareness. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 4241–4247, 2017.

[2613] Yue Fan, Shilei Chu, Wei Zhang, Ran Song, and Yibin Li. Learn by observation: Imitation learning for drone patrolling from videos of a human navigator. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5209–5216, 2020.

[2614] Ilker Bozcan and Erdal Kayacan. Au-air: A multi-modal unmanned aerial vehicle dataset for low altitude traffic surveillance. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8504–8510, 2020.

[2615] András L Majdik, Charles Till, and Davide Scaramuzza. The zurich urban micro aerial vehicle dataset. *The International Journal of Robotics Research*, 36(3):269–273, 2017.

[2616] Zhefan Xu, Xinming Han, Haoyu Shen, Hanyu Jin, and Kenji Shimada. Navrl: Learning safe flight in dynamic environments. *IEEE Robotics and Automation Letters*, 2025.

[2617] Abhik Singla, Sindhu Padakandla, and Shalabh Bhatnagar. Memory-based deep reinforcement learning for obstacle avoidance in uav with limited environment knowledge. *IEEE transactions on intelligent transportation systems*, 22(1):107–118, 2019.

[2618] Yunpeng Gao, Zhigang Wang, Linglin Jing, Dong Wang, Xuelong Li, and Bin Zhao. Aerial vision-and-language navigation via semantic-topo-metric representation guided llm reasoning. *arXiv preprint arXiv:2410.08500*, 2024.

[2619] Fanglong Yao, Yuanchang Yue, Youzhi Liu, Xian Sun, and Kun Fu. Aeroverse: Uav-agent benchmark suite for simulating, pre-training, finetuning, and evaluating aerospace embodied world models. *arXiv preprint arXiv:2408.15511*, 2024.

[2620] Fei Lin, Yonglin Tian, Tengchao Zhang, Jun Huang, Sangtian Guan, and Fei-Yue Wang. Airvista-ii: An agentic system for embodied uavs toward dynamic scene semantic understanding. *arXiv preprint arXiv:2504.09583*, 2025.

[2621] Yatai Ji, Zhengqiu Zhu, Yong Zhao, Beidan Liu, Chen Gao, Yihao Zhao, Sihang Qiu, Yue Hu, Quanjun Yin, and Yong Li. Towards autonomous uav visual object search in city space: Benchmark and agentic methodology. *arXiv preprint arXiv:2505.08765*, 2025.

[2622] Liangqi Yuan, Chuhao Deng, Dong-Jun Han, and Inseok Hwang. Next-generation llm for uav: From natural language to autonomous flight. *arXiv preprint arXiv:2510.21739*, 2025.

[2623] Ruipu Wu, Yige Zhang, Jinyu Chen, Linjiang Huang, Shifeng Zhang, Xu Zhou, Liang Wang, and Si Liu. Aeroduo: Aerial duo for uav-based vision and language navigation. *arXiv preprint arXiv:2508.15232*, 2025.

[2624] François Michel, Martino Trevisan, Danilo Giordano, and Olivier Bonaventure. A first look at starlink performance. In *Proceedings of the 22nd ACM Internet Measurement Conference*, pages 130–136, 2022.

[2625] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Okada, A. Rodriguez, J. M. Romano, and P. R. Wurman. Analysis and observations from the first amazon picking challenge. In *Proc. IEEE Trans. Automation Science and Engineering (T-ASE)*, pages 172–188, 2016.

[2626] A. A. Tubis, A. Radanliev, N. Choudhary, M. R. Abdulsalam, A. Garliñski, and K. M. Panasiuk. Intelligent warehouse in industry 4.0—systematic review. *Sustainability*, 15(9):1–30, 2023.

[2627] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 32(6):1309–1332, 2016.

[2628] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. In *Proc. IEEE IROS*, pages 5135–5142, 2020.

[2629] M. Labbé and F. Michaud. Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation. *Journal of Field Robotics*, 36(2): 416–446, 2019.

[2630] F. Dellaert. Factor graphs and gtsam: A hands-on introduction. In *Proc. Georgia Institute of Technology Technical Report GT-RIM-CP&R-2012-002*, pages 1–56, 2012.

[2631] M. Veloso, J. Biswas, B. Coltin, and S. Rosenthal. Cobots: Robust symbiotic autonomous mobile service robots. In *Proc. IJCAI*, pages 4423–4429, 2015.

[2632] J.-F. Cordeau, G. Laporte, M. W. P. Savelsbergh, and D. Vigo. Vehicle routing. *Handbooks in Operations Research and Management Science*, 14(1):367–428, 2007.

[2633] G. Sartoretti, J. Kerr, Y. Shi, G. Wagner, T. K. S. Kumar, S. Koenig, and H. Choset. Primal: Pathfinding via reinforcement and imitation multi-agent learning. In *Proc. IEEE ICRA*, pages 2112–2119, 2019.

[2634] L. Perron and V. Furnon. Or-tools: Google's open source software suite for optimization, 2019. URL https://arxiv.org/abs/2002.04437.

[2635] M. M. Solomon. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265, 1987.

[2636] I. Sucan and S. Chitta. Moveit! motion planning framework. In *Proc. IEEE ICRA Workshop on Open Source Software*, pages 1–4, 2010.

[2637] P. A. Lasota and J. A. Shah. Analyzing the effects of human-aware motion planning on close-proximity human–robot collaboration. *Human Factors*, 57(1):21–33, 2015.

[2638] J. Mainprice, R. Hayne, and D. Berenson. Goal set inverse optimal control and iterative replanning for predicting human reaching motions in shared workspaces. In *Proc. IEEE ICRA*, pages 826–833, 2016.

[2639] Anca D Dragan, Kenton CT Lee, and Siddhartha S Srinivasa. Legibility and predictability of robot motion. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 301–308, 2013.

[2640] Zebra Technologies. Zebra rfid and barcode scanning platforms (2021) — product documentation, 2021. URL https://www.zebra.com/.

[2641] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. Ros: An open-source robot operating system. In *Proc. ICRA Workshop on Open Source Software*, 2009.

[2642] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *Proc. CVPR*, pages 7167–7176, 2017.

[2643] R. Bajcsy, Y. Aloimonos, and J. K. Tsotsos. Revisiting active perception. *Autonomous Robots*, 42(2): 177–196, 2018.

[2644] Tong Qin, Peiliang Li, and Shaojie Shen. VINS-Mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Transactions on Robotics*, 34(4):1004–1020, 2018.

[2645] Ruofei Du, Eric Turner, Maksym Dzitsiuk, Luca Prasso, Ivo Duarte, Jason Dourgarian, Joao Afonso, Jose Pascoal, Josh Gladstone, Nuno Cruces, Shahram Izadi, Adarsh Kowdle, Konstantine Tsotsos, and David Kim. DepthLab: Real-time 3D interaction with depth maps for mobile augmented reality. In

Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST), pages 829–843, 2020.

[2646] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 36(6), 2017.

[2647] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 36(6), 2017.

[2648] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. MediaPipe Hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*, 2020.

[2649] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2176–2184, 2016.

[2650] Stephan J. Garbin, Yiru Shen, Immo Schuetz, Robert Cavin, Gregory Hughes, and Sachin S. Talathi. OpenEDS: Open eye dataset. *arXiv preprint arXiv:1905.03702*, 2019.

[2651] Jiaxi Jiang, Paul Streli, Huajian Qiu, Andreas Fender, Larissa Laich, Patrick Snape, and Christian Holz. AvatarPoser: Articulated full-body pose tracking from sparse motion sensing. In *European Conference on Computer Vision (ECCV)*, pages 443–460, 2022.

[2652] Yuming Du, Robin Kips, Albert Pumarola, Sebastian Starke, Ali Thabet, and Artsiom Sanakoyeu. Avatars grow legs: Generating smooth human motion from sparse tracking inputs with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 481–490, 2023.

[2653] Richard A. Bolt. "Put-That-There": Voice and gesture at the graphics interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 262–270, 1980.

[2654] Sharif Razzaque, Zachariah Kohn, and Mary C. Whitton. Redirected walking. In *Eurographics Short Presentations*, 2001.

[2655] Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. Foveated 3D graphics. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 31(6), 2012.

[2656] Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. Towards foveated rendering for gaze-tracked virtual reality. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 35(6), 2016.

[2657] J. M. P. van Waveren. The asynchronous time warp for virtual reality on consumer hardware. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology (VRST)*, pages 37–46, 2016.

[2658] Andrew Maimone, Andreas Georgiou, and Joel S. Kollin. Holographic near-eye displays for virtual and augmented reality. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 36(4), 2017.

[2659] Stephen Lombardi, Jason Saragih, Tomas Simon, and Yaser Sheikh. Deep appearance models for face rendering. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 37(4), 2018.

[2660] Richard A. Newcombe, Dieter Fox, and Steven M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 343–352, 2015.

[2661] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. Gaussianavatars: Photorealistic head avatars with rigged 3d gaussians. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20299–20309, 2024.

[2662] Zhe Li, Zerong Zheng, Lizhen Wang, and Yebin Liu. Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19711–19722, 2024.

[2663] Robert Dongas, Kazjon Grace, and Marius Hoggenmüller. Designing to leverage presence in VR rhythm games. *Multimodal Technologies and Interaction*, 7(2):18, 2023.

[2664] Adrian Azzarelli, Nantheera Anantrasirichai, and David R. Bull. Intelligent cinematography: A review of AI research for cinematographic production. *Artificial Intelligence Review*, 58(4):108, 2025. doi:10.1007/s10462-024-11089-3.

[2665] Werner Kritzinger, Matthias Karner, Georg Traar, Jan Henjes, and Wilfried Sihn. Digital twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*, 51(11): 1016–1022, 2018.

[2666] Michael Grieves and John Vickers. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary Perspectives on Complex Systems*, pages 85–113. 2017.

[2667] Edward Glaessgen and David Stargel. The digital twin paradigm for future NASA and U.S. Air Force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, page 1818, 2012.

[2668] Fei Tao, He Zhang, Ang Liu, and Andrew Y. C. Nee. Digital twin in industry: State-of-the-art. *IEEE Transactions on Industrial Informatics*, 15(4):2405–2415, 2019.

[2669] Yuntao Wang, Zhou Su, Shaolong Guo, Minghui Dai, Tom H. Luan, and Yiliang Liu. A survey on digital twins: Architecture, enabling technologies, security and privacy, and future prospects. *IEEE Internet of Things Journal*, 10(17):14965–14987, 2023.

[2670] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H. Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8456–8465, 2023.

[2671] Zhenyu Jiang, Cheng-Chun Hsu, and Yuke Zhu. Ditto: Building digital twins of articulated objects from interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5616–5626, 2022.

[2672] Zoey Chen, Aaron Walsman, Marius Memmel, et al. URDFormer: A pipeline for constructing articulated simulation environments from real-world images. In *Robotics: Science and Systems*, 2024.

[2673] Xuan Li, Yi-Ling Qiao, Peter Yichen Chen, Krishna Murthy Jatavallabhula, Ming Lin, Chenfanfu Jiang, and Chuang Gan. PAC-NeRF: Physics augmented continuum neural radiance fields for geometry-agnostic system identification. In *International Conference on Learning Representations*, 2023.

[2674] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. PhysGaussian: Physics-integrated 3D Gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4389–4398, 2024.

[2675] Zhao Dong et al. Digital twin catalog: A large-scale photorealistic 3D object digital twin dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[2676] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F. Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. ABO: Dataset and benchmarks for real-world 3D object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21126–21136, 2022.

[2677] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10901–10911, 2021.

[2678] Yuanming Hu, Luke Anderson, Tzu-Mao Li, Qi Sun, Nathan Carr, Jonathan Ragan-Kelley, and Frédo Durand. DiffTaichi: Differentiable programming for physical simulation. In *International Conference on Learning Representations*, 2020.

[2679] Krishna Murthy Jatavallabhula, Miles Macklin, Florian Golemo, Vikram Voleti, Linda Petrini, Martin Weiss, Breandan Considine, Jerome Parent-Lévesque, Kevin Xie, Kenny Erleben, et al. gradSim: Differentiable simulation for system identification and visuomotor control. In *International Conference on Learning Representations*, 2021.

[2680] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[2681] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

[2682] Dirk Hartmann, Matthias Herz, and Utz Wever. Model order reduction a key technology for digital twins. In *Reduced-Order Modeling (ROM) for Simulation and Optimization*, pages 167–179. 2018.

[2683] Francisco Chinesta, Elías Cueto, Emmanuelle Abisset-Chavanne, Jean Louis Duval, and Fouad El Khaldi. Virtual, digital and hybrid twins: A new paradigm in data-based engineering and engineered data. *Archives of Computational Methods in Engineering*, 27(1):105–134, 2020.

[2684] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30, 2017.

[2685] Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *International Conference on Robotics and Automation*, pages 3803–3810, 2018.

[2686] Marcel Torne, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. In *Robotics: Science and Systems*, 2024.

[2687] Tianyuan Dai, Josiah Wong, Yunfan Jiang, Chen Wang, Cem Gokmen, Ruohan Zhang, Jiajun Wu, and Li Fei-Fei. Automated creation of digital cousins for robust policy learning. In *Conference on Robot Learning*, 2024.

[2688] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[2689] Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162, 1994.

[2690] Michael G. Kapteyn, Jacob V. R. Pretorius, and Karen E. Willcox. A probabilistic graphical model foundation for enabling predictive digital twins at scale. *Nature Computational Science*, 1(5):337–347, 2021.

[2691] Fei Tao, Meng Zhang, Yushan Liu, and Andrew Y. C. Nee. Digital twin driven prognostics and health management for complex equipment. *CIRP Annals*, 67(1):169–172, 2018.

[2692] Fabian Dembski, Uwe Wössner, Mike Letzgus, Michael Ruddat, and Claudia Yamu. Urban digital twins for smart cities and citizens: The case study of Herrenberg, Germany. *Sustainability*, 12(6):2307, 2020.

[2693] Kean Huat Soon and Victor Hwee Seng Khoo. CityGML modelling for Singapore 3D national mapping. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume 42, pages 37–42, 2017.

[2694] Jorge Corral-Acero, Francesca Margara, Maciej Marciniak, Cristobal Rodero, Filip Loncaric, Yingjing Feng, Andrew Gilbert, Joao F. Fernandes, Hassaan A. Bukhari, Ali Wajdan, et al. The 'digital twin' to enable the vision of precision cardiology. *European Heart Journal*, 41(48):4556–4564, 2020.

[2695] Evangelia Katsoulakis, Qi Wang, Huanmei Wu, Leili Shahriyari, Richard Fletcher, Jinwei Liu, Luke Achenie, Hongfang Liu, Pamela Jackson, Ying Xiao, et al. Digital twins for health: a scoping review. *npj Digital Medicine*, 7(1):77, 2024.

[2696] Cor Verdouw, Bedir Tekinerdogan, Adrie Beulens, and Sjaak Wolfert. Digital twins in smart farming. *Agricultural Systems*, 189:103046, 2021.

[2697] Yang Liu, Chuanchen Luo, Zhongkai Mao, Junran Peng, and Zhaoxiang Zhang. CityGaussianV2: Efficient and geometrically accurate reconstruction for large-scale scenes. In *International Conference on Learning Representations*, 2025.

[2698] Anasse Boutayeb, Iyad Lahsen-cherif, and Ahmed El Khadimi. A comprehensive GeoAI review: Progress, challenges and outlooks. *arXiv preprint arXiv:2412.11643*, 2024.

[2699] Zhu Han, Ce Zhang, Lianru Gao, Zhiqiang Zeng, Bing Zhang, and Peter M Atkinson. Spatio-temporal multi-level attention crop mapping method using time-series sar imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 206:293–310, 2023.

[2700] Ignazio Gallo, Luigi Ranghetti, Nicola Landro, Riccardo La Grassa, and Mirco Boschetti. In-season and dynamic crop mapping using 3d convolution neural networks and sentinel-2 time series. *ISPRS Journal of Photogrammetry and Remote Sensing*, 195:335–352, 2023.

[2701] Mohammadi Sina, Belgiu Mariana, and Stein Alfred. Improvement in crop mapping from satellite image time series by effectively supervising deep neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 198:272–283, 2023.

[2702] Varun Tiwari, Mirela G Tulbure, Júlio Caineta, Mollie D Gaines, Vinicius Perin, Mustafa Kamal, Timothy J Krupnik, Md Abdullah Aziz, and AFM Tariqul Islam. Automated in-season rice crop mapping using sentinel time-series data and google earth engine: A case study in climate-risk prone bangladesh. *Journal of Environmental Management*, 351:119615, 2024.

[2703] João Batista Ribeiro, Renato Rodrigues da Silva, Jocival Dantas Dias, Mauricio Cunha Escarpinati, and André Ricardo Backes. Automated detection of sugarcane crop lines from uav images using deep learning. *Information Processing in Agriculture*, 11(3):385–396, 2024.

[2704] Franciele Morlin Carneiro, Armando Lopes de Brito Filho, Francielle Morelli Ferreira, Getulio de Freitas Seben Junior, Ziany Neiva Brandão, Rouverson Pereira da Silva, and Luciano Shozo Shiratsuchi. Soil and satellite remote sensing variables importance using machine learning to predict cotton yield. *Smart agricultural technology*, 5:100292, 2023.

[2705] Zeeshan Ramzan, HM Shahzad Asif, Irfan Yousuf, and Muhammad Shahbaz. A multimodal data fusion and deep neural networks based technique for tea yield estimation in pakistan using satellite imagery. *IEEe Access*, 11:42578–42594, 2023.

[2706] Jinbang Peng, Dongliang Wang, Wanxue Zhu, Ting Yang, Zhen Liu, Ehsan Eyshi Rezaei, Jing Li, Zhigang Sun, and Xiaoping Xin. Combination of UAV and deep learning to estimate wheat yield at ripening stage: The potential of phenotypic features. *Int. J. Appl. Earth Obs. Geoinformation*, 124: 103494, 2023.

[2707] Yanyu Wang, Ke Zhang, Chunlan Tang, Qiang Cao, Yongchao Tian, Yan Zhu, Weixing Cao, and Xiaojun Liu. Estimation of rice growth parameters based on linear mixed-effect model using multispectral images from fixed-wing unmanned aerial vehicles. *Remote Sensing*, 11(11), 2019.

[2708] Emmanuel Abiodun Abioye, Mohamad Shukri Zainal Abidin, Mohd Saiful Azimi Mahmud, Salinda Buyamin, Mohamad Hafis Izran Ishak, Muhammad Khairie Idham Abd Rahman, Abdulrahaman Okino Otuoze, Patrick Onotu, and Muhammad Shahrul Azwan Ramli. A review on

monitoring and advanced control strategies for precision irrigation. *Comput. Electron. Agric.*, 173: 105441, 2020.

[2709] Anesmar Olino de Albuquerque, Osmar Luiz Ferreira de Carvalho Jr., Cristiano Rosa Silva, Argélica Saiaka Luiz, Pablo Pozzobon de Bem, Roberto Arnaldo Trancoso Gomes, Renato Fontes Guimarães, and Osmar Abílio de Carvalho Júnior. Dealing with clouds and seasonal changes for center pivot irrigation systems detection using instance segmentation in sentinel-2 time series. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 14:8447–8457, 2021.

[2710] Ehsan Raei, Ata Akbari Asanjan, Mohammad Reza Nikoo, Mojtaba Sadegh, Shokoufeh Pourshahabi, and Jan Franklin Adamowski. A deep learning image segmentation model for agricultural irrigation system classification. *Comput. Electron. Agric.*, 198:106977, 2022.

[2711] Harikrishnan Muraleedharan Jalajamony, Midhun Nair, Patricia F. Mead, and Renny Edwin Fernandez. Drone aided thermal mapping for selective irrigation of localized dry spots. *IEEE Access*, 11:7320–7335, 2023.

[2712] Abebaw Alem and Shailender Kumar. Deep learning models performance evaluations for remote sensed image classification. *IEEE Access*, 10:111784–111793, 2022.

[2713] Boan Chen, Quanlong Feng, Bowen Niu, Fengqin Yan, Bingbo Gao, Jianyu Yang, Jianhua Gong, and Jiantao Liu. Multi-modal fusion of satellite and street-view images for urban village classification based on a dual-branch deep neural network. *Int. J. Appl. Earth Obs. Geoinformation*, 109:102794, 2022.

[2714] Sally El Hajjar, Fahed Abdallah, H Kassem, and H Omrani. Long short-term memory and attention models for simulating urban densification. *Sustainable Cities and Society*, 98:104852, 2023.

[2715] Shuting Sun, Lin Mu, Ruyi Feng, Lizhe Wang, and Jijun He. Gan-based LUCC prediction via the combination of prior city planning information and land-use probability. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 14:10189–10198, 2021.

[2716] Alireza Attarhay Tehrani, Omid Veisi, Bahereh Vojdani Fakhr, and Delong Du. Predicting solar radiation in the urban area: A data-driven analysis for sustainable city planning using artificial neural networking. *Sustainable Cities and Society*, 100:105042, 2024.

[2717] Hikmet Canli and Sinan Toklu. Deep learning-based mobile application design for smart parking. *IEEE Access*, 9:61171–61183, 2021.

[2718] Nanqi Ye, Bowen Wang, Michihiro Kita, Ming Xie, and Wenyue Cai. Urban commerce distribution analysis based on street view and deep learning. *IEEE Access*, 7:162841–162849, 2019.

[2719] Dancho Panovski and Titus Zaharia. Long and short-term bus arrival time prediction with traffic density matrix. *IEEE Access*, 8:226267–226284, 2020.

[2720] Noureen Zafar, Irfan Ul Haq, Huniya Sohail, Jawad-ur-Rehman Chughtai, and Muhammad Muneeb. Traffic prediction in smart cities based on hybrid feature space. *IEEE Access*, 10:134333–134348, 2022.

[2721] Yihang Lu, Xianwei Meng, Liqun Peng, Shucai Xu, and Enhong Chen. IODRNN - incremental output decomposition for a valid traffic flow prediction with GNSS data. *Eng. Appl. Artif. Intell.*, 128:107520, 2024.

[2722] Lei Chen, Lu Zhang, Shanshan Cao, Zhiang Wu, and Jie Cao. Personalized itinerary recommendation: Deep and collaborative learning with textual information. *Expert Syst. Appl.*, 144:113070, 2020.

[2723] Erik Rúa, Antón Núñez-Seoane, Pedro Arias, and Joaquín Martínez-Sánchez. Automatic detection to inventory road slopes using open lidar point clouds. *Int. J. Appl. Earth Obs. Geoinformation*, 118: 103225, 2023.

[2724] Sean P. Kearney, Nicholas C. Coops, Suborna Sethi, and Gordon B. Stenhouse. Maintaining accurate, current, rural road network data: An extraction and updating routine using rapideye, participatory

gis and deep learning. *International Journal of Applied Earth Observation and Geoinformation*, 87:102031, 2020.

[2725] Javier Grandio, Belén Riveiro, Daniel Lamas, and Pedro Arias. Multimodal deep learning for point cloud panoptic segmentation of railway environments. *Automation in Construction*, 150:104854, 2023.

[2726] Jianhua Guo, Qingsong Xu, Yue Zeng, Zhiheng Liu, and Xiao Xiang Zhu. Nationwide urban tree canopy mapping and coverage assessment in brazil from high-resolution remote sensing images using deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 198:1–15, 2023.

[2727] Shinjita Das, Qian Chayn Sun, Heng Zhou, et al. Geoai to implement an individual tree inventory: Framework and application of heat mitigation. *Urban Forestry & Urban Greening*, 74:127634, 2022.

[2728] Angus J Carnegie, Harry Eslick, Paul Barber, Matthew Nagel, and Christine Stone. Airborne multispectral imagery and deep learning for biosecurity surveillance of invasive forest pests in urban landscapes. *Urban forestry & urban greening*, 81:127859, 2023.

[2729] Peiqing Lou, Tonghua Wu, Sizhong Yang, Xiaodong Wu, Jianjun Chen, Xiaofan Zhu, Jie Chen, Xingchen Lin, Ren Li, Chengpeng Shang, Dong Wang, Yune La, Amin Wen, and Xin Ma. Deep learning reveals rapid vegetation greening in changing climate from 1988 to 2018 on the Qinghai-Tibet Plateau. *Ecological Indicators*, 148:110020, 2023.

[2730] Shidong Wang, Dunyue Cui, Lu Wang, and Jinyan Peng. Applying deep-learning enhanced fusion methods for improved ndvi reconstruction and long-term vegetation cover study: A case of the danjiang river basin. *Ecological Indicators*, 155:111088, 2023.

[2731] Qinchuan Xin, Jing Li, Ziming Li, Yaoming Li, and Xuewen Zhou. Evaluations and comparisons of rule-based and machine-learning-based methods to retrieve satellite-based vegetation phenology using MODIS and USA national phenology network data. *Int. J. Appl. Earth Obs. Geoinformation*, 93: 102189, 2020.

[2732] Han Ma and Shunlin Liang. Development of the glass 250-m leaf area index product (version 6) from modis data using the bidirectional lstm deep learning model. *Remote sensing of environment*, 273: 112985, 2022.

[2733] Minsu Kim, Dominik Brunner, and Gerrit Kuhlmann. Importance of satellite observations for high-resolution mapping of near-surface no2 by machine learning. *Remote Sensing of Environment*, 264:112573, 2021.

[2734] Ronald Opio, Isaac Mugume, Joyce Nakatumba-Nabende, and Michael Mbogga. Modeling the atmospheric dispersion of so2 from mount nyiragongo. *Journal of African Earth Sciences*, 197:104771, 2023.

[2735] Andrew Rowley and Oktay Karakuş. Predicting air quality via multimodal ai and satellite imagery. *Remote Sensing of Environment*, 293:113609, 2023.

[2736] Dewen Seng, Qiyan Zhang, Xuefeng Zhang, Guangsen Chen, and Xiyuan Chen. Spatiotemporal prediction of air quality based on lstm neural network. *Alexandria Engineering Journal*, 60(2), 2021.

[2737] Jiyunting Sun, J. Pepijn Veefkind, Peter van Velthoven, and Pieternel F. Levelt. Aerosol absorption over land derived from the ultra-violet aerosol index by deep learning. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 14:9692–9710, 2021.

[2738] Saeed Alqadhi, Ahmed Ali Bindajam, Javed Mallick, Swapan Talukdar, and Atiqur Rahman. Applying deep learning to manage urban ecosystems in arid abha, saudi arabia: Remote sensing-based modelling for ecological condition assessment and decision-making. *Heliyon*, 10(4), 2024.

[2739] Lorenzo Gianquintieri, Daniele Oxoli, Enrico Gianluca Caiani, and Maria Antonia Brovelli. Implementation of a geoai model to assess the impact of agricultural land on the spatial distribution of pm2. 5 concentration. *Chemosphere*, 352:141438, 2024.

[2740] Valeriy Shevchenko, Aleksandr Lukashevich, Daria Taniushkina, Alexander Bulkin, Roland Grinis, Kirill Kovalev, Veronika Narozhnaia, Nazar Sotiriadi, Alexander Krenke, and Yury Maximov. Climate change impact on agricultural land suitability: An interpretable machine learning-based eurasia case study. *IEEE Access*, 12:15748–15763, 2024.

[2741] Md Touhidul Islam, Keisuke Yoshida, Satoshi Nishiyama, and Koichi Sakai. Mutual validation of remote hydraulic estimates and flow model simulations using uav-borne lidar and deep learning-based imaging techniques. *Results in Engineering*, 20:101415, 2023.

[2742] Ling Du, Gregory W McCarty, Xia Li, Xin Zhang, Martin C Rabenhorst, Megan W Lang, Zhenhua Zou, Xuesong Zhang, and Audra L Hinson. Drainage ditch network extraction from lidar data using deep convolutional neural networks in a low relief landscape. *Journal of Hydrology*, 628:130591, 2024.

[2743] Shuting Sun, Lin Mu, Ruyi Feng, Yifu Chen, and Wei Han. Quadtree decomposition-based deep learning method for multiscale coastline extraction with high-resolution remote sensing imagery. *Science of Remote Sensing*, 9:100112, 2024.

[2744] Ting On Chan, Simin Zhang, Linyuan Xia, Ming Luo, Jinhua Wu, and Joseph L. Awange. A novel reflectance transformation and convolutional neural network framework for generating bathymetric data for long rivers: A case study on the bei river in south china. *Int. J. Appl. Earth Obs. Geoinformation*, 127:103682, 2024.

[2745] Eyob Betru Wegayehu and Fiseha Behulu Muluneh. Super ensemble based streamflow simulation using multi-source remote sensing and ground gauged rainfall data fusion. *Heliyon*, 9(7):e17982, 2023.

[2746] Rami Al-Ruzouq, Abdallah Shanableh, Ratiranjan Jena, Sunanda Mukherjee, Mohamad Ali Khalil, Mohamed Barakat A Gibril, Biswajeet Pradhan, and Nezar Atalla Hammouri. Hybrid deep learning and remote sensing for the delineation of artificial groundwater recharge zones. *The Egyptian Journal of Remote Sensing and Space Sciences*, 27(2):178–191, 2024.

[2747] Hossein Sahour, Mohamed Sultan, Bassam Abdellatif, Mustafa Emil, Abotalib Z Abotalib, Karem Abdelmohsen, Mehdi Vazifedan, Abdullah T Mohammad, Safaa M Hassan, Mohamed R Metwalli, et al. Identification of shallow groundwater in arid lands using multi-sensor remote sensing data and machine learning algorithms. *Journal of Hydrology*, 614:128509, 2022.

[2748] Viet-Ha Nhu, Duong Cao Phan, Pham Viet Hoa, Pham The Vinh, and Dieu Tien Bui. A new approach based on deep neural networks and multisource geospatial data for spatial prediction of groundwater spring potential. *IEEE Access*, 12:26344–26363, 2024.

[2749] Jae Young Seo and Sang-Il Lee. Predicting changes in spatiotemporal groundwater storage through the integration of multi-satellite data and deep learning models. *IEEE Access*, 9:157571–157583, 2021.

[2750] F Sabzehee, AR Amiri-Simkooei, S Iran-Pour, BD Vishwakarma, and R Kerachian. Enhancing spatial resolution of grace-derived groundwater storage anomalies in urmia catchment using machine learning downscaling methods. *Journal of Environmental Management*, 330:117180, 2023.

[2751] Zhibin Sun, Ni-Bin Chang, Chi-Farn Chen, Chandan Mostafiz, and Wei Gao. Ensemble learning via higher order singular value decomposition for integrating data and classifier fusion in water quality monitoring. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14: 3345–3360, 2021.

[2752] Wenlan Yang, Bolin Fu, Sunzhe Li, Zhinan Lao, Tengfang Deng, Wen He, Hongchang He, and Zhikun Chen. Monitoring multi-water quality of internationally important karst wetland through deep learning, multi-sensor and multi-platform remote sensing images: A case study of guilin, china. *Ecological Indicators*, 154:110755, 2023.

[2753] Sensen Wu, Jin Qi, Zhen Yan, Fangzheng Lyu, Tao Lin, Yuanyuan Wang, and Zhenhong Du. Spatiotemporal assessments of nutrients and water quality in coastal areas using remote sensing

and a spatiotemporal deep learning model. *International Journal of Applied Earth Observation and Geoinformation*, 112:102897, 2022.

[2754] Taleb Vakili and Jamil Amanollahi. Determination of optically inactive water quality variables using landsat 8 data: A case study in geshlagh reservoir affected by agricultural land use. *Journal of Cleaner Production*, 247:119134, 2020.

[2755] Edson Filisbino Freire da Silva, Evlyn Márcia Leão de Moraes Novo, Felipe de Lucia Lobo, Cláudio Clemente Faria Barbosa, Carolline Tressmann Cairo, Mauricio Almeida Noernberg, and Luiz Henrique da Silva Rotta. A machine learning approach for monitoring brazilian optical water types using sentinel-2 msi. *Remote Sensing Applications: Society and Environment*, 23:100583, 2021.

[2756] Dan Niu, Hongshu Che, Chunlei Shi, Zengliang Zang, Hongbin Wang, Xunlai Chen, and Qunbo Huang. A heterogeneous spatiotemporal attention fusion prediction network for precipitation nowcasting. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16: 8286–8296, 2023.

[2757] Qizhao Jin, Xinbang Zhang, Xinyu Xiao, Ying Wang, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Spatiotemporal inference network for precipitation nowcasting with multimodal fusion. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:1299–1314, 2023.

[2758] Chuyao Luo, Xutao Li, and Yunming Ye. Pfst-lstm: A spatiotemporal lstm model with pseudoflow prediction for precipitation nowcasting. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:843–857, 2020.

[2759] Cenker Sengoz, Sheela Ramanna, Scott Kehler, Rushil Goomer, and Paul Pries. Machine learning approaches to improve north american precipitation forecasts. *IEEE Access*, 11:97664–97681, 2023.

[2760] Ratiranjan Jena, Biswajeet Pradhan, Sambit Prasanajit Naik, and Abdullah M Alamri. Earthquake risk assessment in ne india using deep learning and geospatial analysis. *Geoscience Frontiers*, 12(3):101110, 2021.

[2761] Pan Xiong, Cheng Long, Huiyu Zhou, Xuemin Zhang, and Xuhui Shen. Gnss tec-based earthquake ionospheric perturbation detection using a novel deep learning framework. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:4248–4263, 2022.

[2762] Amna Hafeez, Munawar Shah, Muhsan Ehsan, Punyawi Jamjareegulgarn, Junaid Ahmed, M Arslan Tariq, Shahid Iqbal, and Najam Abbas Naqvi. Possible atmosphere and ionospheric anomalies of the 2019 pakistan earthquake using statistical and machine learning procedures on modis lst, gps tec, and gim tec. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14: 11126–11133, 2021.

[2763] Yolanda Torres, José Juan Arranz, Jorge M Gaspar-Escribano, Azadeh Haghi, Sandra Martínez-Cuevas, Belén Benito, and Juan Carlos Ojeda. Integration of lidar and multispectral images for rapid exposure and earthquake vulnerability estimation. application in lorca, spain. *International Journal of Applied Earth Observation and Geoinformation*, 81:161–175, 2019.

[2764] Vivien Zahs, Katharina Anders, Julia Kohns, Alexander Stark, and Bernhard Höfle. Classification of structural building damage grades from multi-temporal photogrammetric point clouds using a machine learning model trained on virtual laser scanning data. *International Journal of Applied Earth Observation and Geoinformation*, 122:103406, 2023.

[2765] Supattra Puttinaovarat and Paramate Horkaew. Flood forecasting system based on integrated big and crowdsource data by using machine learning techniques. *IEEE Access*, 8:5885–5905, 2020.

[2766] Mohammed Moishin, Ravinesh C. Deo, Ramendra Prasad, Nawin Raj, and Shahab A. Abdulla. Designing deep-based learning flood forecast model with convlstm hybrid algorithm. *IEEE Access*, 9: 50982–50993, 2021.

[2767] Wenmei Li, Jiaqi Wu, Huaihuai Chen, Yu Wang, Yan Jia, and Guan Gui. Unet combined with attention mechanism method for extracting flood submerged range. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 15:6588–6597, 2022.

[2768] Tejas Rathod, Vinay Patil, R Harikrishnan, and Priti Shahane. Multipurpose deep learning-powered uav for forest fire prevention and emergency response. *HardwareX*, 16:e00479, 2023.

[2769] Van The Pham, Tuyet Anh Thi Do, Hau Duc Tran, and Anh Ngoc Thi Do. Classifying forest cover and mapping forest fire susceptibility in dak nong province, vietnam utilizing remote sensing and machine learning. *Ecological Informatics*, 79:102392, 2024.

[2770] Xiwen Chen, Bryce Hopkins, Hao Wang, Leo O'Neill, Fatemeh Afghah, Abolfazl Razi, Peter Fulé, Janice Coen, Eric Rowell, and Adam Watts. Wildland fire detection and monitoring using a drone-collected rgb/ir image dataset. *IEEE Access*, 10:121301–121317, 2022.

[2771] Ali Akbar Siddique, Nada Alasbali, Maha Driss, Wadii Boulila, Mohammed S. Alshehri, and Jawad Ahmad. Sustainable collaboration: Federated learning for environmentally conscious forest fire classification in green internet of things (iot). *Internet Things*, 25:101013, 2024.

[2772] Ah Young Cho, Si-eun Park, Duk-jin Kim, Junwoo Kim, Chenglei Li, and Juyoung Song. Burned area mapping using unitemporal planetscope imagery with a deep learning based approach. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, 16:242–253, 2023.

[2773] Siva R. Venna, Amirhossein Tavanaei, Raju N. Gottumukkala, Vijay V. Raghavan, Anthony S. Maida, and Stephen Nichols. A novel data-driven model for real-time influenza forecasting. *IEEE Access*, 7: 7691–7701, 2019.

[2774] Hongping Hu, Haiyan Wang, Feng Wang, Daniel Langley, Adrian Avram, and Maoxing Liu. Prediction of influenza-like illness based on the improved artificial tree algorithm and artificial neural network. *Scientific reports*, 8(1):4895, 2018.

[2775] Fred S Lu, Mohammad W Hattab, Cesar Leonardo Clemente, Matthew Biggerstaff, and Mauricio Santillana. Improved state-level influenza nowcasting in the united states leveraging internet-based data and network approaches. *Nature communications*, 10(1):147, 2019.

[2776] Pi Guo, Tao Liu, Qin Zhang, Li Wang, Jianpeng Xiao, Qingying Zhang, Ganfeng Luo, Zhihao Li, Jianfeng He, Yonghui Zhang, et al. Developing a dengue forecast model using machine learning: A case study in china. *PLoS neglected tropical diseases*, 11(10):e0005973, 2017.

[2777] Adam Sadilek, Stephanie Caty, Lauren DiPrete, Raed Mansour, Tom Schenk Jr., Mark Bergtholdt, Ashish Jha, Prem Ramaswami, and Evgeniy Gabrilovich. Machine-learned epidemiology: real-time detection of foodborne illness at scale. *npj Digit. Medicine*, 1, 2018.

[2778] Yunxiang Zhao, Mingda Hu, Yuan Jin, Fei Chen, Xin Wang, Boqian Wang, Junjie Yue, and Hongguang Ren. Predicting the transmission trend of respiratory viruses in new regions via geospatial similarity learning. *Int. J. Appl. Earth Obs. Geoinformation*, 125:103559, 2023.

[2779] Cecilia Cordeiro da Silva, Clarisse Lins de Lima, Ana Clara Gomes da Silva, Eduardo Luiz Silva, Gabriel Souza Marques, Lucas Job Brito de Araújo, Luiz Antônio Albuquerque Júnior, Samuel Barbosa Jatobá de Souza, Maíra Araújo de Santana, Juliana Carneiro Gomes, et al. Covid-19 dynamic monitoring and real-time spatio-temporal forecasting. *Frontiers in public health*, 9:641253, 2021.

[2780] Sofiane Atek, Cristiano Pesaresi, Marco Eugeni, Corrado De Vito, Vincenzo Cardinale, Massimo Mecella, Antonello Rescio, Luca Petronzio, Aldo Vincenzi, Pasquale Pistillo, et al. A geospatial artificial intelligence and satellite-based earth observation cognitive system in response to covid-19. *Acta astronautica*, 197:323–335, 2022.

[2781] Boxing Zhang, Huan Lei, Yingjie Cai, Zhenyu Zhong, and Zeyu Jiao. COVID-19 contact tracking based on person reidentification and geospatial data. *J. King Saud Univ. Comput. Inf. Sci.*, 35(5):101558, 2023.

[2782] Zhenyu Tan, Xinghua Li, Meiling Gao, and Liangcun Jiang. The environmental story during the COVID-19 lockdown: How human activities affect PM2.5 concentration in china? *IEEE Geosci. Remote. Sens. Lett.*, 19:1–5, 2022.

[2783] James G Kahn, Joshua S Yang, and James S Kahn. 'mobile'health needs and opportunities in developing countries. *Health affairs*, 29(2):252–258, 2010.

[2784] Santosh Kumar, Gregory D. Abowd, William T. Abraham, Mustafa al'Absi, J. Gayle Beck, Duen Horng Chau, Tyson Condie, David E. Conroy, Emre Ertin, Deborah Estrin, Deepak Ganesan, Cho Lam, Benjamin M. Marlin, Clay B. Marsh, Susan A. Murphy, Inbal Nahum-Shani, Kevin Patrick, James M. Rehg, Moushumi Sharmin, Vivek Shetty, Ida Sim, Bonnie Spring, Mani B. Srivastava, and David W. Wetter. Center of excellence for mobile sensor data-to-knowledge (MD2K). *J. Am. Medical Informatics Assoc.*, 22(6):1137–1142, 2015.

[2785] Yu Zhan, Yuzhou Luo, Xunfei Deng, Huajin Chen, Michael L Grieneisen, Xueyou Shen, Lizhong Zhu, and Minghua Zhang. Spatiotemporal prediction of continuous daily pm2. 5 concentrations across china using a spatially explicit machine learning algorithm. *Atmospheric environment*, 155:129–139, 2017.

[2786] Qian Di, Itai Kloog, Petros Koutrakis, Alexei Lyapustin, Yujie Wang, and Joel Schwartz. Assessing pm2. 5 exposures with high spatiotemporal resolution across the continental united states. *Environmental science & technology*, 50(9):4712–4721, 2016.

[2787] Colorado J. Reed, Ritwik Gupta, Shufan Li, Sarah Brockman, Christopher Funk, Brian Clipp, Kurt Keutzer, Salvatore Candido, Matt Uyttendaele, and Trevor Darrell. Scale-MAE: A scale-aware masked autoencoder for multiscale geospatial representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[2788] Johannes Jakubik, Sujit Roy, C. E. Phillips, Paolo Fraccaro, Denys Godwin, Bianca Zadrozny, Daniela Szwarcman, Carlos Gomes, Gabby Nyirjesy, Blair Edwards, et al. Foundation models for generalist geospatial artificial intelligence. *arXiv preprint arXiv:2310.18660*, 2023.

[2789] Fan Liu, Delong Chen, Zhangqingyun Guan, Xiaocong Zhou, Jiale Zhu, Qiaolin Ye, Liyong Fu, and Jun Zhou. RemoteCLIP: A vision language foundation model for remote sensing. *IEEE Transactions on Geoscience and Remote Sensing*, 62, 2024.

[2790] Kartik Kuckreja, Muhammad Sohail Danish, Muzammal Naseer, Abhijit Das, Salman Khan, and Fahad Shahbaz Khan. GeoChat: Grounded large vision-language model for remote sensing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[2791] Kaiyu Li, Zepeng Xin, Li Pang, Chao Pang, Yupeng Deng, Jing Yao, Guisong Xia, Deyu Meng, Zhi Wang, and Xiangyong Cao. SegEarth-R1: Geospatial pixel reasoning via large language model. *arXiv preprint arXiv:2504.09644*, 2025.

[2792] Simone D'Amico, Mathias Benn, and John L Jørgensen. Pose estimation of an uncooperative spacecraft from actual space imagery. *International Journal of Space Science and Engineering 5*, 2(2):171–189, 2014.

[2793] Pedro F Proença and Yang Gao. Deep learning for spacecraft pose estimation from photorealistic rendering. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6007–6013, 2020.

[2794] Albert Garcia, Mohamed Adel Musallam, Vincent Gaudilliere, Enjie Ghorbel, Kassem Al Ismaeil, Marcos Perez, and Djamila Aouada. Lspnet: A 2d localization-oriented spacecraft pose estimation neural network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2048–2056, 2021.

[2795] Sumant Sharma and Simone D'Amico. Neural network-based pose estimation for noncooperative spacecraft rendezvous. *IEEE Transactions on Aerospace and Electronic Systems*, 56(6):4638–4658, 2020.

[2796] Thaweerath Phisannupawong, Patcharin Kamsing, Peerapong Torteeka, Sittiporn Channumsin, Utane Sawangwit, Warunyu Hematulin, Tanatthep Jarawan, Thanaporn Somjit, Soemsak Yooyen, Daniel Delahaye, et al. Vision-based spacecraft pose estimation via a deep convolutional neural network for noncooperative docking operations. *Aerospace*, 7(9):126, 2020.

[2797] Shubham Sonawani, Ryan Alimo, Renaud Detry, Daniel Jeong, Andrew Hess, and Heni Ben Amor. Assistive relative pose estimation for on-orbit assembly using convolutional neural networks. *arXiv preprint arXiv:2001.10673*, 2020.

[2798] Tae Ha Park and Simone D'Amico. Robust multi-task learning and online refinement for spacecraft pose estimation across domain gap. *Advances in Space Research*, 73(11):5726–5740, 2024.

[2799] Sumant Sharma, Connor Beierle, and Simone D'Amico. Pose estimation for non-cooperative spacecraft rendezvous using convolutional neural networks. In *2018 IEEE Aerospace Conference*, pages 1–12, 2018.

[2800] Zibin Liu, Banglei Guan, Yang Shang, Yifei Bian, Pengju Sun, and Qifeng Yu. Stereo event-based, 6-dof pose tracking for uncooperative spacecraft. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–13, 2025.

[2801] Tae Ha Park, Marcus Märtens, Gurvan Lecuyer, Dario Izzo, and Simone D'Amico. Speed+: Next-generation dataset for spacecraft pose estimation across domain gap. In *2022 IEEE aerospace conference (AERO)*, pages 1–15, 2022.

[2802] Mate Kisantal, Sumant Sharma, Tae Ha Park, Dario Izzo, Marcus Märtens, and Simone D'Amico. Satellite pose estimation challenge: Dataset, competition design, and results. *IEEE Transactions on Aerospace and Electronic Systems*, 56(5):4083–4098, 2020.

[2803] Mohamed Adel Musallam, Arunkumar Rathinam, Vincent Gaudillière, Miguel Ortiz del Castillo, and Djamila Aouada. Cubesat-cdt: a cross-domain dataset for 6-dof trajectory estimation of a symmetric spacecraft. In *European Conference on Computer Vision*, pages 112–126, 2022.

[2804] Mohamed Adel Musallam, Vincent Gaudilliere, Enjie Ghorbel, Kassem Al Ismaeil, Marcos Damian Perez, Michel Poucet, and Djamila Aouada. Spacecraft recognition leveraging knowledge of space environment: Simulator, dataset, competition design and analysis. In *2021 IEEE International Conference on Image Processing Challenges (ICIPC)*, pages 11–15, 2021.

[2805] Kevin Black, Shrivu Shankar, Daniel Fonseka, Jacob Deutsch, Abhimanyu Dhir, and Maruthi R Akella. Real-time, flight-ready, non-cooperative spacecraft pose estimation using monocular imagery. *arXiv preprint arXiv:2101.09553*, 2021.

[2806] Tae Ha Park, Sumant Sharma, and Simone D'Amico. Towards robust learning-based pose estimation of noncooperative spacecraft. *arXiv preprint arXiv:1909.00392*, 2019.

[2807] Yadong Shao, Aodi Wu, Shengyang Li, Leizheng Shu, Xue Wan, Yuanbin Shao, and Junyan Huo. Satellite component semantic segmentation: Video dataset and real-time pyramid attention and decoupled attention network. *IEEE Transactions on Aerospace and Electronic Systems*, 59(6):7315–7333, 2023.

[2808] Guangyuan Zhao, Xue Wan, Yaolin Tian, Yadong Shao, and Shengyang Li. 3d component segmentation network and dataset for non-cooperative spacecraft. *Aerospace*, 9(5):248, 2022.

[2809] Zhihao Zhang and Zhaohui Dang. Automatic segmentation annotation of space target using segment anything model and object detection prompts. *IEEE Transactions on Aerospace and Electronic Systems*, 2024.

[2810] Fan Li, Zhaoxiang Zhang, Xuan Wang, Xuanbin Wang, and Yuelei Xu. Exploiting diffusion priors for generalizable few-shot satellite image semantic segmentation. *Remote Sensing*, 17(22):3706, 2025.

[2811] Chaoyan Zhang, Baolong Guo, Nannan Liao, Qiuyun Zhong, Hengyan Liu, Cheng Li, and Jianglei Gong. Star-24k: A public dataset for space common target detection. *KSII Transactions on Internet & Information Systems*, 16(2), 2022.

[2812] LZ Li and T Zhang. Feature detection and recognition of spatial noncooperative objects based on deep learning. *CAAI Trans. Intell. Syst.*, 15(6):1154–1162, 2020.

[2813] Yi Cao, Jinzhen Mu, Xianghong Cheng, and Fengyu Liu. Spacecraft-ds: A spacecraft dataset for key components detection and segmentation via hardware-in-the-loop capture. *IEEE Sensors Journal*, 24 (4):5347–5358, 2024.

[2814] Yunpeng Zhao, Rui Zhong, and Linyan Cui. Intelligent recognition of spacecraft components from photorealistic images based on unreal engine 4. *Advances in Space Research*, 71(9):3761–3774, 2023.

[2815] Hoang Anh Dung, Bo Chen, and Tat-Jun Chin. A spacecraft dataset for detection, segmentation and parts recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2012–2019, 2021.

[2816] Yuxuan Liu, Chunjiang Bian, Hongbin Nie, Shi Chen, and Ziqian Yang. A large-scale synthetic benchmark dataset for non-cooperative space target perception. *Scientific Data*, 12(1):1780, 2025.

[2817] Douglas E Bernard, Gregory A Dorais, Chuck Fry, Edward B Gamble, Bob Kanefsky, James Kurien, William Millar, Nicola Muscettola, P Pandurang Nayak, Barney Pell, et al. Design of the remote agent experiment for spacecraft autonomy. In *1998 IEEE Aerospace Conference Proceedings (Cat. No. 98TH8339)*, volume 2, pages 259–281, 1998.

[2818] Barney Pell, Douglas E Bernard, Steve A Chien, Erann Gat, Nicola Muscettola, P Pandurang Nayak, Michael D Wagner, and Brian C Williams. An autonomous spacecraft agent prototype. In *Proceedings of the first international conference on autonomous agents*, pages 253–261, 1997.

[2819] Mitchell Colby, Logan Yliniemi, and Kagan Tumer. Autonomous multiagent space exploration with high-level human feedback. *Journal of Aerospace Information Systems*, 13(8):301–315, 2016.

[2820] Victor Rodriguez-Fernandez, Alejandro Carrasco, Jason Cheng, Eli Scharf, Peng Mun Siew, and Richard Linares. Language models are spacecraft operators. *arXiv preprint arXiv:2404.00413*, 2024.

[2821] Alejandro Carrasco, Marco Nedungadi, Victor Rodriguez-Fernandez, and Richard Linares. Visual language models as operator agents in the space domain. In *AIAA SCITECH 2025 Forum*, page 1543, 2025.

[2822] Enrico M Zucchelli, Di Wu, Julia Briden, Christian Hofmann, Victor Rodriguez-Fernandez, and Richard Linares. Fine-tuned language models as space systems controllers. *arXiv preprint arXiv:2501.16588*, 2025.

[2823] Alejandro Carrasco, Victor Rodriguez-Fernandez, and Richard Linares. Fine-tuning llms for autonomous spacecraft control: A case study using kerbal space program. *arXiv preprint arXiv:2408.08676*, 2024.

[2824] Matthew Foutter, Daniele Gammelli, Justin Kruger, Ethan Foss, Praneet Bhoj, Tommaso Guffanti, Simone D'Amico, and Marco Pavone. Space-llava: a vision-language model adapted to extraterrestrial applications. *arXiv preprint arXiv:2408.05924*, 2024.

[2825] Alejandro Carrasco, Victor Rodriguez-Fernandez, and Richard Linares. Large language models as autonomous spacecraft operators in kerbal space program. *Advances in Space Research*, 2025.

[2826] Ruben Belo, Cláudia Soares, and Marta Guimarães. Language modeling and retrieval-augmented generation for integration in space operation decision support tools. *58th IAA Symposium on Safety, Quality and Knowledge Management in Space Activities*, 2025.

[2827] Cheyenne Powell and Annalisa Riccardi. Question answering over knowledge graphs for explainable satellite scheduling. *Journal of Aerospace Information Systems*, pages 1–20, 2025.

[2828] Aodi Wu, Haodong Han, Xubo Luo, and Xue Wan. Spacemind: An mcp-based agent architecture fusing large and small models for on-orbit servicing. *Authorea Preprints*, 2025.

[2829] Ross E Allen, Yaron Rachlin, Jessica Ruprecht, Sean Loughran, Jacob Varey, and Herbert Viggh. Spacegym: Discrete and differential games in non-cooperative space operations. In *2023 IEEE Aerospace Conference*, pages 1–12, 2023.

[2830] Mark Maimone, Yang Cheng, and Larry Matthies. Two years of visual odometry on the Mars Exploration Rovers. *Journal of Field Robotics*, 24(3):169–186, 2007.

[2831] Olivier Toupet et al. Enhanced autonomous navigation on the perseverance mars rover. *IEEE Transactions on Field Robotics*, 2025. doi:10.1109/TFR.2025.3636366.

[2832] Masoud S Bahraini, Abdelhafid Zenati, and Nabil Aouf. Autonomous cooperative visual navigation for planetary exploration robots. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9653–9658, 2021.

[2833] Miguel Leiva Vélez, Victor Rodriguez-Fernandez, Hodei Urrutxua, and Nicolas Rozado. A hierarchical decision-making framework for autonomous and adaptive planetary robotics, 09 2025. AAS 25-768 (Preprint).

[2834] Kevin Nickels, Matthew DiCicco, Max Bajracharya, and Paul Backes. Vision guided manipulation for planetary robotics—position control. *Robotics and Autonomous systems*, 58(1):121–129, 2010.

[2835] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *International Conference on 3D Vision (3DV)*, pages 565–571, 2016.

[2836] Fabian Isensee, Paul F. Jaeger, Simon A. A. Kohl, Jens Petersen, and Klaus H. Maier-Hein. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18 (2):203–211, 2021.

[2837] Ali Hatamizadeh, Yucheng Tang, Vishwesh Nath, Dong Yang, Andriy Myronenko, Bennett Landman, Holger R. Roth, and Daguang Xu. UNETR: Transformers for 3d medical image segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 574–584, 2022.

[2838] Ali Hatamizadeh, Vishwesh Nath, Yucheng Tang, Dong Yang, Holger R. Roth, and Daguang Xu. Swin UNETR: Swin transformers for semantic segmentation of brain tumors in MRI images. In *International MICCAI Brainlesion Workshop*, pages 272–284, 2022.

[2839] Jakob Wasserthal, Hanns-Christian Breit, Manfred T. Meyer, Maurice Pradella, Daniel Hinck, Alexander W. Sauter, Tobias Heye, Daniel T. Boll, Joshy Cyriac, Shan Yang, Michael Bach, and Martin Segeroth. TotalSegmentator: Robust segmentation of 104 anatomic structures in CT images. *Radiology: Artificial Intelligence*, 5(5):e230024, 2023.

[2840] Abdel Aziz Taha and Allan Hanbury. Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool. *BMC Medical Imaging*, 15(1):29, 2015.

[2841] Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature Communications*, 15(1):654, 2024.

[2842] Haoyu Wang, Sizheng Guo, Jin Ye, Zhong ying Deng, Junlong Cheng, Tianbin Li, Jianpin Chen, Yanzhou Su, Ziyan Huang, Yiqing Shen, Bin Fu, Shaoting Zhang, Junjun He, and Yu Qiao. SAM-Med3D: Towards general-purpose segmentation models for volumetric medical images. *arXiv preprint arXiv:2310.15161*, 2023.

[2843] Yuxin Du, Fan Bai, Tiejun Huang, and Bo Zhao. SegVol: Universal and interactive volumetric medical image segmentation. In *Advances in Neural Information Processing Systems*, 2024.

[2844] M. Jorge Cardoso, Wenqi Li, Richard Brown, Nic Ma, Eric Kerfoot, Yiheng Wang, Benjamin Murrey, Andriy Myronenko, Can Zhao, Dong Yang, et al. MONAI: An open-source framework for deep learning in healthcare. *arXiv preprint arXiv:2211.02701*, 2022.

[2845] Samuel G. Armato III, Geoffrey McLennan, Luc Bidaut, Michael F. McNitt-Gray, Charles R. Meyer, Anthony P. Reeves, Binsheng Zhao, Denise R. Aberle, Claudia I. Henschke, Eric A. Hoffman, et al. The lung image database consortium (LIDC) and image database resource initiative (IDRI): a completed reference database of lung nodules on CT scans. *Medical Physics*, 38(2):915–931, 2011.

[2846] Arnaud Arindra Adiyoso Setio, Alberto Traverso, Thomas de Bel, Moira S. N. Berens, Cas van den Bogaard, Piergiorgio Cerello, Hao Chen, Qi Dou, Maria Evelina Fantacci, Bram Geurts, et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge. *Medical Image Analysis*, 42:1–13, 2017.

[2847] Florin C. Ghesu, Bogdan Georgescu, Tommaso Mansi, Dominik Neumann, Joachim Hornegger, and Dorin Comaniciu. An artificial agent for anatomical landmark detection in medical images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 229–237, 2016.

[2848] Guha Balakrishnan, Amy Zhao, Mert R. Sabuncu, John Guttag, and Adrian V. Dalca. VoxelMorph: A learning framework for deformable medical image registration. *IEEE Transactions on Medical Imaging*, 38(8):1788–1800, 2019.

[2849] Junyu Chen, Eric C. Frey, Yufan He, William P. Segars, Ye Li, and Yong Du. TransMorph: Transformer for unsupervised medical image registration. *Medical Image Analysis*, 82:102615, 2022.

[2850] Brian B. Avants, Charles L. Epstein, Murray Grossman, and James C. Gee. Symmetric diffeomorphic image registration with cross-correlation: evaluating automated labeling of elderly and neurodegenerative brain. *Medical Image Analysis*, 12(1):26–41, 2008.

[2851] Stefan Klein, Marius Staring, Keelin Murphy, Max A. Viergever, and Josien P. W. Pluim. elastix: a toolbox for intensity-based medical image registration. *IEEE Transactions on Medical Imaging*, 29(1): 196–205, 2010.

[2852] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A. Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M. Summers, et al. The medical segmentation decathlon. *Nature Communications*, 13(1):4128, 2022.

[2853] Bjoern H. Menze, Andras Jakab, Stefan Bauer, Jayashree Kalpathy-Cramer, Keyvan Farahani, Justin Kirby, Yuliya Burren, Nicole Porz, Johannes Slotboom, Roland Wiest, et al. The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024, 2015.

[2854] Yuanfeng Ji, Haotian Bai, Chongjian Ge, Jie Yang, Ye Zhu, Ruimao Zhang, Zhen Li, Lingyan Zhanng, Wanling Ma, Xiang Wan, and Ping Luo. AMOS: A large-scale abdominal multi-organ benchmark for versatile medical image segmentation. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2022.

[2855] Bennett Landman, Zhoubing Xu, J. Igelsias, Martin Styner, Thomas Langerak, and Arno Klein. MICCAI multi-atlas labeling beyond the cranial vault–workshop and challenge. In *Proc. MICCAI Multi-Atlas Labeling Beyond Cranial Vault Workshop Challenge*, 2015.

[2856] Nicholas Heller, Fabian Isensee, Klaus H. Maier-Hein, Xiaoshuai Hou, Chunmei Xie, Fengyi Li, Yang Nan, Guangrui Mu, Zhiyong Lin, Miofei Han, et al. The state of the art in kidney and kidney tumor segmentation in contrast-enhanced CT imaging: Results of the KiTS19 challenge. *Medical Image Analysis*, 67:101821, 2021.

[2857] Olivier Bernard, Alain Lalande, Clement Zotti, Frederick Cervenansky, Xin Yang, Pheng-Ann Heng, Irem Cetin, Karim Lekadir, Oscar Camara, Miguel Angel Gonzalez Ballester, et al. Deep learning techniques for automatic MRI cardiac multi-structures segmentation and diagnosis: is the problem solved? *IEEE Transactions on Medical Imaging*, 37(11):2514–2525, 2018.

[2858] Tobias Heimann and Hans-Peter Meinzer. Statistical shape models for 3d medical image segmentation: a review. *Medical Image Analysis*, 13(4):543–563, 2009.

[2859] Udaranga Wickramasinghe, Edoardo Remelli, Graham Knott, and Pascal Fua. Voxel2Mesh: 3d mesh model generation from volumetric data. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 299–308, 2020.

[2860] Fanwei Kong, Nathan Wilson, and Shawn Shadden. A deep-learning approach for direct whole-heart mesh reconstruction. *Medical Image Analysis*, 74:102222, 2021.

[2861] Ruyi Zha, Yanhao Zhang, and Hongdong Li. NAF: Neural attenuation fields for sparse-view CBCT reconstruction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 442–452, 2022.

[2862] Steven A. Niederer, Joost Lumens, and Natalia A. Trayanova. Computational models in cardiology. *Nature Reviews Cardiology*, 16(2):100–111, 2019.

[2863] Adityo Prakosa, Hermenegild J. Arevalo, Dongdong Deng, Patrick M. Boyle, Plamen P. Nikolov, Hiroshi Ashikaga, Joshua J. E. Blauer, Elyar Ghafoori, Carolyn J. Park, Robert C. Blake, et al. Personalized virtual-heart technology for guiding the ablation of infarct-related ventricular tachycardia. *Nature Biomedical Engineering*, 2(10):732–740, 2018.

[2864] Charles A. Taylor, Timothy A. Fonte, and James K. Min. Computational fluid dynamics applied to cardiac computed tomography for noninvasive quantification of fractional flow reserve: scientific basis. *Journal of the American College of Cardiology*, 61(22):2233–2241, 2013.

[2865] Huifang E. Wang, Marmaduke Woodman, Paul Triebkorn, Jean-Didier Lemarechal, Jayant Jha, Borana Dollomaja, Anirudh Nihalani Vattikonda, Viktor Sip, Samuel Medina Villalon, Meysam Hashemi, et al. Virtual brain twins from MRI for personalized epilepsy surgery. *Science Translational Medicine*, 15(680):eabp8982, 2023.

[2866] Marco Viceconti, Adriano Henney, and Edwin Morley-Fletcher. In silico clinical trials: how computer simulation will transform the biomedical industry. *International Journal of Clinical Trials*, 3(2):37–46, 2016.

[2867] Aldo Badano, Christian G. Graff, Andreu Badal, Diksha Sharma, Rongping Zeng, Frank W. Samuelson, Stephen J. Glick, and Kyle J. Myers. Evaluation of digital breast tomosynthesis as replacement of full-field digital mammography using an in silico imaging trial. *JAMA Network Open*, 1(7):e185474, 2018.

[2868] Andriy Fedorov, Reinhard Beichel, Jayashree Kalpathy-Cramer, Julien Finet, Jean-Christophe Fillion-Robin, Sonia Pujol, Christian Bauer, Dominique Jennings, Fiona Fennessy, Milan Sonka, et al. 3D Slicer as an image computing platform for the quantitative imaging network. *Magnetic Resonance Imaging*, 30(9):1323–1341, 2012.

[2869] François Faure, Christian Duriez, Hervé Delingette, Jérémie Allard, Benjamin Gilles, Stéphanie Marchesseau, Hugo Talbot, Hadrien Courtecuisse, Guillaume Bousquet, Igor Peterlik, and Stéphane Cotin. SOFA: A multi-model framework for interactive physical simulation. In *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, pages 283–321. 2012.

[2870] Aaron Babier, Binghao Zhang, Rafid Mahmood, Kevin L. Moore, Thomas G. Purdie, Andrea L. McNiven, and Timothy C. Y. Chan. OpenKBP: the open-access knowledge-based planning grand challenge and dataset. *Medical Physics*, 48(9):5549–5561, 2021.

[2871] Camilo A Molina, Frank M Phillips, Matthew W Colman, Wilson Z Ray, Majid Khan, Kornelis Poelstra, Larry Khoo, et al. A cadaveric precision and accuracy analysis of augmented reality–mediated percutaneous pedicle implant insertion: presented at the 2020 aans/cns joint section on disorders of the spine and peripheral nerves. *Journal of Neurosurgery: Spine*, 34(2):316–324, 2020.

[2872] Andru P. Twinanda, Sherif Shehata, Didier Mutter, Jacques Marescaux, Michel De Mathelin, and Nicolas Padoy. EndoNet: a deep architecture for recognition tasks on laparoscopic videos. *IEEE Transactions on Medical Imaging*, 36(1):86–97, 2017.

[2873] Max Allan, Jonathan Mcleod, Congcong Wang, Jean Claude Rosenthal, Zhenglei Hu, Niklas Gard, Peter Eisert, Ke Xue Fu, Trevor Zeffiro, Wenyao Xia, et al. Stereo correspondence and reconstruction of endoscopic data challenge. *arXiv preprint arXiv:2101.01133*, 2021.

[2874] Kutsev Bengisu Ozyoruk, Guliz Irem Gokceler, Taylor L. Bobrow, Gulfize Coskun, Kagan Incetan, Yasin Almalioglu, Faisal Mahmood, Eva Curto, Luis Perdigoto, Marina Oliveira, et al. EndoSLAM dataset and an unsupervised monocular visual odometry and depth estimation approach for endoscopic videos. *Medical Image Analysis*, 71:102058, 2021.

[2875] Yuehao Wang, Yonghao Long, Siu Hin Fan, and Qi Dou. Neural rendering for stereo 3D reconstruction of deformable tissues in robotic surgery. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 431–441, 2022.

[2876] Peter Kazanzides, Zihan Chen, Anton Deguet, Gregory S. Fischer, Russell H. Taylor, and Simon P. DiMaio. An open-source research kit for the da Vinci surgical system. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6434–6439, 2014.

[2877] Guang-Zhong Yang, James Cambias, Kevin Cleary, Eric Daimler, James Drake, Pierre E. Dupont, Nobuhiko Hata, Peter Kazanzides, Sylvain Martel, Rajni V. Patel, et al. Medical robotics–regulatory, ethical, and legal considerations for increasing levels of autonomy. *Science Robotics*, 2(4):eaam8638, 2017.

[2878] Azad Shademan, Ryan S. Decker, Justin D. Opfermann, Simon Leonard, Axel Krieger, and Peter C. W. Kim. Supervised autonomous robotic soft tissue surgery. *Science Translational Medicine*, 8(337): 337ra64, 2016.

[2879] Hamed Saeidi, Justin D. Opfermann, Michael Kam, Shuwen Wei, Simon Leonard, Michael H. Hsieh, Jin U. Kang, and Axel Krieger. Autonomous robotic laparoscopic surgery for intestinal anastomosis. *Science Robotics*, 7(62):eabj2908, 2022.

[2880] Xiahai Zhuang, Lei Li, Christian Payer, Darko Stern, Martin Urschler, Mattias P. Heinrich, Julien Oster, Chunliang Wang, Orjan Smedby, Cheng Bian, et al. Evaluation of algorithms for multi-modality whole heart segmentation: an open-access grand challenge. *Medical Image Analysis*, 58:101537, 2019.

[2881] Shih-Cheng Huang, Anuj Pareek, Malte Jensen, Matthew P. Lungren, Serena Yeung, and Akshay S. Chaudhari. Self-supervised learning for medical image classification: a systematic review and implementation guidelines. *npj Digital Medicine*, 6(1):74, 2023.

[2882] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R. Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N. Galtier, Bennett A. Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *npj Digital Medicine*, 3(1):119, 2020.

[2883] Julian N. Acosta, Guido J. Falcone, Pranav Rajpurkar, and Eric J. Topol. Multimodal biomedical AI. *Nature Medicine*, 28(9):1773–1784, 2022.

[2884] Vincent Sitzmann, Julien N. P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit Neural Representations with Periodic Activation Functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[2885] Xudong Chen, Zhun Wei, Maokun Li, and Paolo Rocca. A Review of Deep Learning Approaches for Inverse Scattering Problems (Invited Review). *Progress In Electromagnetics Research*, 167:67–81, 2020.

[2886] Zhun Wei and Xudong Chen. Deep-Learning Schemes for Full-Wave Nonlinear Inverse Scattering Problems. *IEEE Transactions on Geoscience and Remote Sensing*, 57(4):1849–1860, 2019.

[2887] Yuehaw Khoo and Lexing Ying. SwitchNet: a neural network model for forward and inverse scattering problems. *SIAM Journal on Scientific Computing*, 41(5):A3182–A3201, 2019.

[2888] Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian Neural Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[2889] Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho. Lagrangian Neural Networks. *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2020.

[2890] Miles Cranmer, Alvaro Sanchez Gonzalez, Peter Battaglia, Rui Xu, Kyle Cranmer, David Spergel, and Shirley Ho. Discovering symbolic models from deep learning with inductive biases. *Advances in neural information processing systems*, 33:17429–17442, 2020.

[2891] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.

[2892] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier Neural Operator for Parametric Partial Differential Equations. In *International Conference on Learning Representations (ICLR)*, 2021.

[2893] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021.

[2894] Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to Simulate Complex Physics with Graph Networks. In *International Conference on Machine Learning (ICML)*, pages 8459–8468, 2020.

[2895] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W. Battaglia. Learning Mesh-Based Simulation with Graph Networks. In *International Conference on Learning Representations (ICLR)*, 2021.

[2896] Arvind T. Mohan, Nicholas Lubbers, Daniel Livescu, and Michael Chertkov. Spatio-temporal deep learning models of 3D turbulence with physics informed diagnostics. *Journal of Turbulence*, 21(9-10): 484–524, 2020.

[2897] Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1):4950, 2018.

[2898] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[2899] Yuanming Hu, Jiancheng Liu, Andrew Spielberg, Joshua B. Tenenbaum, William T. Freeman, Jiajun Wu, Daniela Rus, and Wojciech Matusik. ChainQueen: A Real-Time Differentiable Physical Simulator for Soft Robotics. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6265–6271, 2019.

[2900] Jörg Behler and Michele Parrinello. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters*, 98(14):146401, 2007.

[2901] Kristof T. Schütt, Pieter-Jan Kindermans, Huziel E. Sauceda, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[2902] Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional Message Passing for Molecular Graphs. In *International Conference on Learning Representations (ICLR)*, 2020.

[2903] Kristof T. Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning (ICML)*, pages 9377–9388, 2021.

[2904] Linfeng Zhang, Jiequn Han, Han Wang, Roberto Car, and Weinan E. Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics. *Physical Review Letters*, 120 (14):143001, 2018.

[2905] Justin S. Smith, Olexandr Isayev, and Adrian E. Roitberg. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chemical Science*, 8(4):3192–3203, 2017.

[2906] Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J. Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics. *Nature Communications*, 14(1):579, 2023.

[2907] Chi Chen and Shyue Ping Ong. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science*, 2(11):718–728, 2022.

[2908] Bowen Deng, Peichen Zhong, KyuJung Jun, Janosh Riebesell, Kevin Han, Christopher J. Bartel, and Gerbrand Ceder. CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence*, 5(9):1031–1041, 2023.

[2909] Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, 2017.

[2910] David Pfau, James S. Spencer, Alexander G. D. G. Matthews, and W. M. C. Foulkes. Ab initio solution of the many-electron Schrödinger equation with deep neural networks. *Physical Review Research*, 2(3): 033429, 2020.

[2911] Jan Hermann, Zeno Schätzle, and Frank Noé. Deep-neural-network solution of the electronic Schrödinger equation. *Nature Chemistry*, 12(10):891–897, 2020.

[2912] Stefan Chmiela, Alexandre Tkatchenko, Huziel E. Sauceda, Igor Poltavsky, Kristof T. Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5):e1603015, 2017.

[2913] Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ulissi, C Lawrence Zitnick, and Abhishek Das. Gemnet-oc: developing graph neural networks for large and diverse molecular simulation datasets. *arXiv preprint arXiv:2204.02782*, 2022.

[2914] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin A. Persson. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013.

[2915] Lowik Chanussot, Abhishek Das, Siddharth Goyal, Thibaut Lavril, Muhammed Shuaibi, Morgane Riviere, Kevin Tran, Javier Heras-Domingo, Caleb Ho, Weihua Hu, Aini Palizhati, Anuroop Sriram, Brandon Wood, Junwoong Yoon, Devi Parikh, C. Lawrence Zitnick, and Zachary Ulissi. Open Catalyst 2020 (OC20) Dataset and Community Challenges. *ACS Catalysis*, 11(10):6059–6072, 2021.

[2916] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018.

[2917] Niklas W. A. Gebauer, Michael Gastegger, and Kristof T. Schütt. Symmetry-adapted generation of 3D point sets for the targeted discovery of molecules. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[2918] Emiel Hoogeboom, Víctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant Diffusion for Molecule Generation in 3D. In *International Conference on Machine Learning (ICML)*, pages 8867–8887, 2022.

[2919] Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. GeoDiff: A Geometric Diffusion Model for Molecular Conformation Generation. In *International Conference on Learning Representations (ICLR)*, 2022.

[2920] Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional Diffusion for Molecular Conformer Generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[2921] Simon Axelrod and Rafael Gómez-Bombarelli. GEOM, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 9(1):185, 2022.

[2922] Tian Xie, Xiang Fu, Octavian-Eugen Ganea, Regina Barzilay, and Tommi Jaakkola. Crystal Diffusion Variational Autoencoder for Periodic Material Generation. In *International Conference on Learning Representations (ICLR)*, 2022.

[2923] Rui Jiao, Wenbing Huang, Peijia Lin, Jiaqi Han, Pin Chen, Yutong Lu, and Yang Liu. Crystal structure prediction by joint equivariant diffusion. *Advances in Neural Information Processing Systems*, 36:17464–17497, 2023.

[2924] Claudio Zeni, Robert Pinsler, Daniel Zügner, Andrew Fowler, Matthew Horton, Xiang Fu, Zilong Wang, Aliaksandra Shysheya, Jonathan Crabbé, Shoko Ueda, et al. A generative model for inorganic materials design. *Nature*, 639(8055):624–632, 2025.

[2925] Andrew W. Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander W. R. Nelson, Alex Bridgland, et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710, 2020.

[2926] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.

[2927] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630(8016):493–500, 2024.

[2928] Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N. Kinch, R. Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

[2929] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[2930] Sjors H. W. Scheres. RELION: implementation of a Bayesian approach to cryo-EM structure determination. *Journal of Structural Biology*, 180(3):519–530, 2012.

[2931] Ali Punjani, John L. Rubinstein, David J. Fleet, and Marcus A. Brubaker. cryoSPARC: algorithms for rapid unsupervised cryo-EM structure determination. *Nature Methods*, 14(3):290–296, 2017.

[2932] Ellen D. Zhong, Tristan Bepler, Bonnie Berger, and Joseph H. Davis. CryoDRGN: reconstruction of heterogeneous cryo-EM structures using neural networks. *Nature Methods*, 18(2):176–185, 2021.

[2933] Patrik L. Ståhl, Fredrik Salmén, Sanja Vickovic, Anna Lundmark, José Fernández Navarro, Jens Magnusson, Stefania Giacomello, Michaela Asp, Jakub O. Westholm, Mikael Huss, et al. Visualization and analysis of gene expression in tissue sections by spatial transcriptomics. *Science*, 353(6294):78–82, 2016.

[2934] Tommaso Biancalani, Gabriele Scalia, Lorenzo Buffoni, Raghav Avasthi, Ziqing Lu, Aman Sanger, Neriman Tokcan, Charles R. Vanderburg, Åsa Segerstolpe, Meng Zhang, et al. Deep learning and alignment of spatially resolved single-cell transcriptomes with Tangram. *Nature Methods*, 18(11): 1352–1362, 2021.

[2935] Mor Nitzan, Nikos Karaiskos, Nir Friedman, and Nikolaus Rajewsky. Gene expression cartography. *Nature*, 576(7785):132–137, 2019.

[2936] Michał Januszewski, Jörgen Kornfeld, Peter H. Li, Art Pope, Tim Blakely, Larry Lindsey, Jeremy Maitin-Shepard, Mike Tyka, Winfried Denk, and Viren Jain. High-precision automated reconstruction of neurons with flood-filling networks. *Nature Methods*, 15(8):605–610, 2018.

[2937] Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. DiffDock: Diffusion Steps, Twists, and Turns for Molecular Docking. In *International Conference on Learning Representations (ICLR)*, 2023.

[2938] Hannes Stärk, Octavian Ganea, Lagnajit Pattanaik, Regina Barzilay, and Tommi Jaakkola. EquiBind: Geometric Deep Learning for Drug Binding Structure Prediction. In *International Conference on Machine Learning (ICML)*, pages 20503–20521, 2022.

[2939] Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, et al. Protein complex prediction with AlphaFold-Multimer. *bioRxiv*, pages 2021–10, 2022.

[2940] Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, et al. De novo design of protein structure and function with RFdiffusion. *Nature*, 620(7976):1089–1100, 2023.

[2941] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Alexis Courbet, Rob J. de Haas, Neville Bethel, et al. Robust deep learning-based protein sequence design using ProteinMPNN. *Science*, 378(6615):49–56, 2022.

[2942] John B. Ingraham, Max Baranov, Zak Costello, Karl W. Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M. Lord, Christopher Ng-Thow-Hing, Erik R. Van Vlack, et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

[2943] Andriy Kryshtafovych, Torsten Schwede, Maya Topf, Krzysztof Fidelis, and John Moult. Critical assessment of methods of protein structure prediction (CASP)—Round XIV. *Proteins: Structure, Function, and Bioinformatics*, 89(12):1607–1617, 2021.

[2944] Renxiao Wang, Xueliang Fang, Yipin Lu, and Shaomeng Wang. The PDBbind database: collection of binding affinities for protein-ligand complexes with known three-dimensional structures. *Journal of Medicinal Chemistry*, 47(12):2977–2980, 2004.

[2945] Dongping Li, Tielong Cai, Tianci Tang, Wenhao Chai, Katherine Rose Driggs-Campbell, and Gaoang Wang. EMMOE: A comprehensive benchmark for embodied mobile manipulation in open environments. *arXiv preprint arXiv:2503.08604*, 2025.

[2946] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, et al. ConceptFusion: Open-set Multimodal 3D Mapping. In *Robotics: Science and Systems (RSS)*, 2023.

[2947] Dhruv Batra, Angel X. Chang, Sonia Chernova, Andrew J. Davison, Jia Deng, Vladlen Koltun, Sergey Levine, Jitendra Malik, Igor Mordatch, Roozbeh Mottaghi, Manolis Savva, and Hao Su. Rearrangement: A Challenge for Embodied AI. *arXiv preprint arXiv:2011.01975*, 2020.

[2948] Luca Weihs, Matt Deitke, Aniruddha Kembhavi, and Roozbeh Mottaghi. Visual Room Rearrangement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5922–5931, 2021.

[2949] Yash Kant, Arun Ramachandran, Sriram Yenamandra, Igor Gilitschenski, Dhruv Batra, Andrew Szot, and Harsh Agrawal. Housekeep: Tidying Virtual Households Using Commonsense Reasoning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 355–373, 2022.

[2950] Gabriel Sarch, Zhaoyuan Fang, Adam W. Harley, Paul Schydlo, Michael J. Tarr, Saurabh Gupta, and Katerina Fragkiadaki. TIDEE: Tidying Up Novel Rooms Using Visuo-Semantic Commonsense Priors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 480–496, 2022.

[2951] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. TidyBot: Personalized Robot Assistance with Large Language Models. *Autonomous Robots*, 47(8):1087–1102, 2023.

[2952] Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander William Clegg, John Turner, et al. HomeRobot: Open-Vocabulary Mobile Manipulation. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2023.

[2953] Peiqi Liu, Yaswanth Orru, Jay Vakil, Chris Paxton, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. OK-Robot: What Really Matters in Integrating Open-Knowledge Models for Robotics. In *Robotics: Science and Systems (RSS)*, 2024.

[2954] Zipeng Fu, Tony Z. Zhao, and Chelsea Finn. Mobile ALOHA: Learning Bimanual Mobile Manipulation with Low-Cost Whole-Body Teleoperation. In *Proceedings of the Conference on Robot Learning (CoRL)*, 2024.

[2955] Dirk Helbing and Péter Molnár. Social Force Model for Pedestrian Dynamics. *Physical Review E*, 51(5): 4282–4286, 1995.

[2956] Changan Chen, Yuejiang Liu, Sven Kreiss, and Alexandre Alahi. Crowd-Robot Interaction: Crowd-Aware Robot Navigation with Attention-Based Deep Reinforcement Learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6015–6022, 2019.

[2957] Peter Trautman and Andreas Krause. Unfreezing the Robot: Navigation in Dense, Interacting Crowds. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 797–803, 2010.

[2958] Haresh Karnan, Anirudh Nair, Xuesu Xiao, Garrett Warnell, Sören Pirk, Alexander Toshev, Justin Hart, Joydeep Biswas, and Peter Stone. Socially Compliant Navigation Dataset (SCAND): A Large-Scale Dataset of Demonstrations for Social Navigation. *IEEE Robotics and Automation Letters*, 7(4): 11807–11814, 2022.

[2959] Shervin Javdani, Henny Admoni, Stefania Pellegrinelli, Siddhartha S. Srinivasa, and J. Andrew Bagnell. Shared Autonomy via Hindsight Optimization for Teleoperation and Teaming. *The International Journal of Robotics Research*, 37(7):717–742, 2018.

[2960] Anca D. Dragan and Siddhartha S. Srinivasa. A Policy-Blending Formalism for Shared Control. *The International Journal of Robotics Research*, 32(7):790–805, 2013.

[2961] Kyle Strabala, Min Kyung Lee, Anca Dragan, Jodi Forlizzi, Siddhartha S. Srinivasa, Maya Cakmak, and Vincenzo Micelli. Toward Seamless Human-Robot Handovers. *Journal of Human-Robot Interaction*, 2(1):112–132, 2013.

[2962] Hsuan-Kung Yang, Tsu-Ching Hsiao, Ryoichiro Oka, Ryuya Nishino, Satoko Tofukuji, and Norimasa Kobori. An embodied AR navigation agent: Integrating BIM with retrieval-augmented generation for language guidance. *arXiv preprint arXiv:2508.16602*, 2025.

[2963] Peter B. L. Meijer. An Experimental System for Auditory Image Representations. *IEEE Transactions on Biomedical Engineering*, 39(2):112–121, 1992.

[2964] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom Yeh. VizWiz: Nearly Real-Time Answers to Visual Questions. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology (UIST)*, pages 333–342, 2010.

[2965] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617, 2018.

[2966] Dragan Ahmetovic, Cole Gleason, Chengxiong Ruan, Kris Kitani, Hironobu Takagi, and Chieko Asakawa. NavCog: A Navigational Cognitive Assistant for the Blind. In *Proceedings of the 18th*

International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI), pages 90–99, 2016.

[2967] Diane J. Cook, Aaron S. Crandall, Brian L. Thomas, and Narayanan C. Krishnan. CASAS: A Smart Home in a Box. *Computer*, 46(7):62–69, 2013.

[2968] Alan K. Bourke, John V. O'Brien, and Gerard M. Lyons. Evaluation of a Threshold-Based Tri-Axial Accelerometer Fall Detection Algorithm. *Gait & Posture*, 26(2):194–199, 2007.

[2969] Richard C. Simpson. Smart Wheelchairs: A Literature Review. *Journal of Rehabilitation Research and Development*, 42(4):423–436, 2005.

[2970] Mahmoud Ghorbel, Joelle Pineau, Richard Gourdeau, Shervin Javdani, and Siddhartha Srinivasa. A Decision-Theoretic Approach for the Collaborative Control of a Smart Wheelchair. *International Journal of Social Robotics*, 10(1):131–145, 2018.

[2971] Tapomayukh Bhattacharjee, Gilwoo Lee, Hanjun Song, and Siddhartha S. Srinivasa. Towards Robotic Feeding: Role of Haptics in Fork-Based Food Manipulation. *IEEE Robotics and Automation Letters*, 4(2): 1485–1492, 2019.

[2972] Tiffany L. Chen, Matei Ciocarlie, Steve Cousins, Phillip M. Grice, Kelsey Hawkins, Kaijen Hsiao, Charles C. Kemp, Chih-Hung King, Daniel A. Lazewatsky, Adam E. Leeper, Hai Nguyen, Andreas Paepcke, Caroline Pantofaru, William D. Smart, and Leila Takayama. Robots for Humanity: Using Assistive Robotics to Empower People with Disabilities. *IEEE Robotics & Automation Magazine*, 20(1): 30–39, 2013.

[2973] Charles C. Kemp, Aaron Edsinger, Henry M. Clever, and Blaine Matulevich. The Design of Stretch: A Compact, Lightweight Mobile Manipulator for Indoor Human Environments. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3150–3157, 2022.

[2974] Zackory Erickson, Vamsee Gangaram, Ariel Kapusta, C. Karen Liu, and Charles C. Kemp. Assistive Gym: A Physics Simulation Framework for Assistive Robotics. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 10169–10176, 2020.

[2975] Xue Qin, Simin Luan, John See, Cong Yang, and Zhijun Li. Embodiedgovbench: A benchmark for governance, recovery, and upgrade safety in embodied agent systems. *arXiv preprint arXiv:2604.11174*, 2026.

[2976] Yuanwen Yue, Theodora Kontogianni, Konrad Schindler, and Francis Engelmann. Connecting the Dots: Floorplan Reconstruction Using Two-Level Queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 845–854, 2023.

[2977] Mani Golparvar-Fard, Feniosky Peña-Mora, and Silvio Savarese. Automated Progress Monitoring Using Unordered Daily Construction Photographs and IFC-Based Building Information Models. *Journal of Computing in Civil Engineering*, 29(1):04014025, 2015.

[2978] Markus Giftthaler, Timothy Sandy, Kathrin Dörfler, Ian Brooks, Mark Buckingham, Gonzalo Rey, Matthias Kohler, Fabio Gramazio, and Jonas Buchli. Mobile Robotic Fabrication at 1:1 Scale: The In Situ Fabricator. *Construction Robotics*, 1(1):3–14, 2017.

[2979] Young-Jin Cha, Wooram Choi, and Oral Büyüköztürk. Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, 32(5): 361–378, 2017.

[2980] Anthony Cioppa, Silvio Giancola, Adrien Deliège, Le Kang, Xin Zhou, Zhiyu Cheng, Bernard Ghanem, and Marc Van Droogenbroeck. SoccerNet-Tracking: Multiple Object Tracking Dataset and Benchmark in Soccer Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3491–3502, 2022.

[2981] Yutao Cui, Chenkai Zeng, Xiaoyu Zhao, Yichun Yang, Gangshan Wu, and Limin Wang. SportsMOT: A Large Multi-Object Tracking Dataset in Multiple Sports Scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9921–9931, 2023.

[2982] Wei Jiang, Juan Camilo Gamboa Higuera, Baptiste Angles, Weiwei Sun, Mehrsan Javan, and Kwang Moo Yi. Optimizing Through Learned Errors for Accurate Sports Field Registration. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 201–210, 2020.

[2983] Scott D. Uhlrich, Antoine Falisse, Łukasz Kidziński, Julie Muccini, Michael Ko, Akshay S. Chaudhari, Jennifer L. Hicks, and Scott L. Delp. OpenCap: Human Movement Dynamics from Smartphone Videos. *PLOS Computational Biology*, 19(10):e1011462, 2023.

[2984] Javier Fernández, Luke Bornn, and Dan Cervone. Decomposing the Immeasurable Sport: A Deep Learning Expected Possession Value Framework for Soccer. In *MIT Sloan Sports Analytics Conference*, 2019.

[2985] Raymond A. Yeh, Alexander G. Schwing, Jonathan Huang, and Kevin Murphy. Diverse Generation for Multi-Agent Sports Games. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4610–4619, 2019.

[2986] Marc Levoy, Kari Pulli, Brian Curless, Szymon Rusinkiewicz, David Koller, Lucas Pereira, Matt Ginzton, Sean Anderson, James Davis, Jeremy Ginsberg, Jonathan Shade, and Duane Fulk. The Digital Michelangelo Project: 3D Scanning of Large Statues. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 131–144, 2000.

[2987] Silvia Sellán, Yun-Chun Chen, Ziyi Wu, Animesh Garg, and Alec Jacobson. Breaking Bad: A Dataset for Geometric Fracture and Reassembly. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2022.

[2988] Theodore Tsesmelis, Luca Palmieri, Marina Khoroshiltseva, Adeela Islam, Gur Elkin Othman, Ofir Itzhak Zhang, Gianluca Scarpellini, Stefano Fiorini, Yaniv Aslan, Luca Moretti, et al. Reassembling the Past: The RePAIR Dataset and Benchmark for Real World 2D and 3D Puzzle Solving. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024.

[2989] Arlen F. Chase, Diane Z. Chase, John F. Weishampel, Jason B. Drake, Ramesh L. Shrestha, K. Clint Slatton, Jaime J. Awe, and William E. Carter. Airborne LiDAR, Archaeology, and the Ancient Maya Landscape at Caracol, Belize. *Journal of Archaeological Science*, 38(2):387–398, 2011.

[2990] Marcello A. Canuto, Francisco Estrada-Belli, Thomas G. Garrison, Stephen D. Houston, Mary Jane Acuña, Milan Kováč, Damien Marken, Philippe Nondedeo, Luke Auld-Thomas, Cyril Castanet, et al. Ancient Lowland Maya Complexity as Revealed by Airborne Laser Scanning of Northern Guatemala. *Science*, 361(6409):eaau0137, 2018.

[2991] Liam Paull, Sajad Saeedi, Mae Seto, and Howard Li. AUV Navigation and Localization: A Review. *IEEE Journal of Oceanic Engineering*, 39(1):131–149, 2014.

[2992] Ryan Eustice, Hanumant Singh, John Leonard, Matthew Walter, and Robert Ballard. Visually Navigating the RMS Titanic with SLAM Information Filters. In *Robotics: Science and Systems (RSS)*, 2005.

[2993] Maxime Ferrera, Vincent Creuze, Julien Moras, and Pauline Trouvé-Peloux. AQUALOC: An Underwater Dataset for Visual-Inertial-Pressure Localization. *The International Journal of Robotics Research*, 38(14):1549–1559, 2019.

[2994] Derya Akkaynak and Tali Treibitz. Sea-Thru: A Method for Removing Water from Underwater Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1682–1691, 2019.

[2995] Md Jahidul Islam, Youya Xia, and Junaed Sattar. Fast Underwater Image Enhancement for Improved Visual Perception. *IEEE Robotics and Automation Letters*, 5(2):3227–3234, 2020.

[2996] Chongyi Li, Chunle Guo, Wenqi Ren, Runmin Cong, Junhui Hou, Sam Kwong, and Dacheng Tao. An Underwater Image Enhancement Benchmark Dataset and Beyond. *IEEE Transactions on Image Processing*, 29:4376–4389, 2020.

[2997] Kakani Katija, Eric Orenstein, Brian Schlining, Lonny Lundsten, Kevin Barnard, Giovanna Sainz, Oceane Boulais, Megan Cromwell, Erin Butler, Benjamin Woodward, and Katherine L. C. Bell. FathomNet: A Global Image Database for Enabling Artificial Intelligence in the Ocean. *Scientific Reports*, 12(1):15914, 2022.

[2998] Franz S. Hover, Ryan M. Eustice, Ayoung Kim, Brendan Englot, Hordur Johannsson, Michael Kaess, and John J. Leonard. Advanced Perception, Navigation and Planning for Autonomous In-Water Ship Hull Inspection. *The International Journal of Robotics Research*, 31(12):1445–1464, 2012.

[2999] Tatjana Chavdarova, Pierre Baqué, Stéphane Bouquet, Andrii Maksai, Cijo Jose, Timur Bagautdinov, Louis Lettry, Pascal Fua, Luc Van Gool, and François Fleuret. WILDTRACK: A Multi-Camera HD Dataset for Dense Unscripted Pedestrian Detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5030–5039, 2018.

[3000] Xiaotian Han, Quanzeng You, Chunyu Wang, Zhizheng Zhang, Peng Chu, Houdong Hu, Jiang Wang, and Zicheng Liu. MMPTRACK: Large-Scale Densely Annotated Multi-Camera Multiple People Tracking Benchmark. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 4860–4869, 2023.

[3001] Yunzhong Hou, Liang Zheng, and Stephen Gould. Multiview Detection with Feature Perspective Transformation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 1–18, 2020.

[3002] Eran Goldman, Roei Herzig, Aviv Eisenschtat, Jacob Goldberger, and Tal Hassner. Precise Detection in Densely Packed Scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5227–5236, 2019.

[3003] Alessio Tonioni and Luigi Di Stefano. Product Recognition in Store Shelves as a Sub-Graph Isomorphism Problem. In *International Conference on Image Analysis and Processing (ICIAP)*, pages 682–693, 2017.

[3004] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1116–1124, 2015.

[3005] Zhongbo Zhang, Jiayi Jin, Yifan Wang, Zaibin Zhang, Haiwen Diao, Lijun Wang, and Huchuan Lu. VABench: Measuring embodied spatial intelligence through visual demonstrations, active perception, and metric control. *arXiv preprint arXiv:2609.19554*, 2026.

[3006] Wes Gurnee and Max Tegmark. Language models represent space and time. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=jE8xbmvFin.

[3007] Zhongang Cai, Yubo Wang, Qingping Sun, Ruisi Wang, Chenyang Gu, Wanqi Yin, Zhiqian Lin, Zhitao Yang, Chen Wei, Oscar Qian, Hui En Pang, Xuanke Shi, Kewang Deng, Xiaoyang Han, Zukai Chen, Jiaqi Li, Xiangyu Fan, Hanming Deng, Lewei Lu, Bo Li, Ziwei Liu, Quan Wang, Dahua Lin, and Lei Yang. Holistic evaluation of multimodal llms on spatial intelligence. *arXiv preprint arXiv:2508.13142*, 2025.

[3008] Rodrigo Gonzalez Laiz and Steffen Schneider. Understanding the limits of latent world models, 2026. URL https://arxiv.org/abs/2610.37588.

[3009] Yun wang, Junbin Xiao, Han Lyu, Yifan Wang, Jing Zuo, Zhanjie Zhang, Hong Huang, Dapeng Wu, and Angela Yao. Keep it in mind: User centric continual spatial intelligence reasoning in

egocentric video streams. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=UfVYPA9VnD.

[3010] Minqi Wang, Jie Peng, Hejun Dong, Ruchang Yao, Puyi Wang, Siyuan Huang, Jihai Zhang, Yafu Li, Qi Dou, Xiaoye Qu, and Yu Cheng. Worldprobe: When video world models fall apart under long-horizon spatial exploration, 2026. URL https://arxiv.org/abs/2610.39721.

[3011] Huizi Cao, Guanyu Xu, Zhixuan Liang, Ming Yin, Jiuxiang Gu, Manling Li, Mengdi Wang, and Shilong Liu. Same space or not: Benchmarking spatial intelligence through cross-view contradictions. *arXiv preprint arXiv:2610.23959*, 2026.

[3012] Aravindh Mahendran, Michael King, Matthew Koichi Grimes, Antoine Yang, Tyler Zhu, Joseph Heyward, Tengda Han, Shiry Ginosar, Chen Sun, Dima Damen, Simon Osindero, Noah Snavely, Simon Lynen, João Carreira, and Viorica Pătrăucean. KilometerVision: A new frontier for large-scale spatial intelligence in VLMs. *arXiv preprint arXiv:2609.39588*, 2026.

[3013] Fang Yu, Xiaoxue Wu, Haiyu Zhang, Yangfu Li, Botong Zhao, Yu Qiao, Tianfan Xue, Yue Lu, and Xinyuan Chen. Grome: Grounding factorized memory into video world models, 2026. URL https://arxiv.org/abs/2610.39812.

[3014] Yan Ki Chan, Yanran Zhang, Wenzhao Zheng, Xiaofeng Wang, Zheng Zhu, Lei Chen, Jie Zhou, and Jiwen Lu. State world models: Execute the rules, generate the pixels. *arXiv preprint arXiv:2610.26811*, 2026.

[3015] Yifei Li, Pengyiang Liu, Yuhang Zang, Zhongyue Shi, Qi Fu, Hongye Hao, and Jiwen Lu. Ovo-s-bench: A hierarchical benchmark for streaming spatial intelligence in multimodal llms, 2026. URL https://arxiv.org/abs/2606.03890.

[3016] Yonghao Yu, Lang Huang, Runyi Li, and Toshihiko Yamasaki. Vision-language-action or world-action? Just Action Models. *arXiv preprint arXiv:2610.37219*, 2026.

[3017] Ayush Mittal and Dhruv Gupta. Calibrated for which policy? conformal uncertainty for fixed world models under intervention. *arXiv preprint arXiv:2610.35877*, 2026.

[3018] Jiayi Luo, Yang Zhang, Liliang Chen, Nan Wang, Chen Gao, Boxiang Qiu, Zhuoxuan Li, Baorui Peng, Chuankang Li, Yuxiang Yan, Si Liu, Qingyun Sun, Shuicheng Yan, Maoqing Yao, Jianxin Li, and Guanghui Ren. Rollback: Experience backtracking in world models for policy improvement. *arXiv preprint arXiv:2610.36068*, 2026.

[3019] Louisa Dahmani and Véronique D Bohbot. Habitual use of gps negatively impacts spatial memory during self-guided navigation. *Scientific reports*, 10(1):6310, 2020.

[3020] Haoyuan Li, Qihang Cao, Huanyuan Zhou, Xin Hu, Sijia Chen, Zihan Guo, Kun Xiang, Yunshuang Nie, Tao Tang, Xinlong Zheng, et al. Geothinker-f: Endogenous predictive world representations for fast spatial intelligence. *arXiv preprint arXiv:2610.20592*, 2026.

[3021] Zhongbo Wu, Jiakai Huang, Zheng Zhang, Siyu Xu, Kaixiang Chen, Xiaohong Liu, Shan You, Chang Xu, and Tao Huang. Pasa-kv: Attacking action-facing visual routes in world action models, 2026.

[3022] Guancheng Pan, Qixin Tan, Yang Yuan, and Andrew Chi-Chih Yao. Algebraic world model: From exploration to proofs, 2026. URL https://arxiv.org/abs/2610.40503.

[3023] Yiming Zhang, Ruoxuan Cao, and Zhihang Zhong. Cocosi: Collaborative cognitive map construction for spatial intelligence, 2026. URL https://arxiv.org/abs/2606.10401.

Author Contributions

Contributions are listed in the order of the author list on the title page. For each author who drafted text, the parts of the handbook they wrote are given in book order, and a co-authored part names its co-author. Authors who reviewed the manuscript and advised on its direction are marked as such.

- **Shengqiong Wu**, University of Oxford:
  - Part I, Introduction and Background, Chapters 1–3 (with Hao Fei)
  - Audio, Section 4.5
  - Object Tracking, Section 6.4.2
  - Physical Reasoning, Section 7.1.2
  - Differentiable Optimization Layers for Spatial Reasoning, Section 7.2.4
  - Reinforcement Learning with Spatial Inductive Biases, Section 7.2.5
  - Hybrid Spatial Reasoning, Section 7.4
  - Spatial Reasoning Benchmarks, Section 7.5
  - 3D Object Synthesis, Section 8.2.1
  - 3D Scene Reconstruction, Section 8.2.3
  - 3D Scene Editing, Section 8.2.5
  - Summary, Challenges, and Future Directions of Video Spatial Generation, Section 8.3.6
  - Introduction to Motion Generation, Section 8.4
  - Programmatic and Symbolic Spatial Generation, Section 8.7
  - Summary and Outlook of Spatial Generation, Section 8.8
  - Aerospace Engineering, Section 11.3
  - Spatial Intelligence for Others, Chapter 14
  - Part V, Discussions and Futures, Chapters 15–17
  - Introductions to Parts II–IV, to Chapters 4 and 6–14, and to the sections of Chapters 6–8
- **Yu Zhao**, National University of Singapore:
  - Height and Distance Estimation, Section 6.2.3
  - Layout Estimation and Room Parsing, Section 6.3.3
  - Mapping and SLAM, Section 6.3.5
  - Surface Normal Estimation, Section 6.2.2
  - Visual Odometry and Camera Pose Estimation, Section 6.2.4
  - Scene Flow Estimation, Section 6.4.1
  - Motion Generation, Sections 8.4.1–8.4.4
  - Autonomous Driving, Section 9.2
  - GeoAI, Section 11.2
- **Tianjie Ju**, National University of Singapore:
  - Foundations: What Spatial Intelligence Requires, Chapter 15
  - Research Gaps and Future Directions, Chapter 16
  - Content Screening
- **Jundong Xu**, National University of Singapore:
  - Geometric and Algebraic Reasoning, Section 7.1.1
  - Topological and Graph-based Reasoning, Section 7.1.3

  - Constraint-based Spatial Solving, Section 7.1.4
  - Rule-based and Programmatic Reasoning, Section 7.1.5
  - Causal Reasoning, Section 7.1.6
  - Warehouse and Industrial Logistics, Section 9.5
- **Bobo Li**, National University of Singapore:
  - Temporal Localization, Section 6.4.3
  - Multi-agent Spatial Reasoning, Section 7.2.6
  - Text-to-image Generation, Section 8.1.1
  - Entertainment, Section 10.2
  - AR/VR/XR, Section 10.1 (with Xin Kong)
- **Jingkang Yang and Shulin Tian**, Nanyang Technological University, Singapore:
  - Chain-of-Thought for Spatial Reasoning, Section 7.3.3
  - Tool-augmented Reasoning, Section 7.3.5
  - Retrieval- & Memory-augmented Spatial Reasoning, Section 7.3.6
  - Agentic Spatial Reasoning, Section 7.3.7
  - Assistant, Section 13.1
- **Daoan Zhang**, University of Rochester:
  - Audio-Visual Spatial Localization, Section 6.5.1
  - 2D Spatial-aware Image Generation, Section 8.1.2
  - 2D Spatial Inpainting & Editing, Section 8.1.3
  - Medicine and Clinical Healthcare, Section 12.1
- **Yanguang Zhao**, National University of Singapore:
  - Keypoint and Landmark Detection, Section 6.1.3
  - Depth Estimation, Section 6.2.1
  - Natural Science, Section 12.2
- **Xuan Ju**, The Chinese University of Hong Kong:
  - Text-to-Video Generation, Section 8.3.1
  - Spatial Controlled Video Generation, Section 8.3.2
  - Temporal Controlled Video Generation, Section 8.3.3
  - Other Controlled Video Generation, Section 8.3.4
  - Video Editing, Section 8.3.5
- **Haoyu Zhao**, The Chinese University of Hong Kong:
  - 3D Representations, Section 4.2
  - 4D Spatial Generation, Section 8.5
- **Yuchen Zhou**, National University of Singapore:
  - Spatial Relation Extraction, Section 6.3.1
  - Scene Graph Generation, Section 6.3.2
  - Action Detection, Section 6.3.4
  - LLM/VLM-centric Spatial Reasoning, Section 7.3.1
  - VLA-based Spatial Reasoning, Section 7.3.2
- **Xiangtai Li**, Nanyang Technological University, Singapore:

  - 2D Representations, Section 4.1
  - Object Detection, Section 6.1.1
  - Segmentation, Section 6.1.2
- **Zhedong Zheng**, University of Macau:
  - Aerial Agents, Section 9.4
- **Joy Hsu**, Stanford University:
  - Domain-Specific Representations, Section 4.6
  - Cross-Modal Spatial Representation, Chapter 5
- **Xin Kong**, Imperial College London:
  - 4D Representations, Section 4.3 (with Chuanxia Zheng)
  - 3D Scene Synthesis, Section 8.2.2
  - AR/VR/XR, Section 10.1 (with Bettina Schlager)
- **Ruining Li**, University of Oxford:
  - World-Model-based Spatial Reasoning, Section 7.3.8 (with Hao Fei)
  - Digital Twins, Section 11.1
- **Cheng Zhang**, Monash University:
  - 3D Panorama Synthesis, Section 8.2.4
- **Chengzu Li**, University of Cambridge:
  - Scene Classification, Section 6.1.4 (with Hao Fei)
  - Implicit Neural Spatial Reasoning, Section 7.2.1 (with Hao Fei)
  - Explicit Structured Neural Reasoning, Section 7.2.2 (with Hao Fei)
  - Geometry-aware Neural Spatial Reasoning, Section 7.2.3 (with Hao Fei)
  - Code/DSL Generation for Spatial Reasoning, Section 7.3.4 (with Hao Fei)
- **Jiafei Duan**, National University of Singapore:
  - Robotic Manipulation, Section 9.1
- **Yu (Bryan) Zhou**, University of California, Los Angeles:
  - Vision-Language Grounding, Section 6.5.2
- **Kenneth Li**, Harvard University:
  - Language Representations, Section 4.4
- **Chuanxia Zheng**, Nanyang Technological University, Singapore:
  - 4D Representations, Section 4.3 (with Xin Kong)
- **Lei Cui**, Microsoft Research: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Furu Wei**, Microsoft Research: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Ziwei Liu**, Nanyang Technological University, Singapore: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Steven K. Feiner**, Columbia University: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Jiebo Luo**, University of Rochester: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook

- **Jianfei Cai**, Monash University: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Shuicheng Yan**, National University of Singapore: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Ming-Hsuan Yang**, University of California, Merced: Reviewing & Advisory, providing comments and feedback that shaped the direction of the handbook
- **Hao Fei**, University of Oxford:
  - Part I, Introduction and Background, Chapters 1–3 (with Shengqiong Wu)
  - Scene Classification, Section 6.1.4 (with Chengzu Li)
  - Pose Estimation, Section 6.2.5
  - Implicit Neural Spatial Reasoning, Section 7.2.1 (with Chengzu Li)
  - Explicit Structured Neural Reasoning, Section 7.2.2 (with Chengzu Li)
  - Geometry-aware Neural Spatial Reasoning, Section 7.2.3 (with Chengzu Li)
  - Code/DSL Generation for Spatial Reasoning, Section 7.3.4 (with Chengzu Li)
  - World-Model-based Spatial Reasoning, Section 7.3.8 (with Ruining Li)
  - World Modeling, Section 8.6
  - Wearable AI System, Section 9.3
  - Figures: drew part of the handbook's illustrations, most notably the teaser figure