ByteDance | Seed

SpatialTree: How Spatial Abilities Branch Out in MLLMs

Yuxi Xiao[▲,★,*], **Longfei Li**[◇,★,*], **Shen Yan**[★], **Xinhang Liu**[★], **Sida Peng**[▲], **Yunchao Wei**[◇], **Xiaowei Zhou**[▲], **Bingyi Kang**[★]

[▲]Zhejiang University, [★]ByteDance Seed, [◇]Beijing Jiaotong University

[†]Project Lead, [*]Equal Contribution

Abstract

Cognitive science suggests that spatial ability develops progressively—from perception to reasoning and interaction. Yet in multimodal LLMs (MLLMs), this hierarchy remains poorly understood, as most studies focus on a narrow set of tasks. We introduce SpatialTree, a cognitive-science-inspired hierarchy that organizes spatial abilities into four levels: low-level perception (L1), mental mapping (L2), simulation (L3), and agentic competence (L4). Based on this taxonomy, we construct the first capability-centric hierarchical benchmark, thoroughly evaluating mainstream MLLMs across 27 sub-abilities. The evaluation results reveal a clear structure: L1 skills are largely orthogonal, whereas higher-level skills are strongly correlated, indicating increasing interdependency. Through targeted supervised fine-tuning, we uncover a surprising transfer dynamic—negative transfer within L1, but strong cross-level transfer from low- to high-level abilities with notable synergy. Finally, we explore how to improve the entire hierarchy. We find that naïve RL that encourages extensive "thinking" is unreliable: it helps complex reasoning but hurts intuitive perception. We propose a simple auto-think strategy that suppresses unnecessary deliberation, enabling RL to consistently improve performance across all levels. By building SpatialTree, we provide a proof-of-concept framework for understanding and systematically scaling spatial abilities in MLLMs.

Date: April 23, 2026
Correspondence: Bingyi Kang, Xiaowei Zhou, Yunchao Wei
Project Page: https://spatialtree.github.io/

1 Introduction

Spatial abilities refer to the capacity to perceive, understand, reason about, and interact with 2D and 3D space, a long-standing topic in cognitive science [15, 48, 51]. In multimodal large language models (MLLMs), these abilities form the cornerstone of Spatial Intelligence (SI), yet remain challenging to study systematically due to their inherent complexity and broad scope [33, 66].

Recent research followed a task-centric trajectory. Early efforts concentrated on spatial tasks within single images [9, 36, 53], such as relative object positioning and size estimation. As capabilities matured, studies expanded into the 3D domain, tackling grounding, detection, and captioning from point clouds [17, 60, 70]. More recently, the advent of multi-view and video-capable models has further diversified the landscape [12, 22, 54, 55, 63, 67], covering a broad spectrum from spatial relevant reasoning to egocentric and dynamic object

[1]Work done during Yuxi Xiao and Longfei Li's internship at Bytedance Seed.

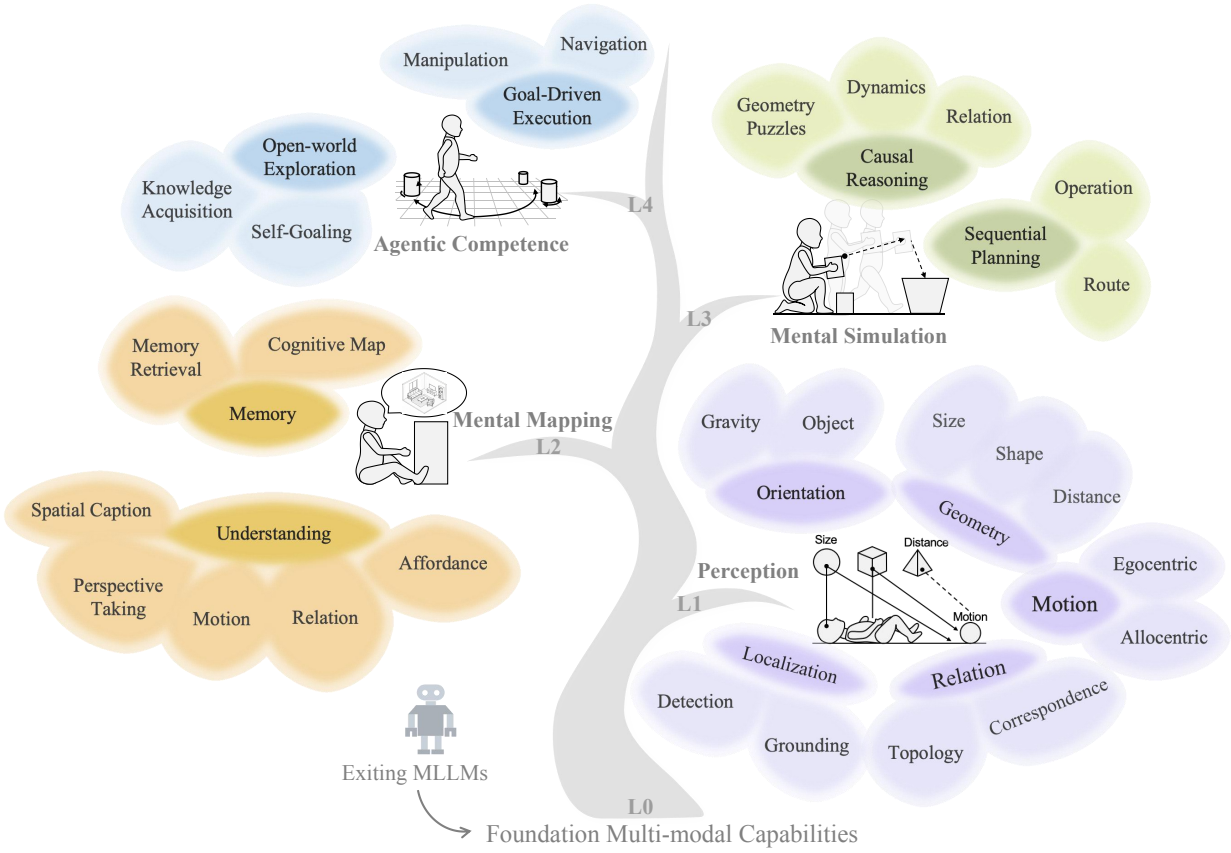


Figure 1 SpatialTree. Inspired by cognitive science, our proposed SpatialTree organizes spatial intelligence into a four-layer hierarchy (L1-L4). Rooted in foundational multi-modal capabilities (L0), the tree progressively branches from Basic perception (L1) to agentic competence (L4).

understanding.

However, these task-centric benchmarks remain fragmented and often treat spatial capabilities as isolated or overlapping skills. This lack of unification makes it difficult to unravel the inherent structure and cross-level dependencies within the proliferation of spatial tasks.

> Can we move beyond disjointed, task-centric evaluations to uncover a compact set of atomic capabilities that reveal how spatial abilities emerge, interact, and transfer?

Drawing inspiration from cognitive science insight that "intelligence is a dynamic structure built through successive stages" [42], we advocate a paradigm shift: moving from fragmented tasks to a capability-centric framework. Specifically, we structure Spatial Abilities as a four level capability tree (Fig. 1):

- **L1 Perception:** This level focuses on native perception of space, capturing raw geometric and physical attributes such as size, distance, and motion, without relying on language or symbolic reasoning.
- **L2 Mental Mapping:** This level maps spatial perception to language, grounding spatial concepts in linguistic semantics and forming language-structured spatial memory.
- **L3 Mental Simulation:** This level supports internal reasoning about space, enabling mental simulation, including causal reasoning about dynamics, relational and geometric problem solving, and sequential planning for actions and navigation.
- **L4 Spatial Agent:** This level executes actions in space, integrating perception, language, and reasoning to interact with the environment, interpret feedback, and complete long-horizon spatial tasks.

To populate this taxonomy, we first reorganize and unify data from prior works [22, 30, 32, 53, 54, 61, 63, 67, 68, 70], which primarily cover specific sub-domains within L1 to L3. To address missing abilities and enrich annotations beyond existing datasets, we construct a Spatial Engine that integrates multiple expert models [29, 34, 38, 52, 57–59, 64]. For L1 to L3, we enhance data diversity by introducing multiple QA formats for the same underlying problem and by incorporating additional sub-capabilities, including orientation estimation, localization, and affordance understanding. In contrast, L4 requires agentic interactions that are largely absent from prior benchmarks. We therefore meticulously curate L4 data across three representative embodiments: character navigation, robot grippers, and human hands. To facilitate MLLM adaptation to agentic tasks, we design an action mapping strategy that discretizes low-level actions into high-level motion primitives, forming an executable action space. In addition, human–object interaction sequences are reformatted into multi-step multiple-choice tasks through manual annotation. Based on the resulting datasets and task formulations, we establish **SpatialTree-bench** to evaluate a broad range of mainstream open-source and commercial MLLMs.

Our evaluation on **SpatialTree-Bench** uncovers a clear hierarchical dependency: while foundational abilities (L1) function independently, higher-level capabilities exhibit strong reliance on these basics. To systematically validate this structure, we move beyond static evaluation to targeted training interventions using both Supervised Fine-Tuning (SFT) and Reinforcement Learning (RL). First, our SFT experiments reveal that spatial skills do not scale uniformly. We observe that while basic abilities struggle with intra-level interference, they serve as critical *stepping stones* for higher-level tasks, unlocking a powerful **multi-ability synergy** when jointly trained. Second, we explore the role of reasoning via RL, leading to a surprising dichotomy between "thinking" and "perceiving". We find that while extensive reasoning is indispensable for complex tasks, it can be detrimental to intuitive perception (e.g., numerical estimation), where over-thinking degrades precision. This necessitates a **auto-think strategy**: suppressing reasoning for immediate perception while encouraging it for complex planning. In summary, our key contributions are:

- We construct **SpatialTree**, the first capability-centric benchmark that organizes spatial intelligence into a rigorous hierarchy, enabling granular diagnosis of model capabilities.
- We reveal the structural transfer dynamics of spatial skills, demonstrating how foundational perception acts as a prerequisite for agentic competence through systematic SFT experiments.
- We identify the trade-off between reasoning and perception in RL training, proposing a differentiated inference strategy that balances intuitive sensing with complex reasoning.

2 Related Work

Spatial Cognitive Modeling. Understanding spatial cognition has long been a central goal in cognitive science and AI. A common insight from classical theories is that spatial abilities are hierarchical, ranging from basic perception and sensorimotor interactions to higher-level reasoning and planning. Piaget [42] highlighted the developmental progression of such abilities, Tolman [51] introduced the idea of cognitive maps to represent environments for flexible navigation, and Kuipers [25, 26] formalized a hierarchical spatial representation linking local perception to global knowledge. More recent symbolic and neural approaches [39, 48] extend these insights to computational models of spatial representation, memory, and reasoning. These studies collectively motivate our SpatialTree, which organizes spatial intelligence into multi-level capabilities, bridging classical theory with systematic computational evaluation.

Multi-modal Large Language Models. The success of GPT-3 [6] and GPT-3.5 [40] demonstrated the potential of large language models for complex linguistic understanding and reasoning. GPT-4V [41] extends GPT-4 [1] with visual inputs, enabling single-image understanding and basic spatial reasoning. Open-sourced models such as LLaVA [31] and QwenVL [2] gradually added multi-image and video capabilities, supporting spatiotemporal reasoning. Reasoning-augmented LLMs, pioneered by OpenAI O1 [21] and DeepSeek-R1 [13], integrate chain-of-thought and reinforcement learning to enhance high-level inference. Building on these advances, GPT-4O [19] and Gemini 2.5 [8] combine perception and reasoning to support complex, agentic decision-making. Collectively, these milestones progressively enable hierarchical spatial intelligence in MLLMs, motivating

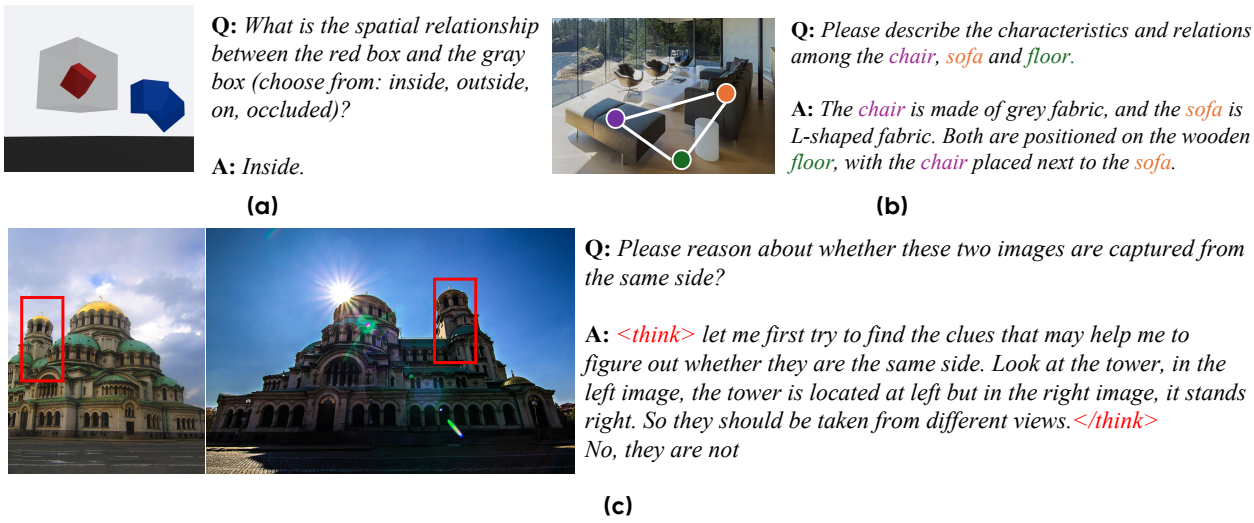


Figure 2 Different Emphasis across Hierarchy Levels. Taking *Relation* in L1, L2, L3 as an example: **(a)** *Relation* in *Perception*, involving basic spatial relations (e.g., inside, outside) **(b)** *Relation* in *Understanding*, describing the attributes and mutual relationships among different objects. **(c)** *Relation* in *Causal Reasoning*, leveraging visual cues and logical inference to solve more complex relational tasks.

structured benchmarks and evaluation frameworks across low-level perception, intermediate reasoning, and high-level agentic competence.

Benchmarks for Spatial Intelligence in MLLMs. Benchmarks for spatial abilities in MLLMs have evolved alongside the models themselves. Early efforts, such as BLINK [9], SpatialEval [53], and 3DSR-Bench [36], focused on evaluating spatial understanding tasks in single images, including distance estimation, relational question answering, and spatial captions. As MLLMs increasingly support multi-frame and video inputs, benchmarks such as VSI-Bench [63] and MMSI-Bench [67] have emerged to evaluate spatial reasoning across multiple views and dynamic scenes. To further enrich task diversity and coverage, Omnispatial [22], SITE [55], and IR3D-Bench [32] extend benchmarks to geometry puzzles, dynamic reasoning, and inverse rendering tasks. Built upon prior efforts, our SpatialTree benchmark systematically organizes spatial abilities into a hierarchical framework, providing the first thorough evaluation across different capabilities.

3 The SpatialTree Taxonomy

In this section, we introduce SpatialTree taxonomy of spatial capabilities. Different levels of the taxonomy emphasize different aspects of spatial ability, as illustrated in Fig. 2. Lower levels focus on intuitive and basic perceptual skills, while higher levels progressively require richer knowledge and more complex logical reasoning. Concrete instantiations are discussed in Sec. 4.

3.1 Perception

We begin with **Perception** (L1), which characterizes the most primitive capacity to sense visual signals from the environment. Rooted in the sensorimotor stage of human development, this level represents the instinctive ability to process spatial information prior to linguistic abstraction. Based on these fundamental biological needs, we categorize perception into five core abilities:

Geometry interprets the physical form and metric properties of the world. Evolutionarily, this allows agents to intuitively decipher three core dimensions: *Distance* (gauging metric and relative depth), *Size* (estimating physical magnitude, area, and volume), and *Shape* (discerning contours, boundaries, and geometric primitives).

These faculties enable immediate judgments, such as deciding if a fruit is small enough to hold or a path is wide enough to traverse.

Motion reflects the innate capacity to process dynamic visual signals over time. This perception is categorized into *Egocentric* (sensing self-motion and heading direction) and *Allocentric* (perceiving the movement and speed of external objects). Interpreting these signals allows humans to instantly determine if an entity is static or mobile, supporting tasks like avoiding obstacles or tracking targets.

Orientation is rooted in the necessity to maintain balance and align with the environment. Humans instinctively sense "up" and "down" via vestibular cues to avoid falling, and visually recognize object poses to interact with them effectively. This category includes *Gravity* (sensing the vertical axis, e.g., knowing which way is up) and *Object* (perceiving poses, e.g., noticing a cup is tilted).

Relation captures the structural continuity and spatial arrangement of the visual world. Beyond isolating individual objects, intuitive perception registers how entities are spatially arranged and whether they correspond across views. This category includes *Topology* (perceiving basic spatial configurations such as *inside*, *outside*, or *overlap*) and *Correspondence* (recognizing the same object or landmark across different viewpoints or visual conditions).

Localization anchors visual stimuli within 2D/3D space. It addresses the fundamental question of where, allowing humans to spot a friend in a crowd or locate keys on a cluttered table. This category includes *Detection* (identifying object presence and spatial extent) and *Grounding* (associating visual observations with spatial positions or coordinates).

3.2 Mental Mapping

Mental Mapping (L2) marks a shift toward alignment with language. It encompasses two key aspects: mapping spatial primitives to semantic concepts, and constructing a language-aligned memory system. Accordingly, we classify Mental Mapping into two main sub-abilities:

Understanding interprets the semantic meaning underlying geometric perception, shifting the focus from mere existence to function and identity. This capability begins by translating visual scenes into linguistic descriptions through *Spatial Captioning*, and by identifying semantic *Relations* (e.g., distinguishing "riding" from "sitting on"). It further elevates *Motion* from raw kinematic cues to semantic interpretation, recognizing purposeful actions rather than simple movement. Crucially, this level includes *Perspective Taking*—the ability to mentally align with alternative viewpoints—and *Affordance* understanding, which identifies the functional possibilities of objects (e.g., recognizing that a handle is graspable), thereby bridging perception with potential future interactions.

Memory extends spatial awareness beyond the instantaneous field of view, enabling the system to retain and update spatial information over time. This capability relies on constructing a **Cognitive Map**, which synthesizes fragmentary observations (e.g., video frames or multi-view images) into a compact and unified global representation. Based on this mental model, the system performs **Memory Retrieval** to ground specific semantic events or moments, allowing it to recall where an object appeared or when a specific action occurred, even if the target is currently occluded or out of sight.

3.3 Mental Simulation

Reasoning and planning prior to action execution are essential components of MLLMs, aligning naturally with the Chain-of-Thought paradigm in language model reasoning. In spatial cognitive science, this process is commonly referred to as mental simulation. We taxonomize mental simulation into two orthogonal directions: causal reasoning and sequential planning.

Causal Reasoning allows MLLMs to model spatial interactions, physical dynamics, and entity relationships within a simulated mental space. It includes reasoning about object geometry (e.g., how shapes interlock in spatial puzzles), motion prediction (e.g., how an object traverses a path), and analyzing semantic–spatial relations (e.g., object A is left of object B). By mentally simulating cause–effect chains in spatial scenarios, MLLMs establish the logical substrate for subsequent planning.

Sequential Planning converts causal insights into coherent, goal-directed action plans expressed in language. It entails designing high-level, step-by-step strategies (e.g., "first move toward the door, then turn right, and finally interact with the handle") and generating abstract routes that respect spatial logic (e.g., "go around the table to reach the sofa"). By chaining linguistic action primitives, MLLMs produce strategic plans that ensure the conceptual sequence aligns with the overarching goal before any low-level execution.

3.4 Agentic Competence

Agentic Competence represents the culmination of spatial intelligence, bridging the gap between cognitive planning and practical execution. It focuses on the agent's ability to translate internal plans into tangible interactions within dynamic environments. This capability is grounded in diverse embodied scenarios, where the model must interpret visual streams to generate precise action sequences. Key tasks include predicting control commands from **video game frames**, deriving manipulation sequences from **robotic arm videos**, and identifying potential movement directions or navigational affordances in **real-world scenes**.

We begin from the ultimate objective of a *Spatial AI Agent* — an MLLM-driven system that integrates multi-modal observations, updates its memory, and selects actions to interact with the 3D world in an intuitive manner. Formally, the agent performs sequential decision-making by modeling:

$$(S_t, A_t, M_t) \sim P_\theta\Big(\cdot \mid O_t, H_{t-1}\Big), \text{where} \quad H_{t-1} = \{(O_0, A_0, M_0), \ldots, (O_{t-1}, A_{t-1}, M_{t-1})\}$$

where $O_t \in \mathcal{O}$ is the current multi-modal observation, $S_t \in \mathcal{S}$ the internal latent state (e.g., goal, plan, or belief), $A_t \in \mathcal{A}$ the chosen action, and $M_t \in \mathcal{M}$ the updated memory representation. MLLMs are expected to output interactive actions executable across 3D environments and embodiments, such as games, simulators, and the physical world. Unlike Vision-Language Action Models (VLAs) decording the low-level control signals in robotics [20], MLLMs take the language as the only interface to link with environments like GUI Agents [43].

4 Instantiating the SpatialTree Benchmark

This section details how the SpatialTree taxonomy is instantiated as a benchmark, covering data curation pipeline (seen in in Fig. 3), and more specific examples of the QAs are demonstrated in Sec. F.

4.1 Data Curation

Data Annotation Framework. As shown in Fig. 3, the data engines are organized hierarchically. At the perception level, we employ a set of expert perception models, including DepthAnything3 [64], SpatialTracker [58, 59], GeoCalib [52], and OrientAnything [57], to extract intermediate perceptual representations, such as depth, correspondences, tracking results, and gravity direction. These representations are then further processed using our designed QA templates and rephrased by LLMs to construct the target QAs. For L2, we leverage 3D reconstruction pipelines (Sec. C.2) to generate BEV maps from videos, which are subsequently captioned and transformed into QAs for cognitive mapping and memory retrieval with multimodal LLMs. For other understanding tasks, such as affordance, we build on partially annotated datasets [35] and augment them by randomly constructing multiple-choice candidates through various visual prompts and by reformulating questions with more abstract descriptions. For L3, we build upon annotated reasoning-related QAs and introduce structured thinking templates, enabling an LLM-based rephraser to augment the original QAs with explicit Chain-of-Thought (CoT) reasoning. For agentic tasks, we curate Internet data covering manipulation and navigation across diverse embodiments (e.g., human hands, robotic grippers, and game characters). We further process the data with our Action-Extraction Pipeline (Sec. C.2), converting embodiment-specific actions into unified key–mouse action sequences suitable for MLLMs. In parallel, we introduce a human annotation pipeline to annotate videos with executable multi-option action sequences.

Data Resources. Our SpatialTree-Bench is constructed by systematically reorganizing numerous recent datasets (detailed in the Appendix) [22, 30, 36, 53, 54, 61, 63, 65, 67, 68, 70] to address their scattered capability coverage and over-reliance on simple questions. We first map each question to our SpatialTree framework and then enhance the evaluation protocol; for instance, complex reasoning tasks from CameraBench and MMSI-Bench are converted to a hybrid *multi-option + LLM-as-a-Judge* format for a finer-grained assessment.

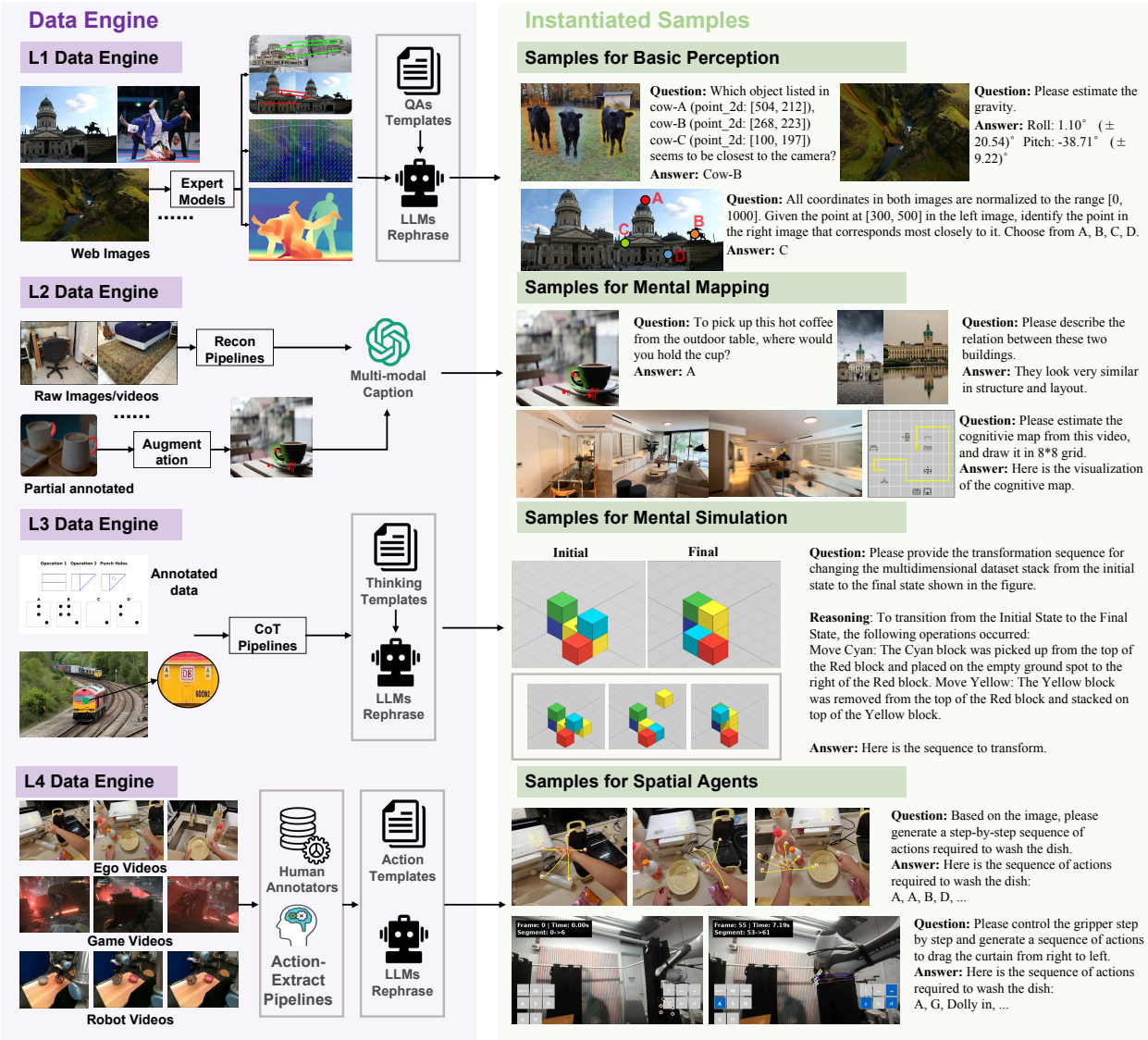


Figure 3 Benchmark Data Engines. Level-specific engines process data and construct QAs.

To fill the remaining capability gaps, we introduce **SpatialPlus**, a new dataset targeting underrepresented abilities (e.g., L1 Orientation, L1 Shape, L2 Spatial Caption) with a primary emphasis on L4 Agentic Competence. We leverage our proprietary **SpatialEngine** to automatically create annotations from a diverse array of video sources, including 3D reconstruction datasets, in-game footage [23], egocentric manipulation videos [18], and robotics data [24]. More implementation details are discussed in Appendix.

4.2 Evaluation Metrics Designs

To facilitate robust evaluation within our hierarchical benchmark, we design diverse metrics tailored to specific downstream tasks. As illustrated in Fig. 4, these metrics are primarily categorized into multiple-choice questions (70.7%), numeric accuracy (e.g., mean relative accuracy), LLM-based evaluation (LLM-as-a-Judge) and specific numeric metrics.

5 A Hierarchical Analysis of Spatial Capabilities

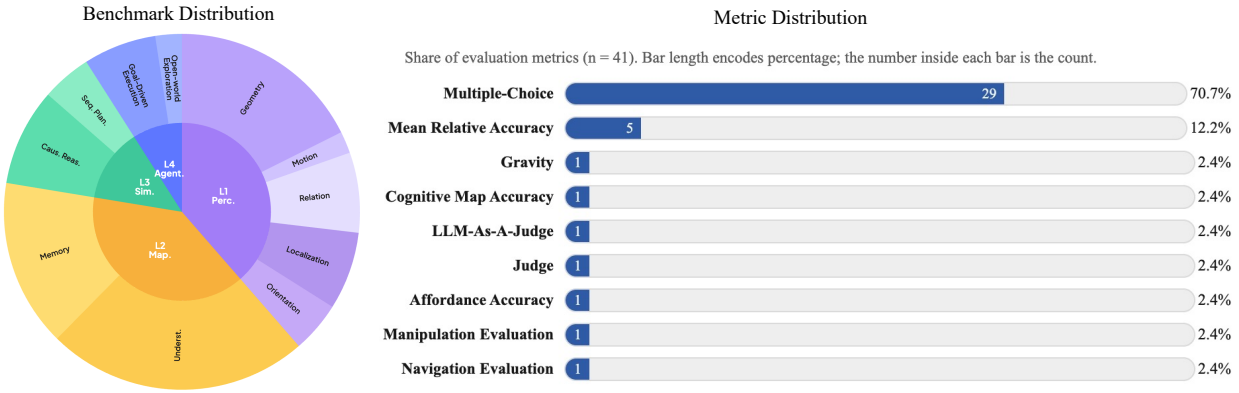


Figure 4 Distribution of Benchmark data and Evaluation Metrics. We analyze the metric usage across 41 tasks in our benchmark. The evaluation relies primarily on multiple-choice questions (70.7%), complemented by task-specific numeric metrics (e.g., cognitive map accuracy) and LLM-as-a-Judge protocols.

5.1 Models and Metrics

Benchmarked Models. We categorize the evaluated MLLMs into three groups: (1) Thinking Models, i.e., models augmented with explicit reasoning or chain-of-thought generation mechanisms (reasoning-augmented), including Gemini 3 Pro [11], Gemini 3 Flash [10], Gemini 2.5 Flash, Gemini 2.5 Pro [8], GLM-4.5V [16], Seed1.5VL [14] and Seed1.8 [46]; (2) Non-Thinking Models, which do not explicitly optimize for reasoning-style generation, such as Gemini 2.5-Pro-Nonthink, Gemini 2.5-Flash-Nonthinking, and GPT-4o [19]; and (3) Open-Source Models, including Qwen2.5-VL [3], Qwen3-VL [62], and Kimi-VL [50], representing recent community-driven multimodal advances. This diverse selection enables comprehensive comparisons across reasoning and non-reasoning paradigms, proprietary and open-source ecosystems, and model scales ranging from 32B to 72B parameters—providing a holistic overview of the current MLLM landscape. A full list of evaluated models is shown in Tab. 1.

Evaluation Metrics. Our evaluation employs a multi-faceted set of metrics tailored to the specific abilities at each level of the SpatialTree. For perception and understanding tasks (L1-L2), we primarily use accuracy-based metrics, such as classification accuracy for object recognition, Mean Squared Error (MSE) for distance estimation, and angular difference for orientation tasks. For higher-level reasoning and planning tasks (L3-L4), we measure task success rates. In the case of agentic tasks (L4), we further analyze the quality of generated actions using metrics like positional error (L2 distance) and orientation error (angular difference) against ground-truth trajectories.

5.2 Overall Performance

We first present the overall performance of all benchmarked models on our proposed SpatialTree-Bench, with detailed results summarized in Tab. 1. In our benchmark, Gemini 3 Flash achieves the best results (57.8) and Qwen3VL-235B get 40.0 leads the benchmark among open-source models.

6 Exploring Ability Dependencies and Hierarchical Transfer

Methods	Rank	Avg.	L1 Perception					L2 Mental Mapping		L3 Mental Simulation		L4 Agentic Competence	
			Geom. [0.40]	Motion [0.15]	Rel. [0.15]	Local. [0.20]	Orient. [0.10]	Underst. [0.70]	Memory [0.30]	Caus. Reas. [0.65]	Seq. Plan. [0.35]	Goal Exec. [0.50]	Open Expl. [0.50]
Non-Thinking Models													
GPT-4o	13	31.9	23.9	38.6	29.8	24.2	36.2	31.2	43.6	29.3	40.5	25.8	39.2
Gemini2.5 Flash NT	10	35.8	31.6	29.3	30.8	35.2	45.4	36.4	53.7	28.4	36.9	27.6	45.7
Gemini2.5 Pro NT	5	41.4	36.2	30.0	33.2	47.0	48.5	43.3	55.2	39.6	47.5	29.2	46.0
Thinking Models													
Seed1.5vl	6	41.3	39.2	36.6	24.6	44.7	44.8	46.5	38.5	36.6	47.0	28.4	25.9
Seed1.8	3	50.3	42.5	42.7	49.9	54.5	52.4	56.3	51.0	49.7	53.6	26.0	70.6
GLM4.5V	9	36.0	35.3	24.0	32.5	34.5	43.7	34.5	34.1	33.8	41.0	26.8	49.7
Gemini2.5-Pro	4	50.1	47.8	32.6	44.4	61.6	47.9	50.5	61.5	47.6	58.2	28.3	63.3
Gemini2.5-Flash	7	41.1	35.6	29.3	40.5	56.2	44.6	43.1	51.4	36.4	49.1	26.9	46.6
Gemini3-Pro	2	56.5	54.5	47.4	62.5	74.0	53.9	60.5	50.3	56.0	68.2	29.9	68.5
Gemini3-Flash	1	57.8	50.1	40.7	62.9	62.4	54.6	62.0	66.8	58.4	68.6	31.6	70.8
Open-source Models													
Qwen2.5VL-7B	15	27.5	17.8	22.6	23.9	20.6	31.6	34.7	15.2	28.4	39.8	24.5	31.1
Qwen2.5VL-32B	14	27.9	21.4	30.0	25.8	22.0	35.1	29.0	21.7	32.8	37.0	14.1	38.6
Qwen2.5VL-72B	12	33.0	24.4	22.0	28.6	37.5	37.3	38.9	35.1	32.6	38.4	23.9	38.6
Qwen3VL-30B	11	35.3	31.9	26.0	32.3	20.7	39.2	37.8	48.1	32.7	44.2	25.8	40.9
Qwen3VL-235B	8	40.0	33.9	27.4	35.1	35.4	38.9	43.7	53.4	37.3	44.7	28.8	48.9
Kimi-VL-A3B	16	24.4	13.8	23.3	31.6	27.0	21.4	24.9	28.3	26.9	27.8	15.7	32.6

Table 1 Our-Bench. Dark gray indicates the best result among all models and light gray indicates the best result among open-source models. NT denotes the non-thinking model. **Avg** is computed using the weights in brackets [·].

To investigate the structure of spatial ability in MLLMs, we analyze dependencies among fine-grained sub-abilities using Pearson correlation coefficients computed from our benchmark scores. A high positive correlation indicates that strong performance on one ability tends to accompany strong performance on another. Fig. 5 presents a heatmap of these correlations across all models. Notably, higher-level capabilities (L3 and L4) exhibit stronger correlations (region A), suggesting that complex tasks such as route planning and causal reasoning rely on overlapping foundational sub-skills. In contrast, lower-level abilities (L1) show weak correlations, indicating they are largely independent. Based on this coarse correlation analysis, we select several underutilized low-level abilities for further investigation. These abilities are explored through Supervised Fine-Tuning (SFT) and explicit prompting to examine their influence and transfer, both within the same level and across higher levels.

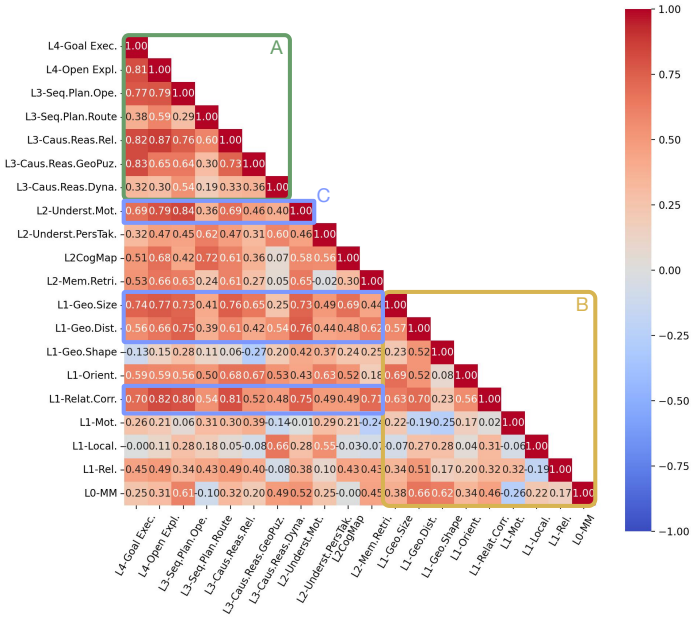


Figure 5 Inter-Capability Dependencies via Pearson Correlation. (A) Correlation matrix among higher-level capabilities (L3 and L4); (B) Correlation matrix among foundational L1 capabilities; (C) Salient low-level abilities influencing higher-level tasks.

6.1 Probing Cross-Ability Transfer via SFT

Finding 1

Cross-Ability Transfer: Single-ability L1 SFT induces cross-level transfer, while yielding limited or slightly negative effects on same-level abilities.

Based on a naive Pearson correlation analysis, we manually select three L1 abilities that exhibit the strongest correlations with higher-level performance: Geometry Distance (*L1-Geo.Dist*), Geometry Size (*L1-Geo.Size*), and Relative Correlation (*L1-Relat.Corr*).

		L1 Perception					L2 Mental Mapping		L3 Mental Simulation		L4 Agentic Competence	
Methods	**Avg.**	**Geom.**	**Motion**	**Rel.**	**Local.**	**Orient.**	**Underst.**	**Memory**	**Caus. Reas.**	**Seq. Plan.**	**Goal Exec.**	**Open Expl.**
Baseline	25.0	20.9	28.6	28.9	24.2	34.2	22.6	21.7	27.2	31.7	22.1	26.5
B+Dist.	24.5	24.1 (+3.2)	26.6 (−2.0)	23.2 (−5.8)	19.6 (−4.6)	34.3 (0.1)	24.6 (+2.0)	21.8 (0.1)	26.1 (-1.1)	30.8 (-0.9)	25.5 (+3.4)	26.1 (-0.4)
B+Corr.	25.2	17.6 (-3.2)	23.9 (−4.7)	30.2 (+1.3)	18.9 (−5.3)	35.6 (+1.4)	21.9 (-0.7)	24.6 (+2.9)	21.8 (−5.4)	33.9 (+2.2)	24.7 (+2.6)	35.9 (+9.4)
B+Size	23.5	24.3 (+3.4)	22.6 (−6.0)	21.4 (−7.5)	21.7 (−2.5)	34.5 (0.3)	21.9 (-0.8)	19.2 (−2.5)	23.4 (−3.8)	30.3 (−1.5)	21.5 (−0.6)	24.3 (-2.2)
B+Dist.+Size+Corr.	26.1	25.5 (+4.6)	29.3 (0.7)	29.4 (0.5)	16.4 (−7.8)	33.7 (0.5)	23.0 (+0.4)	24.2 (+2.5)	25.2 (−2.0)	34.2 (+2.5)	26.0 (+3.9)	28.5 (+2.0)
B+Dist.+Size+Corr.+Mot.	27.3	28.6 (+7.7)	24.6 (−4.0)	20.6 (−8.3)	26.3 (+2.1)	36.0 (+1.8)	22.2 (-0.4)	22.6 (0.9)	28.2 (+1.0)	32.8 (+1.1)	23.3 (+1.1)	35.9 (+9.4)
Baseline+75@(all spat.)	23.6	24.9 (+4.0)	22.6 (−6.0)	25.9 (−3.0)	17.4 (−6.8)	31.2 (−3.0)	22.2 (-0.4)	20.6 (−1.1)	25.7 (−1.5)	30.2 (−1.5)	19.7 (−2.4)	22.8 (−3.7)

Table 2 SFT Comparisons. "B+Dist.", "B+Corr.", and "B+Size" denote the baseline augmented with distance, correspondence, and size tuning data, respectively. Changes are color-coded as notable gains, neutral influence, and drops.

General Data Mixture. To construct the general visual-instruction data, we follow the VST [66] data mixing recipe and combine multimodal datasets from LLaVA-Video [69], LLaVA-NeXT-Interleave [28], and LLaVA-OneVision [27], covering single-image, multi-image, video, and 3D tasks. We then use SpatialEngine to generate ability-specific instruction data, mixed with the general data in a 1:3 ratio specifically for each. To isolate the gains from general data, we use a baseline fine-tuned only on the general data with the same token consumption.

Targeted SFT Data. For *L1-Geo.Dist*, we generate approximately 0.25M distance-relevant QA samples from SUNRGBD [49], Hypersim [45], and Matterport3D [7]. The training data is further augmented with visual prompts and multi-scale transformations to enhance distance reasoning. For *L1-Relat.Corr*, we generate matching data following VST [66], sampling 0.25M examples. Similarly, for *L1-Geo.Size*, we generate 0.25M samples from 3D bounding-box annotated datasets, including SUNRGBD, Hypersim and ArkitScenes [5].

Results and Analysis. As shown in Tab. 6, single-ability SFT on distance, correspondence, or size generally yields negligible gains or even substantial drops in other abilities at the same level. Specifically, `B+Dist.` increases **Geom.** abilities by +3.2, while decreasing **Motion**, **Rel.**, and **Local.** by -2.0, -5.8, and -4.6, respectively. However, it provides non-trivial gains in higher-level abilities, notably **Underst.** (+2.0) and **Goal Exec.** (+3.4). To give a further exploration on how this cability transfer happens and why, we provide a qualitative examples in Fig. 6. After being fine-tuned on distance-QA data, `B+Dist.` can generalize to much more complex distance-related questions in in-the-wild scenarios, including those with novel coordinate prompts and multiple points queried simultaneously. This indicates that the model has learned an awareness of distance rather than overfitting to specific QA templates. Besides, and more intriguingly, the improved distance ability also shows clear cross-level transfer. It benefits higher-level tasks such as robot-arm manipulation, where MLLMs are required to guide the gripper to move, rotate, and open/close in 3D space. A better sense of metric space helps the model generate more reasonable control decisions in the real world.

Finding 2

Multi-ability Synergy: The holistic integration across multiple fundamental abilities achieves synergistic gains far exceeding their individual effects.

Tab. 2 reveals an interesting phenomenon: individual SFT on any single ability—Distance, Size, or Correspondence—has limited impact on overall spatial performance, and can even slightly reduce it (e.g., `B+Dist.` **-0.5**, `B+Corr.` **+0.2**, `B+Size.` **-1.5** relative to the baseline). In contrast, combining all three abilities in a blended SFT (`B+Dist.+Size+Corr.`) yields an overall gain of **+1.1**, surpassing the performance of any individual ability and even exceeding the sum of their separate contributions. Remarkably, for abilities that suffered substantial drops under single-ability SFT—such as L1.Motion (best individual change **-2.0**)—the compositional training produces a positive improvement of **+0.7**.

6.2 Reinforcement Learning

Building on the findings from Supervised Fine-Tuning (SFT), we further investigate the potential of Reinforcement Learning with Verifiable Rewards (RLVR) to scale spatial abilities. Specifically, we employ Group Relative Policy Optimization (GRPO) [47] to align the model's policy with the hierarchical nature of the SpatialTree. Our experiments reveal that a uniform RL strategy is insufficient, conversely a hierarchy-aware

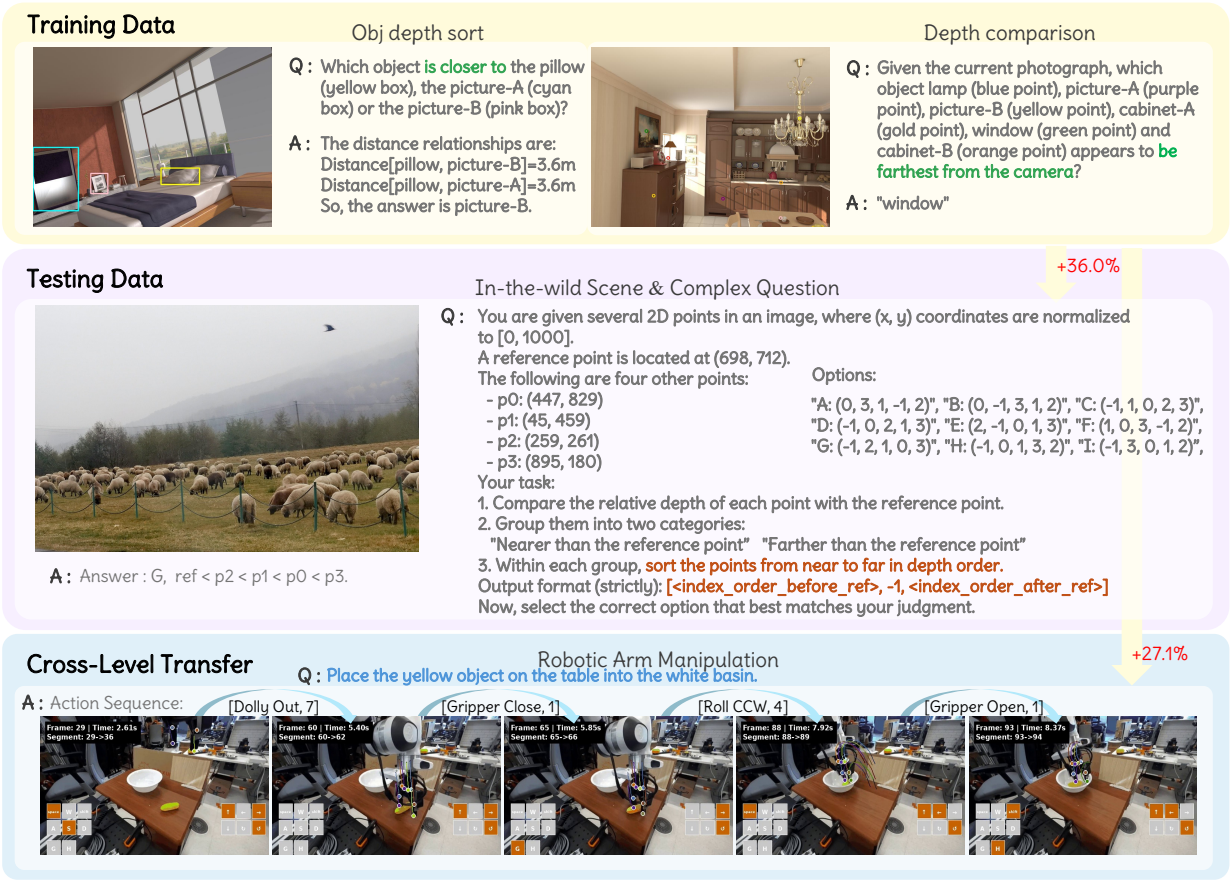


Figure 6 Demonstration of Capability Transfer after Distance SFT. (Top) The model is trained on distance QAs, such as object depth sorting and comparison, just using data from synthetic and indoor scenes. **(Middle)** This learned capability transfers in a zero-shot manner to complex reasoning tasks in unseen, in-the-wild scenes, achieving a **36.0%** performance gain over the baseline. **(Bottom)** Furthermore, the skill exhibits cross-level transfer, enabling the model to perform a robotic arm manipulation task with a **27.1%** performance gain.

		L1 Perception					L2 Mental Mapping		L3 Mental Simulation		L4 Agentic Competence	
Methods	**Avg.**	**Geom.**	**Motion**	**Rel.**	**Local.**	**Orient.**	**Understd.**	**Memory**	**Caus. Reas.**	**Seq. Plan.**	**Goal Exec.**	**Open Expl.**
Qwen2.5-VL-7B	27.5	17.8	22.6	23.9	20.6	21.6	34.7	15.2	28.4	39.8	24.5	31.1
Full RL@think	28.2	19.0 (+1.2)	25.2 (+2.7)	23.6 (-0.4)	20.3 (-0.3)	31.5 (-0.1)	30.6 (-4.1)	13.1 (-2.1)	33.6 (+5.2)	26.8 (-13.0)	29.7 (+5.2)	40.0 (+9.0)
After SFT												
SFT Baseline	27.3	28.6	24.6	20.6	26.3	36.0	22.2	22.6	28.2	32.8	23.3	35.9
L1 RL@think	26.6 (-0.7)	28.4 (0.9)	30.0 (+5.4)	30.5 (+9.9)	19.5 (-6.8)	34.5 (-1.4)	24.9 (+2.8)	18.5 (-4.1)	25.7 (-2.4)	34.0 (+1.1)	24.1 (0.8)	29.6 (-6.3)
L2 RL@think	26.7 (-0.5)	24.4 (-4.1)	22.7 (-1.9)	22.3 (+1.6)	17.0 (-9.4)	35.0 (-0.9)	26.6 (+4.5)	16.1 (-6.5)	29.0 (0.8)	31.4 (-1.4)	25.6 (+2.3)	34.5 (-1.4)
L3 RL@think	27.7 (0.4)	16.2 (-12.3)	24.0 (-0.6)	24.1 (+3.5)	21.4 (-4.9)	38.5 (+2.5)	26.0 (+3.9)	21.8 (-0.8)	31.3 (+3.1)	34.7 (+1.9)	26.5 (+3.2)	38.4 (+2.5)
L4 RL@think	28.5 (+1.2)	23.8 (-4.8)	25.3 (0.7)	22.1 (+1.4)	23.5 (-2.8)	33.9 (-2.1)	25.1 (+3.0)	20.5 (-2.2)	32.0 (+3.8)	34.2 (+1.4)	27.1 (+3.9)	38.8 (+2.9)
Full RL@think	30.1 (+2.9)	29.7 (+1.1)	24.7 (0.1)	27.2 (+6.5)	21.0 (-5.3)	34.8 (-1.2)	27.4 (+5.2)	16.7 (-5.9)	33.6 (+5.5)	37.6 (+4.8)	25.4 (+2.1)	41.7 (+5.8)
Full RL@auto-think	30.8 (+3.6)	31.9 (+3.3)	28.6 (+4.0)	22.0 (+1.3)	23.1 (-3.2)	36.8 (0.8)	28.0 (+5.8)	22.6 (-0.1)	33.5 (+5.4)	35.6 (+2.8)	23.4 (0.1)	44.1 (+8.3)

Table 3 RLVR Comparisons. The table compares the baseline Qwen2.5-VL-7B with the version enhanced by RL on Goal-Exec. tasks. Changes are color-coded as notable gains, neutral influence, and drops.

reward mechanism unlocks significant performance gains.

Limitations of Naive RL on Low-Level Skills. We initially conducted experiments using standard GRPO on individual capabilities selected from each level of the SpatialTree. The results exposed a critical limitations of applying naive RL to spatial tasks, particularly regarding generalization in low-level skills. When optimizing exclusively for single low-level abilities, the model tends to overfit to the specific reward signal. This results in siloed improvements that fail to generalize to other foundational skills and, more critically, provide negligible or even detrimental transfer to high-level capabilities.

Ineffectiveness of Uniform Data Mixing. Attempts to mitigate this by combining datasets from multiple levels or mixing all available data for a unified RL stage yielded only marginal gains. The model struggled to balance the diverse requirements of the benchmark, suggesting that a "one-size-fits-all" reinforcement strategy cannot effectively span the spectrum from atomic perception to complex agentic planning.

Hierarchy-Aware Reward Mechanism. These limitations led us to hypothesize that different levels of spatial intelligence require distinct cognitive modes during training. While high-level reasoning benefits from extensive test-time computation, low-level perception is inherently intuitive and should function as a "fast" system. To test this, we introduced a Hierarchy-Aware Reward mechanism that adjusts the training objective based on the capability level:

- **For Intuitive Perception:** For tasks such as depth estimation, object counting, and orientation, we removed rewards for "thinking processes" and introduced a length penalty. This implicitly discourages the model from over-reasoning on direct visual signals, forcing it to rely on direct visual-text alignment.
- **For Complex Reasoning:** For nested tasks like navigation planning and causal reasoning, we retained and amplified rewards for explicit reasoning steps, encouraging the model to utilize more tokens for intermediate computation.

This hierarchy-aware strategy proved highly effective. As shown in Table 3, the model trained with adaptive rewards significantly outperforms both the baseline and the naive GRPO variants across the entire SpatialTree-Bench. This finding strongly validates the structure of our taxonomy: Spatial Intelligence is not a flat collection of tasks, but a structured hierarchy where foundational perception requires direct alignment, while higher-order competence demands deliberate reasoning.

Strict Evaluation Setup. It is important to note two critical factors in our experimental design that ensure the robustness of our findings:

- Data Decontamination: The robotic arm data samples used for GRPO training are strictly separated from the SpatialTree-Bench testing data. There is no overlap in specific scenes or object configurations between the training and evaluation sets.
- Task and Metric Discrepancy: The training objective is purely maximizing the reward on discrete MCQ selection. In contrast, the SpatialTree-Bench evaluation employs a diverse set of continuous and semantic metrics (e.g., Mean Squared Error for distance, angular error for orientation, and execution success rates for agentic tasks).

Results and Observations. Despite the significant domain gap and the difference in task formulation, the GRPO-tuned Qwen2.5-VL-7B demonstrates notable improvements across multiple levels of the SpatialTree hierarchy compared to its base counterpart. This suggests that the model is not merely memorizing dataset-specific patterns, but is effectively internalizing generalized spatial reasoning policies through the reinforcement learning process. These preliminary findings highlight the potential of RLVR as a scalable pathway for advancing spatial intelligence in MLLMs.

7 Conclusion and Future works

We present SpatialTree, the first capability-centric framework for Spatial Intelligence, organizing abilities into four hierarchical layers. This structure enables analysis of how spatial abilities emerge, compose, and transfer across levels. It also opens opportunities to efficiently scale up spatial intelligence in MLLMs, by strategically leveraging different types of data: identifying which abilities are most effective for pre-training, which can be directly applied in reinforcement learning with minimal additional reasoning data during post-training, and which are acquired through real-world interactions. We believe this could provide a promising path toward advancing spatial intelligence in MLLMs.

References

[1] 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.

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. arXiv preprint arXiv:2309.16609, 2023.

[3] 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.

[4] 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.

[5] 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.

[6] 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. Advances in neural information processing systems, 2020.

[7] 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. International Conference on 3D Vision (3DV), 2017.

[8] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261, 2025.

[9] 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. Springer, 2024.

[10] Google Gemini Team. Gemini 3 flash model card. Technical report, Google DeepMind, 2026. URL https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-Flash-Model-Card.pdf.

[11] Google Gemini Team. Gemini 3 pro model card. Technical report, Google DeepMind, 2026. URL https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-Pro-Model-Card.pdf.

[12] Mohsen Gholami, Ahmad Rezaei, Zhou Weimin, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. arXiv preprint arXiv:2509.06266, 2025.

[13] 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.

[14] 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.

[15] Mary Hegarty. Components of spatial intelligence. In Psychology of learning and motivation, volume 52, pages 265–297. Elsevier, 2010.

[16] Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. arXiv e-prints, pages arXiv–2507, 2025.

[17] 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. Advances in Neural Information Processing Systems, 36: 20482–20494, 2023.

[18] Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. Egodex: Learning dexterous manipulation from large-scale egocentric video, 2025. URL https://arxiv.org/abs/2505.11709.

[19] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. arXiv preprint arXiv:2410.21276, 2024.

[20] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. \π_{0.5}: a vision-language-action model with open-world generalization. arXiv preprint arXiv:2504.16054, 2025.

[21] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. arXiv preprint arXiv:2412.16720, 2024.

[22] 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.

[23] 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. Advances in Neural Information Processing Systems, 37:48955–48970, 2024.

[24] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. arXiv preprint arXiv:2403.12945, 2024.

[25] Benjamin Kuipers. Modeling spatial knowledge. Cognitive science, 2(2):129–153, 1978.

[26] Benjamin Kuipers. The spatial semantic hierarchy. Artificial intelligence, 119(1-2):191–233, 2000.

[27] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326, 2024.

[28] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. arXiv preprint arXiv:2407.07895, 2024.

[29] Xiaodi Li, Pan Xie, Yi Ren, Qijun Gan, Chen Zhang, Fangyuan Kong, Xiang Yin, Bingyue Peng, and Zehuan Yuan. Infinityhuman: Towards long-term audio-driven human. arXiv preprint arXiv:2508.20210, 2025.

[30] Zhiqiu Lin, Siyuan Cen, Daniel Jiang, Jay Karhade, Hewei Wang, Chancharik Mitra, Tiffany Ling, Yuhan Huang, Sifan Liu, Mingyu Chen, et al. Towards understanding camera motions in any video. arXiv preprint arXiv:2504.15376, 2025.

[31] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. Advances in neural information processing systems, 36:34892–34916, 2023.

[32] Parker Liu, Chenxin Li, Zhengxin Li, Yipeng Wu, Wuyang Li, Zhiqin Yang, Zhenyuan Zhang, Yunlong Lin, Sirui Han, and Brandon Y Feng. Ir3d-bench: Evaluating vision-language model scene understanding as agentic inverse rendering. arXiv preprint arXiv:2506.23329, 2025.

[33] Weichen Liu, Qiyao Xue, Haoming Wang, Xiangyu Yin, Boyuan Yang, and Wei Gao. Spatial reasoning in multimodal large language models: A survey of tasks, benchmarks and methods. arXiv preprint arXiv:2511.15722, 2025.

[34] Xinhang Liu, Yuxi Xiao, Donny Y Chen, Jiashi Feng, Yu-Wing Tai, Chi-Keung Tang, and Bingyi Kang. Trace anything: Representing any video in 4d via trajectory fields. arXiv preprint arXiv:2510.13802, 2025.

[35] Hongchen Luo, Wei Zhai, Jing Zhang, Yang Cao, and Dacheng Tao. Learning affordance grounding from exocentric images. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2252–2261, 2022.

[36] Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Celso M de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. arXiv preprint arXiv:2412.07825, 2024.

[37] Xiaofeng Mao, Shaoheng Lin, Zhen Li, Chuanhao Li, Wenshuo Peng, Tong He, Jiangmiao Pang, Mingmin Chi, Yu Qiao, and Kaipeng Zhang. Yume: An interactive world generation model. CoRR, abs/2507.17744, 2025.

[38] Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. Spatiallm: Training large language models for structured indoor modeling. arXiv preprint arXiv:2506.07491, 2025.

[39] Nora Newcombe and Janellen Huttenlocher. Making space: The development of spatial representation and reasoning. MIT press, 2000.

[40] OpenAI. Openai api documentation. https://platform.openai.com/docs/models/gpt-3-5, 2023. Accessed on September 19, 2025.

[41] OpenAI. Gpt-4v(ision) system card, 2023. URL https://cdn.openai.com/papers/GPTV_System_Card.pdf. Accessed: 2025-09-19.

[42] Jean Piaget. The construction of reality in the child. Routledge, 2013.

[43] Yujia Qin, Yining Ye, Junjie Fang, Haoming Wang, Shihao Liang, Shizuo Tian, Junda Zhang, Jiahao Li, Yunxin Li, Shijue Huang, Wanjun Zhong, Kuanye Li, Jiale Yang, Yu Miao, Woyu Lin, Longxiang Liu, Xu Jiang, Qianli Ma, Jingyu Li, Xiaojun Xiao, Kai Cai, Chuang Li, Yaowei Zheng, Chaolin Jin, Chen Li, Xiao Zhou, Minchao Wang, Haoli Chen, Zhaojian Li, Haihua Yang, Haifeng Liu, Feng Lin, Tao Peng, Xin Liu, and Guang Shi. UI-TARS: pioneering automated GUI interaction with native agents. CoRR, abs/2501.12326, 2025.

[44] 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 International Conference on Computer Vision, 2021.

[45] Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Ángel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. IEEE, 2021.

[46] Bytedance Seed. Seed1. 8 model card: Towards generalized real-world agency.

[47] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. arXiv preprint arXiv:2402.03300, 2024.

[48] Shenna Shepard and Douglas Metzler. Mental rotation: effects of dimensionality of objects and type of task. Journal of experimental psychology: Human perception and performance, 14(1):3, 1988.

[49] 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.

[50] Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. arXiv preprint arXiv:2504.07491, 2025.

[51] Edward C Tolman. Cognitive maps in rats and men. Psychological review, 55(4):189, 1948.

[52] 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. Springer, 2024.

[53] Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Yixuan Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. In The Thirty-Eighth Annual Conference on Neural Information Processing Systems, 2024.

[54] Siting Wang, Luoyang Sun, Cheng Deng, Kun Shao, Minnan Pei, Zheng Tian, Haifeng Zhang, and Jun Wang. Spatialviz-bench: Automatically generated spatial visualization reasoning tasks for mllms. arXiv preprint arXiv:2507.07610, 2025.

[55] 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.

[56] 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 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 4909–4916. IEEE, 2020.

[57] Zehan Wang, Ziang Zhang, Tianyu Pang, Chao Du, Hengshuang Zhao, and Zhou Zhao. Orient anything: Learning robust object orientation estimation from rendering 3d models. arXiv:2412.18605, 2024.

[58] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 20406–20417, 2024.

[59] Yuxi Xiao, Jianyuan Wang, Nan Xue, Nikita Karaev, Yuri Makarov, Bingyi Kang, Xing Zhu, Hujun Bao, Yujun Shen, and Xiaowei Zhou. Spatialtrackerv2: 3d point tracking made easy. arXiv preprint arXiv:2507.12462, 2025.

[60] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In European Conference on Computer Vision, pages 131–147. Springer, 2024.

[61] Runsen Xu, Weiyao Wang, Hao Tang, Xingyu Chen, Xiaodong Wang, Fu-Jen Chu, Dahua Lin, Matt Feiszli, and Kevin J Liang. Multi-spatialmllm: Multi-frame spatial understanding with multi-modal large language models. arXiv preprint arXiv:2505.17015, 2025.

[62] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. arXiv preprint arXiv:2505.09388, 2025.

[63] 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 Proceedings of the Computer Vision and Pattern Recognition Conference, pages 10632–10643, 2025.

[64] 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/CVF conference on computer vision and pattern recognition, pages 10371–10381, 2024.

[65] 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.

[66] Rui Yang, Ziyu Zhu, Yanwei Li, Jingjia Huang, Shen Yan, Siyuan Zhou, Zhe Liu, Xiangtai Li, Shuangye Li, Wenqian Wang, Yi Lin, and Hengshuang Zhao. Visual spatial tuning. arXiv preprint arXiv:2511.05491, 2025.

[67] 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.

[68] 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. arXiv preprint arXiv:2506.21458, 2025.

[69] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, April 2024. URL https://llava-vl.github.io/blog/2024-04-30-llava-next-video/.

[70] 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.

Appendix

Contents

1 Introduction **1**

2 Related Work **3**

3 The SpatialTree Taxonomy **4**
- 3.1 **Perception** 4
- 3.2 **Mental Mapping** 5
- 3.3 **Mental Simulation** 5
- 3.4 **Agentic Competence** 6

4 Instantiating the SpatialTree Benchmark **6**
- 4.1 Data Curation 6
- 4.2 Evaluation Metrics Designs 7

5 A Hierarchical Analysis of Spatial Capabilities **7**
- 5.1 Models and Metrics 8
- 5.2 Overall Performance 8

6 Exploring Ability Dependencies and Hierarchical Transfer **8**
- 6.1 Probing Cross-Ability Transfer via SFT 9
- 6.2 Reinforcement Learning 10

7 Conclusion and Future works **12**

A Visualization of Data Sources **18**

B Evaluation Metrics Details **18**

C SpatialPlus: Complementary Data Annotations for SpatialTree **19**
- C.1 Orientations (L1) 19
- C.2 Agentic Competence (L4) 20

D Ability Transfer via Prompting **24**

E Benchmark Metric Aggregation **24**

F More Visualizations for QAs in SpatialTree Bench **25**

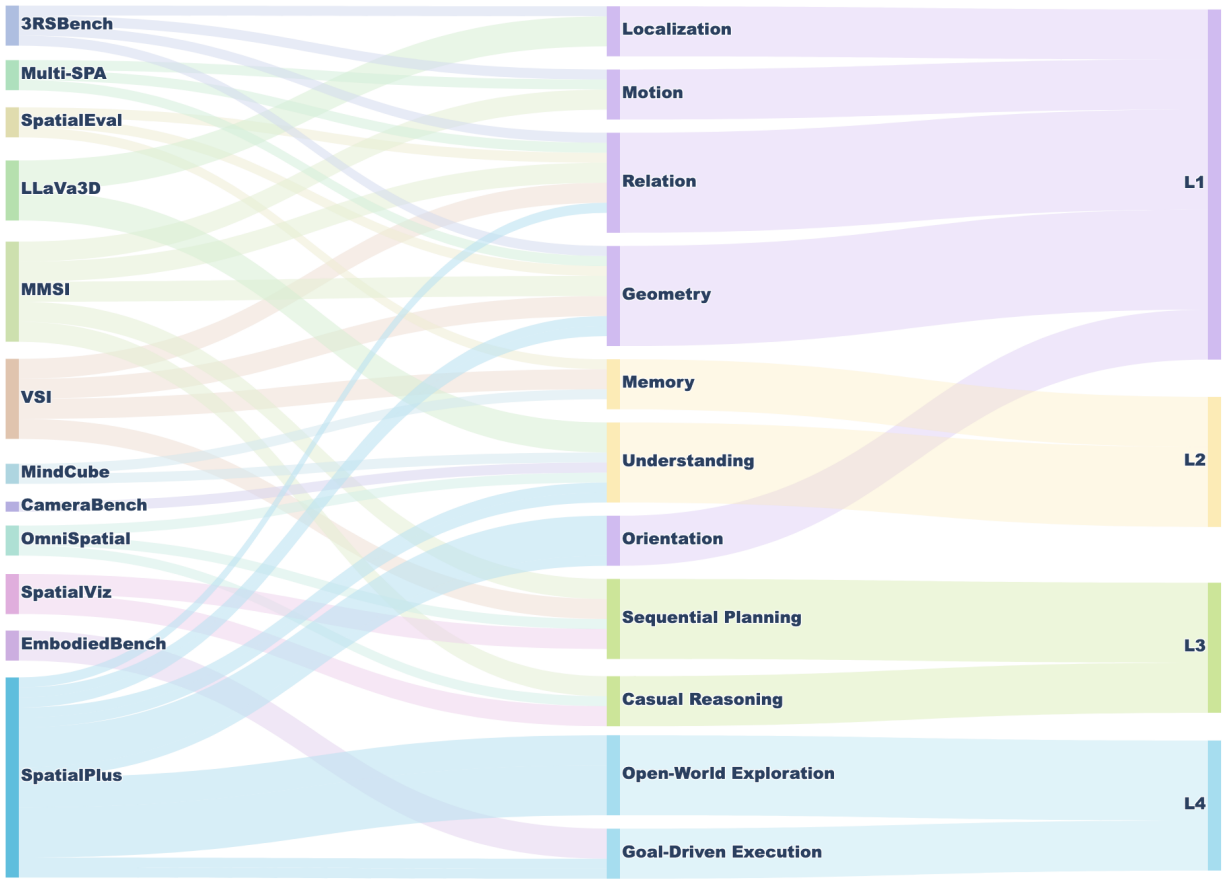


Figure A Construction of SpatialTree-Bench. We build our benchmark by reorganizing various existing datasets and mapping them to our capability tree, where **SpatialPlus**, a complementary dataset are introduced to ensure the capability coverage.

A Visualization of Data Sources

How different datasets contribute to our SpatialTree evaluation is shown in Fig. A.

B Evaluation Metrics Details

Multi-Option QAs. For multi-option question answering, each model is evaluated on its ability to select the correct option from a predefined set. We measure accuracy by comparing the predicted choice against the ground-truth answer. This paradigm captures a model's understanding of spatial relations, object properties, and causal dynamics within a scene, corresponding to the low- and mid-level capabilities in the SpatialTree (L1–L3).

Numeric QAs. Numeric QAs require models to predict continuous quantities such as distances, angles, or 3D coordinates. We evaluate performance using relative error metrics, for example:

$$\text{Relative Error} = \frac{|\hat{y} - y|}{|y|},$$

where $\hat{y}$ is the predicted value and y is the ground truth. This metric ensures that predictions are scaled appropriately across different magnitudes and emphasizes precision in spatial reasoning.

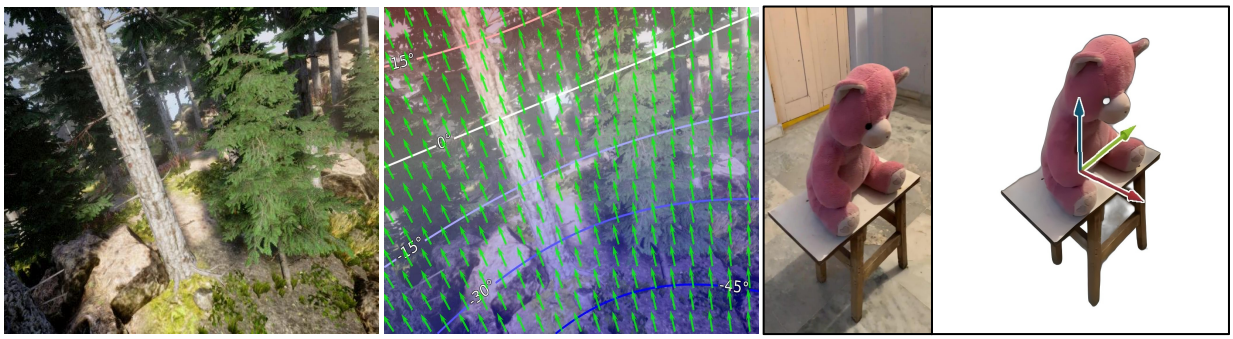


Figure B Orientation Annotations. The left side is the gravity field estimated from GeoCalib [52], while the right side is from OrientAnything.

GPT Judge. For tasks that are open-ended or involve complex reasoning (e.g., trajectory description, action sequence explanation), we leverage a GPT-based judge to assess correctness. The judge evaluates whether the generated response satisfies the task requirements, optionally scoring partial correctness. This approach allows flexible evaluation beyond rigid numeric or multiple-choice formats, especially for mid- and high-level capabilities in L3–L4.

Agentic Evaluation. To assess agentic competence, models are deployed in interactive simulated environments, such as those provided by EmbodiedBench [65]. We evaluate navigation and manipulation tasks along multiple dimensions: success rate in completing the target goal, relative translation accuracy, and directional alignment. For each action step, a combined metric is computed using relative distance and cosine similarity of movement vectors, producing a normalized score in $[0, 1]$. Aggregating scores over all steps yields a comprehensive measure of an agent's ability to plan and execute actions in long-horizon, embodied tasks.

C SpatialPlus: Complementary Data Annotations for SpatialTree

C.1 Orientations (L1)

The Orientation capability, a fundamental yet under-explored area, involves estimating both gravity direction and 3D object orientation. To generate annotations, we leveraged Geocalib [52] for gravity vector estimation and OrientAnything [57] for object poses. We applied these tools to datasets suited for each task: for gravity, we annotated 500 images sampled from the diverse, drone-captured TartanAir [56] dataset; for object orientation, we utilized the object-centric Co3dv2 [44] dataset (Seen in Fig. B). For gravity, the goal is to estimate the camera's orientation relative to the gravity vector, typically represented by the *pitch* and *roll* angles. Formally, let the gravity vector in the world frame be:

$$\mathbf{g}_w = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \tag{1}$$

and let $\mathbf{R}_{cw} \in SO(3)$ denote the rotation from the world frame to the camera frame. The gravity direction in the camera frame is then:

$$\mathbf{g}_c = \mathbf{R}_{cw}\, \mathbf{g}_w. \tag{2}$$

From $\mathbf{g}_c = [g_x, g_y, g_z]^\top$, the pitch and roll angles can be computed as:

$$\text{pitch} = \arctan 2(-g_x, \sqrt{g_y^2 + g_z^2}), \tag{3}$$

$$\text{roll} = \arctan 2(g_y, g_z). \tag{4}$$

Here, *pitch* measures the forward–backward tilt of the camera, while *roll* measures the sideways tilt. To evaluate an MLLM's proficiency in this task, we require the model to analyze the input image and return

these same three parameters in a structured JSON format. An example of our prompt template is shown in Fig. C.

```
{
  "role": "system",
  "content": "You are a vision model specialized in estimating camera orientation from images. Your task
is to infer the gravity direction from the input image by predicting the camera's pitch and roll angles,
as well as the vertical field of view (vFOV). Always output your prediction strictly in the following
JSON format:
  {
    \"pitch\": <float, camera pitch angle in degrees>,
    \"roll\": <float, camera roll angle in degrees>,
    \"vfov\": <float, vertical field of view in degrees>
  }
  "Do not include any additional text or explanation outside of the JSON object."
}
```

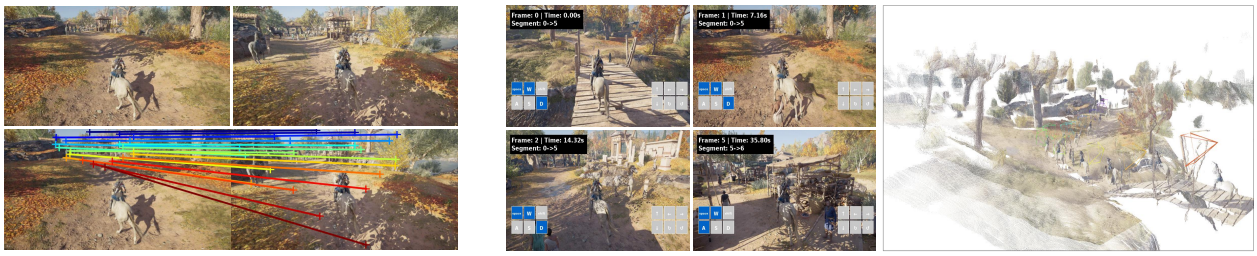
Figure C Prompt template for Orientation Estimation.

For evaluation, we move beyond a simple absolute error metric and adopt a probabilistic approach that accounts for the inherent uncertainty of the ground-truth annotations provided by *Geocalib*. For each predicted parameter (pitch, roll, and vFOV), *Geocalib* also outputs an uncertainty value, which we interpret as the standard deviation (σ_{gt}). We then calculate a normalized similarity score (S) for each parameter using a Gaussian kernel, defined as:

$$S(y_{\text{pred}}, y_{\text{gt}}, \sigma_{\text{gt}}) = \exp\left(-\frac{(y_{\text{pred}} - y_{\text{gt}})^2}{2\sigma_{\text{gt}}^2}\right) \tag{5}$$

where y_{pred} is the MLLM's prediction, y_{gt} is the ground-truth value from *Geocalib*, and σ_{gt} is its associated uncertainty. This scoring function gracefully penalizes deviations from the ground truth: the score is 1 for a perfect match and decays towards 0 as the error increases. Crucially, a larger uncertainty σ_{gt} in the ground truth leads to a slower decay, making the scoring more lenient when the ground truth itself is less certain. The final score for the task is the average of the individual scores for pitch, roll, and vFOV. For object orientation estimation, most of metrics are similar to gravity, and the evaluation are conducted on Azimuth, Polar and Rotation these three angles.

C.2 Agentic Competence (L4)



(a) (b)

Figure D Navigation Data Curation. (a) shows paired images used for evaluation, where MLLMs are expected to move from left to right. (b) illustrates our curation process: reconstructing metric 3D models and camera trajectories, then converting them into actions.

Spatial Action Mapping. In the context of spatial agents, navigation and manipulation represent the most common forms of interaction within 3D environments. We address each with a distinct action space design. For navigation, we conceptualize agent movement as a series of camera motion controls (referring to recent video world models [4, 37]). To enable precise and intuitive control, we decompose complex camera movements

Table A Spatial Action Mapping. This table defines a standardized interface that maps continuous 6-DoF motions and discrete control signals into action primitives with unified parameterization, enabling MLLMs to plan and execute embodied behaviors for agentic competence evaluation.

Primitive	Primitive Term	Category	Description	Action Mapping	Param.	Threshold
P_{truck}	Truck	Translation	Move camera left/right (X-axis)	A/D	v_x	± 0.01 m/s
P_{dolly}	Dolly	Translation	Move camera forward/backward (Z-axis)	W/S	v_z	± 0.01 m/s
P_{pedestal}	Pedestal	Translation	Move camera up/down (Y-axis)	Q/E	v_y	± 0.01 m/s
P_{pan}	Pan	Rotation	Turn camera left/right (yaw)	$\leftarrow / \rightarrow$	ω_y	$\pm 0.5^\circ$/s
P_{tilt}	Tilt	Rotation	Tilt camera up/down (pitch)	$\uparrow / \downarrow$	ω_x	$\pm 0.5^\circ$/s
P_{roll}	Roll	Rotation	Roll camera CW/CCW (roll)	Z/X	ω_z	$\pm 0.5^\circ$/s
O_{gripper}	Gripper	Gripper Control	Open or close the gripper	G/H	State $\in \{0, 1\}$	N/A
$O_{\text{push/pull}}$	Push/Pull	Gesture	Push or pull object along forward axis	P/L	Dir. $\in \{-1, +1\}$	N/A
O_{grab}	Grab	Gesture	Grab or release object	None	State $\in \{0, 1\}$	G/H

into a set of fundamental motion primitives inspired by established cinematography techniques. This approach allows us to translate high-level language instructions (e.g., "move to the left," "look up") into a structured, low-level action space. The six fundamental primitives, their corresponding cinematic terms, degrees of freedom (DoF), and parameterization are detailed in Tab. A.

Formally, the camera trajectories are defined with a series of Camera-to-World (C2W) transformation matrices $\mathcal{T}_{\text{motion}} = \{\mathbf{T}|_0^i, i = 0, 1, \ldots, t\}$, while the camera transformation at each moment is $\mathcal{T}_{i \to i+1} = \mathbf{T}_{i+1}\mathbf{T}_i^{-1},\ i = 0, 1, \ldots, t-1$. Then the continuous camera transformation can be decomposed into different components corresponding to each motion primitive, and discretized into the navigation action A_{nav} using a speed threshold:

$$\begin{aligned} \mathbf{A}_i^{\text{nav}} = \mathbf{T}_{i \to i+1} &= \{\Delta\mathbf{R}, \Delta\mathbf{t}\} \\ &\approx \left\{ \begin{array}{l} t_i \cdot v_i,\ t_k \cdot \omega_k \mid i, k \in \{x, y, z\}, \\ t_i, t_k \in \mathbb{Z}_{\geq 0} \end{array} \right\} \end{aligned} \tag{6}$$

where $\Delta\mathbf{R} = (\Delta R_x, \Delta R_y, \Delta R_z)$ represents the rotation components obtained via Euler decomposition, $\Delta\mathbf{t} = (\Delta t_x, \Delta t_y, \Delta t_z)$ denotes the translation components along the x, y, and z axes, and t_i, t_k are discrete integers ranging from 0 up to the video frame rate (FPS). For manipulation, we focus on two representative scenarios to simplify the problem and enable controlled evaluation: human-hand manipulation and robotic gripper manipulation. For the gripper setting, we include gripper open/close actions along with wrist-level 6-DoF motion. For the human-hand setting, we define a small set of intuitive gesture primitives (i.e., push, pull, grab) seen in Tab. A that capture essential interaction patterns. These manually defined mappings create a unified yet tractable action space for analyzing MLLMs' planning and manipulation competence.

Building on the proposed spatial action mapping, we curate annotated data from diverse sources, including human-hand manipulation videos, navigation video games, robotic arm manipulation datasets, and simulation environments. This unified dataset enables us to evaluate whether MLLMs can accurately plan and execute actions in the defined metric action space. Further implementation details are provided in Sec. 4 and in the experimental section.

Goal-driven Navigation. We leverage our SpatialEngine to get the action annotations as shown in Fig. D. We first extract the metric pose trajectories from the games videos, and convert them into discrete actions with our spatial action mapping, and then we randomly sample several image pairs from the video with the

correspondence checking. For evaluation, the goal is a image, and the MLLMs are supposed to control the character to move to the target positions. We use the prompt template as below:

```
{
  "role": "system",
  "content": "Task Details:\n
  "Analyze Images: Compare the start image <Image 1> and the target image <Image 2> to understand the
required translation and rotation for the robot arm's end-effector.\n"
  "Define Motion: Decompose the movement into 6 steps, each containing one or more elementary
actions.\n"
  "Quantify Actions: For each action, specify an integer step_num that represents its intensity.\n\n"
  "Coordinate System:\n"
  "Right-hand frame attached to the end-effector: +Z forward, +X right, +Y downward.\n\n
  "Action Space:\n"
  "Translation: Dolly In (W), Dolly Out (S), Truck Left (A), Truck Right (D), Pedestal Up (space),
Pedestal Down (shift).\n"
  "Rotation: Pan Left (left arrow), Pan Right (right arrow), Tilt Up (up arrow), Tilt Down (down arrow),
Roll CW (clockwise), Roll CCW (counterclockwise).\n"
  "Special Action: Stay (STOP).\n\n"
  "Step Size:\n"
  "Translation: 0.019375 m/step. Rotation: 0.4509 rad/step.\n\n"
  "Output Format:\n"
  "Return a single JSON object with keys step_1-step_6. Each step contains:\n"
    "actions: list of action symbols\n"
    "step_nums: corresponding integers.\n\n"
  "Example:\n
  {
    \"step_1\": {
      \"actions\": [\"W\", \"A\"],
      \"step_nums\": [5, 2]
    },
    \"step_2\": {
      \"actions\": [\"W\", \"up_arrow\"],
      \"step_nums\": [3, 4]
    }
  }"
}
```

Figure E Prompt of navigation.

In this prompt, translation and rotation steps are computed from the actual movement, while capping the number of steps at 10 to prevent overly long action sequences. To evaluate MLLMs, we compute a normalized metric in the range $[0, 1]$ by combining **relative distance** and **directional accuracy**. Specifically, for each step, let $\Delta\mathbf{p}_{\text{pred}}$ and $\Delta\mathbf{p}_{\text{gt}}$ denote the predicted and ground-truth translation vectors, respectively.

The **relative distance score** is defined as:

$$s_d = \max\left(0, 1 - \frac{\|\Delta\mathbf{p}_{\text{pred}} - \Delta\mathbf{p}_{\text{gt}}\|}{\|\Delta\mathbf{p}_{\text{gt}}\|}\right),$$

and the **directional score** is computed by the cosine similarity:

$$s_\theta = \frac{\Delta\mathbf{p}_{\text{pred}} \cdot \Delta\mathbf{p}_{\text{gt}}}{\|\Delta\mathbf{p}_{\text{pred}}\| \, \|\Delta\mathbf{p}_{\text{gt}}\|}.$$

The final step-wise accuracy is then: $s_{\text{step}} = s_d \cdot \max(0, s_\theta)$

which ensures a value in $[0, 1]$, where 1 indicates perfect alignment in both distance and direction. Aggregating s_{step} across all steps provides a comprehensive measure of the model's precision in executing end-effector motions.

Goal-driven Manipulation For the **Goal-Driven Manipulation** capability, we utilize action annotations from the `Droid` [24] and `EgoDex` [18] datasets. This task requires the MLLM to generate a sequence of precise actions to move a robot end-effector or a human hand from a starting state to a target state, both specified by images. The action space for `Droid` encompasses 7-DoF control: 6-DoF for the end-effector's pose (translation and rotation) and a binary state for the gripper (open/close). A similar action space is adapted for `EgoDex`, controlling wrist pose and finger grip. The MLLM is prompted to generate a sequence of continuous action vectors, as shown in the template below:

```
{
  "role": "system",
  "content": "Task Details:\n
  "Compare the start image <Image 1> and target image <Image 2> to infer the translation and rotation
required for the robot arm's end-effector.\n"
  "Decompose the motion into up to 6 steps, each combining any number of elementary actions.\n\n"
  "Action Space:\n"
  "We define a 7D action vector per step:\n"
  "[dx, dy, dz, d_roll, d_pitch, d_yaw, gripper_state]\n"
  "- Translation (dx, dy, dz): Displacement in meters along +X, +Y, +Z.\n"
  "- Rotation (d_roll, d_pitch, d_yaw): Rotation in radians about +Z, +X, +Y respectively.\n"
  "- gripper_state: 0=open, 1=closed.\n\n"
  "Each dx, dy, dz, d_roll, d_pitch, d_yaw is computed from selected actions and their step_nums:\n"
  "delta_q = step_num * unit_step_size  (translation in meters or rotation in radians)\n\n"
  "Available Actions:\n"
  "W/S: Dolly In/Out (+/-Z)\n"
  "A/D: Truck Left/Right (-/+X)\n"
  "space/shift: Pedestal Up/Down (-/+Y)\n"
  "left_arrow/right_arrow: Pan Left/Right (+/- yaw)\n"
  "up_arrow/down_arrow: Tilt Up/Down (+/- pitch)\n"
  "clockwise/counterclockwise: Roll CW/CCW (+/- roll)\n"
  "STOP: No movement\n\n"
  "Output Format:\n"
  "Return a single JSON object where each step is a key (\"step_1\", \"step_2\", ...).\n"
  "Each step contains:\n"
  "- actions: a list of action symbols\n"
  "- step_nums: a list of integers specifying intensity (1-10)\n"
  "- gripper: 0 or 1 for gripper state\n\n"
  "Example:\n"
  {
    \"step_1\": {
      \"actions\": [\"W\", \"A\"],
      \"step_nums\": [5, 2],
      \"gripper\": 0
    },
    \"step_2\": {
      \"actions\": [\"clockwise\"],
      \"step_nums\": [3],
      \"gripper\": 1
    }
  }
}
```

Figure F Prompt for Goal-Driven Manipulation with 7D Action Representation.

To evaluate the MLLM's performance, we assess the accuracy of the predicted action sequence against the ground truth. For the translational component of the motion, we reuse the step-wise accuracy metric s_{step} from the navigation task, which combines relative distance and directional scores. For the rotational component, we compute a normalized score based on the angular difference between the predicted orientation and the ground truth. Let R_{pred} and R_{gt} be the predicted and ground-truth rotation matrices for a step. The rotational error

angle θ_{err} is calculated from the error rotation matrix $R_{\text{err}} = R_{\text{pred}} R_{\text{gt}}^T$:

$$\theta_{\text{err}} = \arccos\left(\frac{\text{Tr}(R_{\text{err}}) - 1}{2}\right).$$

The **rotation score** s_{rot} is then defined as:

$$s_{\text{rot}} = \max\left(0, 1 - \frac{\theta_{\text{err}}}{\pi}\right),$$

which normalizes the error to a $[0, 1]$ range, where 1 indicates a perfect rotational match. Finally, the **gripper score** s_{gripper} is a binary accuracy (1 if the predicted state matches the ground truth, 0 otherwise). The final score for each step is a weighted combination of these three metrics, providing a holistic evaluation of the model's ability to perform precise, multi-faceted manipulation tasks.

D Ability Transfer via Prompting

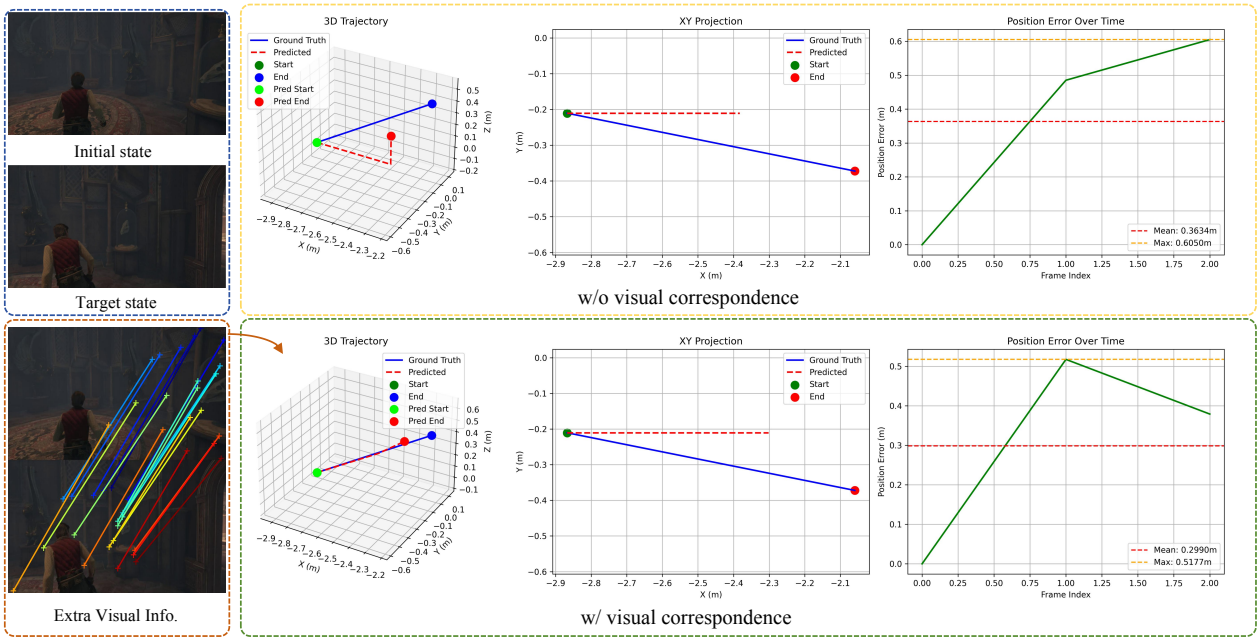


Figure G Correspondence Prompting for Navigation. The correspondence prompt guides Gemini2.5-pro to navigate and move more accurately within 3D environments.

In addition to SFT, we investigate cross-level ability influence through explicit prompting. Specifically, we consider a representative task pair: low-level abilities (L1.Corr, L1.Dist, L1.Size) and a high-level task (L4.Imaged Goal Navigation). Intuitively, correspondence is a necessary component for navigation. Using Gemini2.5-pro, we provide models with explicit prompts derived from matching visualizations, depth, and size context. As shown in Fig. G, correspondence guidance improves target direction recognition, increasing accuracy by **7.1%**, while distance and size prompting yield gains of **5.5%** and **2.1%**, respectively. These results suggest that grounding MLLM reasoning with explicit low-level visual information can substantially enhance performance on complex spatial navigation tasks.

E Benchmark Metric Aggregation

To derive a single, comprehensive score for a model's spatial intelligence, we employ a hierarchical aggregation methodology. This approach is designed to reflect the complex, multi-layered nature of spatial cognition, rather than treating all abilities as equally important. The design is principally guided by established theories

in cognitive psychology, which posit that spatial intelligence is constructed hierarchically, with fundamental perceptual skills forming the bedrock for more abstract reasoning and planning.

Our aggregation framework is built upon the *SpatialTree* structure. The assignment of weights within this tree is determined by a synthesis of theoretical principles and empirical, data-driven insights:

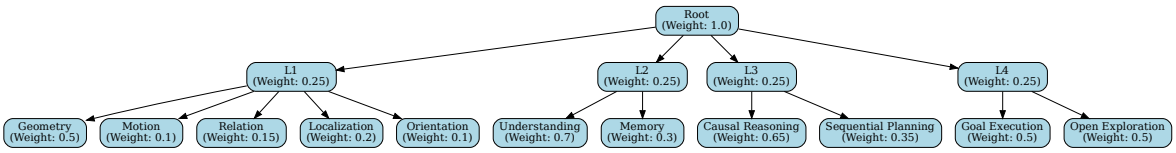


Figure H An illustration of the hierarchical weighting scheme for metric aggregation with in the *SpatialTree*. Each node represents a capability layer, with the assigned weight used for the bottom-up calculation of the final score. The weighting prioritizes foundational perceptual abilities (L1) as they are prerequisites for higher-level cognitive tasks.

Cognitive Hierarchy. In line with cognitive science literature, our weighting scheme prioritizes foundational capabilities, as shown in Fig. H. The L1 layer, which represents low-level spatial perception, is assigned the largest weight, as these skills are prerequisites for almost all higher-level spatial tasks found in L2 (Mental Mapping) and L3 (Mental Simulation).

Empirical Dependency from Correlation Analysis. The theoretical hierarchy is further refined and validated by our empirical findings from the Pearson correlation heatmap (Fig. 5). The heatmap allows us to identify *atomic abilities* that exhibit strong, widespread correlations with a multitude of other skills. These influential abilities are considered more fundamental to the overall spatial intelligence network and are consequently assigned higher weights within their respective sub-trees. This ensures our metric is not just theoretically sound, but also reflects the actual dependencies observed in model performance.

The final score is calculated via a bottom-up, weighted summation. The performance score for any parent node in the tree is the weighted sum of the scores of its immediate children. This process is recursively applied until the root node is reached, yielding a single, principled score that holistically quantifies the spatial intelligence of a given MLLM.

F More Visualizations for QAs in SpatialTree Bench

We declare that Large Language Models (LLMs) were used in a limited capacity during the preparation of this manuscript. Specifically, LLMs were employed for grammar checking, word choice refinement, and typo correction. All core technical contributions, experimental design, analysis, and conclusions are entirely our own. The use of LLMs did not influence the scientific methodology, result interpretation, or theoretical contributions of this research.

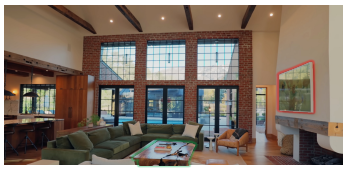
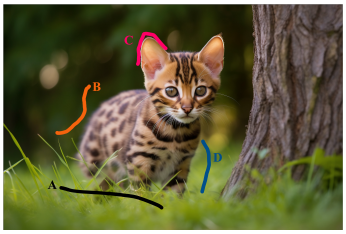
Geometry	
	
Prompt	Which is wider, the width of the painting on the wall or the width of the wooden table next to the sofa?
Answer	The wooden table
Prompt	The coordinates [x , y] are normalized to 0-1 and scaled by 1000, with [0 , 0] at the top-left. The x-axis represents the width, and the y-axis represents the height. What is the depth at the coordinates [389 , 180] in the image (in mm)?
Answer	1915
	
Prompt	Which line is closer to the edge?
Answer	C

Table B Examples of *L1.Geometry*.

Relation	
Prompt	When you were taking the photo in Image 1, where was the exit of the room relative to you?
Answer	Rear left
Prompt	Track [150, 470] from Image 1 to its correspondence in Image 2.
Answer	C

Table C Examples of *L1.Relation*.

Orientation	
Prompt	Analyze the image to determine the vertical field of view (vfov) and calculate the camera'against the vertical axis (gras roll and pitch angles vity).

Answer	`[`
	`"roll_unc": 0.61629718542099,`
	`"pitch_unc": 1.862020492553711,`
	`"vfov_unc": 20.077800750732422`
	`]`

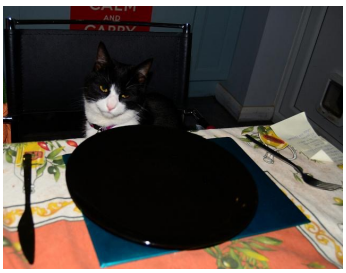
	
Prompt	If I stand at the cat's position facing where it is facing, is the knife in front of me or behind me?
Answer	In front of

Table D Examples of *L1.Orientation*.

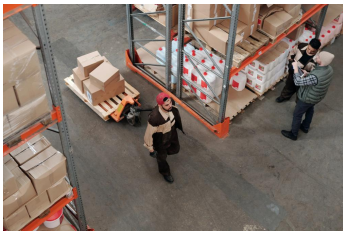
Motion	
Prompt	As shown in the video, which direction is the view moving towards?
Answer	C
Prompt	From view 1 to view 2, in which direction is the race car moving?
Answer	First to the front left, then to the front right.

Table E Examples of *L1.Motion*.

Localization	
Prompt	Please ground mirror in this image. The 3D bounding box is defined in the format: `"bbox_3d": [u_center, v_center, z_center, x_size, y_size, z_size, roll, pitch, yaw]`

Answer	`[`
	` "bbox_3d": [901, 558, 4.87, 0.541, 1.698, 0.243, 179.279, -29.539, 176.489]`
	`]`

Table F Examples of *L1.Localization*.

Understanding	
	

Prompt	From the man wearing the red hat, how many people are on his left?
	`[`
	`A. zero,`
	`B. one,`
	`C. three,`
	`D. two`
	`]`

Answer	D
Prompt	To ride this bicycle along the seawall, where would you place your hands to steer, where would you sit, and where would you put your feet to pedal?
Answer	`[744, 439, 783, 489]`

Table G Examples of *L2.Understanding*.

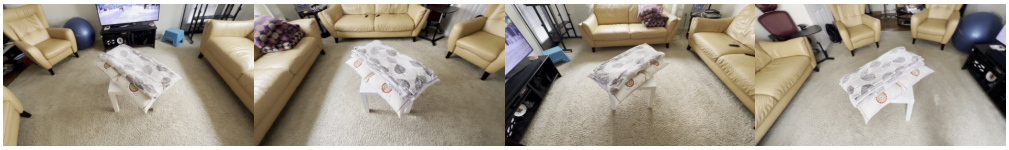
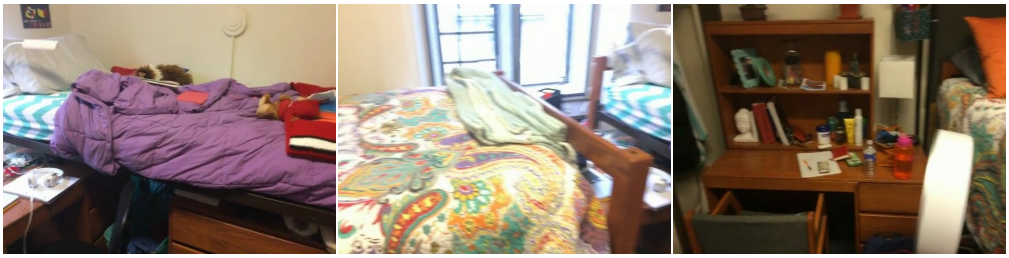
Memory	
	
Prompt	From the man wearing the red hat, how many people are on his left? [A. Leather loveseat with three seat cushions, B. TV, C. Two single sofas, D. Leather loveseat]
Answer	B
	
Prompt	How many bed(s) are in this room?
Answer	2

Table H Examples of *L2.Memory*.

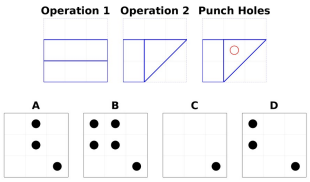
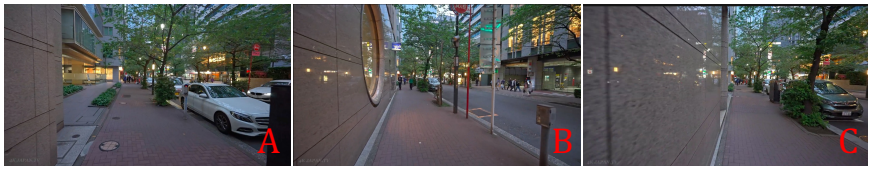
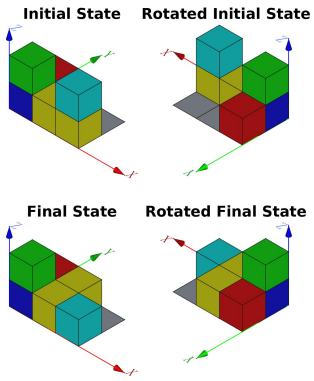
Causal Reasoning	
Prompt	If the dog on the right reaches the camera in 5 s, what is its speed? [A. 14.7m/s B. 1.9m/s C. 21.7m/s D. 11.9m/s]
Answer	B
	
Prompt	A 3x3 grid paper undergoes two folds as shown. A hole is punched in the folded state. Which option (A, B, C, or D) shows the unfolded paper?
Answer	A
Prompt	What object is located immediately to the right of point [710 991] in the second image, just outside of the frame ?
Answer	Wooden Table

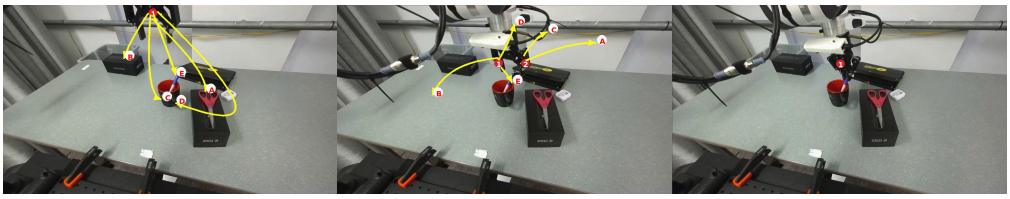
Table I Examples of *L3.Causal Reasoning*.

Sequential Planning	
	
Prompt	The camera is moving forward. Please arrange these three images in chronological order?
Answer	B-C-A
	

Prompt	The top row of images shows different views of the initial state of a cube stack, while the bottom row shows different views of the final state after transformation. During the transformation process, blocks can move one unit in any direction (forward, backward, left, right, up, down). If the target position is empty, the block can move there directly; if the target position already has a block, they swap places. Blocks cannot float in the air. If a block is moved away from a position, any block above it will fall down until reaching a supporting surface. The xyz axes are shown in the diagram, and each block's position can be precisely identified using coordinates (x1,y1,z1). Which of the following transformation sequences can change the cube stack from the initial state to the final state shown in the diagram? Please answer from options A, B, C, or D.
	`A. (1, 0, 0) y+ -- (0, 0, 1) z-`
	`B. (1, 0, 0) x+ -- (1, 0, 0) y+`
	`C. (2, 0, 0) x- -- (1, 0, 0) y+ -- (2, 0, 0) x-`
	`D. (0, 0, 0) x+ -- (0, 1, 0) y- -- (0, 0, 1) y+`

Answer	`C`

Table J Examples of *L3.Sequential Planning*.

L4 Agentic Competence	
	

Prompt	You are an intelligent agent observing a video sequence that depicts a task being performed which is "please get the blue pan out of the bottle". For each image in the sequence that provides candidate action options, select the single most appropriate action to perform in the current state. In the final frame, do not select any action. Instead, determine whether the overall task shown in the sequence has been successfully completed (1) or not completed (0). Please output the selected action for each intermediate frame and the completion flag for the final frame.
	`A. EB1`
	`B. CC1`
	`C. EA0`
	`D. EE1`
	`E. AD0`
	`F. EE0`
	`G. CD0`
	`H. EC0`

Answer	F

Prompt	Which image shows the robot making the most progress towards the task placing the fork to the right side of the orange kitchen wipe?
	`A. First.`
	`B. Second.`
	`C. Third.`
	`D. Fourth`

Answer	`C`

Table K Examples of *L4 Agentic Competence*.

L4 Agentic Competence (Continued)

Prompt

Task: Visual Navigation Action Sequence Generation

You are an expert visual navigation agent. Your task is to generate a sequence of actions to navigate a robot from a starting visual state (Image 2) to a target visual state (Image 3) based on the provided visual information.

Context and Example

We provide three sequential images: Image 1, Image 2, and Image 3. To help you understand the task, we are providing the complete action sequence that navigates from **Image 1** to **Image 2** as an example.

Example Action Sequence from Image 1 to Image 2:

```
{
  "actions": ["Dolly In", "Truck Left", "Pedestal Up",
              "Pan Left", "Roll CW"],
  "step_nums": [0, 3, 4, 5, 0]
}
```

Your Core Task

Now, carefully observe **Image 2** (the starting state) and **Image 3** (the target state). Your mission is to generate the action sequence required to navigate from Image 2 to Image 3.

1. Action Space

You must choose from the following 12 elementary actions. In your output, you **must use the 'symbol' specified** to represent each action.

Category	Action	Sym	Description
Trans.	Dolly In	W	Move forward
	Dolly Out	S	Move backward
	Truck Left	A	Move left
	Truck Right	D	Move right
	Pedestal Up	space	Move up
	Pedestal Down	shift	Move down
Rot.	Pan Left	←	Neg. rot around +Y
	Pan Right	→	Pos. rot around +Y
	Tilt Up	↑	Pos. rot around +X
	Tilt Down	↓	Neg. rot around +X
	Roll CW	↻	Neg. rot around +Z
	Roll CCW	↺	Pos. rot around +Z
Special	Stay	STOP	No movement

2. Step Size Parameters The magnitude of each action is determined by `step_num` combined with a unit step size.

- **Translation**: 0.0626 meters per step.
- **Rotation**: 0.0725 radians per step.

E.g., `action_symbol: "W"` and `step_num: 10` means moving forward by 10×0.0626 meters.

3. Required Output Format Your final output **MUST** be a JSON object containing two keys: `"actions"` and `"step_nums"`. The lengths of both arrays must be identical.

Answer

```
{
  "actions": [
    "Truck Left",
    "Pedestal Up",
    "Pan Left"
  ],
  "step_nums": [
    1,
    4,
    1
  ]
}
```

Table K Examples of *L4 Agentic Competence* (Continued).