ESI-BENCH: TOWARDS EMBODIED SPATIAL INTELLIGENCE THAT CLOSES THE PERCEPTION-ACTION LOOP

Anonymous authors
Paper under double-blind review

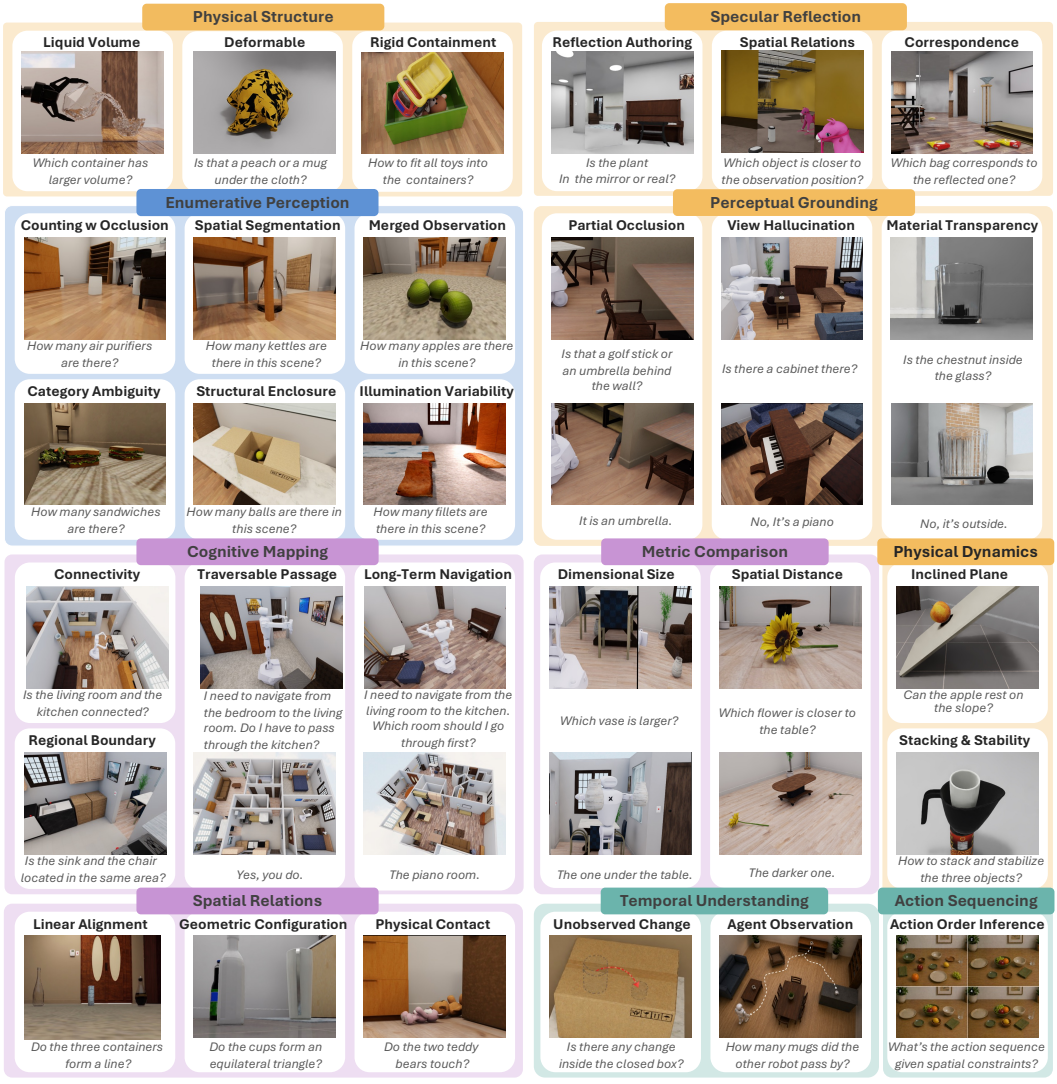


Figure 1: **ESI-BENCH** is a comprehensive benchmark for embodied spatial intelligence, spanning 10 task categories and 29 subcategories organized around Spelke's four core knowledge systems (Spelke & Kinzler, 2007): object representation, layout and geometry, number representation, and agents and goal-directed actions.

ABSTRACT

Spatial intelligence unfolds through a perception–action loop: agents act to acquire observations, and reason about how observations vary as a function of action. Rather than passively processing what is *seen*, they actively uncover what is *unseen* - occluded structure, dynamics, containment, and functionality that cannot be resolved from passive sensing alone. We move beyond prior formulations of spatial intelligence that assume oracle observations by recasting the observer as an actor. We introduce ESI-BENCH, a comprehensive benchmark for embodied spatial intelligence spanning 10 task categories and 29 subcategories built on OmniGibson,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

grounded in Spelke's core knowledge systems. Agents must decide what abilities to deploy — perception, locomotion, and manipulation — and how to sequence them to actively accumulate task-relevant evidence. We conduct extensive experiments on state-of-the-art MLLMs and find that active exploration substantially outperforms passive counterparts, with agents spontaneously discovering emergent spatial strategies without explicit instructions, while random multi-view often adds noise rather than signal despite consuming far more images. Most failures stem not from weak perception but from action blindness: poor action choices lead to poor observations, which in turn drive cascading errors. While explicit 3D grounding stabilizes reasoning on depth-sensitive tasks, imperfect 3D representation proves more harmful than 2D baselines by distorting spatial relations. Human studies further reveal that unlike humans who seek falsifying viewpoints and revise beliefs under contradiction, models commit prematurely with high confidence regardless of evidence quality, exposing a metacognitive gap that neither better perception nor more embodied interaction alone can close. All data, dataset construction codes, and evaluation scripts are available at https://bit.ly/47jGFBX.

1 INTRODUCTION

Perception is often characterized in cognitive science as *perceptually guided action* (Varela et al., 1991; Gibson, 1979): a perception–action loop where knowing how observations change as a function of action (O'Regan & Noë, 2001), and knowing which actions elicit informative sensing, are often more challenging than sensing itself. This is especially critical in spatial intelligence, which concerns not only what is *seen*, but also what is *unseen*. Latent physical properties such as occluded structure, dynamics, containment, and functionality that are inaccessible to passive sensing must be actively revealed through interaction, making spatial intelligence inherently embodied.

We move beyond prior formulations of spatial intelligence that assume passive oracle observations (Liu et al., 2023; Yang et al., 2025a;d) by recasting the observer as an actor. Our work contrasts with prior works in three key ways: (1) *from spatial sensing to spatial competence*, where agents are evaluated not only on what they can perceive, but on whether they know which embodied abilities to deploy to solve spatial tasks; (2) *selective sensing*, where agents must determine which observations are worth acquiring, prioritizing task-relevant information over redundant or uninformative inputs; and (3) *resolving perceptual ambiguities*, where agents must reason through misleading observations to infer hidden spatial structures and underlying physical constraints beyond what is directly observed.

We introduce ESI-BENCH, a comprehensive benchmark for embodied spatial intelligence spanning 10 task categories, 29 subcategories and 3,081 task instances, addressing the critical perception–action gap by focusing on questions that cannot be answered from passive observation alone. Our category design follows Spelke's core knowledge systems (Spelke & Kinzler, 2007), which identify four faculties of spatial intelligence: object representation, layout and geometry, number representation, and agents & goal-driven actions. Building on these theoretical foundations, we conduct human surveys to identify the most challenging spatial tasks that require embodied interaction and manipulation within each faculty, which are distilled into a structured taxonomy spanning diverse forms of spatial reasoning, as illustrated in Figure 1. These tasks only become meaningful when an agent has a body, a belief state, and physical stakes in the outcome: the agent must determine what abilities to deploy (perception, locomotion, manipulation), which actions to take (where to move, what to probe, how to manipulate), and how to execute them in the right order to successfully answer the questions.

We conduct extensive experiments on state-of-the-art MLLMs across three paradigms: passive single-view, passive random multi-view, and active exploration, alongside a ground-truth oracle that separates perception errors from action errors. Our experiments reveal multiple key insights: (1) *active exploration unlocks emergent spatial strategies*: without explicit instruction, active agents spontaneously discover diverse action compositions, driving substantial gains over passive counterparts while passive random multi-view, despite consuming far more images, often adds noise rather than signal; (2) *action blindness dominates perceptual blindness*: for most tasks the bottleneck is not perception but action selection, as models given oracle viewpoints succeed largely, yet certain tasks expose a hard perceptual ceiling that no action can overcome; and (3) *failures cascade and compound*: suboptimal actions produce uninformative views, which trigger worse subsequent actions, creating a compounding chain of errors that cannot be recovered within the step budget.

We further investigate whether explicit 3D representations can help. We find that while explicit 3D grounding stabilizes reasoning on depth-sensitive tasks by recovering information that 2D projections fundamentally lose, imperfect reconstructions prove more harmful than 2D baselines, as geometric artifacts actively distort fine-grained spatial relations and mislead downstream reasoning. Finally, human studies expose a critical gap in epistemic calibration (*i.e.*, a model's confidence matches the quality and uncertainty of its evidence). We observe that unlike humans who actively seek falsifying viewpoints, explore orthogonal angles, and revise their beliefs when contradicted, models commit prematurely with uniformly high confidence regardless of evidence quality, anchoring to first impressions and ignoring contradictory observations, a metacognitive failure that neither better perception nor more embodied interaction alone can close.

2 Related Works

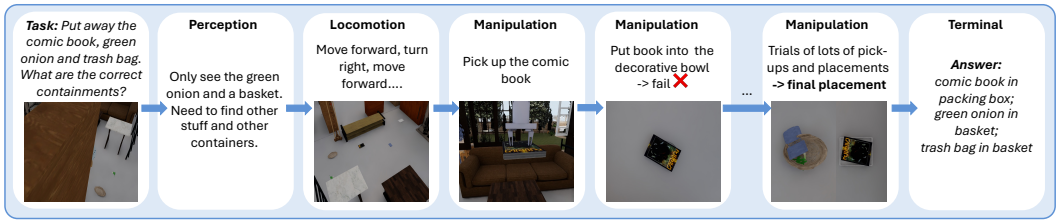
Table 1: Comparison with spatial reasoning and embodied benchmarks. Here, *embodied* means that an agent is situated in an embodied environment and acts through an explicit action space in the environment. **Active?**: whether the benchmark supports active perception. **Action Space**: L = locomotion, P = perception, M = manipulation. **Core Knowledge**: Obj = object, Geom = geometry, Num = number, Phys = physics, Agt = agent. **Hidden**: whether the benchmark targets latent properties such as occlusion, containment, dynamics that are inaccessible from passive observation.

Benchmark	Type	Active?	Action	Hidden	Core
VSR (Liu et al., 2023)	Spatial	✗	P	✗	Geom
BLINK (Fu et al., 2024)	Spatial	✗	P	✗	Obj, Geom, Num
3DSRBench (Ma et al., 2025)	Spatial	✗	P	✗	Obj, Geom
VSI-Bench (Yang et al., 2025a)	Spatial	✗	P	✗	Obj, Geom, Num
MMSI-Bench (Yang et al., 2025d)	Spatial	✗	P	✗	Obj, Geom
MindCube (Wang et al., 2026)	Spatial	✗	P	✓	Geom, Phys
PhysBench (Chow et al., 2025)	Spatial	✗	P	✓	Obj, Geom, Phys
CausalSpatial (Ma et al., 2026)	Spatial	✗	P	✓	Obj, Phys
EmbSpatial-Bench (Du et al., 2024)	Spatial	✗	P	✗	Obj, Geom
OpenEQA (Majumdar et al., 2024)	Embodied	✓	L, P	✗	Obj, Geom
EmbodiedBench (Yang et al., 2025b)	Embodied	✓	L, P, M	✗	Obj, Geom, Agt
EmbodiedEval (Cheng et al., 2025)	Embodied	✓	L, P, M	✗	Obj, Geom, Agt
EXPRESS-Bench (Jiang et al., 2025)	Embodied	✓	L, P	✗	Obj, Geom
ESPIRE (Zhao et al., 2026)	Spatial+Embodied	✗	P, M	✗	Obj, Geom
CHAIN (Wu et al., 2026b)	Embodied	✓	P, M	✓	Obj, Geom, Phys
ESI-Bench (ours)	**Spatial+Embodied**	✓	**L, P, M**	✓	**Obj, Geom, Num, Phys, Agt**

Benchmarks for Spatial Reasoning. Evaluation of spatial intelligence has scaled rapidly, yet most benchmarks still assume fixed observations. VSR (Liu et al., 2023) evaluates spatial relations in single images; BLINK (Fu et al., 2024) and 3DSRBench (Ma et al., 2025) extend to visual perception and fine-grained 3D reasoning; and SpatialScore (Wu et al., 2026a) unifies VGBench and other datasets with a tool-augmented agent. Recent benchmarks broaden inputs to egocentric video in VSI-Bench (Yang et al., 2025a), multi-image consistency in MMSI-Bench (Yang et al., 2025d), partial-observation mental modeling in MindCube (Wang et al., 2026), long-horizon recall and continual counting in Cambrian-S/VSI-SUPER (Yang et al., 2025c), and latent physical structure and object-centric dynamics in PhysBench (Chow et al., 2025) and CausalSpatial (Ma et al., 2026). However, even with richer inputs—images, views, videos, partial coverage, and dynamics—the observation process remains fixed, limiting diagnosis of active information acquisition. ESI-Bench keeps this diagnostic focus while making observation utility depend on the model's own decisions. Table 1 compares ESI-Bench with prior spatial intelligence benchmarks.

Spatial Reasoning Methods in MLLMs. Recent MLLMs improve spatial understanding, but largely assume fixed observations. Geometry-based methods inject 3D priors into 2D backbones: SpatialVLM (Chen et al., 2024) uses synthesized 3D annotations and metric-depth supervision; SpatialBot (Cai et al., 2024) uses RGB-D and depth-oriented QA; SpatialRGPT (Cheng et al., 2024) uses 3D scene graphs and a depth plugin; and Spatial-MLLM (Wu et al., 2025a) and VLM-3R (Fan et al., 2026) add priors from monocular video or reconstructive tuning. Reasoning-based methods make inference explicit: SpatialCoT (Liu et al., 2025) grounds CoT in spatial coordinates, while VILASR (Wu et al., 2025b) combines textual reasoning with visual drawing. These methods improve reasoning over given inputs; `ESI-Bench` tests whether models can choose the observations they need.

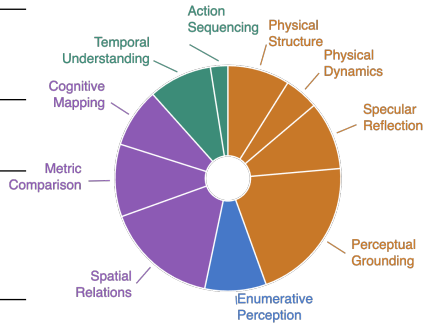
Embodied Evaluation and Active Perception. Another line evaluates models as embodied agents. EmbodiedBench (Yang et al., 2025b) and EmbodiedEval (Cheng et al., 2025) measure navigation,



(a) A dataset example including question, answer and trajectory, from the Rigid Containment sub-category.

Action	Description
Locomotion	
`move_forward` / `move_backward`	Translate agent along viewing axis
`move_left` / `move_right`	Translate agent laterally
`move_up` / `move_down`	Translate agent vertically
Perception	
`turn_left` / `turn_right`	Rotate agent horizontally
`turn_up` / `turn_down`	Rotate agent vertically
Manipulation	
`pick_up obj`	Pick up object
`put obj inside obj`	Put object inside object
`put obj on obj`	Put object on object
`fill obj with water`	Fill object with water
`pour obj from obj to obj`	Pour object from object to object
Terminal	
`answer(`$\hat{y}$`, `c`)`	Commit to final answer with confidence c

(b) ESI-BENCH agent action space.



(c) Distribution of task categories.

Figure 2: Overview of ESI-BENCH: dataset example, agent action space, and task distribution.

interaction, and QA, while OpenEQA (Majumdar et al., 2024) and EXPRESS-Bench (Jiang et al., 2025) study embodied QA and exploration quality, with EAC discouraging disembodied reasoning. EmbSpatial-Bench (Du et al., 2024) and ESPIRE (Zhao et al., 2026) diagnose egocentric spatial reasoning, with ESPIRE separating localization from execution. Active perception methods learn observation selection, from human-demonstration learning in Vision in Action (Xiong et al., 2025) and humanoid head-rotation search in Thinking in 360° (Yu et al., 2025) to viewpoint selection for VLA manipulation in SaPaVe (Liu et al., 2026a) and ActiveVLA (Liu et al., 2026b). Closest to ours, CHAIN (Wu et al., 2026b) evaluates closed-loop physical reasoning in mechanical puzzles, stacking, and packing. `ESI-Bench` complements these works with broader embodied spatial faculties across object, geometry, number, physics, and agent reasoning, including hidden states such as containment, occlusion, transparency, reflection, and unobserved scene change. Table 1 situates these differences.

3 ESI-BENCH

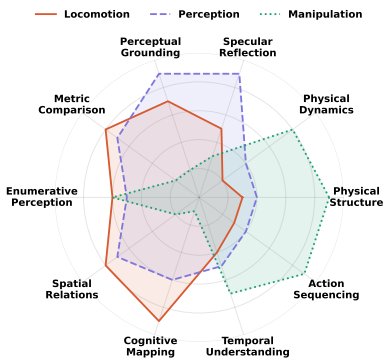
In this section, we introduce ESI-Bench, a comprehensive benchmark comprising 10 task categories, 29 subcategories, and 3,081 task instances. We describe the benchmark setup and task construction pipeline; the full task taxonomy, per-category construction details, and human verification and generator-bias analysis are provided in Appendix A, Appendix P, and Appendix F, respectively.

3.1 BENCHMARK SETUP

Task Definition Each task in ESI-Bench is defined as a tuple $(\mathcal{S}, p_0, q, y^*)$, where $\mathcal{S}$ is a 3D scene instantiated from the BEHAVIOR-1K scene pool with pre-loaded objects, p_0 is the agent's initial pose, q is a natural-language question about a spatial property of the scene, and y^* is the ground-truth answer. We formalize the environment as $\mathcal{E} = \langle \mathcal{S}, \mathcal{A}, \mathcal{O}, T \rangle$, where $\mathcal{A}$ is the action space, $\mathcal{O}$ is the egocentric observation space, and $T : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ governs scene transitions. Given $(\mathcal{S}, p_0, q)$, the agent receives observation $o_t \in \mathcal{O}$ at each timestep, issues action $a_t \in \mathcal{A}$, and induces a trajectory $\tau = (o_0, a_0, o_1, a_1, \ldots)$ until it commits to a final answer $\hat{y}$ within a budget of $T_{\max} = 30$ steps; we validate this budget in Appendix O. The action space spans perception, locomotion, and manipulation. Agents may move through the scene, rotate their viewpoint, interact with objects, and terminate with `answer(`$\hat{y}$`, `c`)`, which commits to an answer $\hat{y}$ with confidence c. We analyze sensitivity to the confidence-based stopping criterion in Appendix M.2. The full action space vocabulary is in Figure 2b; Figure 2a shows an example trajectory; and Appendix N discusses the rationale for high-level action design. Although answers are free-form, the question phrasing implicitly specifies

Category	Definition
Physical Structure	Reveal hidden containment capacity through manipulation
Physical Dynamics	Predict stability and motion under physical forces
Specular Reflection	Infer object relations from mirror reflections
Perceptual Grounding	Resolve viewpoint-dependent perceptual ambiguities
Metric Comparison	Overcome forced-perspective distortions in size and distan
Spatial Relations	Navigate to vantage points that break projective symmetry
Cognitive Mapping	Construct topological representations through locomotion
Enumerative Perception	Count objects despite occlusion, merging, or enclosure
Temporal Understanding	Infer state changes through interaction and observation
Action Sequencing	Reason over action sequences and causal dependencies

(a) ESI-Bench task categories and definitions.



(b) Required action types per category.

Figure 3: ESI-Bench task categories (L). Combination and level of embodied action types (R).

the expected format, such as yes/no for relational tasks, a category for comparisons, an integer for counting, or an ordering for procedural tasks. A response is correct if $\hat{y} = y^*$.

Simulation Environment. We build ESI-BENCH on BEHAVIOR-1K within the OmniGibson simulator. BEHAVIOR-1K provides 51 interactive 3D scenes spanning residential, commercial, and institutional environments, totaling over 300 rooms and 9k object instances across 1,829 categories, with physical properties such as friction, mass, and articulation. OmniGibson, built on NVIDIA Isaac Sim and PhysX 5, supports embodied spatial evaluation through rigid-body contact physics, particle-based fluids, transparency rendering, realistic lighting and reflections, and extended object states such as fill levels and toggled states. For each task instance, we randomly sample a BEHAVIOR-1K scene and select rooms based on room type and task-category requirements from a combined room-object list. We load the room into OmniGibson, allow physics to settle, and query the simulator state to extract a structured scene graph with object bounding boxes, categories, spatial relationships, room assignments, and states such as fillable capacity, toggled state, and contact flags. This scene graph supports scenario construction by providing the object inventory for task-relevant selection, the spatial layout for computing agent and camera poses, and the geometric and object-state ground truth for deriving labels. Appendix L discusses the rationale for using such simulation environment.

3.2 TASK CONSTRUCTION

Task Proposal. GPT-5 is prompted with the scene graph alongside task category requirements to select task-relevant objects from a random sample of 200 candidate categories drawn from the full BEHAVIOR-1K inventory, applying task-specific physical criteria to choose the most appropriate categories and resolve a specific model instance per category. Beyond object selection, GPT-5 also determines the initial positions of both the objects and the agent within the scene, and generates a ground-truth action trajectory providing the optimal sequence of actions needed to resolve the task. The selected objects and their spatial configuration implicitly define the task, with the ground-truth answer y^* derived directly from the resulting scene state.

Scene Instantiation. Given GPT-5-proposed object selections and initial positions, we load objects into the scene. Before placement, we check conflicts with existing scene content using bbox intersection tests. Objects are then placed on supporting surfaces via physics-based kinematic sampling and settle for a fixed number of simulation steps. After settling, we validate each configuration through stability checks from re-queried bboxes, per-view object existence checks using segmentation masks, and contact-flag validation when applicable. Configurations failing any check are rejected.

Agent Trajectory Collection. The agent is initialized at the GPT-5-proposed pose, sampled using controlled randomization and predefined placement rules designed to withhold the scene configuration and properties. At initialization, we apply the same verification battery used for scene instantiation: per-view object existence checks, bbox re-querying, and contact-flag validation when applicable. Proposed actions are executed step by step, with each step rendered as an egocentric observation and verified using the same checks. Trajectories failing verification at any step are discarded.

Metadata Saving. Upon completing trajectory collection, we save each task instance to a structured JSON file containing the scene, room, floor, per-object category and model instance, verified initial

position and quaternion, agent initial pose and quaternion p_0, per-view object existence flags, question, ground-truth answer, and action trajectory. This metadata provides a self-contained, reproducible record of the task instance that can be directly reloaded into the BEHAVIOR environment.

Human Verification and Generator Bias Audit. All generated task instances are reviewed by human annotators using rendered per-step observations and metadata. Annotators verify three criteria: correctness, ensuring the initial state and trajectory are physically valid; answerability, ensuring the task is solvable through interaction and spatially unambiguous; and non-triviality, ensuring the task cannot be solved from visual bias or prior knowledge alone and requires genuine spatial uncertainty. Each instance is independently reviewed by three annotators, with disagreements resolved by majority vote. Instances failing any criterion are discarded. Because GPT-5 is used as a proposal engine for objects, placements, questions, and trajectories, we further audit the generated tasks for possible linguistic and object-category biases. Appendix F reports human verification protocol, verification scores, inter-annotator agreement, rejection analyses, shortcut baselines, diversity statistics, and comparisons between GPT- and matched human-generated tasks. These results show that GPT-generated tasks are high-quality, exhibit limited shortcut bias, and have similar difficulty to human-generated tasks.

3.3 Task Categories and Statistics

ESI-Bench comprises 3,081 tasks across 10 categories and 29 subcategories. The category definitions are shown in Figure 3a. Due to space constraints, sub-category definitions is provided in Appendix A. Figure 2c reports the task distribution, and Figure 1 provides concrete examples for each subcategory. Per-category subcategory distributions are shown in Appendix P. Each category demands a distinct combination of embodied abilities, as illustrated in Figure 3b.

4 Experiments

4.1 Models and Evaluation Setup

We evaluate models across four paradigms, organized by the degree of action and perceptual access granted to the agent: (1) **Passive Single-View** provides a single fixed observation from the initial pose, establishing a baseline under conditions identical to existing spatial benchmarks; (2) **Passive Multi-View** provides a set of 30 views along a randomly-sampled trajectory starting at the initial pose in the scene, matching the maximum active trajectory budget $T_{\max} = 30$, to broadly cover the full environment and simulate exhaustive passive scene coverage without any agent active action or viewpoint selection; (3) **Active Exploration** places the agent the initial pose with full access to the action space, requiring it to gather evidence through deliberate movement and interaction before committing to a final answer; (4) **Ground-Truth Passive** provides the sequence of views rendered along the ground-truth action trajectory, serving as an oracle ablation that *separates perception errors from action errors*. Comparing (1) vs. (2) reveals whether more passive views helps; comparing (2) vs. (3) isolates the benefit of active action-guided over passive exhaustive coverage; comparing (3) vs. (4) isolates whether failures stem from the agent's inability to select informative actions or from perceptual limitations given the perfect views themselves.

We evaluate two families of models. 2D VLMs include GPT-6 Astra, GPT-5 and Gemini 3.1 Pro, each taking egocentric visual observations as input, evaluated across all four paradigms. 3D-augmented models include VGGT+Gemini, where explicit 3D scene representations are reconstructed from multi-view observations via VGGT, from which scene graphs are constructed and provided to the language model; and Ground-Truth 3D+Gemini, where perfect point clouds derived directly from simulator state are used to construct scene graphs provided to the language model, serving as an oracle ablation for 3D grounding. Human performance is additionally collected on all tasks to establish a human upper bound. The full human study protocol is described in Appendix C. All models are evaluated zero-shot with no task-specific fine-tuning, and results are reported in Table 2 as accuracy on matched subsets to ensure fair cross-paradigm comparison, with qualitative examples in Figure 4 and Appendix E. We additionally evaluate open-source models with and without task-specific fine-tuning in Appendix M.4. More model details including prompting templates are in Appendix M. Given the recent release of GPT-6, we prioritized its comprehensive evaluation in the 2D VLM setting across all four paradigms. We also performed a preliminary evaluation of 3D-augmented GPT-6 on a subset of tasks, using GPT-6 for both 3D scene reconstruction and spatial reasoning under active exploration setting. These additional results are presented in Appendix G.

Table 2: **ESI-BENCH results (%)** across passive, active, and oracle paradigms for 2D+VLMs, 3D+LLMs, and humans. **bold** denotes the highest non-oracle score, excluding GT Passive, GT 3D, and humans; **blue** denotes the highest score with GT Passive included, excluding GT 3D and humans.

	2D + VLM												3D + LLM								Human Performance			
	GPT-6-Astra				GPT-5				Gemini 3.1 Pro				VGGT + Gemini				GT 3D + Gemini							
Category/ Subcategory	Passive Single	Passive Multiple	Active Multiple	GT Passive	Passive Single	Passive Multiple	Active Multiple	GT Passive	Passive Single	Passive Multiple	Active Multiple	GT Passive	Passive Single	Passive Multiple	Active Multiple	GT Passive	Passive Single	Passive Multiple	Active Multiple	GT Passive	Passive Single	Passive Multiple	Active Multiple	GT Passive
Perceptual Grounding																								
View Hallucination	36.2	33.8	69.1	86.9	11.7	20.2	60.1	87.8	39.9	32.9	68.1	**91.1**	56.3	59.0	**76.1**	87.6	61.6	60.6	74.1	89.0	52.2	45.8	80.8	83.8
Partial Occlusion	33.7	31.6	56.8	80.0	32.6	26.3	47.4	86.3	30.5	32.6	**70.5**	**88.4**	51.6	48.4	53.7	86.3	56.8	55.8	72.6	91.6	37.9	34.7	80.0	87.4
Transparency	43.1	45.0	64.2	91.7	30.3	36.7	**66.1**	**96.3**	44.0	45.0	52.3	88.0	37.4	29.4	31.8	90.9	27.8	31.2	60.4	100	41.3	44.0	93.6	97.2
Physical Structure																								
Rigid Containment	52.5	55.0	60.0	95.0	45.0	42.5	42.5	95.0	47.5	40.0	**67.5**	**97.5**	27.5	37.5	57.5	72.5	45.0	42.5	65.0	95.0	47.5	42.5	92.5	100
Liquid Volume	75.0	74.3	**81.6**	83.1	66.2	66.2	**81.6**	86.0	69.9	67.6	80.9	**86.8**	65.4	56.7	71.3	77.2	74.5	74.5	83.1	86.8	79.4	66.9	89.7	86.8
Deformable	50.0	50.0	**58.2**	76.5	42.9	41.8	55.1	75.5	34.7	41.8	43.9	78.6	43.9	42.9	49.0	**81.6**	46.9	54.1	99.0	99.0	57.1	60.2	76.5	82.7
Physical Dynamics																								
Stacking & Stability	41.6	40.4	69.7	**92.1**	34.8	37.1	62.9	86.5	38.2	36.0	**78.7**	84.3	34.8	39.3	55.1	62.9	27.2	33.7	68.5	86.5	36.0	39.3	84.3	86.5
Inclined Plane	59.0	60.7	82.0	**91.8**	57.4	60.7	77.0	86.9	65.6	62.3	**83.6**	88.5	67.2	62.3	80.3	86.9	63.9	63.9	83.6	91.8	60.7	62.3	83.6	83.6
Specular Reflection																								
Reflection Authoring	81.8	78.8	**86.9**	**93.9**	68.7	70.7	70.3	73.7	60.6	60.6	64.9	67.0	52.5	54.6	53.5	55.6	55.6	58.6	55.6	60.6	94.9	88.9	96.0	96.0
Spatial Relations	33.6	31.9	**56.6**	**66.4**	50.4	38.9	54.8	58.4	43.4	42.5	44.2	46.9	41.6	39.8	41.6	43.4	38.9	35.4	54.2	41.6	78.8	78.8	84.1	87.6
Correspondence	42.0	44.3	**56.8**	**64.8**	39.8	51.1	52.3	56.8	37.5	40.9	42.0	42.0	37.5	43.2	48.9	48.9	31.8	39.8	42.3	48.9	85.2	80.7	89.8	92.0
Spatial Relations																								
Linear Alignment	26.6	44.7	58.5	**78.7**	27.7	31.9	42.6	60.6	47.9	44.6	**67.0**	77.7	45.7	38.3	53.2	59.6	73.4	73.4	84.0	89.4	51.1	39.4	73.4	79.8
Geometric Config	**31.6**	24.9	23.0	**57.2**	25.3	20.4	26.0	26.0	27.5	22.3	27.5	44.6	9.9	11.3	18.6	32.6	70.8	73.6	88.0	100	32.4	33.5	74.3	86.3
Physical Contact	52.5	62.5	**72.5**	80.8	40.0	41.7	64.2	**90.0**	60.8	55.8	70.0	74.2	59.2	41.7	59.2	78.2	65.8	67.5	70.8	72.5	35.8	40.8	88.3	90.8
Metric Comparison																								
Dimensional Size	33.5	34.7	**69.5**	**89.8**	42.5	44.9	67.7	80.3	44.3	41.3	68.3	82.6	50.4	40.1	59.3	80.8	58.6	60.5	69.9	85.9	48.5	56.9	82.6	91.9
Spatial Distance	**66.4**	63.8	**66.4**	**83.6**	53.9	49.1	58.6	73.7	52.6	50.5	59.9	80.3	50.7	47.3	55.0	71.4	59.6	61.2	64.5	96.7	57.9	59.9	69.1	78.9
Cognitive Mapping																								
Connectivity	50.0	51.7	63.3	83.3	68.3	70.0	68.3	78.3	51.7	48.3	60.0	**85.0**	66.7	66.7	**73.3**	83.3	65.0	65.0	71.7	86.7	68.3	68.3	81.7	91.7
Traversable Passage	63.3	66.7	66.7	75.0	68.3	66.7	71.7	73.3	66.7	61.7	**73.3**	**78.3**	68.3	63.3	66.7	**78.3**	71.7	73.3	71.7	80.0	70.0	71.7	78.3	85.0
Regional Boundary	65.0	62.5	**66.3**	66.3	65.0	63.8	65.0	**67.5**	63.8	62.5	65.0	65.0	65.0	65.0	62.5	**67.5**	60.0	68.8	62.5	68.8	61.3	62.5	67.5	70.0
Navigation	41.7	40.0	35.0	43.3	40.0	38.3	41.7	50.0	33.3	35.0	33.3	41.7	36.7	40.0	**43.3**	**51.6**	36.7	41.7	43.3	40.0	35.0	36.7	51.7	61.7
Enumerative																								
Counting w Occlusion	3.3	3.3	**20.0**	56.7	3.3	3.3	13.3	56.7	3.3	3.3	10.0	**63.3**	0.0	0.0	13.3	53.3	33.3	80.0	83.3	100.0	6.7	10.0	53.3	76.7
Spatial Segmentation	3.3	3.3	**33.3**	63.3	3.3	6.7	26.7	63.3	3.3	3.3	16.7	**70.0**	3.3	3.3	23.3	56.7	43.3	80.0	66.7	86.7	13.3	23.3	63.3	70.0
Merged Observation	51.7	48.3	61.7	68.3	38.3	35.0	51.7	50.0	43.3	51.7	61.7	**75.0**	**63.3**	51.7	55.0	65.0	51.7	63.3	93.3	100.0	60.0	60.0	65.0	73.3
Category Ambiguity	8.3	10.0	8.3	43.3	8.3	10.0	8.3	41.7	13.3	13.3	15.0	**48.3**	21.7	**26.7**	**26.7**	46.7	18.3	76.7	40.0	93.3	23.3	25.0	51.7	61.7
Structural Enclosure	5.0	12.5	**25.0**	**67.5**	5.0	10.0	22.5	**67.5**	2.5	0.0	10.0	52.5	0.0	0.0	12.5	52.5	40.0	60.0	50.0	100.0	10.0	22.5	42.5	77.5
Illumination	8.0	20.0	28.0	56.0	6.0	22.0	22.0	46.0	12.0	16.0	22.0	**58.0**	10.0	28.0	**30.0**	40.0	20.0	84.0	42.0	96.0	20.0	34.0	62.0	66.0
Temporal Understand																								
Unobserved Change	33.1	33.8	**56.1**	**77.0**	40.5	41.2	51.4	**77.0**	37.2	37.8	47.3	74.5	37.8	41.9	45.8	76.6	39.9	42.6	46.5	74.5	39.4	39.4	70.8	81.0
Agent Observation	48.1	49.6	**58.6**	**83.5**	40.6	30.1	51.0	72.7	37.6	36.1	**58.6**	65.4	32.7	27.9	34.2	56.5	36.3	33.3	76.7	87.8	38.3	42.1	83.5	90.2
Action Sequencing																								
Action Order	41.6	41.6	**63.6**	**72.7**	36.4	37.7	41.6	67.5	44.2	46.8	54.5	**72.7**	35.1	31.1	50.6	58.4	44.2	46.8	51.9	75.3	40.3	41.6	74.0	81.8

4.2 WHAT REALLY HOLDS SPATIAL INTELLIGENCE BACK: SEEING OR ACTING?

Table 2 reports accuracy across baselines under passive and active paradigms. Without any instruction, active agents *spontaneously discover emergent spatial strategies*: to determine whether a chestnut is inside a glass, agents independently develop four distinct approaches, moving behind the object, repositioning top-down, picking it up, and pouring it out (Figure 4a), none of which were prescribed. These emergent abilities drive consistent and substantial gains across tasks: on View Hallucination, from 36.2% to 69.1% for GPT-6; and on Rigid Containment, from 47.5% to 67.5% for Gemini. In contrast, passive multi-view provides negligible or negative gains despite consuming far more images (*e.g.*, GPT-5 drops from 53.9% to 49.1% on Spatial Distance compared with the single-view passive baseline), confirming that observation quantity without selective action often adds noise rather than signal. Figure 5 further supports this distinction: active exploration reaches correct answers with only a small number of task-conditioned steps, suggesting that the benefit comes from selective evidence acquisition rather than simply increasing the number of observations. We quantify the diversity of emergent strategies in Appendix H and further analyze passive multi-view through failure analysis, view-count ablations, and stronger coverage-, heuristic-, and oracle-based passive baselines in Appendix K.

Ground-truth passive trajectories reveal that the bottleneck is largely in action selection, not perception: GPT-5 improves from 42.5% under active exploration to 95.0% with ground-truth trajectory views on Rigid Containment, and GPT-6 from 64.2% to 91.7% on Transparency. Geometric Configuration stand as exceptions where oracle views provide limited benefit: GPT-5 reaches only 26.0% on Geometric Configuration under ground-truth passive, and models consistently fail to determine whether three objects form an equilateral triangle even from the perfect viewpoint (Figure 4c). These

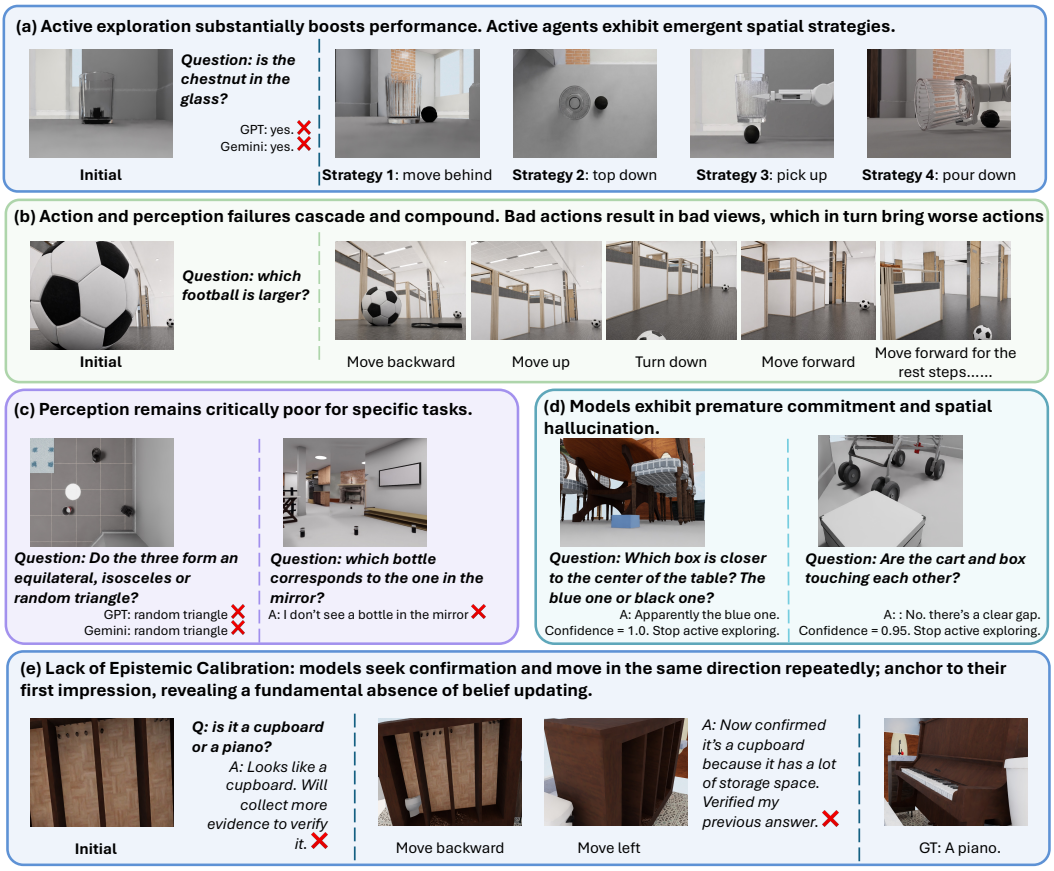


Figure 4: **Qualitative Study**, showing success & failure modes and reasons behind model behavior.

cases indicate hard perceptual limits that no action strategy can overcome. Together, the active-to-oracle gap shows that action and perception failures cascade and compound. Bad actions lead to bad views, which in turn induce worse subsequent actions that cannot be recovered within the step budget (Figure 4b). *Notably, even GPT-6-Astra, one of the most capable current models, remains weak under passive observation, achieves only modest improvements over GPT-5 under active exploration, and still exhibits a substantial gap to ground-truth passive trajectories*, further strengthening our claims. We further disentangle this gap through controlled component ablations and human error attribution over action selection, memory, belief updating, and stopping in Appendix M.1. These analyses identify action selection as the largest individual error source.

Action Blindness Dominates Perceptual Blindness, while Their Coupling Drives Failure Cascades

- Without explicit instruction, agents spontaneously acquire spatial strategies, yielding substantial gains over passive multi-view, which often adds noise rather than signal.
- For most tasks, perception is not the bottleneck: with the right viewpoint, the agent succeeds; yet some tasks hit a hard perceptual ceiling that no action can overcome.
- Suboptimal actions produce bad views, which in turn produce worse actions, cascading into reasoning failures unrecoverable within the step budget.

4.3 When Does 3D Help, and When Does It Hurt?

From Table 2, GT 3D+Gemini substantially improves performance when the underlying 3D representation is accurate: on Geometric Configuration, it reaches 70.8% compared with 27.5% for Gemini, a 43.3-point gain; on Counting w Occlusion, it reaches 33.3% compared with 3.3%, a 30.0-point gain. These gains suggest that explicit geometry can resolve spatial ambiguities that are difficult to infer from 2D observations alone. However, imperfect 3D reconstruction can be actively harmful: VGGT+Gemini drops to 9.9% on Geometric Configuration compared with Gemini's 27.5%, and to 0.0% on Counting w Occlusion compared with 3.3%. Rather than merely failing to help, noisy reconstructions distort fine-grained spatial relations and cause the LLM to reason over a corrupted

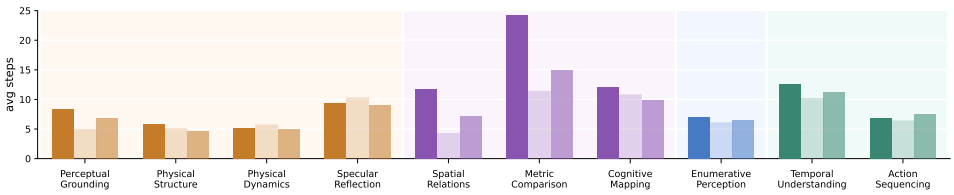


Figure 5: Average active steps to reach a correct answer.Left to right: GPT-5, Gemini, GPT-6

scene graph, making 3D augmentation a high-variance strategy that amplifies both successes and failures. We quantitatively analyze VGGT failure modes, including object duplication, hallucination, and spatial-relation corruption, in Appendix I. In our preliminary experiment using GPT-6 for both 3D reconstruction and reasoning (Sec. G), it outperforms VGGT+Gemini but still falls substantially behind GT 3D+Gemini, suggesting current models still struggle to recover accurate 3D spatial information.

3D Helps When Geometry Is Perfect, But Imperfect Reconstruction Actively Misleads

- Explicit 3D grounding improves performance on tasks where depth and occlusion render 2D projections fundamentally ambiguous.
- Imperfect 3D reconstruction degrades performance, proving more harmful than 2D baselines.

4.4 How Far Are Models From Human-Level Spatial Reasoning?

Table 2 shows that human and model performance are often comparable when agents are given fixed observations, either from the initial passive view or from the ground-truth trajectory. This suggests that the perceptual gap between humans and models, while present, may be smaller than commonly assumed. In some categories, models even match or exceed human passive performance. Under ground-truth trajectories, GPT-6 surpasses humans in Inclined Plane, Stacking and Distance. Yet the divergence emerges sharply under active exploration: humans dramatically outperform models by knowing what to look for and when to stop, with active performance much closer to ground-truth passive performance (*e.g.*, on Transparency, humans reach 93.6% versus 64.2% for GPT-6 and 52.3% for Gemini). Examining model and human trajectories, we find humans exhibit stronger epistemic caution: they gather more observations before committing, actively seek viewpoints that could falsify their current hypothesis, and reduce confidence under ambiguity. Figure 4d shows models terminate exploration prematurely, committing with high confidence after only a few steps regardless of evidence ambiguity, producing spatial hallucinations that contradict the scene state. This overconfidence is compounded by a directional bias in action selection: rather than seeking views that could disprove an initial impression, models repeatedly move in the same direction, accumulating redundant rather than informative observations. As shown in Figure 4e, when asked whether an object is a cupboard or a piano, the model initially guesses cupboard, then moves backward and left only to confirm that belief rather than challenge it, asserting the wrong answer with increased confidence despite never seeking a falsifying viewpoint. Models anchor rigidly to their first impression rather than revising, exposing a failure of epistemic calibration that is fundamentally distinct from perceptual ability and cannot be resolved by stronger visual encoders or additional exploration steps. To make this model-human gap more concrete, Appendix J operationalizes it through trajectory-level quantitative measures of view diversity, contrastive-view seeking, and belief revision.

Lack of Epistemic Calibration: Models Can See But Do Not Know When They Have Seen Enough

- Models commit prematurely with high confidence; humans treat uncertainty as a signal to keep looking, not a reason to answer.
- Humans seek viewpoints that falsify their hypothesis; models seek confirmation and move in the same direction repeatedly.
- Humans revise beliefs when contradicted; models anchor to their first impression, revealing a fundamental absence of belief updating.

Conclusion We introduced ESI-Bench, a spatial intelligencebenchmark requiring agents to acquire evidence through perception, locomotion, and manipulation. Across 10 categories and 29 subcategories, active exploration improves over passive observation, but models still struggle with action selection. ESI-Bench provides a testbed for studying agents that close the perception-action loop.

AI USE STATEMENT

We use generative AI tools to assist with literature review, language refinement and code inspection. The authors verified the references, scientific claims, experimental results, and AI-assisted artifacts, and take responsibility for the final submission.

REPRODUCIBILITY STATEMENT

All data, dataset construction codes, and model codes are available at https://bit.ly/47jGFBX. The paper and supplementary material also describe the implementation details.

REFERENCES

Wenxiao Cai, Yaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. *arXiv preprint arXiv:2406.13642*, 2024. 3

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14455–14465, June 2024. 3

An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision language models, 2024. URL https://arxiv.org/abs/2406.01584. 3

Zhili Cheng, Yuge Tu, Ran Li, Shiqi Dai, Jinyi Hu, Shengding Hu, Jiahao Li, Yang Shi, Tianyu Yu, Weize Chen, Lei Shi, and Maosong Sun. Embodiedeval: Evaluate multimodal llms as embodied agents, 2025. URL https://arxiv.org/abs/2501.11858. 3

Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Guizilini, and Yue Wang. Physbench: Benchmarking and enhancing vision-language models for physical world understanding, 2025. URL https://arxiv.org/abs/2501.16411. 3

Mengfei Du, Binhao Wu, Zejun Li, Xuanjing Huang, and Zhongyu Wei. Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models, 2024. URL https://arxiv.org/abs/2406.05756. 3, 4

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Huaizhi Qu, Shijie Zhou, Dilin Wang, Zhicheng Yan, Hongyu Xu, Justin Theiss, Tianlong Chen, Jiachen Li, Zhengzhong Tu, Zhangyang Wang, and Rakesh Ranjan. Vlm-3r: Vision-language models augmented with instruction-aligned 3d reconstruction, 2026. URL https://arxiv.org/abs/2505.20279. 3

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, 2024. URL https://arxiv.org/abs/2404.12390. 3

James J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, 1979. 2

Kaixuan Jiang, Yang Liu, Weixing Chen, Jingzhou Luo, Ziliang Chen, Ling Pan, Guanbin Li, and Liang Lin. Beyond the destination: A novel benchmark for exploration-aware embodied question answering, 2025. URL https://arxiv.org/abs/2503.11117. 3, 4

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning, 2023. URL https://arxiv.org/abs/2205.00363. 2, 3

Mengzhen Liu, Enshen Zhou, Cheng Chi, Yi Han, Shanyu Rong, Liming Chen, Pengwei Wang, Zhongyuan Wang, and Shanghang Zhang. Sapave: Towards active perception and manipulation in vision-language-action models for robotics, 2026a. URL https://arxiv.org/abs/2603.12193. 4

Yuecheng Liu, Dafeng Chi, Shiguang Wu, Zhanguang Zhang, Yaochen Hu, Lingfeng Zhang, Yingxue Zhang, Shuang Wu, Tongtong Cao, Guowei Huang, Helong Huang, Guangjian Tian, Weichao Qiu, Xingyue Quan, Jianye Hao, and Yuzheng Zhuang. Spatialcot: Advancing spatial reasoning through coordinate alignment and chain-of-thought for embodied task planning, 2025. URL https://arxiv.org/abs/2501.10074. 3

Zhenyang Liu, Yongchong Gu, Yikai Wang, Xiangyang Xue, and Yanwei Fu. Activevla: Injecting active perception into vision-language-action models for precise 3d robotic manipulation, 2026b. URL https://arxiv.org/abs/2601.08325. 4

Wenxin Ma, Chenlong Wang, Ruisheng Yuan, Hao Chen, Nanru Dai, S. Kevin Zhou, Yijun Yang, Alan Yuille, and Jieneng Chen. Causalspatial: A benchmark for object-centric causal spatial reasoning, 2026. URL https://arxiv.org/abs/2601.13304. 3

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso M de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark, 2025. URL https://arxiv.org/abs/2412.07825. 3

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Sasha Sax, and Aravind Rajeswaran. Openeqa: Embodied question answering in the era of foundation models. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3, 4

J. Kevin O'Regan and Alva Noë. A sensorimotor account of vision and visual consciousness. *Behavioral and Brain Sciences*, 24(5):939–973, 2001. 2

Elizabeth S. Spelke and Katherine D. Kinzler. Core knowledge. *Developmental Science*, 10(1):89–96, 2007. 1, 2

Francisco J. Varela, Evan Thompson, and Eleanor Rosch. *The Embodied Mind: Cognitive Science and Human Experience*. MIT Press, 1991. 2

Qineng Wang, Baiqiao Yin, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, Saining Xie, Jiajun Wu, Li Fei-Fei, and Manling Li. Mindcube: Spatial mental modeling from limited views, 2026. URL https://arxiv.org/abs/2506.21458. 3

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence, 2025a. URL https://arxiv.org/abs/2505.23747. 3

Haoning Wu, Xiao Huang, Yaohui Chen, Ya Zhang, Yanfeng Wang, and Weidi Xie. Spatialscore: Towards comprehensive evaluation for spatial intelligence, 2026a. URL https://arxiv.org/abs/2505.17012. 3

Junfei Wu, Jian Guan, Kaituo Feng, Qiang Liu, Shu Wu, Liang Wang, Wei Wu, and Tieniu Tan. Reinforcing spatial reasoning in vision-language models with interwoven thinking and visual drawing, 2025b. URL https://arxiv.org/abs/2506.09965. 3

Yuhao Wu, Maojia Song, Yihuai Lan, Lei Wang, Zhiqiang Hu, Yao Xiao, Heng Zhou, Weihua Zheng, Dylan Raharja, Soujanya Poria, and Roy Ka-Wei Lee. From perception to action: An interactive benchmark for vision reasoning, 2026b. URL https://arxiv.org/abs/2602.21015. 3, 4

Haoyu Xiong, Xiaomeng Xu, Jimmy Wu, Yifan Hou, Jeannette Bohg, and Shuran Song. Vision in action: Learning active perception from human demonstrations, 2025. URL https://arxiv.org/abs/2506.15666. 4

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, 2025a. URL https://arxiv.org/abs/2412.14171. 2, 3

Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, Heng Ji, Huan Zhang, and Tong Zhang. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents, 2025b. URL https://arxiv.org/abs/2502.09560. 3

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Daohan Lu, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-s: Towards spatial supersensing in video, 2025c. URL https://arxiv.org/abs/2511.04670. 3

Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, Dahua Lin, Tai Wang, and Jiangmiao Pang. Mmsi-bench: A benchmark for multi-image spatial intelligence, 2025d. URL https://arxiv.org/abs/2505.23764. 2, 3

Heyang Yu, Yinan Han, Xiangyu Zhang, Baiqiao Yin, Bowen Chang, Xiangyu Han, Xinhao Liu, Jing Zhang, Marco Pavone, Chen Feng, Saining Xie, and Yiming Li. Thinking in 360°: Humanoid visual search in the wild, 2025. URL https://arxiv.org/abs/2511.20351. 4

Yanpeng Zhao, Wentao Ding, Hongtao Li, Baoxiong Jia, and Zilong Zheng. Espire: A diagnostic benchmark for embodied spatial reasoning of vision-language models, 2026. URL https://arxiv.org/abs/2603.13033. 3, 4

CONTENTS

1 Introduction **2**

2 Related Works **3**

3 ESI-BENCH **4**

3.1 Benchmark Setup 4

3.2 Task Construction 5

3.3 Task Categories and Statistics 6

4 Experiments **6**

4.1 Models and Evaluation Setup 6

4.2 What Really Holds Spatial Intelligence Back: Seeing or Acting? 7

4.3 When Does 3D Help, and When Does It Hurt? 8

4.4 How Far Are Models From Human-Level Spatial Reasoning? 9

A Per-Category Task Description **16**

B Per-Category Task Distribution **16**

C Human Study Protocol **16**

D More Benchmark Examples **17**

E More Qualitative Examples **19**

F Human Verification and Generator Bias Analysis **19**

F.1 Human Verification Protocol 20

F.2 Inter-Annotator Agreement and Rejection Analysis 21

F.3 Verification Scores for GPT-5-Generated Tasks 21

F.4 Bias Audit for GPT-5-Generated Tasks 22

F.5 Comparison with Human-Generated Tasks 23

G GPT-6 3D Experiments **24**

H Diverse Emergent Behaviors in Active Exploration **24**

I Failure Analysis of VGGT-Based 3D Augmentation **25**

J Operationalizing the Metacognitive Gap **26**

K Analysis of Passive Multi-View Evaluation **27**

K.1 Number of Passive Views 27

K.2 Stronger Passive-to-Oracle Baselines 28

L Why a Synthetic Simulator Benchmark? 28

M Active Agent Prompting and Action Selection 29

M.1 Component Ablations and Failure Attribution 30

M.2 Stopping-Threshold Sensitivity 30

M.3 Failure Attribution by Action Type 31

M.4 Open-Source Models and Task-Specific Fine-Tuning 31

N Why High-Level Actions? 32

O Step Budget Ablation 32

P Per-Category Task Construction Details 33

P.1 Physical Structure: Material Transparency 33

P.2 Physical Structure: Liquid Volume 34

P.3 Physical Structure: Deformable 34

P.4 Physical Dynamics: Inclined Plane 34

P.5 Physical Dynamics: Stacking and Stability 35

P.6 Metric Comparison: Spatial Distance 35

P.7 Metric Comparison: Dimensional Size 35

P.8 Spatial Relations: Physical Contact 36

P.9 Spatial Relations: Linear Alignment and Geometric Configuration 36

P.10 Physical Structure: Rigid Containment 36

P.11 Action Sequencing: Action Order Inference 37

P.12 Temporal Understanding: Agent Observation 37

P.13 Temporal Understanding: Unobserved Changes 37

P.14 Perceptual Grounding: Partial Occlusion 38

P.15 Perceptual Grounding: View Hallucination 38

P.16 Specular Reflection : Reflection Authorization 38

P.17 Specular Reflection : Spatial Relationship 39

P.18 Specular Reflection : Scene Correspondence 39

P.19 Enumerative Perception : Counting w Occlusion 40

P.20 Enumerative Perception : Spatial Segmentation 40

P.21 Enumerative Perception : Merged Observation 40

P.22 Enumerative Perception : Category Ambiguity 41

P.23 Enumerative Perception : Structural Enclosure 41

P.24 Enumerative Perception : Illumination Variability 41

P.25 Cognitive Mapping : Connectivity 41

P.26 Cognitive Mapping : Traversable Passage 42
P.27 Cognitive Mapping : Regional Boundary 42
P.28 Cognitive Mapping : Long-Term Navigation 42

A PER-CATEGORY TASK DESCRIPTION

In Table A, we show the detailed definitions of each sub-category in our ESI-BENCH.

Table 3: **ESI-BENCH task taxonomy and definitions** of 10 categories spanning 29 subcategories. Colors correspond to Spelke's four core knowledge systems: object representation, layout and geometry, number representation, and agents and goal-directed actions.

Category	Subcategory	Description	Example
Physical Structure	Rigid Containment	Plan the placement of multiple objects across multiple containers	How can all the toys be fit into the boxes?
	Liquid Volume	Compare liquid-holding capacity of containers	Which container has larger volume?
	Deformable	Infer structure from deformable object	What's under the sweater, apple or mug?
Physical Dynamics	Inclined Plane	Predict object motion and stability on slopes	Can the apple sit stable on the slope?
	Stacking & Stability	Whether objects stack or balance given shape, mass and geometry	How to stack and stabilize these three objects?
Specular Reflection	Reflection Authoring	Distinguish real objects from mirror reflections	Is the object real or reflected?
	Spatial Relations	Infer relations among objects across mirror and real-world views	What relations does the mirror reveal?
	Correspondence	Identify which objects appear in mirror given the real-world scene	Which of the 3 snacks appears in the mirror?
Perceptual Grounding	Partial Occlusion	Reason about objects hidden behind other scene elements	Golf Stick or Umbrella behind the Wall?
	View Hallucination	Detect objects whose visibility changes critically with viewing angle	Cabinet or piano from this view?
	Material Transparency	Reason about objects seen through transparent surfaces	Is the object inside the glass or not?
Metric Comparison	Dimensional Size	Compare relative sizes of objects	Which vase is larger?
	Spatial Distance	Compare relative distances with respect to a reference object	Which flower is closer to the table?
Spatial Relations	Linear Alignment	Whether objects are arranged along a common axis	Do the glasses form a line?
	Geometric Configuration	Identify the shape formed by a set of objects	Do the cups form an equilateral triangle?
	Physical Contact	Detect whether two or more objects are in direct contact	Are the teddy bears touching each other?
Cognitive Mapping	Connectivity	Whether two locations or regions are mutually reachable	Is region A connected to C?
	Traversable Passage	Identify navigable corridors or passageways between regions	Passage between the rooms?
	Regional Boundary	Identify and delineate distinct functional spatial regions	Boundaries of this region?
	Long-Term Navigation	Plan multi-step navigation trajectories toward a distant goal	What room is behind the wall?
Enumerative Perception	Counting w Occlusion	Count objects partially obscured by other scene elements	How many balls are under the blanket?
	Spatial Segmentation	Count objects separated across distinct spatial regions	One or two cylinders separated by the post?
	Category Ambiguity	Count visually similar objects requiring fine-grained distinction	How many apples among the balls?
	Merged Observation	Count groups that appear visually merged from a single viewpoint	How many separate stacks of books?
	Illumination Variability	Count objects under challenging or non-uniform lighting	Objects in the dim scene?
	Structural Enclosure	Count objects hidden within enclosed or covered spaces	Objects in the microwave?
Temporal Understanding	Unobserved Change	Infer scene changes that occurred during an unobserved interval	What changed when I looked away?
	Agent Observation	Reason about scene dynamics from other agents	Other robot's world model?
Action Sequencing	Action Order Inference	Determine the correct procedural ordering of a sequence of actions	What is the action sequence for assembly?

B PER-CATEGORY TASK DISTRIBUTION

In Figure 6, we show the detailed distribution of sub-categories inside each cateogyr.

C HUMAN STUDY PROTOCOL

Participants and compensation. We recruited 28 participants via internal volunteer sign-up, all with normal or corrected-to-normal vision, and all was not exposed to such tasks before, ensuring fairness. Each participant was compensated at a rate above local minimum wage. The study was conducted in accordance with institutional guidelines; no IRB approval was required as the study involved no sensitive data collection and participants interacted only with simulated environments. The participants evaluate on the full benchmark like our models.

Interface. Participants interacted with a web-based interface displaying the egocentric view of an agent inside an OmniGibson scene. At each step, the participant selected one action from the same discrete action vocabulary available to evaluated models: locomotion actions (`move_forward`, `move_backward`, `move_left`, `move_right`, `move_up`, `move_down`), perception actions (`turn_left`, `turn_right`, `turn_up`, `turn_down`), manipulation actions (`pick_up`, `put_inside`, `put_on`, `fill_with_water`, `pour`), and the terminal action `answer`$(\hat{y}, c)$ which commits to a final free-form answer with an associated confidence score $c \in [0, 1]$. The step budget was capped at $T_{\max} = 30$, matching the budget available to all evaluated models. We strictly guarantee that the human interface is exactly the same as active model interface with the same pixel-level egocentric rendering and action discretization affects.

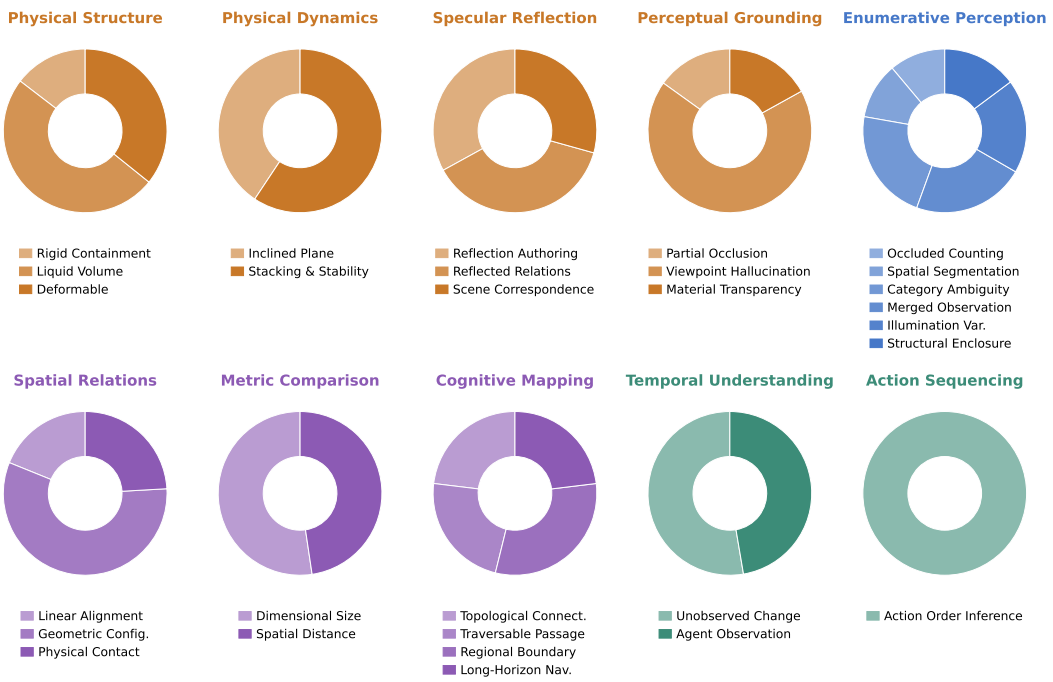


Figure 6: Subcategory distribution within each of the 10 ESI-BENCH task categories.

Task assignment. Each participant completed a subset of tasks drawn from at least 6 of the 10 ESI-BENCH categories, with tasks sampled to balance category coverage across participants. Each task instance was completed by at least 5 participants independently; disagreements on the ground-truth answer were resolved by a third annotator.

Instructions. Participants were shown a brief written tutorial with two worked examples before beginning. They were instructed to explore freely, gather as much evidence as they deemed necessary, and answer only when confident. No hints about the correct answer or the nature of the spatial challenge were provided.

Data collected. For each task instance and participant we recorded: the full action sequence and egocentric frame at each step, the final answer and confidence score, the step at which `answer` was issued, and any intermediate confidence ratings logged during exploration.

D MORE BENCHMARK EXAMPLES

In figure 7, we show more data points / examples of our ESI-BENCH datasets.

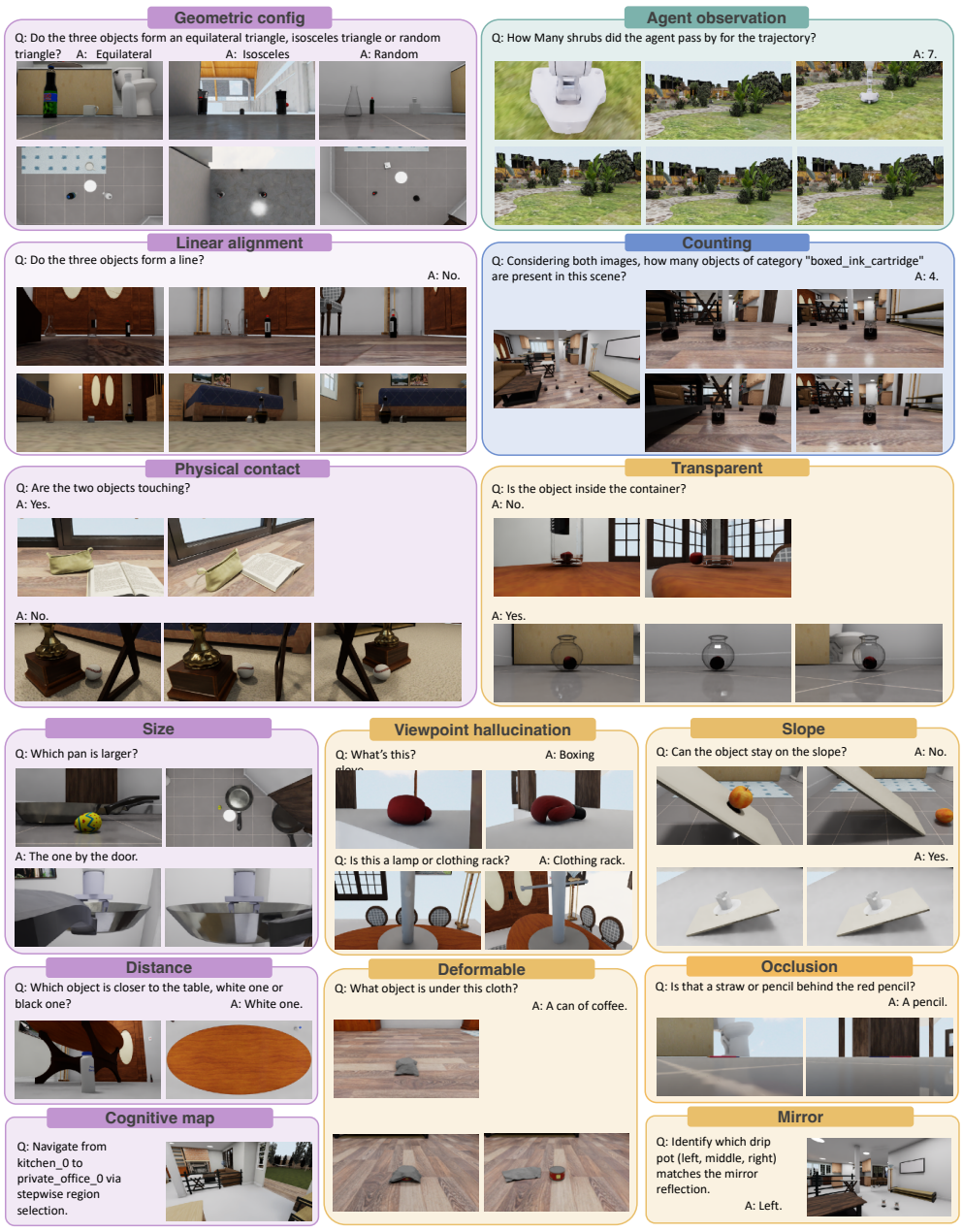


Figure 7: Additional benchmark examples from ESI-BENCH, organized by core knowledge systems: object representation, layout and geometry, number representation, and agents and goal-directed actions.

E More Qualitative Examples

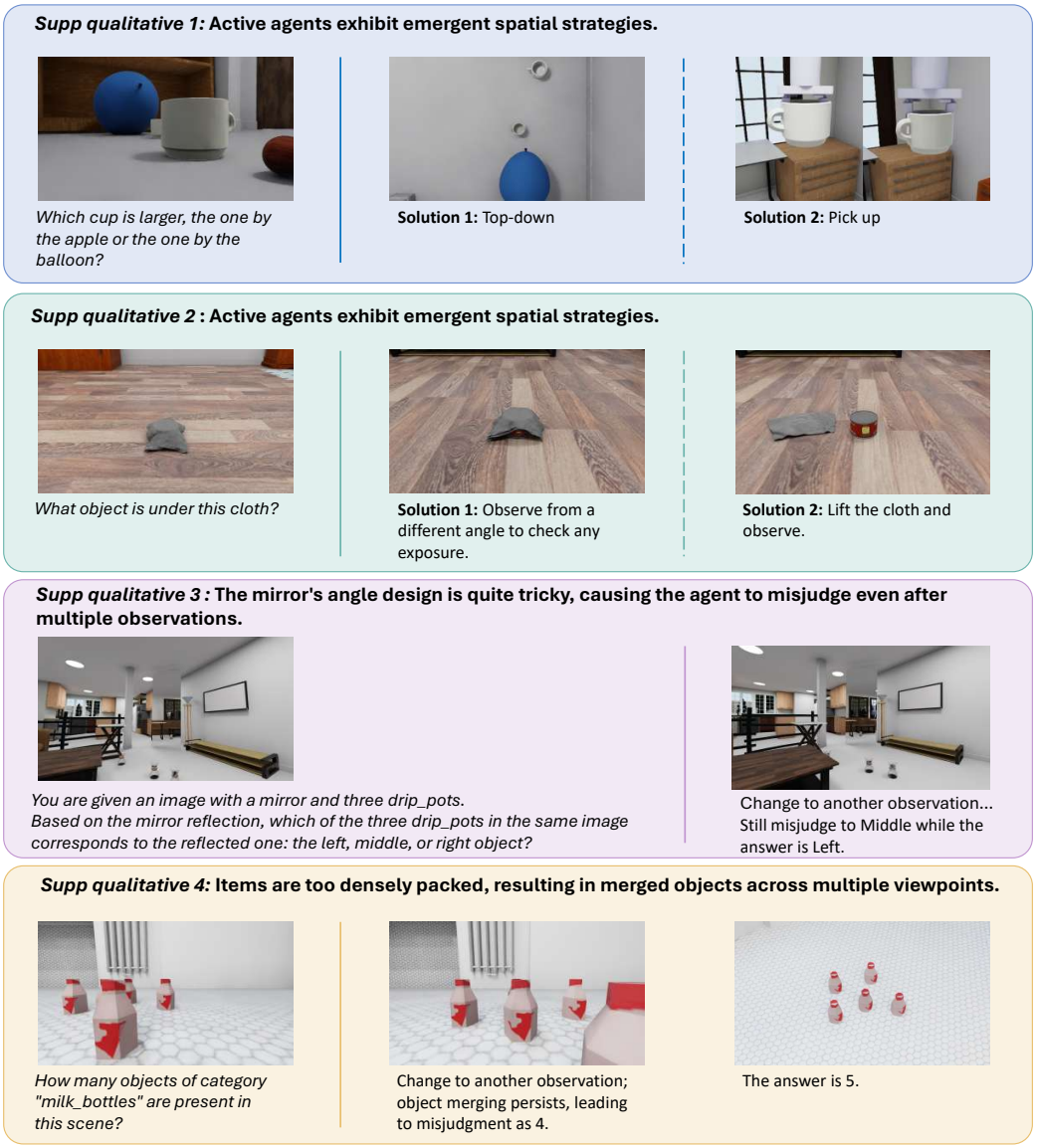


Figure 8: Additional qualitative examples illustrating emergent agent behaviors and failure modes: (1) active agents employ emergent spatial strategies such as top-down viewing and object manipulation to compare sizes; (2) active agents explore alternative viewpoints or interact with the environment to identify hidden objects; (3) tricky mirror angle designs cause persistent misjudgment; (4) densely packed objects lead to merged perception across viewpoints, resulting in undercounting.

F Human Verification and Generator Bias Analysis

This appendix provides additional details on human verification and on our analysis of possible generator bias. GPT-5 is used only as a scalable proposal engine during dataset construction: it proposes candidate objects, object placements, initial agent poses, natural-language questions, and candidate action trajectories from simulator scene graphs. The final benchmark instances, however, are not accepted directly from GPT-5. Each candidate is instantiated and executed in OmniGibson, filtered

through simulator-state checks, and then verified by human annotators using rendered observations and simulator metadata. This ensures that the final benchmark is grounded in physical simulation and human judgment rather than relying on GPT-5 as an automatic source of correctness.

Because LLM-assisted benchmark construction may introduce systematic artifacts, we further audit GPT-5-generated tasks for possible linguistic and object-category. The purpose of this analysis is to test whether models could solve ESI-BENCH by exploiting superficial regularities rather than actively acquiring task-relevant evidence. We evaluate shortcut baselines that remove visual observations and action histories, measure answer/object/template diversity, and compare model accuracy on matched GPT-5-generated and human-generated task subsets.

F.1 HUMAN VERIFICATION PROTOCOL

After simulator-based filtering, every candidate task is reviewed by human annotators. Human verification is designed to ensure that each retained instance is **correct**, **answerable**, and **non-trivial**. Annotators are not asked to judge whether a task is interesting in the abstract; instead, they follow a structured checklist tied to the simulator state, rendered observations, and the available action space.

For each candidate task, annotators are shown: (1) the natural-language question and proposed ground-truth answer; (2) the initial egocentric observation; (3) rendered observations along the verified trajectory; (4) the final simulator state after executing the trajectory; and (5) task metadata, including object categories, object states, spatial relations, containment/contact flags, visibility masks, and room assignments when needed. This allows annotators to verify both the visual evidence available to an agent and the simulator-derived state used to determine the answer.

Annotators evaluate each instance along three axes.

Correctness. Correctness measures whether the task is physically and semantically consistent with the simulator state. Annotators verify that the relevant objects are successfully instantiated, that object placements are physically plausible, that the proposed trajectory can be executed without invalid transitions, and that the proposed answer matches the simulator-derived state. For example, a containment task is marked correct only if the target object is actually inside the container according to simulator state and the rendered observations are consistent with this state. A contact task is marked correct only if the contact relation is supported by simulator contact flags and is visually plausible in the rendered scene. Instances are rejected if objects float, intersect unnaturally, disappear, fail to settle, or if the proposed answer contradicts the simulator state.

Answerability. Answerability measures whether the question has a unique answer that can be obtained through the available action space. Annotators verify that the required evidence can be acquired through locomotion, viewpoint control, or manipulation within the step budget. A task is rejected if the relevant object is never observable, if the answer depends on subjective interpretation, if multiple answers are equally plausible, or if the available actions cannot reveal the necessary evidence. For example, a question about whether an object is inside a closed container is answerable only if the agent can inspect, open, move, pour, or otherwise interact with the relevant objects in a way that reveals the hidden state.

Non-triviality. Non-triviality measures whether the task genuinely requires embodied evidence acquisition. Annotators reject tasks that can be answered from the initial observation alone, from obvious category priors, or from superficial visual cues unrelated to the intended spatial reasoning. For example, an occlusion task is rejected if the hidden object is already fully visible at a perfect angle in the first frame. The goal is to retain tasks where the agent must actively decide what information to acquire.

Each instance is independently reviewed by 3 annotators. For each verification axis, annotators assign a binary pass/fail label and may provide a short rejection reason. An instance is retained only if it passes all three axes. When annotators disagree, the disagreement is resolved by majority vote. Rejected instances are removed from the benchmark and are not used in evaluation.

F.2 Inter-Annotator Agreement and Rejection Analysis

To further quantify the reliability of human verification, we measure agreement on the answerability criterion using both raw percent agreement and Fleiss' κ. Table 4 shows substantial agreement across all categories, with particularly high agreement for Physical Structure, Physical Dynamics, and Action Sequencing. This indicates that whether a task admits a unique, obtainable answer can be assessed consistently by independent annotators.

Category	Agreement (%)	Fleiss' κ
Perceptual Grounding	84.8	0.68
Physical Structure	95.2	0.91
Physical Dynamics	92.4	0.85
Specular Reflection	81.9	0.61
Spatial Relations	88.7	0.76
Metric Comparison	87.6	0.73
Cognitive Mapping	83.5	0.65
Enumerative Perception	87.0	0.72
Temporal Understanding	89.2	0.77
Action Sequencing	93.1	0.87

Table 4: Inter-annotator agreement on task answerability.

We also analyze why candidate tasks are rejected. Table 5 reports the distribution of primary rejection reasons within each category. Most rejections arise from practical scene-construction constraints such as unsuitable object sizes, limited interaction space, object ambiguity, or physical infeasibility. Multiple-answer ambiguity is concentrated in categories where alternative physical configurations or routes can arise, rather than being widespread across the benchmark.

Category	Multiple Ans.	Obj. Size	Limited Space	Obj. Ambig.	Room Ambig.	Phys. Infeas.	Question Qual.	Others
Perceptual Grounding	0	26	21	31	0	4	13	5
Physical Structure	6	28	18	9	4	23	8	4
Physical Dynamics	10	13	19	13	0	29	9	7
Specular Reflection	0	32	16	26	0	3	19	4
Spatial Relations	0	21	28	14	0	16	13	8
Metric Comparison	0	39	16	18	0	13	10	4
Cognitive Mapping	19	0	16	5	31	19	6	4
Enumerative Perception	0	25	11	29	8	12	13	2
Temporal Understanding	0	18	29	11	6	4	25	7
Action Sequencing	5	11	41	6	0	23	10	4

Table 5: Primary rejection reasons for candidate tasks, reported as percentages within each category.

Among candidates rejected specifically because multiple valid answers were plausible, most are removed automatically by simulator-based verification before human review (Table 6). Across the affected categories, 123 cases are automatically rejected and 18 are rejected during human verification, for 141 total multiple-answer rejections.

Category	Self-Reject	Human Reject	Total
Physical Structure	28	5	33
Physical Dynamics	35	3	38
Cognitive Mapping	46	7	53
Action Sequencing	14	3	17
Total	123	18	141

Table 6: Candidate tasks rejected because multiple valid answers were plausible.

F.3 Verification Scores for GPT-5-Generated Tasks

Table 7 reports category-level human verification scores for GPT-5-generated task proposals after simulator filtering. Correctness, answerability, and non-triviality are reported separately, and *Overall*

Pass denotes the percentage of instances that pass all three checks. The high verification scores indicate that GPT-5-generated proposals are generally consistent with simulator state, answerable through embodied interaction, and non-trivial after the validation pipeline.

Category	Correctness	Answerability	Non-triviality	Overall Pass
Perceptual Grounding	94.7	93.8	91.6	88.9
Physical Structure	91.3	90.5	88.2	84.6
Physical Dynamics	89.8	88.7	86.9	82.1
Specular Reflection	93.4	92.1	89.5	86.7
Spatial Relations	92.6	91.4	87.8	85.2
Metric Comparison	95.1	94.3	90.7	89.4
Cognitive Mapping	96.4	95.8	92.6	91.2
Enumerative Perception	87.9	86.5	83.9	78.6
Temporal Understanding	90.7	89.9	87.1	83.6
Action Sequencing	88.5	87.3	85.4	80.9
Average	92.0	91.0	88.4	85.1

Table 7: Human verification scores for GPT-5-generated task proposals after simulator filtering. Correctness measures physical and semantic consistency with the simulator state; answerability measures whether the question has a unique answer obtainable through the available action space; non-triviality measures whether the task requires embodied evidence acquisition rather than being solvable from the initial observation or priors. Overall pass denotes the percentage of instances that pass all three checks.

The verification scores also reveal where automatic task proposal is most challenging. Categories involving physical interaction and hidden state changes, such as Physical Dynamics, Enumerative Perception, Temporal Understanding, and Action Sequencing, have lower overall pass rates than categories such as Cognitive Mapping and Metric Comparison. This is expected because these categories require more precise control over physical state, occlusion, object interaction, or temporal dependencies. Human verification therefore plays an important role in removing invalid, ambiguous, or insufficiently interactive instances before evaluation.

F.4 Bias Audit for GPT-5-Generated Tasks

A central concern for LLM-assisted benchmark construction is that generated tasks may contain systematic artifacts. In ESI-Bench, such artifacts could appear in two forms. First, **linguistic bias**: question wording might reveal the answer. Second, **object-category bias**: particular object categories might correlate with particular labels. We therefore evaluate shortcut baselines and dataset-level diversity statistics.

Linguistic bias. We test whether task questions alone contain enough information to predict the answer. For each task, we remove all visual observations, simulator metadata, and action history, and provide only the natural-language question. If GPT-5-generated questions contained strong linguistic artifacts, question-only accuracy would be substantially above majority baselines. Instead, question-only performance remains close to chance and below passive perception, indicating that question wording alone is insufficient to solve the tasks.

Object-category bias. We next test whether task-relevant object categories reveal the answer. We provide the model with the question and the unordered list of task-relevant object categories, but no object positions, rendered observations, action history, or simulator state. This metadata-only setting tests whether certain objects are systematically associated with particular answers. Metadata-only performance is slightly higher than question-only performance, as expected, because object categories provide weak priors about plausible physical interactions. However, it remains below passive single-view performance and far below active exploration, indicating that object identity alone does not provide a reliable shortcut.

Table 8 shows a clear separation between shortcut baselines, passive perception, and active exploration. The average question-only accuracy is 36.6%, and metadata-only accuracy is 39.3%, both below passive single-view accuracy at 42.5%. Active exploration further improves to 56.9%. This ordering

Category	Question-Only	Metadata-Only	Passive Single	Active
Perceptual Grounding	26.1	28.3	32.8	66.3
Physical Structure	31.5	35.0	50.7	64.1
Physical Dynamics	49.2	51.0	51.9	81.2
Specular Reflection	47.2	49.5	47.2	50.4
Spatial Relations	30.8	35.1	45.4	54.8
Metric Comparison	49.2	50.8	48.5	64.1
Cognitive Mapping	47.6	50.1	53.9	57.9
Enumerative Perception	24.3	26.1	13.0	22.6
Temporal Understanding	35.7	38.1	37.4	53.0
Action Sequencing	24.8	29.3	44.2	54.5
Average	36.6	39.3	42.5	56.9

Table 8: Shortcut baseline analysis for GPT-5-generated tasks. Question-only uses only the natural-language question. Metadata-only uses the question and unordered task-relevant object categories, but no observations, positions, action history, or simulator state. Passive Single and Active report Gemini 3.1 category-level averages from the main evaluation. Shortcut baselines remain below active exploration, suggesting that benchmark performance is not explained by linguistic or object-category artifacts.

indicates that language and object-category priors do not explain the benchmark results: visual evidence is needed even for passive performance, and action-guided evidence acquisition provides an additional gain beyond passive observation. The only exception is Enumerative Perception, where passive single-view is especially low because counting tasks often require resolving occlusion, enclosure, or merged observations; nevertheless, active exploration still substantially outperforms all shortcut baselines.

Category	Answer Balance	Object Diversity
Perceptual Grounding	0.92	0.87
Physical Structure	0.89	0.84
Physical Dynamics	0.95	0.82
Specular Reflection	0.93	0.85
Spatial Relations	0.90	0.88
Metric Comparison	0.96	0.86
Cognitive Mapping	0.94	0.80
Enumerative Perception	0.88	0.89
Temporal Understanding	0.91	0.83
Action Sequencing	0.87	0.81
Average	0.92	0.85

Table 9: Bias and diversity statistics for GPT-5-generated tasks. Answer balance is normalized answer entropy; object diversity is normalized entropy over task-relevant object categories. Higher values indicate less concentration around a single answer or object set.

Table 9 further shows that GPT-5-generated tasks are not dominated by a small set of labels, objects, or spatial templates. Answer distributions are balanced across categories, with an average normalized entropy of 0.92. Object diversity is also high, reflecting the use of BEHAVIOR-1K object inventories, random 200-category subsampling, and category-specific filtering. However, even these categories retain substantial variation in relative placement and action requirements.

F.5 Comparison with Human-Generated Tasks

As an additional check, we compare GPT-5-generated tasks with a matched subset of human-generated tasks. Human annotators are given the same category definitions, action space, and simulator scene context, and are asked to author natural-language questions that require embodied spatial reasoning. We then construct matched GPT-5-generated and human-generated subsets with the same category distribution and evaluate representative agents on both.

The purpose of this comparison is not to use human-generated tasks as an additional benchmark split, but to test whether GPT-5-generated tasks introduce a systematic difficulty bias. If GPT-5-generated tasks were substantially easier or harder than human-authored ones, we would expect large performance differences between the two subsets. Instead, as shown in Table 10, model accuracies are broadly similar across task sources, with average gaps within a few percentage points.

Model	GPT-5-Generated	Human-Generated	Absolute Gap
GPT-5 Active	51.6	48.9	2.7
Gemini 3.1 Active	56.9	53.4	3.5
VGGT + Gemini Active	51.6	48.1	3.5
GT 3D + Gemini Active	68.0	63.6	4.4
Human Active	75.3	72.1	3.2
Average	60.7	57.2	3.5

Table 10: Accuracy comparison between GPT-5-generated and human-generated task subsets under active exploration. The gaps remain within a few percentage points, suggesting that GPT-5-generated tasks have similar difficulty to human-generated tasks.

Overall, these results support three conclusions. First, GPT-5-generated task proposals achieve high human verification scores after simulator filtering, indicating that they are generally correct, answerable, and non-trivial. Second, question-only and metadata-only baselines remain below passive single-view performance and far below active exploration, suggesting that benchmark performance cannot be explained by linguistic or object-category shortcuts. Third, model accuracies on GPT-5-generated and human-generated subsets are similar, with differences mostly within a few percentage points. This suggests that GPT-5 can serve as a scalable proposal mechanism without substantially distorting benchmark difficulty, provided that generated tasks are filtered through simulator execution and human verification.

G GPT-6 3D Experiments

As GPT-6 was released only a few days before the ICLR submission deadline, we prioritized evaluating its 2D performance. Nevertheless, we also conducted a preliminary evaluation on selected ESI-Bench subcategories using GPT-6 for both 3D scene reconstruction and spatial reasoning under active exploration setting. As shown in Table 11, the preliminary results suggest that stronger models can improve 3D-based spatial reasoning, occasionally outperforming image-based GPT-6. However, performance remains substantially below ground-truth 3D representations on several geometry-intensive tasks, suggesting that SOTA models are still not fully satisfactory when it comes to accurate 3D reconstruction - although they show cool results.

Table 11: **Illustrative 3D baseline experiment planning.** Active exploration performance (%) across selected ESI-Bench subcategories. The 3D + GPT-6 results are hypothetical planning values, not experimental measurements.

Subcategory	GPT-6 2D	VGGT + Gemini	GPT-6 3D	GT 3D + Gemini
Geometric Configuration	23.0	18.6	30.2	88.0
Structural Enclosure	25.0	12.5	27.4	50.0
Spatial Segmentation	33.3	23.3	26.8	66.7
Counting w Occlusion	20.0	13.3	17.6	83.3
Linear Alignment	58.5	53.2	60.3	84.0
Merged Observation	61.7	55.0	58.7	93.3
Dimensional Size	69.5	59.3	65.2	69.9

H Diverse Emergent Behaviors in Active Exploration

Active exploration not only improves accuracy, but also reveals that agents often develop diverse ways of resolving the same type of spatial uncertainty. Since ESI-Bench does not provide category-

Subcategory	# Emergent Strategy Clusters
Partial Occlusion	4
Material Transparency	6
Rigid Containment	3
Liquid Volume	3
Physical Contact	4
Spatial Distance	3
Dimensional Size	4

Table 12: Human-annotated diversity of emergent behaviors in successful active trajectories. For each representative subcategory, annotators group trajectories into distinct behavioral clusters. The table reports the number of clusters discovered, without imposing predefined strategy labels.

specific exploration policies, prescribed trajectories, or step-by-step instructions, these behaviors are not hand-coded. They emerge from the interaction between the question, the current egocentric observation, and the embodied action space.

To quantify this diversity, we conduct a human annotation study over successful active trajectories. For each representative subcategory, annotators group trajectories according to their high-level behavioral pattern, without being given predefined strategy labels. We then report the number of distinct strategy clusters that emerge within each subcategory. A larger number indicates that agents found multiple qualitatively different behavioral routes to acquire the evidence needed for the same class of question.

Table 12 shows that emergent behavior diversity varies substantially across subcategories. They admit multiple possible routes to the answer: the agent can acquire evidence through different viewpoints, different interactions, or different action orders. This suggests that active exploration is not merely executing a single canonical policy, but can produce several self-developed behavioral patterns for the same underlying spatial problem.

This analysis complements the aggregate accuracy results. Accuracy shows that active exploration helps; strategy clustering shows that the improvement is not simply due to receiving more images. Agents can develop multiple behavioral routes for acquiring task-relevant evidence, even without being told how to explore. At the same time, diversity alone does not imply human-like epistemic control. As discussed in Experiments section, models may still commit prematurely or fail to revise beliefs under contradiction.

I Failure Analysis of VGGT-Based 3D Augmentation

The VGGT-based 3D augmentation results raise an important question: why can reconstructed 3D scene graphs hurt performance even when explicit 3D information should in principle help? In our experiments, `VGGT + Gemini` underperforms the 2D Gemini baseline on several fine-grained spatial tasks, with the most striking case appearing in Counting w Occlusion. We attribute this degradation to errors introduced during 3D reconstruction and scene-graph construction, which can corrupt the symbolic input provided to the language model.

We manually inspect representative `VGGT + Gemini` failures and observe three recurring error modes. The dominant failure is **object duplication**: partially observed or depth-ambiguous objects are reconstructed as multiple fragments or repeated instances, causing the downstream scene graph to over-count objects. This is especially harmful for Enumerative Perception tasks, where the final answer depends directly on the number of object instances. A second failure mode is **object hallucination**: noisy geometry occasionally produces spurious object proposals or assigns object labels to background fragments. A third failure mode is **spatial-relation corruption**: inaccurate depth estimates distort relative positions, contact relations, containment, and near/far comparisons, leading the language model to reason over an incorrect spatial structure.

These errors suggest that the poor performance of `VGGT + Gemini` is not caused by 3D information being unhelpful in general. Rather, it reflects a mismatch between the precision required by ESI-Bench and the reliability of current reconstructed scene graphs. Ground-truth 3D representations

Failure Mode	Main Effect	Affected Tasks	%
Object duplication	One partially observed object is reconstructed as multiple fragments or repeated instances, causing over-counting.	Counting w Occlusion; Merged Observation	41
Object hallucination	Noisy geometry or background fragments are converted into spurious object proposals.	Enumerative Perception; Perceptual Grounding	31
Spatial-relation corruption	Depth errors distort relative position, contact, containment, and near/far relations.	Spatial Relations; Metric Comparison; Physical Structure	28

Table 13: Common failure modes of VGGT-based 3D augmentation. Manual inspection attributes 41% of representative failures to object duplication, 31% to object hallucination, and 28% to spatial-relation corruption.

improve performance on several depth-sensitive tasks, showing that accurate 3D structure can be beneficial. However, when reconstruction is noisy, the resulting scene graph can become more misleading than a 2D observation, because the language model tends to treat the graph as a reliable symbolic description of the scene. This highlights a key challenge for 3D-augmented MLLMs: improving not only reconstruction quality, but also uncertainty-aware scene-graph construction that can distinguish reliable 3D evidence from ambiguous or fragmented geometry.

J Operationalizing the Metacognitive Gap

Our main experiments identify a metacognitive gap between humans and current MLLMs: humans tend to seek falsifying viewpoints, delay commitment under ambiguity, and revise beliefs when contradicted, whereas models often stop early with high confidence. In the main paper, we analyze this gap qualitatively through representative trajectories. Here we clarify how this behavior can be operationalized using trajectory-level measures.

We define three measurable axes of metacognitive behavior. First, **evidence sufficiency** measures whether the agent gathers sufficiently diverse observations before answering. This can be approximated by the diversity of viewpoints visited and whether the agent acquires observations that directly reveal the task-relevant evidence. Second, **falsification seeking** measures whether the agent selects actions that could disconfirm its current hypothesis, rather than only actions that preserve or confirm the initial belief. This can be estimated by annotating whether a new view is redundant, confirmatory, or contrastive with respect to the previous observation. Third, **belief revision** measures whether the agent changes its answer after later observations contradict earlier impressions.

Axis	Definition	Signal
Evidence sufficiency	Whether the agent gathers sufficiently diverse observations before answering.	View diversity
Falsification seeking	Whether actions seek evidence that could disconfirm the current hypothesis.	Contrastive views
Belief revision	Whether the agent changes its answer after contradictory evidence.	Answer updates

Table 14: Operational measures for the metacognitive gap observed in ESI-Bench.

We further compute these measures on active trajectories. **View diversity** is the percentage of trajectories containing observations from multiple meaningfully distinct viewpoints. **Contrastive view rate** is the percentage of trajectories containing at least one observation that disambiguates between competing hypotheses or reveals evidence against the initial hypothesis. **Belief revision rate** is the percentage of trajectories where the predicted answer changes after additional observations.

Table 15 shows that humans acquire more diverse viewpoints, seek contrastive evidence more often, and revise their beliefs more frequently after new observations. In contrast, models exhibit lower

Agent	View Diversity	Contrastive View Rate	Belief Revision Rate
Human Active	71.8	62.7	41.3
Gemini 3.1 Active	43.5	31.5	18.9
GPT-5 Active	39.2	28.7	16.4

Table 15: Trajectory-level measures of the metacognitive gap. Humans acquire more diverse and contrastive evidence, and revise beliefs more often than current MLLMs. All values are percentages.

rates of falsification-seeking behavior and belief revision. These measures are complementary to accuracy: accuracy captures whether the final answer is correct, while metacognitive metrics capture whether the agent knows when its evidence is insufficient and how to acquire better evidence. A full benchmark for metacognition is beyond the scope of this work, but ESI-BENCH provides the interaction traces needed to study these behaviors systematically.

K ANALYSIS OF PASSIVE MULTI-VIEW EVALUATION

Our experiments show that passive multi-view input often fails to improve over passive single-view input, and in some categories even degrades performance. This finding should be interpreted carefully. The passive multi-view baseline uses 30 randomly sampled views to match the active exploration step budget, but it is not intended to represent an optimal multi-view reasoning system. Instead, it asks whether simply providing more unselected observations is sufficient to close the gap between passive perception and active evidence acquisition.

The degradation can arise from several sources. First, **view selection randomness**: random views may miss the diagnostic region while still adding many irrelevant images. Second, **image overload**: current MLLMs may struggle to integrate 30 views, especially when only a few contain task-relevant evidence. Third, **lack of temporal grounding**: passive views are unordered and not tied to the agent's own actions, making it harder to infer how observations relate to spatial movement. Fourth, **conflicting evidence**: redundant or partially occluded views may introduce apparent contradictions that distract the model from the useful observation.

Source	Possible Effect
Random view selection	Views may cover the scene broadly but miss the diagnostic evidence needed for the question.
Image overload	Many irrelevant images can dilute the few useful observations and increase reasoning burden.
No action grounding	Views are not connected to self-selected actions, making spatial integration harder.
Conflicting evidence	Occluded or partial views can introduce misleading cues across images.

Table 16: Possible reasons why passive multi-view input may fail to improve over passive single-view input.

This analysis suggests that the weakness of passive multi-view should not be read as evidence that multi-view reasoning is inherently unhelpful. A stronger baseline could use learned view selection, oracle-selected diagnostic views, view ranking, or explicit cross-view spatial alignment. Our results instead show that unstructured view accumulation is not a substitute for active exploration: the agent must not only receive more observations, but decide which observations are worth acquiring.

K.1 NUMBER OF PASSIVE VIEWS

The 30-view Passive Multi-View setting matches the observation budget used by Active Exploration, but this does not by itself establish whether 30 passive images are optimal. We therefore vary the number of randomly sampled passive views from 10 to 50. Performance improves substantially from 10 to 20 views, but saturates thereafter and slightly degrades at 40–50 views (Table 17). Thus, the

gap is not explained by choosing an unusually small passive observation budget; simply supplying more random views does not continue to improve performance.

Views	10	20	30	40	50
All (%)	28.5	39.0	39.7	37.7	38.4

Table 17: Passive Multi-View image-count ablation.

K.2 Stronger Passive-to-Oracle Baselines

We further expand the passive comparison from the original baselines to a spectrum of stronger viewpoint-selection strategies. **Random Camera Pose** independently samples 30 camera positions and orientations. **Random Trajectory** is the original Passive Multi-View baseline. **Full Scene Coverage** places 30 cameras evenly around the scene rim and points them toward the scene center. **Max-Information Gain–Scene** greedily selects viewpoints that maximize newly revealed scene content using OmniGibson visibility. **Max-Information Gain–Object** instead maximizes newly revealed geometry of task-relevant candidate objects. **Pose-Grounded Oracle** uses known task-object bounding boxes to place cameras around the relevant objects. **Ground-Truth Passive Trajectory** renders observations along the expert-verified task-solving trajectory. Except for the pure-random baselines, these settings use progressively more privileged scene or task information and should therefore be interpreted as increasingly strong oracle references rather than realistic passive observers.

Category	**Rand. Pose**	**Rand. Traj.**	**Coverage**	**Max-Info Scene**	**Max-Info Obj.**	**Pose Oracle**	**GT Passive**	**Active**
Perceptual Grounding	35.5	33.1	42.5	36.9	64.5	84.6	91.2	67.0
Physical Structure	51.1	54.3	55.5	47.1	59.5	70.4	85.4	65.7
Physical Dynamics	48.7	46.7	50.7	42.0	78.7	82.0	86.0	80.7
Specular Reflection	45.3	48.0	47.3	41.3	48.3	51.7	52.1	50.4
Spatial Relations	31.3	34.6	33.7	34.5	39.6	43.8	58.0	45.2
Metric Comparison	42.0	45.7	49.2	38.6	63.3	71.5	81.5	64.3
Cognitive Mapping	53.9	52.7	55.8	52.7	56.9	57.3	67.3	58.5
Enumerative Perception	17.0	18.1	28.5	17.0	25.2	52.2	60.7	25.6
Temporal Understanding	34.2	37.0	40.6	40.2	47.3	66.6	70.2	52.7
Action Sequencing	44.2	46.8	45.5	39.0	46.8	59.7	72.7	54.5

Table 18: Expanded passive-to-oracle baselines. All values are accuracy percentages. Coverage-oriented passive strategies remain below Active Exploration overall, while task-relevant oracle information substantially closes the gap.

These results sharpen the interpretation of the passive comparison. Active Exploration consistently outperforms purely random passive observation. Scene-level coverage alone does not recover the gap: Full Scene Coverage and Max-Information Gain–Scene often remain substantially below Active Exploration. In contrast, once viewpoint selection is given task-relevant oracle information, Max-Information Gain–Object and Pose-Grounded Oracle close much of the gap and can exceed Active Exploration. The key distinction is therefore not simply the number of observations, but whether the acquired observations are relevant to the task.

L Why a Synthetic Simulator Benchmark?

ESI-Bench is built in simulation rather than collected directly in the real world because the goal of the benchmark is controlled diagnosis of embodied spatial intelligence. Many of our tasks depend on hidden physical or spatial variables—containment, contact, fill level, occlusion, room connectivity, object count, and unobserved state changes—whose ground truth must be known exactly. In real-world data, these variables are difficult to label exhaustively and often require additional sensors, manual inspection, or assumptions about the scene. In contrast, simulation provides direct access to object poses, segmentation masks, contact flags, containment relations, room assignments, and physical states, allowing us to construct unambiguous labels and reject invalid instances.

Simulation also allows us to control the source of difficulty. ESI-Bench is designed to test whether an agent knows *what evidence to acquire*, not whether it can handle uncontrolled sensor noise,

imperfect calibration, or real-world deployment constraints. By using OmniGibson and BEHAVIOR-1K scenes, we can systematically vary occlusion, viewpoint ambiguity, object placement, physical interaction, and action requirements while keeping the task answer verifiable. This makes it possible to separate perception errors from action-selection errors through diagnostic settings such as passive single-view, passive multi-view, active exploration, and ground-truth passive trajectories.

We do not claim that simulator performance directly transfers to real-world robotics. Rather, ESI-BENCH serves as a controlled testbed for identifying failure modes that would be difficult to isolate in the real world. A real-world version would introduce additional challenges such as sensor noise, imperfect actuation, reconstruction errors, and environment variability. These are important directions, but they would obscure the central question studied here: whether current MLLMs can close the perception-action loop by actively selecting informative observations. The synthetic simulator setting therefore provides a necessary first step toward rigorous, reproducible, and fine-grained evaluation of embodied spatial intelligence.

M ACTIVE AGENT PROMPTING AND ACTION SELECTION

In the active exploration setting, the evaluated MLLM controls the agent through the same discrete action space used by human participants. At each timestep, the model receives the task question, the current egocentric observation, the previous action-observation history, and the full action vocabulary. It is then prompted to select exactly one next action. The selected action is executed in OmniGibson, producing the next egocentric observation. This loop continues until the model issues the terminal `answer(`$\hat{y}$`, `c`)` action or reaches the maximum budget of $T_{\max} = 30$ steps.

Action-selection prompt. We use the following prompt template for active exploration. The same template is used across task categories and models, with only the task question, observation history, and available objects filled in from the current instance.

Active Exploration Prompt

You are an embodied agent in a 3D environment. Your goal is to answer the question by actively exploring the scene.
You may take one action at each step. Use actions only when they help you gather evidence.
Do not answer until you have enough evidence.
Question: {question}
Current observation: You are given the current egocentric image.
Action history: {previous actions and brief observation summaries}
Available actions:

- `move_forward`, `move_backward`, `move_left`, `move_right`, `move_up`, `move_down`
- `turn_left`, `turn_right`, `turn_up`, `turn_down`
- `pick_up(obj)`
- `put(obj, inside, obj)`
- `put(obj, on, obj)`
- `fill(obj, with_water)`
- `pour(obj, from, obj, to, obj)`
- `answer(answer, confidence)`

Choose exactly one next action from the available action space.
Return your response in the following format:
`Reason:` briefly explain what evidence you need next.
`Action:` one valid action.

Action parsing and execution. Model outputs are parsed into the nearest valid action string. If the output contains an invalid action, malformed object reference, or multiple actions, we reprompt the model once with the same observation and a reminder to output exactly one valid action. If the

second response remains invalid, the step is counted as invalid and the agent remains in place. For manipulation actions, object names are matched to visible or previously observed object identifiers when possible. Failed physical executions, such as attempting to pick up an unreachable object or placing an object into an incompatible container, are executed as simulator failures and included in the trajectory, since such failures reflect the agent's action-selection behavior.

M.1 Component Ablations and Failure Attribution

The active loop contains several coupled mechanisms beyond action selection: access to interaction history, explicit failure handling, belief updating, and confidence-based stopping. To quantify their contributions, we remove each prompted component individually while keeping the remaining pipeline unchanged. Table 19 shows that all three components contribute, with the largest degradation occurring when belief updating is removed.

Category	Full	w/o History	w/o Failure Handling	w/o Belief Updating
Perceptual Grounding	67.0	60.2	63.3	58.5
Physical Structure	65.7	57.7	61.7	59.1
Physical Dynamics	80.7	75.3	77.3	70.0
Specular Reflection	50.4	44.7	47.0	46.3
Spatial Relations	45.2	39.8	42.2	41.0
Metric Comparison	64.3	59.3	61.4	55.8
Cognitive Mapping	58.5	53.5	54.6	49.2
Enumerative Perception	25.6	22.2	23.3	19.6
Temporal Understanding	52.7	46.3	49.5	48.0
Action Sequencing	54.5	50.7	52.0	47.4

Table 19: Component ablation of the active exploration pipeline.

We complement this intervention with human error attribution on Gemini active trajectories. Annotators assign each failure to action selection, memory/integration, belief updating, or stopping. As shown in Table 20, action selection is the largest single error source in every category, but it is not the only contributor. We therefore interpret "action blindness" as a major contributor to the active-oracle gap rather than a fully isolated explanation; these components are inherently coupled because action selection depends on remembered observations and current beliefs.

Category	Action Selection	Memory	Belief Updating	Stopping
Perceptual Grounding	52	19	23	6
Physical Structure	56	16	21	7
Physical Dynamics	58	13	20	9
Specular Reflection	50	15	29	6
Spatial Relations	54	17	22	7
Metric Comparison	53	18	23	6
Cognitive Mapping	45	33	17	5
Enumerative Perception	51	24	15	10
Temporal Understanding	48	20	26	6
Action Sequencing	61	14	17	8

Table 20: Human attribution of active-exploration failures. Values are percentages of attributed failures within each category.

M.2 Stopping-Threshold Sensitivity

We also vary the confidence threshold used to terminate active exploration. Accuracy peaks at the default threshold of 0.8 (Table 21), while both earlier stopping and substantially more conservative stopping reduce performance. This supports 0.8 as an empirically grounded operating point and indicates that the qualitative overconfidence failures are not an artifact of a single arbitrary threshold.

Threshold	0.5	0.6	0.7	0.8	0.9	0.95	0.99
All (%)	47.82	52.41	55.16	56.12	54.28	50.87	51.64

Table 21: Stopping-threshold sensitivity for Active Exploration.

M.3 Failure Attribution by Action Type

To diagnose the role of the action space more finely, we attribute action-selection failures to perception, locomotion, or manipulation. Table 22 shows strong category-specific structure: Specular Reflection is dominated by perception failures, Cognitive Mapping by locomotion failures, and Physical Structure, Physical Dynamics, and Action Sequencing by manipulation failures. Thus, the benchmark exposes distinct bottlenecks within the broader action-selection problem rather than treating all embodied actions as interchangeable viewpoint changes.

Category	**Perception**	**Locomotion**	**Manipulation**
Perceptual Grounding	49	39	12
Physical Structure	21	12	67
Physical Dynamics	20	12	68
Specular Reflection	58	27	15
Spatial Relations	46	46	8
Metric Comparison	39	47	14
Cognitive Mapping	37	58	5
Enumerative Perception	39	44	17
Temporal Understanding	36	39	25
Action Sequencing	28	16	56

Table 22: Failure attribution across perception, locomotion, and manipulation actions. Values are percentages.

M.4 Open-Source Models and Task-Specific Fine-Tuning

The main benchmark is designed for zero-shot evaluation of general-purpose MLLMs. To test whether the observed difficulty is merely due to lack of exposure to the benchmark-specific action space, we additionally evaluate two open-source 4B models before and after task-specific fine-tuning. Fine-tuning produces mixed category-level changes rather than a consistent improvement (Table 23). We therefore treat this as a proof-of-concept analysis rather than the primary protocol: larger-scale embodied training may improve these capabilities, but the current results suggest that the benchmark is not trivially solved by limited task-specific adaptation.

Category	**Qwen3-VL 4B**	**Qwen3-VL 4B FT**	**Gemma 3 4B IT**	**Gemma 3 4B FT**
Perceptual Grounding	21.9	20.1	8.8	9.9
Physical Structure	44.6	42.7	43.0	44.3
Physical Dynamics	36.7	39.3	33.3	36.0
Specular Reflection	16.7	19.3	9.0	9.3
Spatial Relations	40.7	39.3	18.6	21.9
Metric Comparison	41.4	42.6	31.4	28.8
Cognitive Mapping	28.1	27.3	24.2	23.1
Enumerative Perception	23.0	23.0	22.2	21.5
Temporal Understanding	36.7	31.7	33.8	34.5
Action Sequencing	16.9	19.5	16.9	15.6

Table 23: Active Exploration results for open-source models before and after task-specific fine-tuning. All values are percentages.

N WHY HIGH-LEVEL ACTIONS?

ESI-BENCH uses a high-level discrete action space rather than low-level motor control because the benchmark is designed to diagnose embodied spatial reasoning, not visuomotor control. Many tasks require agents to decide *what evidence to acquire*: whether to move to another viewpoint, inspect behind an occluder, manipulate a container, compare distances, or test a physical relation. These decisions operate at the level of information-seeking behavior. Introducing low-level control would add additional sources of failure, such as grasp instability, collision recovery, locomotion drift, controller tuning, and actuation noise, making it difficult to determine whether an error comes from spatial reasoning or motor execution.

The high-level action space therefore acts as a controlled abstraction. It preserves the core perception-action loop—agents still choose where to move, what to inspect, what to manipulate, and when to answer—while reducing confounds from robot-specific control policies. This makes the evaluation more diagnostic: when an agent fails, the failure is more likely to reflect poor evidence acquisition, action selection, belief updating, or spatial reasoning, rather than low-level controller failure.

We acknowledge that low-level control is an important dimension of embodied intelligence. However, it is orthogonal to the main question studied in ESI-BENCH. A low-level version of the benchmark could evaluate whether agents can translate spatial plans into continuous motor commands, handle imperfect actuation, and recover from execution errors. We view this as a complementary direction. ESI-BENCH intentionally factors out low-level control so that the benchmark can focus on the higher-level spatial competence required to close the perception-action loop.

O STEP BUDGET ABLATION

We analyze how the maximum exploration budget affects active performance. Since active exploration gives the agent additional opportunities to acquire evidence, it is important to check whether performance continues to improve simply by allowing more steps. We therefore evaluate Gemini 3.1 Active with step budgets from 5 to 50, increasing the budget in intervals of 5 steps.

Figure 9 shows that performance improves rapidly from 5 to 15 steps, begins to saturate around 15–20 steps, and becomes nearly flat after 30 steps. Increasing the budget beyond 40 steps does not improve accuracy and even leads to a slight decrease. This suggests that the useful part of active exploration is not unbounded observation accumulation, but targeted evidence acquisition within the first few informative actions. Once the agent has acquired the relevant views or interactions, additional steps mostly introduce redundant or distracting observations.

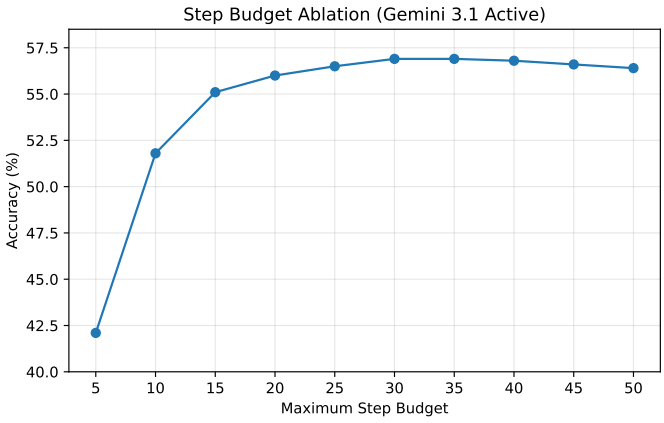


Figure 9: Step budget ablation for Gemini 3.1 Active. Performance rises quickly up to 15–20 steps, saturates around 30 steps, and slightly decreases after 40 steps.

This saturation pattern motivates our choice of $T_{\max} = 30$ in the main experiments. A 30-step budget is large enough to support multi-step navigation, viewpoint adjustment, and manipulation, while

avoiding unnecessarily long trajectories that add computational cost without improving performance. The slight degradation at larger budgets also suggests that longer exploration is not always beneficial: models may revisit redundant views, accumulate conflicting evidence, shift too much from task-relevant regions, or delay commitment even after sufficient information has already been observed.

P PER-CATEGORY TASK CONSTRUCTION DETAILS

Each task category in ESI-BENCH is instantiated through a dedicated pipeline script that (i) selects task-relevant objects, (ii) places them in a physically valid configuration, (iii) renders a fixed set of egocentric camera views, and (iv) records ground-truth labels together with per-view segmentation visibility flags. All scripts share a common infrastructure: OmniGibson scenes are loaded restricted to a single room, object categories are drawn from a 200-category subsample of the BEHAVIOR-1K inventory, GPT-5 is queried for task-appropriate category selection where semantic judgment is required, and physics is settled for a fixed number of simulation steps before any rendering or label extraction occurs.

Initial Agent Viewpoint Setup. The initial agent viewpoint for each category is constructed to be task-diagnostic: for containment tasks, the agent camera is placed collinear with the container and object so that the object is occluded behind or inside the container from view 0; for distance and size tasks, the initial azimuth is set along the direction of one of the objects or reference-anchored object, creating forced-perspective ambiguity; for contact tasks, the agent views the contact zone head-on where the gap is smallest; for stacking tasks, the initial agent sensor is placed at near-ground height where vertical ordering is hardest to judge; and for geometric configuration tasks, the initial view is taken from the centroid at grazing height where collinearity and triangle shape are indistinguishable. In each case the initial observation is deliberately insufficient to resolve the question, motivating the need for active exploration. **Instances where task-relevant objects are not visible in any rendered view under the per-view segmentation check described above are discarded.**

Below we describe the construction procedure for each of the ten task categories.

P.1 PHYSICAL STRUCTURE: MATERIAL TRANSPARENCY

Object selection. A transparent container is sampled uniformly at random from a human-curated list of container categories; its model instance is drawn from the BEHAVIOR-1K inventory. A small object is sampled from a separate human-curated list, with model resolution via the inventory.

Scale. The small object is rescaled so that its largest bounding-box dimension equals $\min(\text{Uniform}(l, u),\ 0.5 \times d_{\min}^{\text{container}})$ metres, where $d_{\min}^{\text{container}}$ is the smallest of the container's three inventory bounding-box dimensions at unit scale, ensuring the object is always physically smaller than the container's narrowest cross-section.

Placement. The container is placed on any surface object whose name contains "table", falling back to the room floor. For positive instances, containment is attempted via OmniGibson's `Inside` object state; if that fails, the small object is teleported to 25% of the container's interior height and settled for 30 simulation steps, with a bounding-box containment check applied. If the directly behind position is occupied, up to 8 lateral offsets of 0.05 m each are tried with vacancy checks against all scene-graph bounding boxes; a live-AABB correction pass then adjusts the final position so the gap is exactly 0.01 m. Instances that fail placement at both attempts (positive) or all lateral candidates (negative) are discarded.

Agent intitial spotion and view. For positive instances the initial azimuth is fixed along $-Y$; for negative instances it is set opposite to the container-to-small-object axis, so that view 0 always maximises occlusion of the small object behind the container.

Ground-truth label. Binary: *inside* (positive) or *not inside* (negative).

P.2 PHYSICAL STRUCTURE: LIQUID VOLUME

Object selection. Fillable container pairs are selected from a human-precomputed manifest such that their measured particle capacities differ by at least 200 particles (`TARGET_CAP_DIFF`). Capacities are measured offline via a water-dip pipeline: each container is submerged in a particle-based water system simulated with GPU dynamics in BEHAVIOR, particle positions are clustered using a connected-component graph over a KD-tree with radius equal to twice the particle radius, and the largest connected component count is recorded as the fillable capacity.

Pouring Simulation. When water from one container is poured to another, we first fill that container with water particles using BEHAVIOR's dipping system. When it is poured into another container, we transfer the water particles from current container to target container, producing one of two scenarios: complete transfer or overflow.

Ground-truth label. The container with the larger offline-measured particle capacity.

P.3 PHYSICAL STRUCTURE: DEFORMABLE

Object selection. We first use GPT-5 to generate an object pool, and then filter it to retain portable objects with an average radius between 5 and $15\,\mathrm{cm}$. For each question, we randomly sample one target object from this precomputed pool. Three distractor categories are sampled from the remaining categories to form a four-way multiple-choice question. We also sample one usable cloth asset with a side length no larger than $50\,\mathrm{cm}$, prioritizing cloth models marked as recommended.

Object placement. The target object is placed on the selected room floor at a free location with a fixed clearance from nearby obstacles. Candidate positions are sampled from the traversability map when available, and otherwise from free floor locations. The object is dropped slightly above the floor and allowed to settle. The cloth is then positioned above the settled object center and dropped with a downward velocity until it covers the object.

Agent initial view. The main camera is placed at a fixed azimuth around the covered object and points toward the cloth-covered target. Its distance and height are adjusted according to the cloth footprint. We use a fixed field of view of 70°.

Ground-truth label. The ground-truth label is the semantic category of the object hidden under the cloth. The answer choices contain the target category and three distractor categories, shuffled into options A–D.

P.4 PHYSICAL DYNAMICS: INCLINED PLANE

Object selection. A room from the BEHAVIOR-1K scene pool is loaded; candidate object categories are drawn from a room-object compatibility list, excluding floors, ceilings, and walls. A single object is selected and placed on a procedurally generated slope. **The mass of the object is randomly shifted from its original mass.**

Slope construction. A rigid box primitive with half-extents $(x, 0.15, 0.01)$ m is tilted at an angle drawn uniformly from $[10^\circ, 45^\circ]$ from horizontal, where the slope base half-extent x is drawn uniformly from $[0.20, 0.30]$ m. Static friction is drawn uniformly from $[0.3, 3.0]$; dynamic friction is drawn uniformly from $[0.3, f_s]$ where f_s is the sampled static friction, ensuring the physical constraint $f_d \leq f_s$ is always satisfied. Both coefficients are applied to the slope and object via an Isaac Sim physics material with restitution 0.1. The physics timestep is $1/240$ s; action timestep is $1/60$ s.

Simulation and label derivation. When the object is placed at the slope surface and the simulation runs for 30 steps. Slide is detected if the object's XY displacement exceeds 0.03 m; fall is detected if the Z coordinate drops more than 0.05 m below the slope bottom. The ground-truth label (stable / slides / falls) is derived from these thresholds.

Agent initial viewpoint. The agent stands The initial view captures the object stands near the slope on the floor.

Ground-truth label. Stable or not.

P.5 Physical Dynamics: Stacking and Stability

Object selection. GPT-5 receives a 200-category randomly-sampled of the BEHAVIOR-1K inventory (different for every data point) and selects 2 or 3 objects that are "solid, not soft/flexible/liquid, and plausibly stackable."

Scale normalization. All selected objects are scaled to a uniform XY footprint equal to the largest natural XY extent among them, with Z scaled proportionally. This controls for trivial size mismatches and isolates shape-based stability.

Placement. For each permutation, the bottom object is placed on the floor via `sample_kinematics("onTop", floor)`, subsequent objects are placed on top via the same primitive, and physics is settled for 60 steps. Stability is then assessed along two axes. First, an `is_on_top` check verifies that the upper object's AABB minimum Z exceeds the lower object's AABB minimum Z by at least 0.02 m, and that the XY intersection-over-union of the two AABBs is at least 0.1, confirming the upper object is both elevated above and spatially overlapping the lower one. Second, a tilt check computes the dot product of the object's world-up vector with the global $+Z$ axis; a stack layer is considered upright only if this dot product exceeds 0.9, corresponding to a tilt of less than roughly 26° from vertical.

Ground-truth label. The stable stacking order(s) among all permutations tried, derived from `is_on_top()` outcomes after physics settling.

P.6 Metric Comparison: Spatial Distance

Object selection. A reference object already present in the loaded room scene graph is selected based on clearance: all four corners of its AABB are evaluated for clearance to the nearest other object's bounding-box centre; at least two corners with clearance > 0.5 m are required. GPT-5 selects two similar-category objects to serve as the near and far distractors.

Placement. The near object (`obj_near`) is placed m from one qualifying corner outward along the diagonal unit vector. The far object (`obj_far`) is placed with a slightly longer distance m from a second qualifying corner. Both objects use identity quaternion orientation.

Agent initial position and view. The agent is positioned that from its initial view, one object appears much larger than the other and blocking some view, making it hard to deduct distance.

Ground-truth label. `obj_near` (which one is closer) or `obj_far` (which one is far).

P.7 Metric Comparison: Dimensional Size

Object selection. GPT-5 selects a medium-sized household object category (not tiny, not large furniture) from a 200-category random-sampled subsample (different each time). Two instances of the same model are loaded: `task_obj1` at scale 1.0 and `task_obj2` at scale drawn uniformly from $[1.2, 1.3]$.

Reference objects. Two visually distinct reference objects are placed within 0.05 m of each task object's bounding-box boundary, with each reference at least 0.30 m from the other task object's boundary. References serve as unambiguous spatial anchors differentiating the two task objects.

Agent initial position and view. The agent is positioned that from its initial view, one object is much closer than the other and blocking some view, making it hard to deduct size.

Ground-truth label. The reference object to which the object is larger / samller.

P.8 Spatial Relations: Physical Contact

Object selection. GPT-5 selects 2 medium-to-small-sized objects (e.g., teddy bear, book, basketball) from a randomly-sampled 200-category subsample, avoiding tiny items (cups, pens) and large furniture.

Placement. For positive (touching) instances, the two objects are placed in physical contact: `obj2_xmin` $=$ `obj1_xmax` (zero X gap, faces meeting) and `obj2_ymin` $=$ `obj1_ymax` -0.05 m (confirming contact along Y). A small random X drift of ± 0.03 m is applied per instance for positional variety. After placement, a pixel-level mask adjacency check via dilation overlap of the two instance segmentation masks verifies that contact is visually confirmed. For negative (non-touching) instances, the same initial placement is attempted but an iterative correction loop pushes `obj2` outward in X until both the X and Y face-to-face gaps exceed 0.05 m, ensuring a clearly visible separation. In both cases a live-AABB verification pass confirms the final gap values before the instance is accepted.

Initial agent position and view. The agent is placed in a view which can be delusive (e.g., appears contacting when it's not).

Ground-truth label. Binary: the two objects are in contact or not.

P.9 Spatial Relations: Linear Alignment and Geometric Configuration

Object selection. For both Linear Alignment and Geometric Configuration, GPT-5 selects 3 small-footprint objects (table-top or hand-held size, with some Z height) from a random-sampled 200-category subsample, avoiding large furniture, vehicles, and mats.

Linear Alignment placement. The three objects are placed collinearly along a random room axis, spaced uniformly. Positive instances use strict collinearity; negative instances introduce controlled lateral offsets that break alignment from most viewpoints.

Geometric Configuration placement. Objects are placed at the vertices of an equilateral triangle with side length drawn uniformly from $[0.4, 0.65]$ m. A size-compatibility check is applied: if the ratio of the largest to smallest object XY footprint exceeds 2.0, the object set is resampled (up to 5 attempts), preventing degenerate configurations where one object dominates the triangle vertex.

Agent initial position and view. The agent is placed at ground level, $0°$ orbit from the centroid of the three objects, where collinearity and triangle shape are indistinguishable from a single frontal observation.

Ground-truth labels. Linear Alignment: binary (collinear / not collinear). Geometric Configuration: categorical (equilateral / isosceles / random triangle).

P.10 Physical Structure: Rigid Containment

Object selection. GPT-5 selects one fillable object category and three container categories of varying sizes (small, fit, large) from a 200-category subsample. Two additional non-fillable objects are independently sampled and scaled as a pair to jointly fill the large container, with each object's XY extent at the large container. With 50% probability a third additional object is sampled and scaled to fit the small container, giving 3 or 4 objects to be assigned in total, each with a unique target container.

Pick-place simulation. Each object is placed into its target container via a two-stage kinematic pipeline. First, placement is attempted, settling physics for 30 steps and verifying success via a bounding-box containment check (object AABB fully within container AABB). If that fails, a fallback

attempt uses with the same settling and verification. A robot grasp-and-release primitive is executed before each placement to simulate realistic pick-and-place manipulation. Robot eye-camera frames are captured before pickup, before placement, and after placement, providing egocentric views of each manipulation stage.

Ground-truth label. The correct assignment of each object to its target container.

P.11 Action Sequencing: Action Order Inference

Object selection. GPT-5 designs a 2- or 3-object placement hierarchy (e.g., plate → bowl → food item). For each role in the hierarchy, several candidate instances are loaded: one that clearly fits its target and one that barely fails, with scales chosen to make the size difference physically meaningful but visually subtle. Others are confusables.

Layout. All objects are placed on the floor or table, with the fixed object at the centre and candidate pairs spread out left and right.

Ground-truth label. A spatial relation string specifying which candidate should be placed at each step of the correct action sequence.

P.12 Temporal Understanding: Agent Observation

Object selection. GPT-5 selects 1–3 object categories to be placed as hidden objects along a robot navigation path within the scene.

Agent setup. Two agents operate simultaneously: a navigating robot that traverses a planned path through the scene, and a static observer agent fixed at the start position at 5.05.0 5.0 m height with ceiling removed, always looking at the navigating robot. The navigating robot's start and goal positions are sampled from traversable pixels of the BEHAVIOR-1K traversability map; the shortest path is validated to have at least 10 waypoints, with start/goal resampled up to 20 times if the dry-run reachability check fails.

Object placement. Hidden objects are placed at surface-appropriate positions along the confirmed path via `OnTop` state sampling, such that the navigating robot passes within 1.01.0 1.0 m of each.

Observer agent position and view. The static observer agent is fixed at the start position at 5.05.0 5.0 m height with ceiling removed, always oriented to face the navigating robot. It records what objects the navigating robot passes during its trajectory, serving as the ground-truth witness whose observations form the basis of the question posed to the evaluated model.

Ground-truth label. The count and categories of hidden objects the navigating robot passes within the proximity threshold, as observed by the static agent.

P.13 Temporal Understanding: Unobserved Changes

Object selection. We have three question types for this task: change detection, change identification and current-state reasoning. For each question, we sample one or more boxes depending on the task type. Change detection and change identification use a single box, while current-state reasoning uses three colored boxes. Box contents are sampled from a precomputed count-target object pool generated by GPT-5. Depending on the sampled change type, the content may be kept the same, replaced by another category, removed, or added to an initially empty box.

Object placement. Boxes are placed on the selected room floor using the hidden-in-box placement routine. The placement respects the room bounding box, floor geometry, nearby blockers, and traversability constraints. For each phase, the corresponding content object is spawned and placed inside its assigned box; empty boxes contain no object. We involve two rendering phases to represent the time change. Phase 1 is rendered first, then its contents are removed and Phase 2 contents are spawned and rendered.

Agent initial view. Each phase is rendered from a primary viewpoint chosen to maximize visible boxes. Candidate camera positions are sampled around the room boundary and filtered by clearance and traversability. The selected camera looks toward the average box location.

Ground-truth label. The ground-truth label is derived directly from the sampled phase states. For change detection, the answer is whether any box changed. For change identification, the answer is the target box's change type: `replace`, `remove`, `add`, or `no_change`. For current-state reasoning, the answer is the colored box whose content changed. Ground-truth close-up images are rendered by opening the boxes in each phase.

P.14 Perceptual Grounding: Partial Occlusion

Object selection. A target object is sampled from a gpt-generated and human-curated manifest of occlusion-sensitive categories and some confusable categories corresponding to the target categories when occluded. A compatible occluder is sampled from a precomputed compatibility map, ensuring the occluder is semantically plausible alongside the target. The occluder is scaled per-axis so that each of its bounding-box dimensions equals the target's corresponding dimension multiplied by a scale factor of 1.1, making it slightly larger than the target and thus capable of producing meaningful partial occlusion.

Placement. Both objects are placed on a valid table surface via `OnTop` state sampling. The occluder is positioned at a corner of the target's bounding box so that from the canonical hard view it partially obscures the target.

Agent initial viewpoint. The agent is placed at 15° lateral offset from the perfectly collinear occluder-to-target axis, at the same height as the target centre, looking horizontally at the target. This near-collinear position maximises ambiguity: the target is partially hidden but not fully obscured, making it impossible to identify with certainty from the initial view.

Ground-truth label. The category of the partially occluded target object.

P.15 Perceptual Grounding: View Hallucination

Object selection. A single object is sampled from the same human-curated candidate pool used for occlusion tasks. A confusable category map loaded from pre-curated records (1000 categories and their confusable cateogires by GPT), for each category, which other categories it is visually confusable with and under what viewing conditions, providing the basis for the question posed to the evaluated model.

Placement. The object is placed on a randomly sampled support surface (table or floor) via `OnTop` state sampling.

Agent initial viewpoint. The initial agent position is chosen from the set of azimuths at which the object is most view-ambiguous.

Ground-truth label. The true category of the object, contrasted against its confusable alternatives recorded in the metadata.

P.16 Specular Reflection : Reflection Authorization

Object selection. We first use GPT to obtain a list of placeable object categories from the BEHAVIOR-1K object inventory. We then filter out categories that are inherently reflective or structurally fixed, such as mirrors, walls, floors, doors, windows, and ceilings. From the remaining candidates, we retain only objects whose average-volume-derived equivalent spherical radius falls within 0.02–0.20 meter, ensuring that the selected objects are suitable for stable floor placement and mirror-reality reasoning. Finally, for each question, the target object category is randomly sampled using the scene name, room name, and question index as sampling context.

Mirror and initial-view setup. For each question, one mirror is first placed on a free floor position in the target room. The mirror is oriented to approximately face the room center, with a random yaw perturbation of up to 30°. Its pose defines a geometric mirror plane, represented by a mirror center, a 2D normal direction, and a tangent axis. These parameters are used to compute whether an object should appear as a mirror reflection. After the mirror pose is fixed, the initial camera pose is selected based on the mirror placement: the camera is sampled from free floor positions on the front side of the mirror, preferring viewpoints approximately 2.0 m away from the mirror with a small lateral offset and oriented to look at the mirror center. A candidate initial view is accepted only if the line of sight from the camera to the mirror is not occluded by scene objects or non-traversable regions, ensuring that the mirror is clearly visible in the initial image.

Object placement. Each question places one real object on the floor. For positive examples, the object is sampled from positions whose reflected ray intersects the mirror plane within the mirror width and remains visible from the camera through the mirror. The object must lie at least 1.0 m from the mirror and must not be directly visible to the camera. For negative examples, the object is instead placed in a direct-only position: it is visible to the camera in the real world, but the geometric reflection test confirms that it is not visible through the mirror.

Ground-truth label. The ground-truth answer is a binary label indicating whether the queried object is seen through the mirror. Positive questions use the answer "Yes" when the object is mirror-visible but not directly visible. Negative questions use the answer "No" when the object is directly visible but has no valid mirror reflection.

P.17 Specular Reflection : Spatial Relationship

Object selection. We got the same object pool as P.16. For each question, two object categories are randomly sampled from this filtered pool.

Mirror and initial-view setup. We use the same mirror pose setup and initial-view setup as P.16.

Object placement. Each question places two real objects on the floor, denoted as object A and object B. Candidate object positions are sampled from free floor locations whose geometric mirror reflections are visible from the camera. For each candidate, the reflected ray from the camera through the mirror must intersect the mirror plane within the mirror width, remain inside the camera field of view, and pass both object-level and traversability-map occlusion checks. The object must also lie on the valid front side of the mirror and within the allowed mirror-depth range.

Ground-truth label. The ground-truth answer is a categorical label, either "A" or "B". It is computed by comparing the 3D Euclidean distance from the observation position to the real position of object A and object B. If object A is closer or tied, the answer is "A"; otherwise, the answer is "B".

P.18 Specular Reflection : Scene Correspondence

Object selection. We got the same object pool as P.16. For each question, two object categories are randomly sampled from this filtered pool.

Mirror and initial-view setup. We use the same mirror pose setup and initial-view setup as P.16.

Object placement. Each question places three identical real objects on the floor. The target object is sampled along the reflected viewing ray induced by the camera-to-mirror direction, at a distance of approximately 1.0–2.0 m from the mirror. The other two objects are placed on a line approximately perpendicular to this reflected ray, with offsets chosen so that the target object appears as the left, middle, or right object in the rendered image. All three placements must lie inside the valid floor region, avoid collisions with the mirror and scene blockers, and satisfy a minimum separation constraint. A small random yaw jitter is applied to each object to reduce visual degeneracy while preserving category and model identity. All candidate objects will be visible in our view.

Ground-truth label. The ground-truth answer is a categorical label from $\{\text{left}, \text{middle}, \text{right}\}$. It is determined by the rendered-image horizontal position of the target real object, which is the object placed on the reflected mirror ray.

P.19 Enumerative Perception : Counting w Occlusion

Object selection. We first use GPT to obtain a list of placeable common object pool from the BEHAVIOR-1K object inventory. Then we filter the pool by physical size: the representative model must have all bounding-box edges no larger than 0.25 m, and cannot be so tiny that all three edges are below 0.05 m. This produces countable household objects that are small enough to be hidden by ordinary furniture, while remaining visible once the agent moves to a useful viewpoint.

Agent initial viewpoint. The agent position is taken from the input or from the loaded scene. If unavailable, we use the center of the selected floor as a default observer position. Images are rendered from the agent location with a camera height of about 1.25 m above the floor and a downward pitch of $30°$.

Occluder selection and Object placement. We collect candidate occluders from the loaded OmniGibson room using their runtime bounding boxes and categories. We keep objects that are large enough to hide a small target object, such as cabinets, shelves, tables, sofas, counters, and similar furniture.

For each valid occluder, we place the count object behind it with respect to the agent's initial position. Specifically, we compute the direction from the agent to the occluder center, then place the target slightly beyond the occluder along this direction. We also require the line segment from the agent to the target to intersect the occluder's bounding box, so that the target is occluded from the initial viewpoint.

Ground-truth label. The ground-truth label is the number of target objects that are successfully placed in the scene. We give four number options for each question.

P.20 Enumerative Perception : Spatial Segmentation

Object selection. We remain the same object pool in P.19.

Divider selection and Object placement. For this task, we use chair-like objects as natural visual dividers, including chairs, armchairs, stools, and swivel chairs. Structural or non-blocking objects such as floors, walls, doors, windows, lights, and pictures are excluded.

For each valid divider, we place the count object at the horizontal center of the divider's bounding box, on the floor underneath or within the divider region. This creates a case where the object is spatially separated or visually broken up by the surrounding furniture structure.

Agent initial viewpoint. The same agent initial view as P.19.

Ground-truth label. The same ground-truth label as P.19.

P.21 Enumerative Perception : Merged Observation

Object selection. We remain the same object pool in P.19.

Object placement. We create a visually merged counting case by placing multiple instances of the same count object in a compact cluster. We first sample a free floor position from the room, ensuring that it lies inside the valid room region and does not collide with blocking objects.

Starting from this base position, we add the remaining target objects nearby with small random offsets. The offsets are constrained so that neighboring objects stay very close to each other, and in some cases objects are stacked on top of nearby ones. This produces a group where individual instances may visually merge into one another from the agent's viewpoint.

Agent initial viewpoint. The same agent initial view as P.19.

Ground-truth label. The same ground-truth label as P.19.

P.22 Enumerative Perception : Category Ambiguity

Object selection. We first maintain the same objest pool as P.19. To create semantic ambiguity, we compute CLIP similarity scores for all object pairs in the pool. For each object category, we keep its top-3 most similar categories as candidate confusers. For each question, we first sample a counting object category. Then, for each counting object added to the scene, we also add one confuser object, randomly selected from its top-3 candidate confusers.

Object placement. We first sample valid free-floor positions inside the room, requiring each position to lie on the selected floor, avoid collisions with blocking objects, and remain sufficiently far from the agent. For each candidate, we place one target object at the sampled free position. We then search for a nearby valid position for the confuser object using small offsets around the target location. The confuser is placed close to the target object, but still at a collision-free floor position.

Agent initial viewpoint. The same agent initial view as P.19.

Ground-truth label. The same ground-truth label as P.19.

P.23 Enumerative Perception : Structural Enclosure

Object selection. We first maintain the same objest pool as P.19. For container object, we fixed them with several types of box, to make sure they all can open and close.

Object placement. Each container is instantiated in the room and placed either on a valid support surface, such as a table, or on a free floor position if no suitable support is available.

We then randomly choose a subset of these containers to contain the target objects. For each selected container, the target object position is set near the center of the container's bounding box, and the container is kept closed when possible. Containers not selected by the random sampling remain empty.

Agent initial viewpoint. The same agent initial view as P.19.

Ground-truth label. The same ground-truth label as P.19.

P.24 Enumerative Perception : Illumination Variability

Object selection. We first maintain the same objest pool as P.19.

Object placement. For the light-change task, we first sample valid free-floor positions inside the room. Target objects are placed at these collision-free positions. We then adjust the ambient light intensity and add directional lights to selected objects, creating views with different illumination conditions while keeping the object positions fixed.

Agent initial viewpoint. The same agent initial view as P.19.

Ground-truth label. The same ground-truth label as P.19.

P.25 Cognitive Mapping : Connectivity

Agent initial viewpoint. The connect task selects one navigable render point for each room from the traversability map, preferring free cells with high clearance and near the room center. From this point, it captures eight evenly spaced room reference views. Each question then attaches the views of its required rooms, and uses the first available room view as the main visual context.

Question Generation. The script extracts room regions from the scene segmentation map and samples navigable candidate points for each room. It then generates question: whether two rooms are connected. For each candidate room pair, it calls `scene.get_shortest_path` between sampled points and projects the returned world path back to a room sequence using the segmentation map.

Ground-truth label. For pair connectivity, the label is `Yes` if a valid shortest path exists between the two rooms, and `No` otherwise.

P.26 Cognitive Mapping : Traversable Passage

Agent initial viewpoint. The same as P.25.

Question Generation. The script extracts room regions from the scene segmentation map and samples navigable candidate points for each room. It then generates question: whether the shortest path between two rooms passes through a third room. For each candidate room pair, it calls `scene.get_shortest_path` between sampled points and projects the returned world path back to a room sequence using the segmentation map.

Ground-truth label. For shortest-path-via-region questions, the label is `Yes` if the queried via-room appears inside the projected shortest room sequence, excluding the source and target rooms; otherwise it is `No`.

P.27 Cognitive Mapping : Regional Boundary

Object Selection. The script collects objects in the scene after filtering out robots, structural/background categories, doors, floors, walls, ceilings, mirrors, and oversized objects. Only objects referenced by generated questions are rendered, with four object-centric views captured around the object.

Agent initial viewpoint. The same as P.25.

Question Generation. The script extracts room regions from the segmentation map. It then assigns valid scene objects to regions and generates three question types: whether an object belongs to a region, whether two objects are in the same region, and which of two regions an object is closer to.

Ground-truth label. For object-in-region questions, the label is `Yes` if the queried region matches the object's assigned region, and `No` otherwise. For same-region questions, the label is `Yes` if the two objects have the same assigned region. For closer-region questions, the answer is the region with the smaller distance from the object center to the region bbox, with region-center distance used as a tiebreaker.

P.28 Cognitive Mapping : Long-Term Navigation

Agent initial viewpoint. The same as P.25.

Question Generation. The script extracts room regions from the segmentation map. It then searches reachable room pairs with `scene.get_shortest_path`, ranks them by path distance, and keeps the farthest connected pairs. For each selected pair, it generates two question types: `navigation_actions`, which asks for turn/action choices along the route, and `navigation_regions`, which asks for the next region to explore at each step.

Ground-truth label. The ground truth is derived from the selected shortest path and its projected region sequence. For `navigation_regions`, each answer is the next region in the path sequence. For `navigation_actions`, go-to-region steps are fixed by the same sequence, while turn-choice answers are computed from the change in heading between consecutive regions and labeled as `turn left`, `turn right`, or `turn back`.