ESPIRE: A Diagnostic Benchmark for Embodied Spatial Reasoning of Vision-Language Models

Yanpeng Zhao[†] [⋆] [✉] Wentao Ding[⋆] Hongtao Li[⋆] Baoxiong Jia Zilong Zheng [✉]
State Key Laboratory of General Artificial Intelligence, BIGAI

https://spatigen.github.io/espire.io/ https://github.com/spatigen/espire

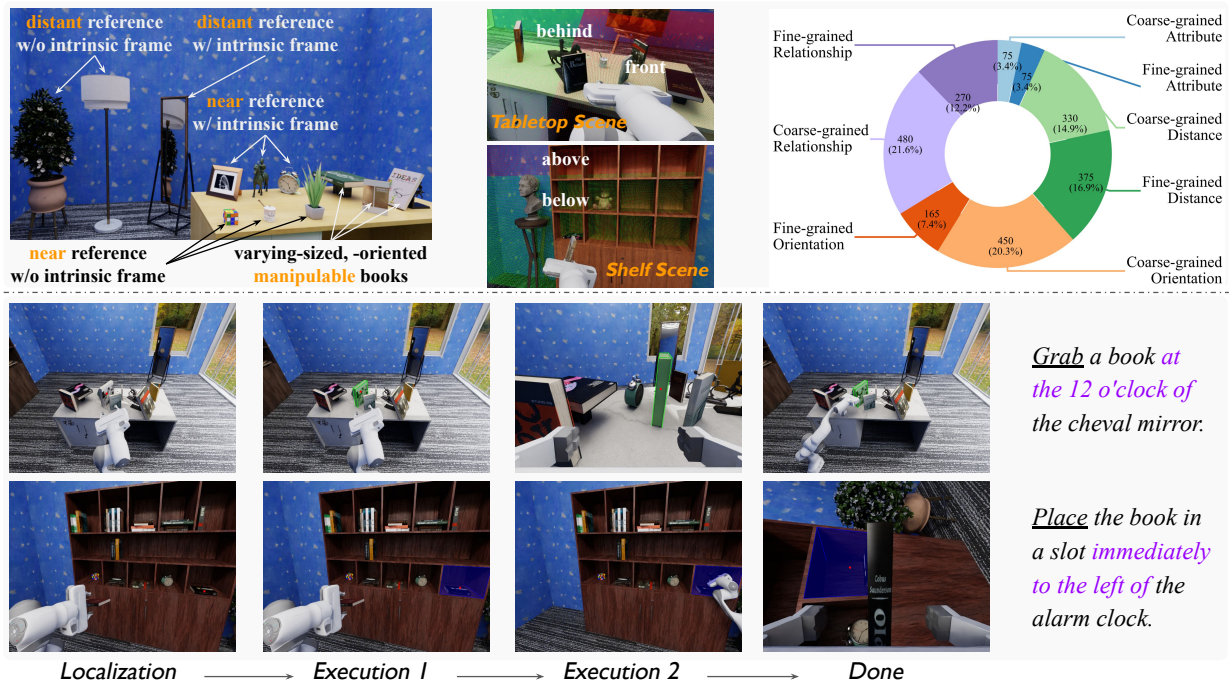

Figure 1. ESPIRE: a simulated physical world. **Top:** the spatial world of ESPIRE covers key factors of spatial reasoning like spatial aspects (e.g., relationship and distance), reference frames, reference objects (§4.1). It features a tabletop scene for *pick* tasks and a shelf scene for *place* tasks (§4.2) and supports reasoning at varying granularities (see Table 10 in Appendix A.2). **Bottom:** example ESPIRE tasks that all inherently rely on spatial reasoning.

Abstract

A recent trend in vision-language models (VLMs) has been to enhance their spatial cognition for embodied domains. Despite progress, existing evaluations have been limited both in paradigm and in coverage, hindering rapid, iterative model development. To address these limitations, we propose ESPIRE*, a diagnostic benchmark for embodied spatial reasoning.* ESPIRE *offers a simulated world that physically grounds VLMs and evaluates them on spatial-reasoning-centric robotic tasks, thus narrowing the gap between evaluation and real-world deployment. To adapt VLMs to robotic tasks, we decompose each task into localization and execution, and frame both as generative problems, in stark contrast to predominant discriminative evaluations (e.g., via visual-question answering) that rely on distractors and discard execution. This decomposition further enables a fine-grained analysis beyond passive spatial reasoning toward reasoning to act. We systematically design* ESPIRE *both at the instruction level and at the environment level, ensuring broad coverage of spatial reasoning scenarios. We use* ESPIRE *to diagnose a range of frontier VLMs and provide in-depth analysis of their spatial reasoning behaviors.*

Contact: yannzhao.ed@gmail.com, {dingwentao,lihongtao}@bigai.ai.
†: Project Lead. ⋆: Core Contributor. ✉: Corresponding Author.

1. Introduction

Spatial cognition goes beyond perception; it enables reasoning and interaction with the 3D physical world, forming the foundation for embodied agents. While pivotal, current machine learning models—and in particular, vision-language models (VLMs)—still lag behind humans in this capacity [13, 23, 31], limiting applications in embodied domains such as robotic navigation and manipulation [16, 18, 19]. To bridge the gap, extensive efforts have been devoted to enhancing the spatial intelligence of VLMs [7, 9, 37, 40, 54, 56, 59].

Despite the remarkable progress, the evaluation of spatially intelligent VLMs remains limited. First, most existing benchmarks are static, adopting multiple-choice visual-question answering (VQA), though this facilitates automatic evaluation, the reliance on distractors renders them prone to biases. Moreover, VQA departs from practical scenarios, where VLM agents must *proactively* act upon given instructions in 3D rather than *passively* selecting an answer from a predefined set. Though more reliable real-world evaluations have been explored, the dependence on specific hardware and handcrafted tasks hinders their scalability and reproducibility [40, 54].

Recently, some have eschewed discriminative VQA and proposed *pointing*, a generative evaluation methodology that requires models to locate the target object/space by generating points in 2D pixel space [54, 59], but the execution phase that typically follows localization in robotics tasks has been overlooked or overly simplified. Others have attempted to address execution while circumventing the limitations of real-world evaluation using simulated environments [26, 30, 37, 51]. Yet, both directions lack a systematic design of evaluation tasks that supports detailed analysis of spatial reasoning across different aspects (e.g., relationships and distances) and granularities (e.g., relative vs. precise distance).

To address these limitations, we propose ESPIRE, a simulation-based benchmark for embodied spatial reasoning with physically-grounded VLMs. Since VLMs are inherently not trained to act, to adapt them for robotics tasks, we decompose each task into localization (which identifies manipulable targets) and execution (which performs the corresponding actions), and frame them as goal position and goal pose generation, respectively. This fully generative, unified evaluation paradigm extends passive spatial reasoning toward acting upon understanding, thus reducing the gap between evaluation and real-world deployment.

To serve our diagnostic purpose, we propose a systematic task design that enables assessment and analysis of the native spatial reasoning of VLMs across varying spatial aspects and granularities. We follow a hierarchical design philosophy, ensuring that the evaluation is spatial-centric and has a broad coverage. Specifically, we first identify three primary factors that characterize spatial reasoning: (1) spatial aspects, including attributes, relationships, distances, and orientations, (2) reference objects, including oriented and non-oriented, and (3) reference frames, including relative, intrinsic, and absolute. A particular configuration of these factors defines a context for spatial reasoning. For example, '*place the book behind the picture frame*' requires reasoning about 'positional relationship (behind)' relative to an 'oriented reference (picture frame)' using the 'intrinsic frame (attached to the picture frame)'. Within a given context, we curate tasks to examine reasoning across different granularities, e.g., fine-grained orientations in '*grab a book to the 2 o'clock of the picture frame*' and precise distances in '*grab a book within 1.2 meters of you*.' To the best of our knowledge, this systematic design supports the most comprehensive, fine-grained analysis that existing benchmarks lack.

We build ESPIRE on Isaac Sim [36] that provides realistic physics simulation, and incorporate necessary measures to reduce *sim-to-real* gaps. ESPIRE offers a total of 148 spatial-reasoning types for localization and covers typical *pick* and *place* actions, enabling a focus on VLM-oriented, embodied native spatial reasoning while maintaining sufficient challenges in tool-free execution. Combined with randomly sampled environments of varied clutter degrees, this provides broad coverage of spatial-centric *reasoning* and *acting*. To support scalable task generation, we represent task instructions in functional programs that can be executed on 3D scene graph representations of environment states and yield ground-truth targets.

We use ESPIRE to evaluate a diverse suite of VLMs, spanning proprietary, open-access, unified, and spatially-enhanced models. We find that VLMs perform much better in localization than in execution, indicating good passive spatial understanding but limited capacity for acting-oriented spatial reasoning. Among all spatial aspects, orientation reasoning poses the greatest challenge in both stages, suggesting a critical deficiency in grounding 3D rotational geometry. Overall, these findings highlight promising avenues for advancing the spatial cognition of VLMs. *We emphasize that* ESPIRE *is not intended to replace real-world evaluation, but to complement it with a scalable, reproducible alternative that facilitates rapid, iterative model improvement.*

In summary, our contributions are the following:

- ESPIRE, a diagnostic benchmark for embodied spatial reasoning of VLMs in physically-grounded photorealistic environments.
- A generative evaluation paradigm that unifies 3D localization and execution, bridging the gap between passive spatial understanding and actional spatial reasoning
- A systematic robotic task design that enables fine-grained diagnosis across diverse spatial reasoning contexts and granularities.
- Experiments and analysis that quantify key bottlenecks in 3D rotational geometry and suggest future directions for enhancement.

Table 1. Comparisons of spatial-reasoning benchmarks. 'Text Gen.' and 'Point Gen.' indicate that models produce answers in natural language and 2D points, respectively. 'Fully Gen.' denotes that models generate positions and rotations in 3D. 'Tool-Free' means no external tools are used, thus assessing the *intrinsic* spatial reasoning of VLMs.

Benchmark	Localization & Execution	Evaluation		Systematicity	Physically-Grounded	Diagnostic	Clutter Level
		Paradigm	Tool-Free				
image- and video-based							
Blink [13]	✗	VQA	✓	✗	✗	✗	*high*
CV-Bench [46]	✗	VQA	✓	✗	✗	✗	*high*
VSI-Bench [50]	✗	VQA	✓	✗	✗	✗	*high*
WHERE2PLACE [54]	✗	Point Gen.	✓	✗	✗	✗	*high*
SpatialVQA [7]	✗	VQA, Text Gen.	✓	✗	✗	✗	*high*
SpatialRGPT-Bench [9]	✗	Text Gen.	✗	✗	✗	✗	*high*
RoboSpatial-Home [40]	✗	VQA, Point Gen.	✓	✗	✗	✗	*high*
Point-Bench [10]	✗	Point Gen.	✓	✗	✗	✗	*high*
simulation-based							
Open6DOR [11]	✓	VQA	✗	✗	✓	✗	*low*
EB-Manipulation [51]	✓	Fully Gen.	✗	✗	✓	✗	*low*
ESPIRE (*ours*)	✓	***Fully Gen.***	✓	✓	✓	✓	***high***

2. Related work

Spatial reasoning with vision-language models. Extensive research has sought to boost the spatial intelligence of VLMs. Some rely on enhanced prompting mechanisms for improved 3D spatial reasoning [29, 33], while many others adopt a data-centric method; in other words, they integrate 3D scene representations (e.g., depth maps and point clouds) into VLMs [37, 56]. Meanwhile, many benchmarks have been proposed to evaluate their 2D and 3D spatial reasoning ability, including SpatialVQA [7], RoboSpatial-Home [40], VSI-Bench [50], and many others [6, 8, 9, 13, 23, 31, 46, 54, 56, 57]. But these benchmarks are limited by their static nature and lack of systematic spatial-centric design. In addition, they predominantly adopt VQA-style evaluations, which are often prone to linguistic biases. In contrast, we propose a systematic task design and a unified generative paradigm, shifting the focus toward active, embodied evaluation.

Simulation-based evaluation through robotic tasks. Unlike human-assisted real-world evaluation, simulation-based approaches allow for more scalable and reproducible evaluation of robotics models, and have been widely used to assess robot policies in domains such as navigation and manipulation [11, 15, 21, 34, 39, 41, 43, 53, 55]. Due to the inherent limitations of simulators, substantial discrepancies exist between simulated observations and real-world observations. To bridge the gap, researchers have been improving physics engines and enhancing synthesis mechanisms to approximate real-world perceptions [1, 36, 45, 48]. Though there have been simulated environments, such as LIBERO [30], CALVIN [34], SIMPLER [26], and EmbodiedBench [51] for *real-to-sim* evaluation, they are limited in overly simplified scenes and tasks or reliance on external tools. In addition, none of them provides a systematic design of spatial-centric reasoning tasks and supports comprehensive diagnoses.

Foundation models for robotics manipulation. Foundation models, including pre-trained LLMs and VLMs, have been applied to robotic manipulation. Early work focuses primarily on task planning while relying on predefined primitives to achieve robot control [12, 20, 28, 49, 58]. Recently, many have attempted to generate trajectories, i.e., sequences of poses, for motion planning [17–19, 37, 54] and devise agentic frameworks for reasoning and acting [14]. Following the unified design philosophy, more recent efforts have focused on developing integrated vision-language-action models (VLAs) that can directly generate low-level action sequences as control policies [3–5, 27, 35, 47, 52], but their success hinges on the underlying spatial reasoning of their vision-language components, we focus on diagnosing VLMs to isolate and identify the specialized spatial inductive biases that are required to inform and improve future unified architectures.

6-DoF object rearrangement. 6-DoF object rearrangement involves predicting a goal state of an object that is described in SE(3) and satisfies the given instruction. With a motion planner, such a formulation enables zero-shot transfer of foundation models from perception to execution [18, 24]. The approaches to 6-DoF tasks can be roughly divided into generative- and discriminative-based. Generative methods solve for a goal translation and rotation of a directional vector under certain constraints [17, 19], while discriminative approaches generate random candidates and use a critic to filter and select the best goal pose [11, 24]. We follow the generative paradigm and prompt VLMs to generate a goal pose and ground it in the simulated physical world.

3. Spatial-centric evaluation of embodied VLMs

We propose evaluating the spatial cognition of VLMs through robotics tasks situated in a simulated physical world, narrowing the gap between evaluation and real-world deploy-

ment. To adapt VLMs for robotics tasks, we decompose each task into two sequential subtasks: localization and execution, formulate them as generative tasks, and ensure that spatial reasoning is the key factor.

- **Localization** refers to locating a target that is specified in a given instruction from the paired scene, such as the 'book' in '*pick up the farthest book*' and the 'empty spot' in '*place the book in an empty spot*'. We follow [54] and [59] and formulate it as a *pointing* task that produces 2D coordinates on scene images.
  Evaluation Metric. We measure model performance using accuracy, defined as the fraction of correct localizations. Unlike discriminative VQA-style evaluations that rely on distractors for automatic metrics, our generative formulation allows for directly comparing the predicted point against the target segmentation mask.
- **Execution** follows the localization stage to execute actions (e.g., *pick* or *place*) in the physically grounded environment. Since VLMs cannot directly produce low-level control actions, we simplify execution as a 6-DoF task that predicts the goal pose, including goal position and orientation prediction, in SE(3). We again formulate goal position prediction as a *pointing* task.
  Evaluation Metric. We measure model performance using acceptance rate, defined as the fraction of physically achieved poses. The acceptability of a predicted pose is assessed by a motion planner like cuRobo [42], making VLMs physically grounded.

In both tasks, native spatial reasoning is inherently needed since VLMs are required to generate positions and orientations in 3D, without relying on external tools. The shared *pointing* formulation between localization and execution further bridges spatial reasoning for understanding and for acting.

4. The ESPIRE benchmark

We propose ESPIRE, a simulated environment that provides a suite of robotics tasks for diagnosing spatial-centric reasoning (see Figure 1). We design ESPIRE systematically both in instructions (§4.1) and environments (§4.2), ensuring a broad coverage of spatial reasoning scenarios, enabling scalable robotic task generation (§4.3), and supporting targeted analysis across contexts and granularities.

4.1. Spatial reasoning tasks

Task specification. We group spatial reasoning tasks into four broad classes by the spatial aspects they require to reason about: *relationships*, *distances*, *attributes* (e.g., dimensions and volumes), and *orientations*. A spatial reasoning task typically involves describing an object in relation to another (e.g., '*grab the book to your left*'), thus relying on a frame of reference. Following [25], we consider three types of reference frames: *relative*, *intrinsic*, and *absolute* frames. The choice of reference frame depends on the reference object, e.g.. intrinsic-oriented objects like 'picture frame' that have a clear front face naturally support intrinsic frames, whereas non-oriented objects like 'sphere ball' do not. Moreover, the reference frame may vary with linguistic specifications, e.g., '*pick up a book on the left of the picture frame*' exhibits ambiguity since both a relative frame and an intrinsic frame can be used, but attaching the clause '*relative to the picture frame's front*' makes the intrinsic frame the only valid interpretation.

To disentangle this complexity, we identify three key factors that characterize spatial reasoning: spatial aspect (S), reference frame (F), and reference object (O); we define their combination $C = (S, F, O)$ as the task specification. A particular configuration of these factors specifies a context for spatial reasoning. For example, $c = (\text{relationship}, \text{intrinsic}, \text{table})$ requires using the *intrinsic* frame of the *table* to carry out relationship reasoning; an instance of it can be '*grab a book on the left of the table*.' This disentanglement lets us focus on designing tasks that target reasoning at varying granularities like *left*, *leftmost*, *second leftmost*, and *to your 11 o'clock*.

Instruction representation. We associate each task instruction with a 3-tuple $T = (C, A, P)$, where C denotes the task specification, $A \in \{\text{pick}, \text{place}\}$ represents execution, and P indicates localization. We represent P as a functional program [22] that can be evaluated on the 3D scene graph representation G of a given environment state and produces a list of valid answers, i.e., objects to be manipulated or spaces to be filled. Crucially, A functional program is composed of atomic functions and defines a reasoning chain, such as finding a specific object $\texttt{unique}(\texttt{filter}(O, G))$ and querying the objects to its left $\texttt{filterRel}(\text{left}, \texttt{unique}(\texttt{filter}(O, G)))$. This enables flexible control of the task complexity by varying the number of reasoning hops.

Instruction families. We define an instruction family on top of a task $T = (C, A, P)$ by associating it with a set of task templates that represent different linguistic expressions of the functional program P. Supposing $C = (\text{distance}, \text{viewer}, \text{intrinsic})$, $A =$ 'Pick', and a template '*[A] a book among the books [R] you*', we can create an instruction, which queries the distance between a book and the viewer, by binding the variable $[R]$ with a type of distance reasoning (e.g., Closest or Furthest). Using the same variable $[R]$, the functional program P can be formed as:

$$\texttt{filterDist}([R], \texttt{filter}(\text{book}, G), \text{viewer})$$

We curate a total of 148 spatial-reasoning task types, distributed across 65 instruction families, including 31 'pick'

instruction families and 34 'place' instruction families. For each instruction family, we manually write 3-4 templates to enhance linguistic diversity. Though functional programs enable multi-hop compositional reasoning, we limit reasoning up to 3 hops, as our primary focus is on spatial rather than compositional reasoning.[1] In practice, we find that a small number of spatial reasoning hops already poses challenges for existing multimodal foundation models.

4.2. Simulation environment

We simulate two task environments in ESPIRE: tabletop and shelf scenes. Both are constructed systematically using a diverse array of photorealistic objects and various spatial layouts and environmental factors like lighting and clutter. This design ensures that our environments provide a comprehensive instantiation of the task specification C, yielding diverse instances that challenge model reasoning across multiple levels of granularity (refer to Appendix A.2 for detailed scene configurations).

Environment representation and generation. We initialize each environment from a random state, which is represented by a 3D scene graph that consists of nodes as objects and edges as spatial relationships. All objects are annotated with ground-truth information, including sizes, dimensions, and poses relative to a predefined absolute reference frame. We generate the initial state of an environment by sampling a random 3D scene graph and rendering it in Isaac Sim [36], ensuring that the environment is physically valid. We adjust the minimum margin of objects and the dimensions of shelf slots; this mitigates the visual ambiguity of spatial aspects and accommodates sufficient, physically feasible tasks in the environment. The Franka robot is initialized in a random pose. We equip it with an on-wrist camera that provides an egocentric view and supplement it with two fixed-position cameras that provide global views of the tabletop and shelf scenes, respectively (referred to as world views). To increase variety and realism, we add external lights. We randomly sample and initialize the positions and orientations of all cameras and external lights.

Reducing the real-to-sim visual gaps. Visual gaps mainly arise from distribution shifts in texture, material, lighting, and camera configurations. Instead of performing complex visual-matching mitigation as in SimplerEnv [26], we employ a more scalable strategy that focuses on enhancing the diversity of the environment: we use annotated 3D assets with realistic textures and tune their sizes to reflect their real-world counterparts. For essential background assets like the tabletop and shelf, we randomly assign textures derived from real-world materials. Combined with randomization in lighting and camera poses, this produces a diverse and visually realistic set of environments (see details in Appendix A.2 and a discussion on sim-to-real relevance in Appendix A.3).

4.3. Simulation tasks

A simulation task is defined by a pair of an environment state and a task instruction. We generate 'pick' and 'place' tasks sequentially. First, we sample and render an environment. The Franka robot is always initialized in a position suitable for performing 'pick' tasks, so we start with 'pick' task generation, and 'place' task generation follows the same procedure. For each variable in a given instruction family, after sampling a random type, we perform value filtering. This is particularly useful for the reference-object variable, as not all reference objects appear in the task space visible from the world view. Once all variables are bound and instantiated, we obtain the final functional program and execute it on the 3D scene graph representation of the visible portion of the environment state. The yielded answers are further verified using a motion planner. We only retain those that correspond to feasible manipulations.[2] Finally, we randomly select a task template from the given task family and instantiate it into a natural language instruction.

5. Experiments

5.1. Experimental setups

Evaluated models. We consider a diverse range of multimodal foundation models, including proprietary VLMs like Gemini2.5-Pro [44], public general-purpose VLMs like instruction-tuned Qwen3-VL [2] and InternVL3 [60], and spatial-reasoning enhanced VLMs like RoboBrain2.0 [38].

Evaluation tasks. Each task family is paired with at least 15 different scenes, leading to around 15 trials on average. We define the difficulty of a task as the complexity of the accompanying scene. Specifically, we categorize the tabletop and shelf tasks into three difficulty levels: easy, medium, and hard. For tabletop tasks, the three levels correspond to scenes that contain 1-2, 3-5, and 6-8 books on the table, respectively.[3] For shelf tasks, difficulty is defined as the fullness of the associated shelf: easy, medium, and hard correspond to shelves where one-third, two-thirds, and all slots are occupied, respectively (refer to Table 14 in Appendix A.5 for illustrations).

[1] Nonetheless, ESPIRE can be readily extended by increasing the number of spatial reasoning hops.

[2] We assume the robot can move freely in 3D, with both locomotion across the ground plane and vertical motion along the global up-axis. This relaxation facilitates reliable execution when using VLMs and a large task space that broadens the coverage of spatial reasoning scenarios.

[3] The book number best correlates with task complexity, but overall complexity is driven by instructions and environmental factors like object number and pose, light, and texture.

Table 2. The localization accuracy (%), acceptance rate (%) in execution, and overall task success rate (%) across different VLMs.

Models	Pick			Place		
	accuracy	acceptance	success	accuracy	acceptance	success
w/o reflection						
Gemini2.5-Pro	**57.72**	**63.93**	**34.06**	50.61	28.36	5.68
InternVL3-78B	28.31	63.01	17.26	23.66	40.94	9.67
RoboBrain2.0-7B	**57.72**	18.81	10.87	**50.70**	15.68	8.64
Qwen3-VL-30B-A3B	54.43	62.56	32.15	45.54	**43.47**	**20.00**
Qwen3-VL-8B	47.03	63.20	29.32	35.71	37.31	12.41
Qwen3-VL-235B-A22B	51.96	52.79	26.76	47.42	41.22	19.34
w/ reflection						
Qwen3-VL-30B-A3B	54.52	27.85	17.08	51.92	23.94	13.80
Qwen3-VL-8B	58.63	24.38	15.07	54.08	12.02	6.67
Qwen3-VL-235B-A22B	**64.29**	**36.71**	**23.20**	**59.72**	**25.45**	**15.40**

Evaluation settings. Our evaluation suite offers a total of 2,220 tasks, consisting of 1,095 *pick* tasks and 1,125 *place* tasks. We limit the number of attempts to 3 for localization and 5 for execution. If localization fails, we randomly select a gold target for execution; otherwise, we use the target localized by the model for execution. We consider non-reflection and reflection settings. In the non-reflection setting, the initial observation is provided by the world-view, while all subsequent observations are obtained from the ego-view. In the reflection setting, the model additionally receives as input its reflections from the previous failed attempt (refer to Algorithms 2 and 3 in Appendix A.4). To enhance reflection in execution, both the world-view and the ego-view are provided. We prompt models to output a 2D point in pixel space while providing them with ground-truth depth. Depending on the settings, we may provide ground-truth rotations of pitch, yaw, and roll or prompt models to generate them directly.

5.2. Main results

In general, proprietary VLMs like Gemini2.5-Pro show the strongest performance under most metrics, while public VLMs like the Qwen3-VL series is narrowing the gaps and even outperform Gemini2.4-Pro in execution on *place* tasks. Interestingly, larger models do not necessarily lead to better performance, e.g., Qwen3-VL-30B with 3B activated parameters outperforms Qwen3-VL-8B and Qwen3-VL-235B (w/ 22B activated parameters). Moreover, all models demonstrate decent localization accuracy except InternVL3-78B, which has an accuracy below 30% (see Table 2), possibly because the native multimodal pre-training adopted by InternVL3-78B for aligning vision and language is not as effective as widely-used stage-by-stage alignment learning [32]. Despite that, it is surprising to see that InternVL3-78B performs much better (e.g., >40%) in execution than in localization. While RoboBrain2.0-7B achieves impressive localization performance, it is mostly because its post-training involves extensive generic spatial reasoning tasks, but, unfortunately, this does not transfer to improved performance in execution (e.g., <20%) that requires acting-oriented spatial reasoning about 3D rotational geometry.

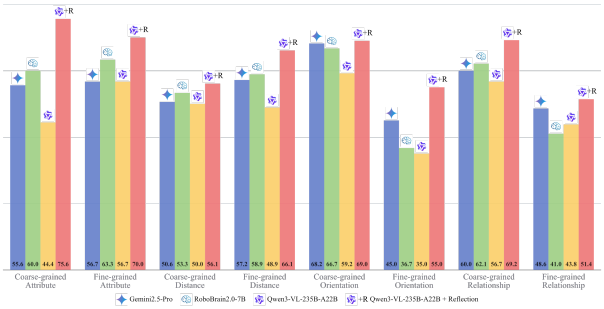

Figure 2. Localization performance across spatial aspects and granularities on *pick* tasks.

'Place' is generally harder than 'pick.' Compared with *pick* tasks, *place* tasks impose much stricter acceptance conditions. Specifically, when predicting a pose for the placement of a book, the model needs to consider additional constraints of the target space, especially when the target space is partially occupied. Moreover, the model usually suffers much more from occlusion when it gets closer to the target space, making it harder to predict a relatively-center point of the space or recover from a non-ideal position, whereas in pick tasks, the model only needs to align with any one of all graspable faces (e.g., the spine or the top edge), and it can optionally move to a better position to facilitate pose prediction (refer to our qualitative studies in Table 8).

Reflection is helpful for localization but does not necessarily help with execution. While reflection improves localization performance of all Qwen3-VL models (see Table 1), it does not yield a comparable improvement in execution; conversely, it degrades execution performance. This is likely because strong 3D rotation understanding is the key factor for execution and forms the foundation for reflection; however, as we will see in an analysis of rotation predictions (see Table 6), current VLMs are weak in this capacity, suggesting that future work is well-suited for curating rotation-reasoning data for fine-tuning.

5.3. Analysis

The systematic design of ESPIRE enables fine-grained analysis of model behavior. We demonstrate this by examining spatial reasoning performance across spatial aspects and task difficulty levels, and by analyzing model behavior during successful task execution.

Table 3. Localization accuracy (%) across four primary spatial aspects.

Models	Accuracy			
	attribute	distance	orientation	relationship
Pick				
Gemini2.5-Pro	56.00	53.89	**63.81**	**56.52**
InternVL3-78B	29.33	21.67	33.65	30.14
RoboBrain2.0-7B	**61.33**	**56.11**	60.95	55.65
Qwen3-VL-30B-A3B	57.33	50.83	57.14	55.07
Qwen3-VL-8B	42.67	40.28	53.97	48.70
Qwen3-VL-235B-A22B	49.33	49.44	54.60	52.75
AVERAGE	49.33	45.37	54.02	49.81
Place				
Gemini2.5-Pro	53.33	**46.09**	**48.75**	**55.06**
InternVL3-78B	28.00	16.23	21.25	30.62
RoboBrain2.0-7B	**69.33**	45.80	47.08	53.58
Qwen3-VL-30B-A3B	48.00	38.55	39.58	54.57
Qwen3-VL-8B	44.00	28.41	31.67	42.82
Qwen3-VL-235B-A22B	62.67	37.68	46.26	53.58
AVERAGE	47.20	33.33	37.17	47.03

Table 5. The average number of attempts to succeed in localization and execution, and average distance (meter) between the target and end-effector upon execution success and before execution success. 'Rank' indicates model ranking in execution.

Models	#Localization	#Move	Dist. at success	Dist. before success	Rank
Pick					
Gemini2.5-Pro	1.20	2.54	0.07	0.47	1
InternVL3-78B	1.05	2.56	0.05	0.48	3
RoboBrain2.0-7B	1.36	2.54	0.05	0.50	6
Qwen3-VL-30B-A3B	1.16	2.49	0.06	0.38	4
Qwen3-VL-8B	1.17	2.41	0.05	0.42	2
Qwen3-VL-235B-A22B	1.18	2.53	0.05	0.40	5
Place					
Gemini2.5-Pro	1.42	3.27	0.26	0.75	5
InternVL3-78B	1.08	2.07	0.24	0.97	3
RoboBrain2.0-7B	1.59	2.98	0.24	0.85	6
Qwen3-VL-30B-A3B	1.33	2.12	0.24	0.95	1
Qwen3-VL-8B	1.30	2.10	0.24	0.97	4
Qwen3-VL-235B-A22B	1.28	2.16	0.23	0.91	2

Localization performance across spatial aspects. We group results by spatial aspects (see Table 3 and Figure 2). Overall, all models perform worse on 'distance' than on the other spatial aspects, across *pick* and *place* tasks, indicating that current VLMs lack the capacity for precise distance understanding. Among them, Gemini2.5-Pro and RoboBrain2.0-7B exhibit relatively stronger overall performance while showing smaller performance variations across spatial aspects, likely because they have been specifically fine-tuned on related spatial reasoning tasks; this is explicitly the case for RoboBrain2.0-7B.

Model performance across task difficulty levels. We further group results by task difficulty levels (see Table 4). Unsurprisingly, on *pick* tasks, most models demonstrate a decrease in both localization execution with increasing task difficulty, except InternVL3-78B and RoboBrain2.0-7B that, in some cases, perform slightly better on harder tasks. Similarly, on *place* tasks, both the localization performance and the execution performance negatively correlate with the task difficulty. Still, there are exceptions like InternVL3-78B and Qwen3-VL-30B-A3B.

Table 4. Performance across difficulty levels.

Models	Accuracy (%)			Acceptance (%)		
	easy	medium	hard	easy	medium	hard
Pick						
Gemini2.5-Pro	60.78	**60.98**	52.04	**70.96**	60.98	60.71
InternVL3-78B	24.85	29.00	30.61	60.78	65.85	**62.24**
RoboBrain2.0-7B	**62.57**	56.10	**55.10**	21.56	15.72	19.39
Qwen3-VL-30B-A3B	**62.57**	53.39	48.87	65.87	**66.67**	55.87
Qwen3-VL-8B	58.08	42.82	41.58	64.97	63.96	60.97
Qwen3-VL-235B-A22B	59.58	52.57	44.90	58.68	55.56	45.15
Place						
Gemini2.5-Pro	**57.46**	48.21	46.11	36.06	28.37	20.46
InternVL3-78B	25.35	22.31	23.34	51.55	39.12	31.99
RoboBrain2.0-7B	52.68	**52.62**	**46.69**	18.31	16.80	11.82
Qwen3-VL-30B-A3B	47.61	44.08	44.96	**52.39**	**43.80**	**34.01**
Qwen3-VL-8B	37.18	36.74	33.14	42.25	40.06	29.39
Qwen3-VL-235B-A22B	51.83	48.48	41.79	48.17	43.53	31.70

Prerequisites for successful execution. Next, we analyze the prerequisites that are strongly associated with successful execution. To this end, we compute the average number of attempts used to achieve successful localization and execution, and the average distances between the target and the end-effector upon execution success and before execution success (see Table 5). Though there is no clear correlation between execution success and pre-success distance, we find that InternVL3-78B and Qwen3-VL-30B-A3B, which are relatively better at execution, tend to be far-sighted (e.g., w/ an average distance of 48cm) and near-sighted (e.g., w/ an average distance of 38cm), respectively. Interestingly, the pre-success distance in *place* tasks is usually twice that in *pick* tasks, presumably because, in *place* tasks, the robot needs to stay reasonably far away from the target space to mitigate occlusion. Moreover, in *place* tasks, strong models like the Qwen3-VL series often require a moderate number of moves; that is, they tend to try multiple times (i.e., around 2.1) before making the final successful execution. In contrast, models that try much more times (i.e., around 3) are usually weaker in execution, e.g., RoboBrain2.0-7B fails spectacularly because it struggles in acting-oriented spatial reasoning.

Apart from the above quantitative analysis, we present qualitative analysis of both successful runs and failed runs in Table 8, Appendix A.4.

5.4. Ablation of rotation prediction

At the core of execution lies the prediction of rotations along the pitch, roll, and yaw axes. This reflects the model's capability for 3D geometric reasoning and its understanding of object affordances, as the predicted rotations are further composed into a goal pose for execution. To better understand the intrinsic capability of VLMs for rotation prediction, we ablate the set of rotation axes to be predicted using Qwen3-VL-235B-A22B.

Table 6. Acceptance rate with rotation axes generated by Qwen3-VL-235B-A22B. Unchecked axes indicate that ground-truth rotations are used. For place tasks, results are reported on tasks with and without explicit pose constraints (C).

Pitch	Yaw	Roll	Pick (%)	Place (%)	
				w/o C	w/ C
			52.73	37.74	43.33
✓			20.91	26.42	26.67
	✓		28.18	29.25	25.00
		✓	30.91	35.85	23.33
✓	✓		4.55	23.58	25.00
✓		✓	13.64	33.02	16.67
	✓	✓	11.82	32.08	10.00
✓	✓	✓	3.64	24.53	16.00

Specifically, rather than using the ground-truth angles derived from the predicted grasping face of the target book/space, we instruct the VLM to directly generate rotation angles for pitch, yaw, and roll. We randomly sample 110 *pick* and 106 *place* tasks from the test suite for this ablation study. Since most *place* tasks impose no constraints on the final pose, we additionally include 60 *place* tasks with explicit pose constraints (e.g., '*place the book at a tilt of 60 degrees.*').

Interestingly, pitch and roll appear to be the key factors for *pick* tasks and constrained *place* tasks, respectively (see Table 6). We conjecture that *pick* tasks require the model to select a feasible grasping face, which largely depends on the pitch axis, whereas constrained *place* tasks require the model

to determine a deviation from the upright direction, primarily governed by the roll axis. As expected, execution becomes harder as more axes need to be predicted. In particular, the pitch-yaw combination adversely affects *pick* the most, while yaw-roll has the largest impact on constrained *place* tasks.

5.5. Human study

As discussed in Section 4.1, spatial reasoning tasks may exhibit ambiguity when the intended reference frame is not explicitly specified and must be inferred. For example, given the instruction '*grab a book to the left of the picture frame,*' an agent must determine whether to interpret the relation using the intrinsic frame or the relative frame. To investigate to what extent VLMs exhibit frame preferences similar to humans, we use near oriented objects, distant oriented objects, and the table as the reference, construct 91 *pick* tasks involving ambiguous frames, and collect responses from five human participants. Then, we measure human-model agreement by computing the Spearman's rank correlation.

Among the three types of oriented reference objects, humans and models show strong agreement when the table or distant objects are used as references (i.e., w/ high positive correlations above 0.8) but disagree (i.e., w/ negative correlations) when the reference is a near object such as an alarm clock. In such cases, we find that humans tend to prefer the intrinsic frame of the reference object while models favor the relative frame. We hypothesize that when the reference and target objects (i.e., books) are of comparable size, humans perceive the reference as an oriented object with salient geometric cues, making its intrinsic frame more accessible. In contrast, VLMs appear to struggle with object-centric orientation inference and therefore default to the relative frame.

Table 7. Agreement of humans and models on reference frames. References are categorized into table, near, and distant objects.

Model	Near Obj.	Distant Obj.	Table
Gemini2.5-Pro	-0.573 $\pm$ 0.634	0.8 $\pm$ 0.274	1.0 $\pm$ 0.0
RoboBrain2.0-7B	-0.674 $\pm$ 0.242	0.8 $\pm$ 0.274	1.0 $\pm$ 0.0
Qwen3-VL-30B-A3B	-0.100 $\pm$ 0.652	0.8 $\pm$ 0.274	1.0 $\pm$ 0.0
Qwen3-VL-8B	-0.674 $\pm$ 0.242	0.8 $\pm$ 0.274	1.0 $\pm$ 0.0
Qwen3-VL-235B-A22B	-0.573 $\pm$ 0.634	0.8 $\pm$ 0.274	1.0 $\pm$ 0.0

5.6. Efficiency of ESPIRE

Our analysis regarding the running time of ESPIRE reveals two primary sources of latency: API calls and model inference. API response time is largely affected by network stability, whereas the model inference time is determined by the model size and the hardware used for deployment. Taking RoboBrain2.0-7B [38], when running on an RTX 4090 machine, a single inference takes an average of 9.25 seconds. Another source of latency comes from execution that involves motion planning and environment update. Specifically, the average time for executing a move request is about 18.12 seconds in our experiments on a workstation equipped with an NVIDIA RTX 4090 GPU.

6. Discussion and future work

ESPIRE is the first simulated physical environment designed for the diagnostic evaluation of spatial cognition in VLMs, featuring spatial-centric robotic tasks that are explicitly designed to be scalable and diverse. To evaluate VLMs that cannot directly produce low-level control actions, we have reformulated robotic tasks into localization and execution. While future VLAs are supposed to integrate the two phases, we deliberately prioritize diagnosis over integration. This design choice is further motivated by existing agentic frameworks that decouple reasoning and acting, using VLMs for high-level spatial reasoning and VLAs or controllers for action execution [14]. By isolating the reasoning stage, our framework provides a 'microscope' to identify where spatial reasoning chains break, offering a concrete roadmap for the specialized spatial inductive biases that future architectures will require.

A limitation of ESPIRE is: it is restricted to indoor scenes. Despite our systematic design, it does not cover spatial reasoning scenarios that arise only outdoors, such as reasoning with larger units of measure (e.g., *kilometer*), reasoning with larger-sized reference objects (e.g., *trees*), and reasoning using the global reference frame (e.g., *south* or *east*). Nonetheless, ESPIRE readily supports such extensions, for example, by making outdoor reference objects visible through glass walls.

Beyond that, ESPIRE opens several new avenues for the development and analysis of spatially intelligent VLMs. For example, ESPIRE allows for designing long-horizon tasks that require multi-step spatial reasoning, leading to many interesting model analyses, including the modeling of dependencies between reasoning steps and the role of memory in long-horizon spatial reasoning. Moreover, since 'pick' and 'place' tasks typically occur sequentially in robotics, but are performed in different workspaces in ESPIRE, it is well-suited for extending it to evaluate mobile manipulation.

7. Conclusion

We have presented ESPIRE, a simulated environment that provides an evaluation suite for embodied spatial reasoning with vision-language models. ESPIRE evaluates VLMs on robotic tasks in a physically grounded setting, thus mitigating the gap between evaluation and practical deployment. By breaking down each task into localization and execution, ESPIRE provides a unified evaluation of passive spatial reasoning and action-oriented spatial reasoning. We systematically design ESPIRE to simulate a diverse range of spatial reasoning scenarios, enabling a comprehensive analysis across spatial aspects and at multiple levels of granularity. Our experimental results and analysis reveal future directions for enhancing VLMs in spatial reasoning.

References

[1] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3674–3683, 2018. 3

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. 5, 4

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2024. 3

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-1: Robotics transformer for real-world control at scale. In *Robotics: Science and Systems*, 2023.

[5] Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions, 2025. 3

[6] Wenxiao Cai, Yaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. *CoRR*, abs/2406.13642, 2024. 3

[7] 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)*, pages 14455–14465, 2024. 2, 3

[8] Kaiyuan Chen, Shuangyu Xie, Zehan Ma, Pannag R Sanketi, and Ken Goldberg. Robo2vlm: Visual question answering from large-scale in-the-wild robot manipulation datasets, 2025. 3

[9] 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. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 2, 3

[10] Long Cheng, Jiafei Duan, Yi Ru Wang, Haoquan Fang, Boyang Li, Yushan Huang, Elvis Wang, Ainaz Eftekhar, Jason Lee, Wentao Yuan, Rose Hendrix, Noah A. Smith, Fei Xia, Dieter Fox, and Ranjay Krishna. Pointarena: Probing multimodal grounding through language-guided pointing, 2025. 3

[11] Yufei Ding, Haoran Geng, Chaoyi Xu, Xiaomeng Fang, Jiazhao Zhang, Songlin Wei, Qiyu Dai, Zhizheng Zhang, and He Wang. Open6dor: Benchmarking open-instruction 6-dof object rearrangement and a vlm-based approach. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7359–7366, 2024. 3

[12] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-e: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, pages 8469–8488. PMLR, 2023. 3

[13] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XXIII*, page 148–166, Berlin, Heidelberg, 2024. Springer-Verlag. 2, 3

[14] Gemini-Robotics-Team. Gemini robotics 1.5: Pushing the frontier of generalist robots with advanced embodied reasoning, thinking, and motion transfer, 2025. 3, 8

[15] Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. Maniskill2: A unified benchmark for generalizable manipulation skills. In *The Eleventh International Conference on Learning Representations*, 2023. 3

[16] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual language maps for robot navigation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10608–10615, 2023. 2

[17] Haoxu Huang, Fanqi Lin, Yingdong Hu, Shengjie Wang, and Yang Gao. Copa: General robotic manipulation through spatial constraints of parts with foundation models. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9488–9495, 2024. 3

[18] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. In *Proceedings of The 7th Conference on Robot Learning*, pages 540–562. PMLR, 2023. 2, 3

[19] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. In *8th Annual Conference on Robot Learning*, 2024. 2, 3

[20] Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander T Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as i can, not as i say: Grounding language in robotic affordances. In *6th Annual Conference on Robot Learning*, 2022. 3

[21] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020. 3

[22] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4

[23] Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. 2, 3

[24] Ivan Kapelyukh, Yifei Ren, Ignacio Alzugaray, and Edward Johns. Dream2real: Zero-shot 3d object rearrangement with vision-language models. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024. 3

[25] Stephen C. Levinson. *Space in Language and Cognition: Explorations in Cognitive Diversity*. Cambridge University Press, 2003. 4, 1

[26] Xuanlin Li, Kyle Hsu, Jiayuan Gu, Oier Mees, Karl Pertsch, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. In *8th Annual Conference on Robot Learning*, 2024. 2, 3, 5

[27] Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[28] Jacky Liang, Wenlong Huang, Fei Xia, Peng Xu, Karol Hausman, Brian Ichter, Pete Florence, and Andy Zeng. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500, 2023. 3

[29] Xun Liang, Xin Guo, Zhongming Jin, Weihang Pan, Penghui Shang, Deng Cai, Binbin Lin, and Jieping Ye. Enhancing spatial reasoning through visual and textual thinking, 2025. 3

[30] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, qiang liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. 2, 3

[31] Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023. 2, 3

[32] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 6

[33] Chenyang Ma, Kai Lu, Ta-Ying Cheng, Niki Trigoni, and Andrew Markham. Spatialpin: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3d priors. In *Advances in Neural Information Processing Systems*, pages 68803–68832. Curran Associates, Inc., 2024. 3

[34] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022. 3

[35] Oier Mees, Dibya Ghosh, Karl Pertsch, Kevin Black, Homer Rich Walke, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024. 3

[36] NVIDIA. Isaac Sim, 2025. 2, 3, 5

[37] Zekun Qi, Wenyao Zhang, Yufei Ding, Runpei Dong, Xinqiang Yu, Jingwen Li, Lingyun Xu, Baoyu Li, Xialin He, Guofan Fan, Jiazhao Zhang, Jiawei He, Jiayuan Gu, Xin Jin, Kaisheng Ma, Zhizheng Zhang, He Wang, and Li Yi. Sofar: Language-grounded orientation bridges spatial reasoning and object manipulation, 2025. 2, 3

[38] BAAI RoboBrain-Team, Mingyu Cao, Huajie Tan, Yuheng Ji, Xiansheng Chen, Minglan Lin, Zhiyu Li, Zhou Cao, Pengwei Wang, Enshen Zhou, Yi Han, Yingbo Tang, Xiangqi Xu, Wei Guo, Yaoxu Lyu, Yijie Xu, Jiayu Shi, Mengfei Du, Cheng Chi, Mengdi Zhao, Xiaoshuai Hao, Junkai Zhao, Xiaojie Zhang, Shanyu Rong, Huaihai Lyu, Zhengliang Cai, Yankai Fu, Ning Chen, Bolun Zhang, Lingfeng Zhang, Shuyi Zhang, Dong Liu, Xi Feng, Songjing Wang, Xiaodan Liu, Yance Jiao, Mengsi Lyu, Zhuo Chen, Chenrui He, Yulong Ao, Xue Sun, Zheqi He, Jingshu Zheng, Xi Yang, Donghai Shi, Kunchang Xie, Bochao Zhang, Shaokai Nie, Chunlei Men, Yonghua Lin, Zhongyuan Wang, Tiejun Huang, and Shanghang Zhang. Robobrain 2.0 technical report, 2025. 5, 8

[39] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10737–10746, 2020. 3

[40] Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. Robospatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 15768–15780, 2025. 2, 3

[41] Sanjana Srivastava, Chengshu Li, Michael Lingelbach, Roberto Martín-Martín, Fei Xia, Kent Elliott Vainio, Zheng Lian, Cem Gokmen, Shyamal Buch, Karen Liu, Silvio Savarese, Hyowon Gweon, Jiajun Wu, and Li Fei-Fei. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments. In *Proceedings of the 5th Conference on Robot Learning*, pages 477–490. PMLR, 2022. 3

[42] Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, Nathan Ratliff, and Dieter Fox. Curobo: Parallelized collision-free robot motion generation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8112–8119, 2023. 4

[43] Andrew Szot, Alex Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Chaplot, Oleksandr Maksymets, Aaron Gokaslan, Vladimir Vondrus, Sameer Dharur, Franziska Meier, Wojciech Galuba, Angel Chang, Zsolt Kira, Vladlen Koltun, Jitendra Malik, Manolis Savva, and Dhruv Batra. Habitat 2.0: training home assistants to rearrange their habitat. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2021. Curran Associates Inc. 3

[44] Gemini 2.5 Team, Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, and Others. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. 5, 4

[45] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012. 3

[46] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In *Advances in Neural Information Processing Systems*, pages 87310–87356. Curran Associates, Inc., 2024. 3

[47] Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified vision-language-action model, 2025. 3

[48] Fei Xia, Amir R. Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9068–9079, 2018. 3

[49] Bing Xie, Xiangming Xi, Xinan Zhao, Yuhan Wang, Wei Song, Jianjun Gu, and Shiqiang Zhu. Chatgpt for robotics: A new approach to human-robot interaction and task planning. In *ICIRA (5)*, pages 365–376, 2023. 3

[50] 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, 2024. 3

[51] 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. In *Proceedings of the 42nd International Conference on Machine Learning*, pages 70576–70631. PMLR, 2025. 2, 3

[52] Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Se June Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, Lars Liden, Kimin Lee, Jianfeng Gao, Luke Zettlemoyer, Dieter Fox, and Minjoon Seo. Latent action pretraining from videos. In *The Thirteenth International Conference on Learning Representations*, 2025. 3

[53] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning*, pages 1094–1100. PMLR, 2020. 3

[54] Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction in robotics. In *8th Annual Conference on Robot Learning*, 2024. 2, 3, 4

[55] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, and Johnny Lee. Transporter networks: Rearranging the visual world for robotic manipulation. In *Proceedings of the 2020 Conference on Robot Learning*, pages 726–747. PMLR, 2021. 3

[56] Weichen Zhang, Zile Zhou, Zhiheng Zheng, Chen Gao, Jinqiang Cui, Yong Li, Xinlei Chen, and Xiao-Ping Zhang. Open3dvqa: A benchmark for comprehensive spatial reasoning with multimodal large language model in open space, 2025. 2, 3

[57] Baining Zhao, Ziyou Wang, Jianjie Fang, Chen Gao, Fanhang Man, Jinqiang Cui, Xin Wang, Xinlei Chen, Yong Li, and Wenwu Zhu. Embodied-r: Collaborative framework for activating embodied spatial reasoning in foundation models via reinforcement learning, 2025. 3

[58] Peiyuan Zhi, Zhiyuan Zhang, Muzhi Han, Zeyu Zhang, Zhitian Li, Ziyuan Jiao, Baoxiong Jia, and Siyuan Huang. Closed-loop open-vocabulary mobile manipulation with gpt-4v. *CoRR*, abs/2404.10220, 2024. 3

[59] Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Wang, Tiejun Huang, Lu Sheng, and Shanghang Zhang. Roborefer: Towards spatial referring with reasoning in vision-language models for robotics, 2025. 2, 4, 3

[60] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao,

Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models, 2025. 5