SEEK-AND-VIEW REASONING FOR MULTI-VIEW SPATIAL UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing approaches to multi-view spatial reasoning operate largely on sparse input views. Vision-language models (VLMs) are thus restricted to understand a scene and infer spatial relations within these fixed views, leading to fragile cross-view alignment and geometry-to-language bottleneck. To address these issues, we formulate a novel *Seek-and-View* reasoning approach to find implicit cross-view spatial evidence by locating a question-relevant view to support the spatial reasoning. To realize this approach, we propose Vantage, a training-free model-agnostic reasoning framework that pairs a VLM with a 3D foundation model: a viewpoint-grounded reasoning stage for question analysis and view planning, followed by a geometry-grounded evidence augmentation stage to effectively synthesize and incorporate visual evidence into the final reasoning. Comprehensive experiments on six VLMs demonstrate consistent improvements on five benchmarks without fine-tuning. Overall, by revealing spatial evidence through view-grounded reasoning, Vantage can largely reduce reliance on language-based cross-view alignment and improve multi-view spatial understanding. Code will be released.

1 INTRODUCTION

Multi-view spatial understanding (Gholami et al., 2026; Zhang et al., 2026a; Wang et al., 2026c; Ke et al., 2026) requires models to capture the structure of a scene from partial observations that are fixed and sparse. Accurate cross-view alignment is therefore essential. Especially, spatial questions cannot be answered simply from individual views and instead require associating evidence from different views. Despite rapid progress in vision-language models (VLMs), their spatial reasoning over multiple views remains limited and prone to hallucination (Fu et al., 2024).

Recent VLMs therefore seek to strengthen geometric reasoning capabilities by verbalizing the geometry during reasoning (Ke et al., 2026; Jia et al., 2026; Li et al., 2026) or by injecting geometric priors into the model (Zheng et al., 2026; Zhang et al., 2026c; Hu et al., 2026), both aimed at improving the model's awareness of the underlying scene structure (see the illustrations in Figure 1(a)). While improving over shallow visual recognition, both approaches rely on the VLM to align views and perform the reasoning explicitly in language, leading to the following two issues. First is *fragile cross-view alignment*. Strong semantic and contextual reasoning ability does not inherently enforce structured and globally consistent geometric constraints required across views, leaving the alignment implicit and error-prone. Second is *the geometry-to-language bottleneck*. Upon compressing the spatial geometry into discrete semantic descriptions, the reasoning would potentially overlook fine-grained spatial information (Wu et al., 2026; Chen et al., 2026b). More fundamentally, both approaches remain within a **View-and-Reason** paradigm, in which the reasoning is fixed on the given observations. Humans, in contrast, can conceptualize the three-dimensional structure of a scene and seek a view to reveal the visible information for resolving the spatial relation. Such a visual thinking process leads us to ask: *Can a model decide where to look, then reason better from that view?*

To explore this idea, we formulate a novel **Seek-and-View** paradigm, in which a model first seeks a viewpoint that consolidates evidence from different views into an observation for supporting the spatial reasoning. Our new paradigm alleviates the burden on semantic textual reasoning for global scene understanding through question-relevant, evidence-seeking, and reasoning-oriented visual evidence acquisition. Recent studies explore a think-with-image paradigm for visual question answering (VQA), where the model reasons with additional visual cues obtained by zooming, drawing

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.

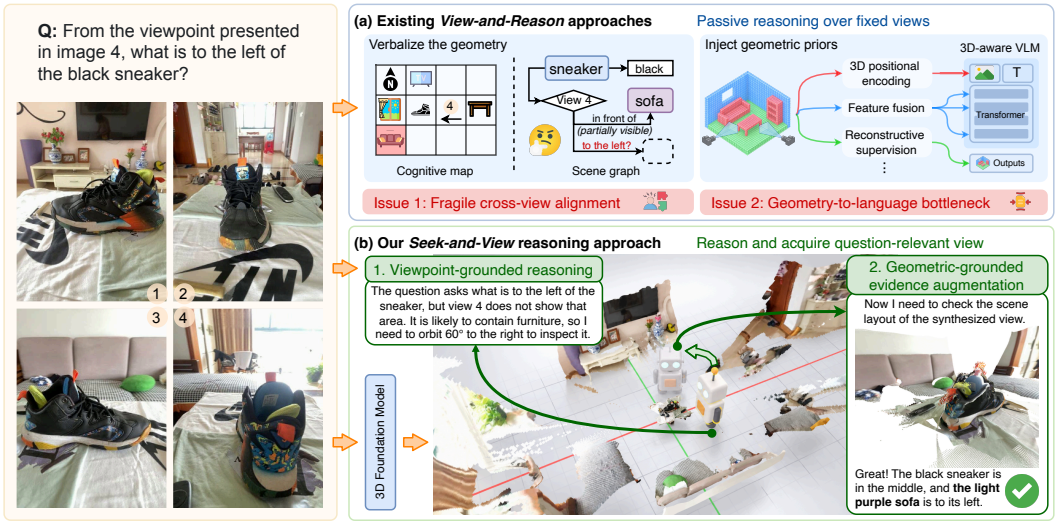


Figure 1: (a) Existing approaches reason over fixed views, suffering from fragile cross-view alignment and the geometry-to-language bottleneck. (b) Our novel Seek-and-View reasoning paradigm seeks a question-relevant view (see bottom right) to make the spatial evidence directly observable.

auxiliary lines, or estimating depth (Gupta & Kembhavi, 2023; Surís et al., 2023; OpenAI, 2025; Qin et al., 2025; Wu et al., 2026). Yet, the underlying visual thoughts are still an in-view understanding confined to the fixed observations. Beyond fixed observations, we aim to realize a Seek-and-View paradigm, in which the challenges are (i) to locate a question-relevant viewpoint, and (ii) to effectively contextualize the acquired visual evidence for subsequent reasoning.

To address these challenges, we propose **Vantage**, a two-stage training-free and model-agnostic framework that instantiates the Seek-and-View paradigm by incorporating a VLM with a 3D foundation model (3DFM). In the first *viewpoint-grounded reasoning* stage, the VLM analyzes the question to identify the necessary observation(s) and plans a viewpoint to acquire the intended evidence. In the second *geometry-grounded evidence augmentation* stage, the 3DFM produces a new view of the scene and takes it with the given question and proper guidance to support the final VLM reasoning. In this way, the 3DFM externalizes the cross-view alignment by reconstructing a consistent scene representation, meanwhile the VLM plans the viewpoint and continues reasoning with the additional visual evidence acquired from the new view; see Figure 1(b).

Our evaluations focus on multi-view spatial VQA tasks whose answers are determined by the geometric relations within static real-world scenes. Across five spatial reasoning benchmarks and six different VLMs, Vantage is able to consistently boost the performance of all the six VLMs without any fine-tuning, with a mean accuracy gain of 6.7%. These results highlight that the difficulty in current multi-view spatial understanding stems not only from limited geometric reasoning capability but also from passive reasoning on fixed input observations. Further, we analyze failure cases and reveal the other limitations of existing VLMs. Our main contributions are threefold:

- We introduce a novel Seek-and-View reasoning paradigm for multi-view spatial understanding, in which we propose to seek cross-view spatial evidence through a question-relevant viewpoint rather than reasoning solely on fixed input observations.
- We propose Vantage, a two-stage training-free framework to realize Seek-and-View with a viewpoint-grounded reasoning and a geometry-grounded evidence augmentation, explicitly coupling VLM reasoning with a 3D foundation model.
- Through extensive experiments across six VLMs and five benchmarks, we demonstrate the effectiveness of Seek-and-View, revealing its strength to boost the VLM performance and also the limitations of existing VLMs under this reasoning paradigm.

2 Related Work

Multi-View Spatial Reasoning Existing research has enhanced VLMs for multi-view spatial reasoning largely in two directions. The first verbalizes scene geometry into textual or symbolic structures, such as cognitive maps, scene graphs, or bounding boxes to support downstream reasoning (Lee et al., 2025; Ke et al., 2026; Wang et al., 2026c; Li et al., 2026; Tao et al., 2026; Cai et al., 2026; Cheng et al., 2026; Choi et al., 2026). This approach, however, demands labor-intensive per-scene annotations and subsequent supervised fine-tuning. The second direction injects geometric priors directly into the model (Wang et al., 2025a; Zheng et al., 2026; Janjua et al., 2026; Hu et al., 2026; Zhang et al., 2026b;c), *e.g.*, by fusing features from geometric encoders, redesigning positional encoding with 3D coordinates, or introducing reconstructive training objectives. Despite these advances, the visual observations are typically fixed, leaving the model to reason over a predetermined set of views. In contrast, our Seek-and-View reasoning paradigm adapts the observation itself by acquiring a task-specific view tailored to the question, while keeping the core VLM training-free.

Thinking with images paradigm Thinking with images (OpenAI, 2025) is an emerging paradigm that externalizes intermediate reasoning into visual operations such as sketching, marking, zooming, and depth estimation for mathematical geometry and 2D spatial tasks (Gupta & Kembhavi, 2023; Surís et al., 2023; Hu et al., 2024; Qin et al., 2025; Zhang et al., 2026e; Yang et al., 2026d; Wu et al., 2026). Related agentic approaches further treat the VLM as a tool-calling agent, augmenting reasoning via various external tools (Chen et al., 2026b;a; Cho et al., 2026; Wang et al., 2026b).

Spatial reasoning can leverage a world model to generate unseen observations for iterative exploration (Yang et al., 2026c; Cao et al., 2026; Zhang et al., 2026d; Zhu et al., 2026; Liu et al., 2026; Zhang et al., 2026f). This approach, however, incurs substantial inference cost, while requiring dedicated fine-tuning to maintain cross-view consistency. Other approaches (Yang et al., 2026a; Zhang et al., 2026h; Jin et al., 2026) construct stitched panoramic views or top-down scene representations to try to preserve the scene consistency. Yet, these representations are typically generated as generic global views, rather than tailored to expose the spatial evidence needed for a specific question. Our approach instead performs a question-relevant view seeking, allowing the geometric backend to acquire the desired visual evidence to support the subsequent spatial reasoning.

3D Foundation Models for View Synthesis Recent 3D foundation models (3DFMs) (Wang et al., 2025b; Lin et al., 2025; Wang et al., 2025c; 2026a) have made feed-forward recovery of scene geometry increasingly practical. G3T (Kani & Snavely, 2026) further introduces gravity-aligned point maps, providing an upright scene coordinate frame. By reconstructing a consistent scene representation and estimating camera poses, 3DFMs externalize the cross-view alignment and allow viewpoint planning and view synthesis of the scene. Such a view synthesis differs from conventional novel view synthesis, which aims to faithfully render a scene from a specified pose. Our setting instead seeks a viewpoint mainly for exposing the spatial evidence to support the reasoning. Hence, the synthesized view need not be photorealistic. Its core value is to preserve the underlying scene and reveal necessary visual evidence needed for answering the given question.

3 Method

Vantage is a two-stage pipeline, as shown in Figure 2. Its name reflects the key idea, *i.e.*, the model seeks a vantage viewpoint from which the spatial relation relevant to answering the given question can become visible to support the reasoning. In this section, we first overview the Seek-and-View reasoning paradigm in Section 3.1, followed by the method details in Sections 3.2 and 3.3.

3.1 Overview of Seek-and-View Reasoning Paradigm

Given a spatial question q and a set of sparse, unposed input views $\mathcal{V} = \{v_i\}_{i=1}^{N}$, we consider visual question answering (VQA), in which the observed input views $\mathcal{V}$ collectively provide the scene information for answering question q. However, the relevant spatial relation for answering question q may not be directly observable solely from any individual view in $\mathcal{V}$.

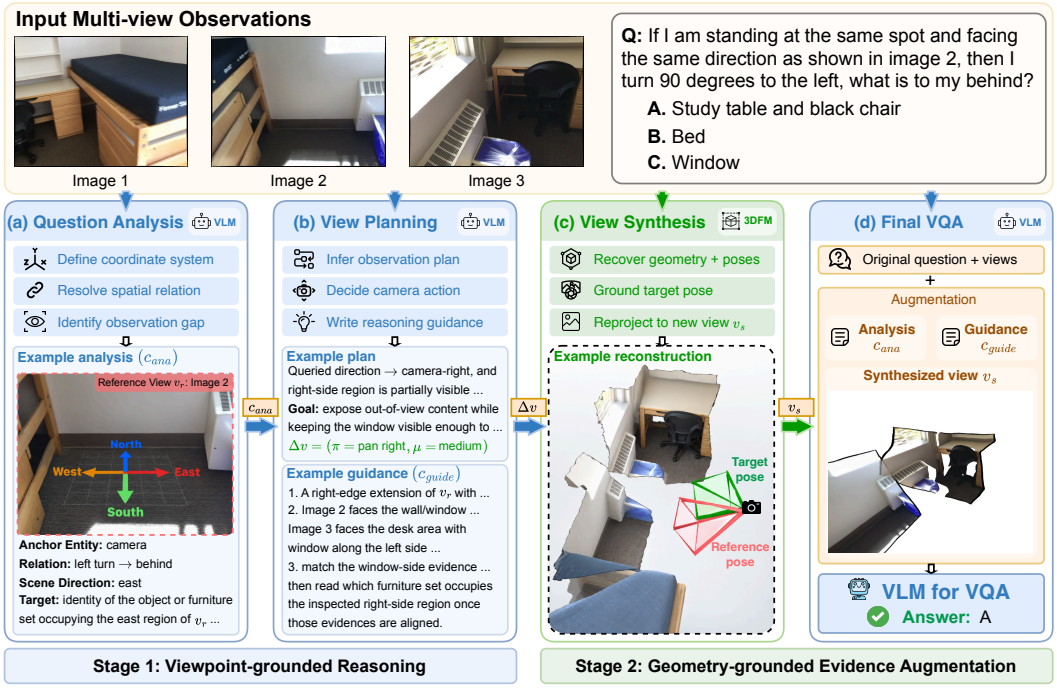


Figure 2: **Overview of Vantage pipeline.** Stage 1: Viewpoint-grounded reasoning employs an VLM (in blue) to (a) produce an analysis on the input question (top-right), select reference view v_r, identify queried spatial relation, then (b) plan a question-relevant camera action Δv on reference view v_r and generate reasoning guidance for supporting view interpretation. Stage 2: (c) Geometry-grounded evidence augmentation synthesizes the planned view v_s from the 3DFM reconstruction (in green) and provides it together with the original images, viewpoint analysis c_{ana}, and reasoning guidance c_{guide} to support (d) the final reasoning.

Seeking. Directly using a VLM to predict the camera pose of the target view is unreliable (Zhang et al., 2026f). Yet, introducing a metric 3D coordinate system could further escalate the uncertainty in scale, orientation, and alignment. We thus formulate view planning as a relative camera action grounded in one of the observed views to more reliably control the viewpoint. A reasoner $\mathcal{H}$ analyzes the question q and available observations $\mathcal{V}$ to select a reference view v_r and a relative camera action Δv, while retaining intermediate reasoning contexts for subsequent answer prediction:

$$\mathcal{H}(q, \mathcal{V}) \rightarrow (v_r, \Delta v, c_{\text{ana}}, c_{\text{guide}}), \tag{1}$$

where c_{ana} captures the question-oriented analysis of the current observations, and c_{guide} contains the planned view adjustment and its underlying rationale.

Viewing. Before answering the input question q, we obtain the view through a geometry-grounded synthesis: a 3D foundation model reconstructs the scene and its camera poses from the unposed input views $\mathcal{V}$, and the planned action Δv is applied to the reference view v_r to reproject the reconstruction to produce the requested view v_s. The synthesized view v_s is then incorporated into the original VQA $(q, \mathcal{V})$, together with the intermediate reasoning contexts $(c_{\text{ana}}, c_{\text{guide}})$, to support the subsequent spatial reasoning. We detail the reconstruction and reprojection in Section 3.3.

3.2 Viewpoint-Grounded Reasoning

Instead of unconstrained reasoning, we structure view planning around question analysis and action planning in viewpoint-aware perspective. In this way, we can resolve spatial relations not directly observable from any individual view and determine how to adjust the viewpoint to reveal the spatial relations. However, grounding such semantic reasoning into an executable camera action remains challenging. We therefore decompose viewpoint-grounded reasoning into the following two steps:

Question analysis: *Establishing a semantic directional frame from the reference view.* A natural starting point for multi-view reasoning is to find a view whose scene contents are most relevant to the question. Here, we employ a VLM to first select a reference view v_r from the input views $\mathcal{V}$ conditioned on the question q. It then then identifies the anchor entity, namely the camera viewpoint, object, or region in the reference view that constitutes one side of the queried spatial relation. Directional terms such as left, right, front, and behind may refer to different frames of reference while sharing the same linguistic expressions (Yang et al., 2026b; Wang et al., 2026c). This creates ambiguity in how a directional relation should be spatially grounded. To reduce this ambiguity, the VLM establishes a semantic directional frame grounded in reference view v_r. By default, the forward direction of v_r is treated as north, unless an explicit directional convention is specified by the question. This semantic frame is maintained throughout the reasoning process to provide a consistent interpretation of directional relations without requiring explicit metric 3D coordinates.

Resolving Spatial Relations into Observation Needs. The VLM identifies the queried entity or region and the spatial relation with respect to the anchor. It then resolves this relation within the established semantic directional frame, translating object-relative or camera-relative directions into a unified scene direction; see Figure 2 (Stage 1). From the reference view v_r, the model examines the currently visible evidence and determines which spatial relation remains visible or verifiable. Together, these elements form the question analysis context c_{ana}, which summarizes the problem diagnosis and provides the reasoning basis for subsequent viewpoint planning.

View Planning: *From Observation Need to Camera Action.* The observation need identified above is then translated into a camera action. However, it is expressed semantically, whereas acquiring a new view requires a camera pose defined in the reconstructed 3D space. We therefore provide the VLM with a predefined action set of 17 camera motions spanning rotation, translation, and hybrid transformations, each parameterized by three coarse magnitude levels: small, medium, and large. For example, a rightward rotation corresponds to 30°, 60°, and 90°, respectively. Conditioned on question q, input views $\mathcal{V}$, question analysis context c_{ana}, and an action-planning prompt that specifies this action set, the VLM predicts a camera adjustment relative to the reference view v_r:

$$\Delta v = (\pi, \mu), \tag{2}$$

where π denotes the selected camera action and μ its magnitude level. After the 3DFM estimates the pose of the reference view, the geometric backend maps Δv to the corresponding numerical pose transformation in the reconstructed coordinate frame. This structured interface allows the VLM to reason about how to adjust the viewpoint without directly predicting the metric camera pose, while leaving precise geometric grounding to the reconstruction backend. Section 4.3 shows the benefit of this structured action interface and the Appendix provides the action set and planning prompt.

Preserving the Reasoning Context. The synthesized view alone does not convey why the viewpoint was selected or which spatial cues should be examined. To explicitly convey this planning intent (Wang et al., 2026b), the VLM also generates the reasoning guidance context c_{guide} within the same view planning call. The guidance describes the expected spatial configuration, identifies relevant visual anchors in the input views, and specifies the spatial cues to inspect or combine in the synthesized view. Since it is generated before view synthesis, it records the intended use of the new observation without assuming that the expected evidence is present. The resulting c_{guide} is passed to the final VQA stage as auxiliary context, preserving the connection between the planned observation and the original visual evidence.

3.3 Geometry-Grounded Evidence Augmentation

In this stage, we ground the planned viewpoint in the reconstructed 3D scene to synthesize the view. We then use this geometry-grounded observation with proper guidance from the previous stage to augment the original VQA for continuous reasoning through the following two steps.

View Synthesis: *Realizing the Requested Observation.* It is important that the planned semantic action preserves its intended orientation with respect to the ground plane. For example, the action in Figure 2 requires the camera to pan 60° to the right while remaining horizontally aligned with the ground plane. This motivates gravity-aligning the reconstructed scene to provide a canonical world frame for converting semantic actions into geometric camera transformations. The 3DFM $\mathcal{G}$ first recovers the scene geometry and camera poses from the unposed input views $\mathcal{V}$:

$$\mathcal{G}(\mathcal{V}) = (\mathcal{Z}, \mathcal{P}), \tag{3}$$

where $\mathcal{Z}$ denotes the reconstructed scene as a point cloud and $\mathcal{P} = \{p_i\}_{i=1}^{N}$ the estimated camera poses. The planned action Δv is then deterministically mapped to a 6-DoF camera transformation relative to the reference view pose p_r in this gravity-aligned reconstruction, with the scene geometry providing the scale required to instantiate its coarse magnitude. Reprojecting the point cloud from the resulting target pose gives the synthesized view:

$$v_s = \mathcal{R}\left(\mathcal{Z}, p_r \oplus \Delta v\right), \tag{4}$$

where $\mathcal{R}$ denotes the reprojection operator and $\oplus$ the relative camera transformation induced by Δv. Since v_s only rearranges pixels observed in the input views, scene surfaces not covered by any input view remain empty, so the synthesized view supplements rather than replaces the input observations. More implementation details of view synthesis are provided in the Appendix.

Final VQA: *Answering with Contextual Augmentation.* For final answer prediction, we augment the original inputs with the synthesized view v_s and the reasoning contexts produced during question analysis and view planning. Specifically, c_{ana} preserves the resolved directional frame and observation need, while c_{guide} conveys how the synthesized view should be interpreted with respect to the relevant anchors and spatial cues. The VLM $\mathcal{F}$ then answers the question with the synthesized view and the reasoning contexts:

$$\hat{y} = \mathcal{F}\left(q, \mathcal{V}, v_s, c_{\text{ana}}, c_{\text{guide}}\right), \tag{5}$$

where $\hat{y}$ is the predicted answer. The original input views remain available throughout this stage, since the synthesized view may be incomplete, allowing the model to combine evidence from both the observed views and the synthesized observation. These three augmentation components, namely the synthesized view, question analysis, and reasoning guidance, are evaluated separately in our ablation study in Section 4.3.

4 Experiments

4.1 Setup

Implementation Details. We evaluate Vantage on six VLMs for a wide range of model sizes, *i.e.*, GPT-5.4 (OpenAI, 2026), Gemma-4-31B (Team et al., 2026), Qwen3.6-27B (Qwen Team, 2026), InternVL3-8B (Zhu et al., 2025), Qwen3-VL-8B, and Qwen3-VL-4B (Yang et al., 2025). For the open-source models, we run inference using vLLM (Kwon et al., 2023) under the official configurations with two NVIDIA H200 GPUs. We employ G3T (Kani & Snavely, 2026) as the default 3D foundation model (3DFM) for gravity-aligned scene reconstruction and camera pose estimation, and deploy it on the same two GPUs.

Benchmarks. We primarily evaluate on multi-view spatial VQA in static scenes, excluding tasks that involve temporal dynamics or simulated environments. Our main evaluation covers MindCube-tiny (Wang et al., 2026c), the Positional Relationship and Attribute subsets of MMSI-Bench (Yang et al., 2026b), and the Multi-view Reasoning subset of BLINK (Fu et al., 2024). To evaluate the broader applicability of the Seek-and-View paradigm, we additionally include the Perspective Taking subset of OmniSpatial (Jia et al., 2026) for single-view spatial understanding, and the Dynamic Rotation and Dynamic Translation subsets of SPINBench (Zhang et al., 2026g), where objects or subjects rotate or translate while the background scene remains static.

We compare with the state-of-the-art methods (Cai et al., 2026; Chen et al., 2026b) on three multi-view benchmarks. Following the evaluation protocols of prior work (Wang et al., 2026c; Chen et al., 2026b), we report accuracy (%) for each subset and the average accuracy (%) across subsets. Unless otherwise specified, all evaluations use the benchmark subsets described above. Details on each benchmark are provided in the Appendix.

4.2 Main Results

Consistent Gains Across Benchmarks. Table 1 shows that Vantage consistently improves the average accuracy of all six VLMs without additional training. The gains extend to strong base performance, with Qwen3.6-27B improving from 90.5% to 93.5% on the MindCube-tiny Rotation subset. More substantial gains are observed when cross-view reasoning remains challenging, as Gemma-4-31B improves from 51.0% to 82.0% on the same subset. Beyond the multi-view benchmarks, Vantage also improves five of the six models on OmniSpatial, where viewpoint adjustment

Table 1: **Main results on five benchmarks.** For each base model, the better result is shown in **bold**. The final two columns report the average accuracy across all five benchmarks and the relative improvement. †Results are reported as in their original papers.

Model	MindCube-tiny				MMSI-Bench			BLINK	OmniSpatial	SPINBench			Avg.	% Gain
	Rot.	Ard.	Amg.	All	PR.	Attr.	All	MV	Pers.	D.Rot	D.Tr	All		
Training-Scaled Spatial Model														
SenseNova-SI-1.5-InternVL3-8B	92.5	90.8	94.5	93.24	47.1	37.7	45.2	63.9	52.0	43.6	41.7	43.0	59.5	-
Model-Scaled Spatial Agent														
GCA (Qwen3-VL-235B-A22B-Thinking)†	82.0	61.8	59.8	64.2	52.8	45.0	51.2	-	58.6	-	-	-	-	-
Vantage on base models														
Qwen3-VL-4B	33.0	34.0	20.3	26.0	27.4	30.0	27.9	41.4	42.1	48.2	**64.7**	53.2	38.1	–
+ Vantage	**35.0**	**46.8**	**36.8**	**38.9**	**34.7**	30.0	**33.7**	**42.1**	**47.2**	**49.3**	62.8	**53.4**	**43.1**	+13.1
Qwen3-VL-8B	36.5	**33.2**	28.8	31.3	**31.4**	24.6	30.1	44.4	43.9	**44.2**	73.7	**53.2**	40.6	–
+ Vantage	36.5	28.8	**31.2**	**31.6**	29.9	**30.8**	30.1	**45.1**	**45.1**	42.5	**75.0**	52.5	**40.9**	+0.7
InternVL3-8B	**35.5**	**47.2**	32.8	36.8	29.3	24.6	28.4	54.1	**44.0**	47.0	**53.2**	48.9	42.4	–
+ Vantage	31.5	41.2	**42.2**	**39.9**	**30.8**	**27.7**	**30.2**	**55.6**	43.9	**52.1**	49.4	**51.3**	**44.2**	+4.2
Qwen3.6-27B	90.5	**71.2**	50.5	63.0	42.5	41.5	42.3	28.6	52.6	**85.3**	98.1	**89.2**	55.1	–
+ Vantage	**93.5**	70.0	**60.0**	**68.8**	**44.3**	**46.9**	**44.8**	**29.3**	**64.7**	83.6	98.1	88.0	**59.1**	+7.3
Gemma-4-31B	51.0	**63.6**	50.2	53.5	35.4	**42.3**	36.8	36.1	51.7	59.8	**96.8**	71.1	49.8	–
+ Vantage	**82.0**	56.0	**57.7**	**61.9**	**36.2**	40.8	**37.1**	**44.4**	**55.1**	**64.9**	94.2	**73.9**	**54.5**	+9.4
GPT-5.4	38.5	50.8	43.3	44.2	33.3	38.5	34.4	48.1	47.4	49.0	82.7	59.3	46.7	–
+ Vantage	**61.5**	**56.8**	**50.8**	**54.3**	**41.8**	38.5	**41.1**	**54.1**	**53.7**	**53.8**	**86.5**	**63.9**	**53.4**	+14.3

Table 2: **Cross-source composition results.** *Self* denotes steps executed by the evaluated model, while *Cross* denotes steps replaced by Qwen3.6-27B. Bold indicates the highest accuracy within each model group. Δ denotes the accuracy change relative to model + Vantage.

Model	Question Analysis	View Planning	MindCube-tiny		MMSI-Bench		BLINK		OmniSpatial	
			Acc.	Δ	Acc.	Δ	Acc.	Δ	Acc.	Δ
Qwen3-VL-4B	–	–	26.0	−12.9	27.9	−5.8	41.4	−0.7	42.1	−5.1
	Self	Self	38.9	–	33.7	–	**42.1**	–	47.2	–
	Cross	Self	**51.0**	+12.1	35.3	+1.6	37.6	−4.5	**50.6**	+3.4
	Cross	Cross	49.9	+11.0	**36.2**	+2.5	36.8	−5.3	48.1	+0.9
Qwen3-VL-8B	–	–	31.3	−0.3	30.1	0.0	44.4	−0.7	43.9	−1.2
	Self	Self	31.6	–	30.1	–	**45.1**	–	45.1	–
	Cross	Self	50.6	+19.0	**35.1**	+5.0	36.1	−9.0	**51.3**	+6.2
	Cross	Cross	**50.8**	+19.2	34.8	+4.7	33.1	−12.0	49.0	+3.9
Qwen3.6-27B	–	–	63.0	−5.8	42.3	−2.5	28.6	−0.7	52.6	−12.1
	Self	Self	**68.8**	–	**44.8**	–	**29.3**	–	**64.7**	–

can expose inferred 3D structures through parallax and provide auxiliary depth cues. On the dynamic SPINBench subsets, Vantage improves four of the six models, suggesting that the sought view helps to provide spatial cues for object-motion reasoning. Overall, these results demonstrate the consistent benefits of Vantage through viewpoint-grounded reasoning and geometry-grounded evidence augmentation to enhance the spatial reasoning.

Comparison with State-of-the-Art Methods. We also compare with state-of-the-art methods on three multi-view benchmarks. These approaches achieve strong performance through either extensive spatial-task training or tool-integrated agents built on substantially larger VLMs. In contrast, Vantage achieves its improvements with a training-free and diagnostic pipeline, reaching comparable or superior performance on several subsets without scaling task-specific training or model capacity for reliable tool calling. This highlights the potential of Seek-and-View as a complementary direction for advancing multi-view spatial reasoning.

Analysis and Planning Are Critical to Seek-and-View. The gains vary across models and do not increase monotonically with model size. To study further into this behavior, we compose the two steps in Stage 1, *i.e.*, question analysis and view planning, using different model sources, while keeping Stage 2 unchanged. As shown in Table 2, the additional gains for the two smaller models depend strongly on the quality of question analysis and view planning, particularly for Qwen3-VL-8B. Directly replacing both steps with outputs from the larger Qwen3.6-27B does not always yield the best performance. Instead, stronger results are obtained often when the smaller model performs view planning based on the question analysis by Qwen3.6-27B. This suggests that the camera action and reasoning guidance should be aligned with the reasoning needs of the model for final VQA.

Table 3: **Question analysis (QA) step ablation.** QA-skip plans views without the analysis; QA-only augments final VQA with analysis alone.

Model	Setting	MindCube-tiny	MMSI-Bench
Qwen3-VL-4B	Baseline	26.0	27.9
	Vantage	**38.9**	**33.7**
	QA-skip	30.0	26.8
	QA-only	34.8	33.3
Qwen3-VL-8B	Baseline	31.3	30.1
	Vantage	**31.6**	30.1
	QA-skip	31.2	**31.6**
	QA-only	27.5	28.5
Qwen3.6-27B	Baseline	63.0	42.3
	Vantage	**68.8**	**44.8**
	QA-skip	65.0	43.3
	QA-only	63.4	42.2
Gemma-4-31B	Baseline	53.5	36.8
	Vantage	**61.9**	37.1
	QA-skip	58.7	**37.3**
	QA-only	57.9	36.5

Table 4: **Action interface ablation.** We compare action selection, 6-DoF prediction, random actions and view interpolation on MindCube-tiny.

Model	Setting	Acc.	Δ
Qwen3-VL-4B	Action selection (ours)	**38.9**	–
	6-DoF prediction	34.6	−4.3
	Random action	33.0	−5.9
	View interpolation	32.1	−6.8
Qwen3-VL-8B	Action selection (ours)	31.6	–
	6-DoF prediction	31.2	−0.4
	Random action	31.8	+0.2
	View interpolation	**32.4**	+0.8
Qwen3.6-27B	Action selection (ours)	**68.8**	–
	6-DoF prediction	67.4	−1.4
	Random action	63.1	−5.7
	View interpolation	64.3	−4.5
Gemma-4-31B	Action selection (ours)	**61.9**	–
	6-DoF prediction	61.5	−0.4
	Random action	55.2	−6.7
	View interpolation	54.6	−7.3

Table 5: **Final VQA context ablation.** Checkmarks indicate the component provided to the original VQA. Bold and italic denote the best and second-best average accuracy within each model, respectively. The final two columns report the absolute and relative improvements over its baseline.

Model	Augmentation			Benchmark			Avg.	Abs Δ	Relative Δ
	View	Analysis	Guidance	MindCube-tiny	MMSI-Bench	OmniSpatial			
Qwen3-VL-4B				26.0	27.9	42.1	32.0	–	–
		✓	✓	34.5	32.1	43.1	36.6	+4.6	+14.4
	✓			29.7	32.7	43.5	35.3	+3.3	+10.3
	✓	✓		38.2	34.0	42.4	*38.2*	+6.2	+19.4
	✓		✓	36.7	31.4	42.1	36.7	+4.7	+14.7
	✓	✓	✓	38.9	33.7	47.2	**39.9**	**+7.9**	**+24.7**
Gemma-4-31B				53.5	36.8	51.7	47.3	–	–
		✓	✓	56.9	37.6	53.6	49.4	+2.1	+4.4
	✓			56.8	34.5	55.4	48.9	+1.6	+3.4
	✓	✓		59.3	36.4	55.3	*50.3*	+3.0	+6.3
	✓		✓	60.9	35.4	51.7	49.3	+2.0	+4.2
	✓	✓	✓	61.9	37.1	55.1	**51.4**	**+4.1**	**+8.7**

4.3 Ablation Studies

We evaluate the design of Vantage through ablations of its question analysis step, action interface, and final VQA context. Reconstruction backend comparisons are provided in the Appendix.

Role of the Question Analysis. As shown in Table 3, we compare the full pipeline with *QA-skip*, which plans the view without question analysis, and *QA-only*, which uses question analysis alone to augment the final VQA. For Qwen3-VL-4B, removing QA reduces accuracy from 38.9% to 30.0% on MindCube-tiny and from 33.7% to 26.8% on MMSI-Bench. Although QA alone improves over the baseline in several cases, Vantage outperforms both variants in most settings. This indicates that the QA step supports both view seeking and subsequent final reasoning.

Role of the Action Interface. Table 4 compares alternative strategies for determining the viewpoint. The 6-DoF variant directly predicts a numerical camera pose, the random variant samples from the predefined action set, and view interpolation synthesizes an intermediate observation between neighboring views. Across all four models, 6-DoF prediction underperforms our action-selection interface, while random actions and view interpolation provide limited gains overall. Notably, view interpolation only surpasses our method for Qwen3-VL-8B. These results support structured action selection as a more reliable interface between semantic planning and executable camera motion, and suggest that the gains of Seek-and-View come from seeking question-relevant viewpoints rather than simply adding denser or more continuous observations.

Role of the Visual and Textual Context. Table 5 examines how the synthesized view v_s, question analysis c_{ana}, and reasoning guidance c_{guide} contribute to the final VQA step. All preceding steps are kept fixed, with only the context supplied to the final VQA being varied. Using the synthesized view alone improves the average accuracy across the three benchmarks for all four models. Combining the view with textual context generally yields further gains, while the full combination consistently

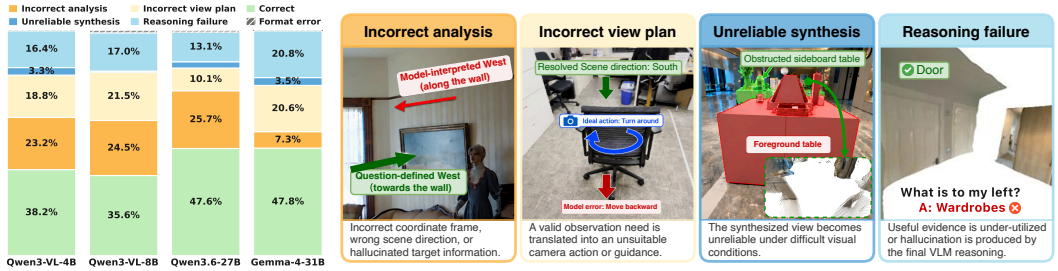


Figure 3: **Major failure types of Vantage.** *Left*: Macro-averaged failure analysis. *Right*: Representative failure cases by type. More detailed visualizations of these representative failure cases are provided in the Appendix.

outperforms the setting using only question analysis and reasoning guidance (row 2). These results indicate that visual and textual context play complementary roles: the synthesized view provides additional spatial evidence, while the analysis and guidance preserve the observation objective and help the model interpret this evidence during the final reasoning.

4.4 Discussion

Our training-free model-agnostic framework enables a diagnostic analysis of where current VLMs + Seek-and-View still fail. We audit 600 incorrect predictions across four models and three multi-view benchmarks. Figure 3 summarizes the estimated prevalence of different failure types and representative cases. Detailed categorization and estimation procedures are provided in the Appendix.

What Limits View Seeking? The dominant failures arise in the viewpoint-grounded reasoning stage. *Incorrect analysis* and *incorrect view plan* together account for most residual errors, indicating that deciding *what should be seen* and *how to reach it* remains the main bottleneck. As illustrated in Figure 3 (right), these failures reflect errors in analyzing the spatial context or in translating a valid observation need into an executable camera action. These results suggest that effective viewpoint seeking depends on both accurate question interpretation and reliable grounding from semantic intent to geometric action. This trend is further supported by the cross-source results in Table 2.

What Limits Reasoning from the Sought View? Failures at this stage mainly arise from *unreliable synthesis* or *reasoning failure*. Unreliable synthesis is rare and mostly occurs under challenging visual conditions, such as reflections, transparency, severe occlusion, close-up views, darkness, blur, or low resolution. This suggests that Seek-and-View tolerates imperfect rendering, as long as the synthesized view preserves sufficient geometric and semantic evidence for reasoning. Reasoning failures occur when the final VLM under-utilizes useful evidence in the sought view or produces an inconsistent reasoning trace. Thus, the sought view provides additional visual evidence to support cross-view alignment, while reliable spatial reasoning remains necessary for accurate answering.

5 Conclusion

We formulated Seek-and-View, a new reasoning paradigm that seeks question-relevant viewpoints to make spatial evidence distributed across views more directly observable. This paradigm alleviates the burden on semantic textual reasoning for global scene understanding through question-relevant, evidence-seeking and reasoning-oriented visual evidence acquisition. To instantiate Seek-and-View, we proposed Vantage, which integrates a VLM with a 3D foundation model through a viewpoint-grounded reasoning stage and a geometry-grounded evidence augmentation stage. Across five benchmarks and six VLMs, this training-free and model-agnostic framework consistently improves spatial reasoning. The results suggest that more robust multi-view spatial understanding does not always require strengthening reasoning over fixed observations; rather, seeking question-relevant visual evidence supports cross-view alignment and enhances subsequent reasoning.

Limitation & Future work. As a training-free framework, Vantage may yield rollouts of varying quality depending on the capabilities of the underlying VLM and 3DFM. Future work could leverage high-quality rollouts for end-to-end optimization toward more reliable spatial reasoning.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on the conceptual framing of the research, research methodology and experimental design. Generative AI tools were also used to assist with software development and debugging, the creation and refinement of scientific figures, literature search and summarization, and drafting and editing portions of the manuscript for clarity and readability. We did not use generative AI tools to generate synthetic datasets, formulate mathematical claims, construct or prove theoretical results, or clean or reformat datasets; these tasks are either not applicable to this work or were performed without generative AI assistance.

All AI-assisted research suggestions were independently evaluated by the authors against the relevant literature and empirical evidence. AI-assisted code was reviewed and tested, and scientific figures and manuscript text produced or modified with AI assistance were manually reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Zhongang Cai, Ruisi Wang, Chenyang Gu, Fanyi Pu, Junxiang Xu, Yubo Wang, Wanqi Yin, Zhitao Yang, Chen Wei, Tongxi Zhou, et al. Scaling spatial intelligence with multimodal foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7879–7890, 2026.

Meng Cao, Xingyu Li, Xue Liu, Ian Reid, and Xiaodan Liang. Spatialdreamer: Incentivizing spatial reasoning via active mental imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7176–7187, 2026.

Jiahua Chen, Qihong Tang, Weinong Wang, and Qi Fan. Enhancing mllm spatial understanding via active 3d scene exploration for multi-perspective reasoning. *arXiv preprint arXiv:2604.06725*, 2026a.

Zeren Chen, Xiaoya Lu, Zhijie Zheng, Pengrui Li, Lehan He, Yijin Zhou, Jing Shao, Bohan Zhuang, and Lu Sheng. Geometrically-constrained agent for spatial reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 38689–38699, 2026b.

An-Chieh Cheng, Yang Fu, Yatai Ji, Ligeng Zhu, Guanqi Zhan, Zhuoyang Zhang, Zhaojing Yang, Song Han, Yao Lu, Pavlo Molchanov, et al. Grounded 3d-aware spatial vision-language modeling. *arXiv preprint arXiv:2605.30307*, 2026.

Seokju Cho, Ryo Hachiuma, Abhishek Badki, Hang Su, Byung-Kwan Lee, Chan Hee Song, Sifei Liu, Subhashree Radhakrishnan, Seungryong Kim, Yu-Chiang Frank Wang, et al. Spatialclaw: Rethinking action interface for agentic spatial reasoning. *arXiv preprint arXiv:2606.13673*, 2026.

Jiho Choi, Seonho Lee, Seojeong Park, and Hyunjung Shim. Dense reward for multi-view 3d reasoning with global maps and local views. *arXiv preprint arXiv:2606.23557*, 2026.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pp. 148–166. Springer, 2024.

Mohsen Gholami, Ahmad Rezaei, Zhou Weimin, Sitong Mao, Shunbo Zhou, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. In *International Conference on Learning Representations*, volume 2026, pp. 101989–102021, 2026.

Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14953–14962. IEEE, 2023.

Wenbo Hu, Jingli Lin, Yilin Long, Yunlong Ran, Lihan Jiang, Yifan Wang, Chenming Zhu, Runsen Xu, Tai Wang, and Jiangmiao Pang. G$ˆ2$vlm: Geometry grounded vision language model with unified 3d reconstruction and spatial reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9535–9546, June 2026.

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and Ranjay Krishna. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. *Advances in Neural Information Processing Systems*, 37:139348–139379, 2024.

Muhammad Kamran Janjua, Hugo Silva, Di Niu, and Bahador Rashidi. Don't show pixels, show cues: Unlocking visual tool reasoning in language models via perception programs. *arXiv preprint arXiv:2604.12896*, 2026.

Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, Xinqiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. In *International Conference on Learning Representations*, volume 2026, pp. 35634–35670, 2026.

Zhao Jin, Rong-Cheng Tu, Jingyi Liao, Wenhao Sun, Xiao Luo, Shunyu Liu, and Dacheng Tao. Spazer: Spatial-semantic progressive reasoning agent for zero-shot 3d visual grounding. *Advances in Neural Information Processing Systems*, 38:165549–165576, 2026.

Bharath Raj Nagoor Kani and Noah Snavely. G3t up! gravity aligned coordinate frames simplify pointmap processing. *arXiv preprint arXiv:2605.27372*, 2026.

Fucai Ke, Zhixi Cai, Boying Li, Long Chen, Beibei Lin, Weiqing Wang, Pari Delir Haghighi, Gholamreza Haffari, and Hamid Rezatofighi. View2space: Studying multi-view visual reasoning from sparse observations. *arXiv preprint arXiv:2603.16506*, 2026.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Phillip Y Lee, Jihyeon Je, Chanho Park, Mikaela Angelina Uy, Leonidas Guibas, and Minhyuk Sung. Perspective-aware reasoning in vision-language models via mental imagery simulation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9241–9251, 2025.

Xudong Li, Mengdan Zhang, Peixian Chen, Jiaxi Tan, Zihao Huang, Jingyuan Zheng, Yan Zhang, Xiawu Zheng, Xing Sun, and Rongrong Ji. Omniview-space: Reinforcing spatial reasoning via multi-perspective spatial mapping. *arXiv preprint arXiv:2607.00881*, 2026.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Yifan Liu, Fangneng Zhan, Kaichen Zhou, Yilun Du, Paul Pu Liang, and Hanspeter Pfister. Abstract 3d perception for spatial intelligence in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 38647–38656, 2026.

OpenAI. Thinking with images. `https://openai.com/index/thinking-with-images/`, April 2025.

OpenAI. Introducing GPT-5.4. `https://openai.com/index/introducing-gpt-5-4/`, 2026.

Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and XuDong Wang. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens. *arXiv preprint arXiv:2511.19418*, 2025.

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27b dense model, April 2026. URL `https://qwen.ai/blog?id=qwen3.6-27b`.

Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11854–11864. IEEE, 2023.

Xingjian Tao, Yiwei Wang, Yujun Cai, Yifan Song, and Jing Tang. Viewfusion: Structured spatial thinking chains for multi-view reasoning. In *European Conference on Computer Vision*, pp. 135–151. Springer, 2026.

Gemma Team, Sherif El Abd, Vaibhav Aggarwal, Robin Algayres, Alek Andreev, Olivier Bachem, Ian Ballantyne, Cormac Brick, Victor Cărbune, Michelle Casbon, et al. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Alexander Veicht, Paul-Edouard Sarlin, Philipp Lindenberger, and Marc Pollefeys. Geocalib: Learning single-image calibration with geometric optimization. In *European Conference on Computer Vision*, pp. 1–20. Springer, 2024.

Haochen Wang, Yucheng Zhao, Tiancai Wang, Haoqiang Fan, Xiangyu Zhang, and Zhaoxiang Zhang. Ross3d: Reconstructive visual instruction tuning with 3d-awareness. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9275–9286. IEEE, 2025a.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025b.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω. *arXiv preprint arXiv:2605.15195*, 2026a.

Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Yiping Lu, Zhengyuan Yang, Lijuan Wang, et al. Vagen: Reinforcing world model reasoning for multi-turn vlm agents. *Advances in Neural Information Processing Systems*, 38:172871–172933, 2026b.

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. In *The Fourteenth International Conference on Learning Representations*, 2026c. URL https://openreview.net/forum?id=0FhrtdKLtD.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025c.

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. *Advances in Neural Information Processing Systems*, 38:143297–143330, 2026.

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

Qian Yang, Ankur Sikarwar, Huy Le, Le Zhang, Zhuan Shi, Perouz Taslakian, and Aishwarya Agrawal. How and what to imagine? visual thinking in unified multimodal models for cross-view spatial reasoning. *arXiv preprint arXiv:2605.27310*, 2026a.

Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, et al. Mmsi-bench: A benchmark for multi-image spatial intelligence. In *International Conference on Learning Representations*, volume 2026, pp. 157051–157088, 2026b.

Yuncong Yang, Jiageng Liu, Zheyuan Zhang, Siyuan Zhou, Reuben Tan, Jianwei Yang, Yilun Du, and Chuang Gan. Mindjourney: Test-time scaling with world models for spatial reasoning. *Advances in Neural Information Processing Systems*, 38:109855–109885, 2026c.

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33510–33520, 2026d.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Jilin Mei, Chunhui Chen, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, et al. From flatland to space: Teaching vision-language models to perceive and reason in 3d. *Advances in Neural Information Processing Systems*, 38, 2026a.

Jian Zhang, Shijie Zhou, Bangya Liu, Achuta Kadambi, and Zhiwen Fan. Spatialstack: Layered geometry-language fusion for 3d vlm spatial reasoning. *arXiv preprint arXiv:2603.27437*, 2026b.

Shihua Zhang, Qiuhong Shen, Shizun Wang, Tianbo Pan, and Xinchao Wang. Make geometry matter for spatial reasoning. In *European Conference on Computer Vision*, pp. 231–249. Springer, 2026c.

Wanyue Zhang, Wenxiang Wu, Wang Xu, Jiaxin Luo, Helu Zhi, Yibin Huang, Shuo Ren, Zitao Liu, and Jiajun Zhang. World2vlm: Distilling world model imagination into vlms for dynamic spatial reasoning. *arXiv preprint arXiv:2604.26934*, 2026d.

Wenhao Zhang, Yuexiang Xie, Yuchang Sun, Yanxi Chen, Guoyin Wang, Yaliang Li, Bolin Ding, and Jingren Zhou. On-policy rl meets off-policy experts: Harmonizing supervised fine-tuning and reinforcement learning via dynamic weighting. In *International Conference on Learning Representations*, volume 2026, pp. 120693–120726, 2026e.

Yanbing Zhang, Bo Wang, Jianhui Liu, Nan Jiang, Jiaxiu Jiang, Haoze Sun, Yijun Yang, Shenghe Zheng, Lin Song, Haoyang Huang, et al. Thinking with novel views: A systematic analysis of generative-augmented spatial intelligence. *arXiv preprint arXiv:2605.10588*, 2026f.

Yuyou Zhang, Radu Corcodel, Chiori Hori, Anoop Cherian, and Ding Zhao. Spinbench: Perspective and rotation as a lens on spatial reasoning in vlms. In *International Conference on Learning Representations*, volume 2026, pp. 70072–70141, 2026g.

Zaibin Zhang, Yuhan Wu, Lianjie Jia, Yifan Wang, Zhongbo Zhang, Yijiang Li, Binghao Ran, Fuxi Zhang, Zhuohan Sun, Yizhuang Peng, et al. Think3d: Thinking with space for spatial reasoning. *arXiv preprint arXiv:2601.13029*, 2026h.

Duo Zheng, Yanyang Li, Liwei Wang, et al. Learning from videos for 3d world: Enhancing mllms with 3d vision geometry priors. *Advances in neural information processing systems*, 38:20560–20586, 2026.

Chenming Zhu, Jingli Lin, Yilin Long, Peizhou Cao, Tai Wang, Jiangmiao Pang, and Xihui Liu. Thinking with imagination: Agentic visual spatial reasoning with world simulators. *arXiv preprint arXiv:2606.06476*, 2026.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

APPENDIX

The appendix is organized as follows:

- Section A provides additional design and implementation details of Vantage, including the action set, view synthesis pipeline, gravity alignment, reconstruction backend, and prompt templates.
- Section B provides detailed descriptions of the evaluation benchmarks and subset selection.
- Section C presents additional qualitative results and the detailed failure analysis protocol.

A MORE DETAILS OF VANTAGE

This section provides additional details of Vantage, covering the action space, view synthesis procedure, gravity alignment study, and prompt templates.

Table 6: **Action set for view planning.** Each semantic action is deterministically mapped to a geometric transformation with three magnitude levels (small, medium, and large), except for *turn around*, which uses a fixed 180° rotation. Translation magnitudes are expressed in scene unit s.

Category	Action	Transformation	Small	Medium	Large
Rotation	Pan left	$+\Delta\mathrm{yaw}$	30°	60°	90°
	Pan right	$-\Delta\mathrm{yaw}$	30°	60°	90°
	Tilt up	$+\Delta\mathrm{pitch}$	30°	60°	90°
	Tilt down	$-\Delta\mathrm{pitch}$	30°	60°	90°
	Turn around	$\Delta\mathrm{yaw}$		180°	
Translation	Move right	$+\Delta x$	$0.40s$	$0.80s$	$1.40s$
	Move left	$-\Delta x$	$0.40s$	$0.80s$	$1.40s$
	Pedestal down	$+\Delta y$	$0.30s$	$0.55s$	$0.90s$
	Pedestal up	$-\Delta y$	$0.30s$	$0.55s$	$0.90s$
	Move forward	$+\Delta z$	$0.40s$	$0.80s$	$1.40s$
	Move backward	$-\Delta z$	$0.40s$	$0.80s$	$1.40s$
Hybrid	Orbit left	Orbit around centroid, left	30°	60°	90°
	Orbit right	Orbit around centroid, right	30°	60°	90°
	Orbit up	Orbit around centroid, upward	30°	60°	90°
	Orbit down	Orbit around centroid, downward	30°	60°	90°
	Bird's-eye view	$+\Delta\mathrm{pitch}$	40°	55°	70°
	Worm's-eye view	$-\Delta\mathrm{pitch}$	40°	55°	70°

A.1 Action Set for View Planning

We provide the complete action set used for view planning in Table 6, together with the corresponding magnitude levels. We adopt the camera convention $+\mathrm{yaw}$ for leftward rotation, $+\mathrm{pitch}$ for upward tilt, and $+x$, $+y$, and $+z$ for rightward, downward, and forward translation, respectively. Rotation actions modify only the viewing direction, whereas translation actions move the camera by a distance specified in scene units. For hybrid transformations, orbit actions rotate the camera around the centroid of the reconstructed point cloud, while bird's-eye and worm's-eye views introduce additional pitch rotations to obtain elevated or lowered viewpoints. Since the reconstructed point cloud is not metrically scaled, absolute translation distances are not directly transferable across scenes. We therefore define one scene unit s as the mean Euclidean distance from the reference camera center to the remaining input camera centers, providing a scene-adaptive scale that reflects the spatial spread of the observed viewpoints.

A.2 View Synthesis Pipeline

Given the target camera extrinsics determined above, we further specify the camera intrinsics for view synthesis. In particular, the field of view (FoV) is chosen to preserve sufficient scene coverage after the planned camera transformation. For rotation and translation actions, we use a fixed FoV of 120° to provide a sufficiently wide observation. For hybrid transformations, which can substantially change the camera position relative to the scene, we adapt the FoV to maintain coverage of the reconstructed scene. For example, a bird's-eye transformation may change the apparent scale of a compact scene, and the FoV is adjusted accordingly to retain informative coverage. The output follows the aspect ratio of the reference view, while reprojection is performed at twice the target width and height before downsampling to the resolution of the reference view.

Table 7 illustrate the effect of the FoV while keeping the upstream viewpoint-grounded reasoning fixed. Reducing the diagonal FoV from 120° to 80° lowers accuracy by 0.7 and 2.5 % for Qwen3.6-27B and Gemma-4-31B, respectively. This suggests that FoV affects how much spatial context can be captured within the synthesized view, which is particularly important for complex real-world scenes where relevant evidence may span a broader spatial region.

Table 7: Field-of-view ablation on MindCube.

Model	**FoV**	**MindCube-tiny**	Δ
Qwen3.6-27B	120° (ours)	**68.8**	–
	80°	68.1	-0.7
Gemma-4-31B	120° (ours)	**61.9**	–
	80°	59.4	-2.5

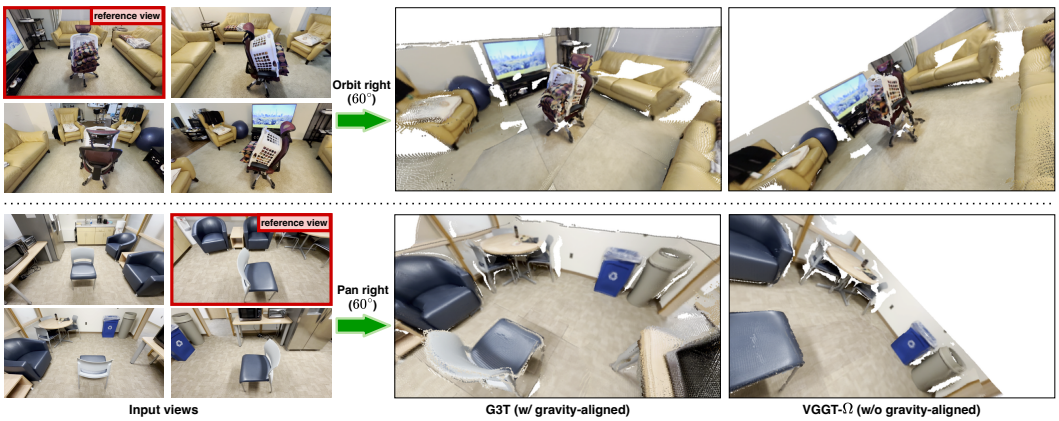


Figure 4: **Effect of gravity alignment on view synthesis.** Given the same input views, reference view, and planned camera action, we compare view synthesis using a gravity-aligned G3T reconstruction and a non-gravity-aligned VGGT-Ω reconstruction. Gravity alignment provides a ground-consistent coordinate frame for executing relative camera actions, while an unaligned reconstruction can cause the realized viewpoint to deviate from the intended ground-relative motion.

Table 8: Reconstruction backends ablation.

Model	Backend	MindCube-tiny	MMSI-Bench	Avg.
Qwen3-VL-4B	G3T	**38.9**	**33.7**	**36.3**
	VGGT-Ω + GeoCalib	37.4	30.7	34.1
Qwen3-VL-8B	G3T	31.6	**30.1**	**30.9**
	VGGT-Ω + GeoCalib	31.6	29.9	30.8
Qwen3.6-27B	G3T	**68.8**	**44.8**	**56.8**
	VGGT-Ω + GeoCalib	67.2	42.2	54.7
Gemma-4-31B	G3T	61.9	**37.1**	49.5
	VGGT-Ω + GeoCalib	**63.5**	35.6	**49.6**

Our action interface assumes a ground-aligned coordinate frame, as camera motions such as forward, backward, and lateral translation are defined relative to the ground plane. Accordingly, the reconstructed scene should be aligned with gravity before applying the planned camera transformation. Figure 4 illustrates this effect by applying the same planned action under different reconstruction configurations, showing how gravity alignment influences the realized camera motion.

Reconstruction Backend. Table 8 compares G3T (Kani & Snavely, 2026) with VGGT-Ω (Wang et al., 2026a) combined with GeoCalib (Veicht et al., 2024), representing two routes to the gravity-aligned reconstruction required by our action interface. G3T directly produces a gravity-aligned reconstruction, whereas the alternative first reconstructs the scene with VGGT-Ω and then uses GeoCalib on the reference view to estimate the gravity direction for alignment. Both backends yield comparable downstream performance. We use G3T as the default backend, as it integrates gravity alignment directly into reconstruction and avoids the additional calibration step, while the VGGT-Ω + GeoCalib pipeline requires approximately $1.4\times$ the inference time.

A.3 Prompt Templates

Prompts 1–4 provide the prompt templates used throughout Vantage pipeline.

Prompt 1 is used for viewpoint analysis. Given the input views $\mathcal{V}$ and question q, the VLM produces a structured, viewpoint-centric analysis c_{ana} that identifies the reference view, anchor entity, frame of reference, and observation need. At this step, the model does not answer the question or select a camera action.

Prompt 2 is used for view planning. Conditioned on the input views, question, and viewpoint analysis, the VLM selects a relative camera action Δv with respect to the reference view and generates

the reasoning guidance c_{guide} for subsequent answering. The action is represented by a predefined action type and magnitude level, rather than direct numerical camera parameters.

Prompt 3 is used for final VQA. The synthesized view v_s is appended to the original input views, and the VLM receives the original question together with c_{ana} and c_{guide} as additional context for final answering.

Prompt 4 defines the fallback setting triggered by a format error. In this case, the VLM answers using only the original input views together with the available textual context, namely the question analysis and, if available, the reasoning guidance.

B Benchmark Details

This section provides additional descriptions of the benchmarks summarized in Table 1 of the main paper, including the evaluated subsets and our subset selection criteria.

B.1 Benchmark Descriptions

MindCube-tiny. MindCube evaluates spatial mental modeling from limited multi-view observations, covering spatial relations that require reasoning about object positions, orientations, and hypothetical viewpoint changes (Wang et al., 2026c). We evaluate on the MindCube-tiny split and report its Rotation, Among, and Around subsets.

MMSI-Bench. MMSI-Bench evaluates multi-view spatial intelligence over real-world scenes, covering positional relations among cameras, objects, and regions, as well as attribute, motion, and multi-step reasoning (Yang et al., 2026b). We focus on the subsets centered on static multi-view spatial reasoning and exclude the Motion and Multi-Step Reasoning subsets. The former involves temporal changes across observations, while the latter places greater emphasis on compositional reasoning over multiple spatial steps. Both introduce additional reasoning factors beyond the setting targeted by Seek-and-View, which focuses on seeking question-relevant visual evidence when spatial information is fragmented across sparse observations or difficult to express and resolve through language alone.

BLINK. BLINK is a diagnostic benchmark covering wide perception-demanding visual tasks, including multi-view reasoning, relative depth, visual correspondence, and spatial relations (Fu et al., 2024). We use only its Multi-view Reasoning subset, which requires reasoning across observations from different viewpoints and is therefore most aligned with our multi-view spatial setting.

OmniSpatial. OmniSpatial evaluates spatial reasoning across four broad dimensions: dynamic reasoning, complex spatial logic, spatial interaction, and perspective taking (Jia et al., 2026). We include only the Perspective Taking subset, which evaluates egocentric, allocentric, and hypothetical viewpoint reasoning from a single observed scene. The remaining categories primarily emphasize temporal dynamics, interaction and planning, or abstract geometric and pattern reasoning, and are not designed around multi-view observations of real-world scenes.

SPINBench. SPINBench is a diagnostic benchmark for spatial reasoning under perspective and geometric transformations, covering translation, rotation, relative pose, and viewpoint change (Zhang et al., 2026g). We additionally evaluate on the Dynamic Rotation and Dynamic Translation subsets, where an object or subject undergoes controlled rotation or translation while the surrounding scene largely remains static. These subsets provide a limited extension beyond our core static multi-view setting by testing whether Seek-and-View can also support spatial reasoning when the queried entity itself changes pose or position.

C Additional Qualitative Analysis

This section provides additional qualitative analyses of Vantage, including representative success and failure cases shown in Figures 6–9 . We also provide further details of the failure analysis protocol used in the main paper.

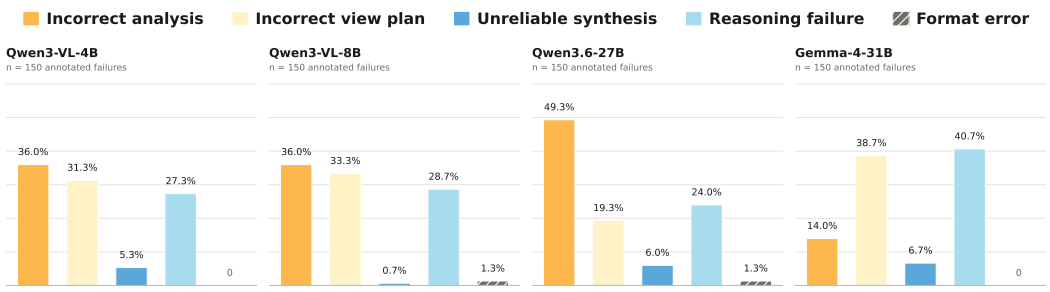


Figure 5: **Failure distribution by model.** For each model, we annotate 150 incorrect predictions sampled across the three multi-view benchmarks and categorize them into five failure types. Bars indicate the proportion of annotated failures falling into each category.

Failure Analysis Protocol. We randomly sample 50 incorrect predictions for each of four models on each of three multi-view benchmarks, yielding 600 annotated cases in total. Each case is assigned to one of five failure categories: *Incorrect analysis*, *Incorrect view plan*, *Unreliable synthesis*, *Reasoning failure*, or *Format error*. Figure 5 reports the resulting per-model failure distributions. To account for differences in model accuracy, we weight the sampled proportion of each failure type by the corresponding empirical error rate and then macro-average the resulting estimates across benchmarks. The aggregated results are reported in Figure 3 of the main paper.

Failure Case Visualizations. Figures 10–13 provide detailed visualizations of the representative cases shown in Figure 3 (right), covering the four major failure types and illustrating how errors arise at different stages of the Vantage pipeline.

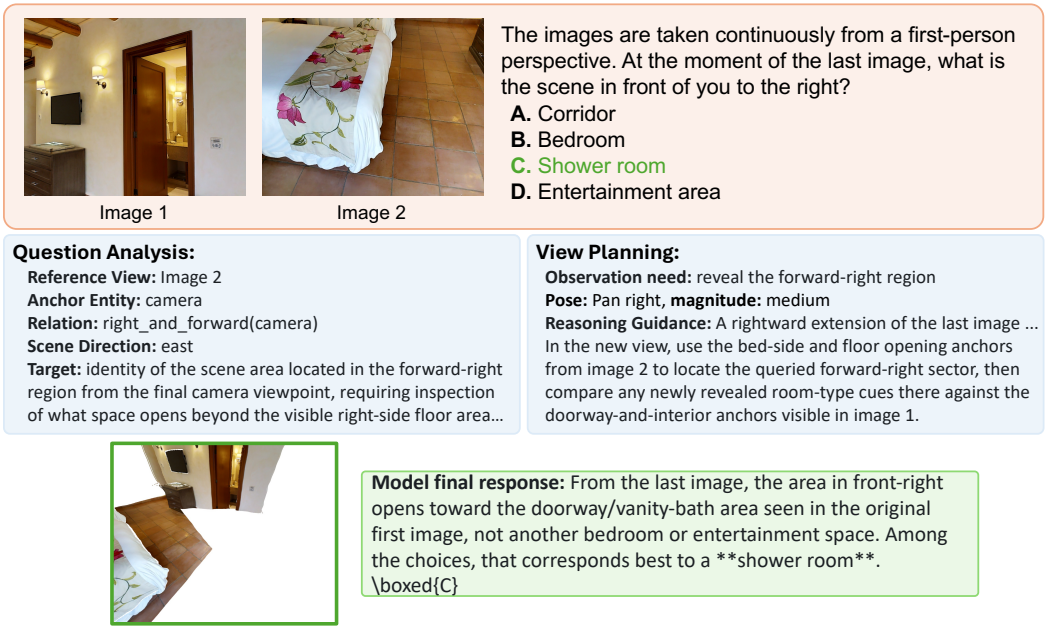


Figure 6: Qualitative example of Vantage with GPT-5.4. For clarity, we simplify the original model outputs and present only the key intermediate information relevant to the reasoning process.

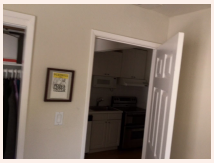


Image 1

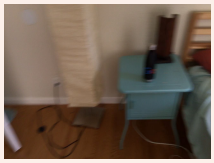


Image 2

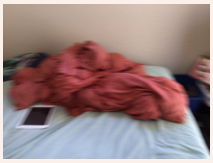


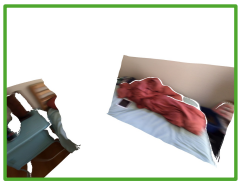
Image 3

Based on these three images: If I am standing at the same spot and facing the same direction as shown in image 3, what is to my left?

A. Door
B. Bed and red blanket
C. Bedside table

Question Analysis:
Reference View: Image 3
Anchor Entity: camera
Relation: left_of(camera)
Scene Direction: west
Target: identity of the scene/object occupying the person's left side when facing as in image 3, requiring inspection of the leftward adjacent region around the bed from the image 3.

View Planning:
Observation need: expose out-of-view content at the left side
Pose: Pan left, **magnitude:** small
Reasoning Guidance: A left-edge extension of the bed-facing reference view ... In the new rendered view, use the bed-and-red-blanket anchor from image 3 together with the bed-side furniture anchor from image 2, and check where that bed-adjacent region appears within this left-revealing continuation...



Model final response: C. Bedside table

Figure 7: Qualitative example of Vantage with GPT-5.4.


Image 1


Image 2

Based on these two views showing the same scene: in which direction did I move from the first view to the second view?

A. Directly left
B. Diagonally forward and right
C. Directly right
D. Diagonally forward and left

Question Analysis:
Reference View: Image 1
Anchor Entity: camera
Relation: movement(reference_viewpoint, second_view)
Scene Direction: need scene overview
Target: direction of camera translation relative to the scene layout, inferred from how persistent objects and background structures shift between the two views.

View Planning:
Observation need: reveal global scene layout
Pose: bird's eye view, **magnitude:** medium
Reasoning Guidance: A top-down overview of the same room, ... locate the window-sink kitchen side identified by image 1 and the partition side identified by image 2, then read where the table anchor and the two scene-facing directions fall within that shared overhead layout...



Model final response: B. Diagonally forward and right

Figure 8: Qualitative example of Vantage with GPT-5.4.

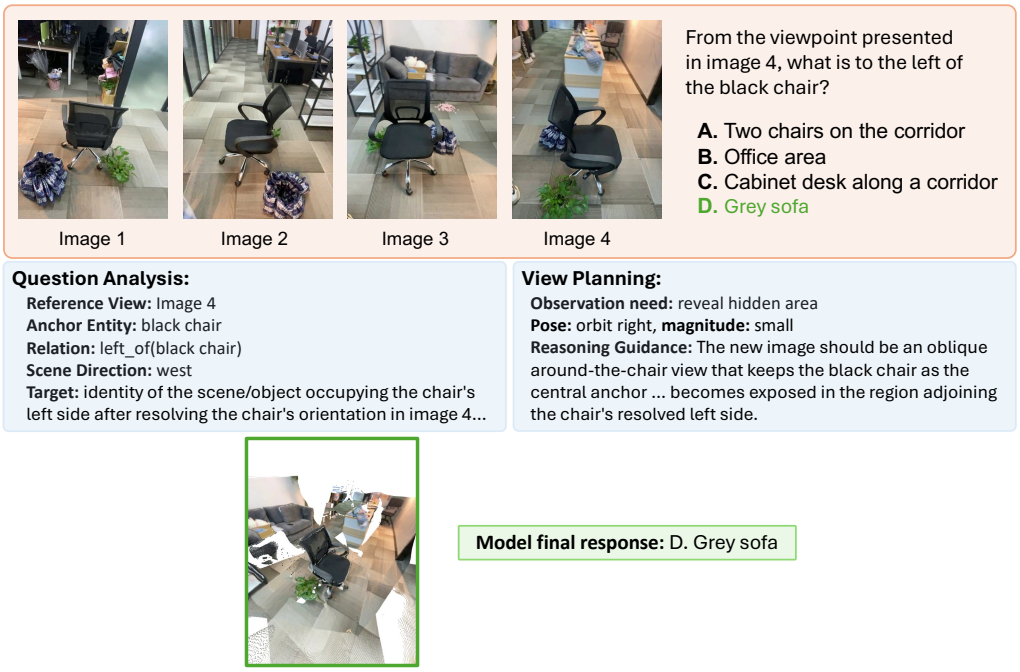


Figure 9: Qualitative example of Vantage with GPT-5.4.

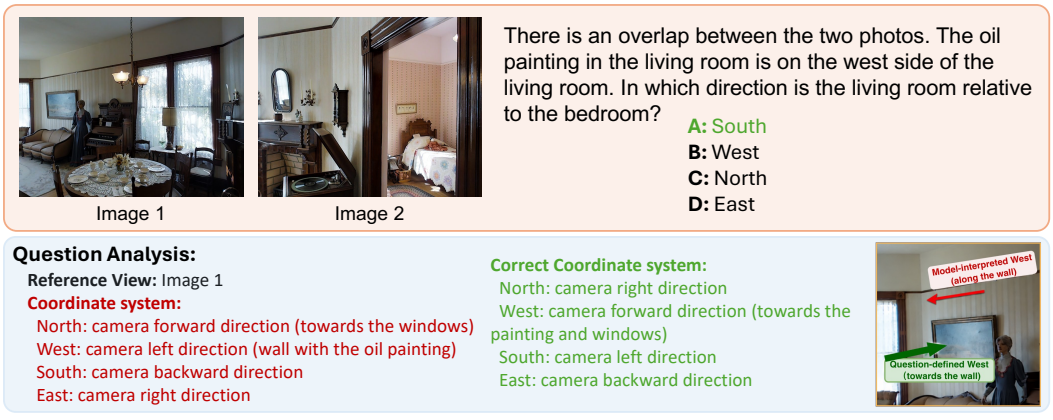


Figure 10: **Incorrect analysis.** Qualitative failure case from Qwen3.6-27B.

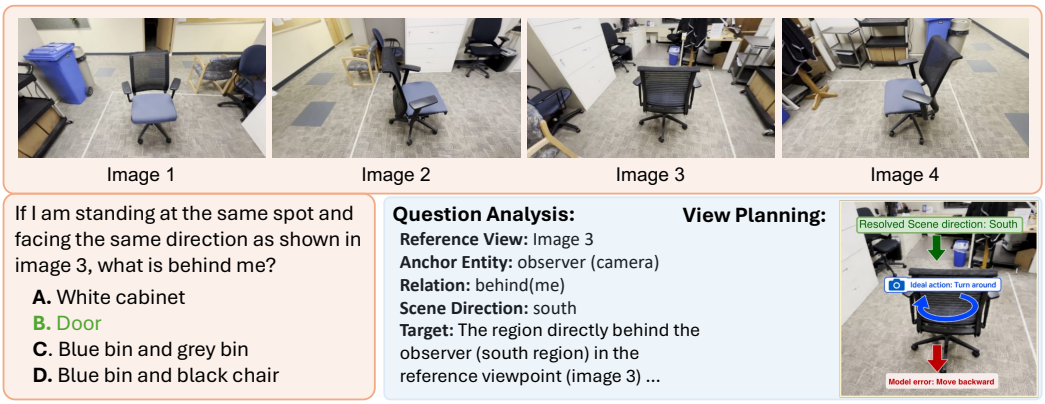


Figure 11: **Incorrect view plan.** Qualitative failure case from Qwen3-VL-4B.

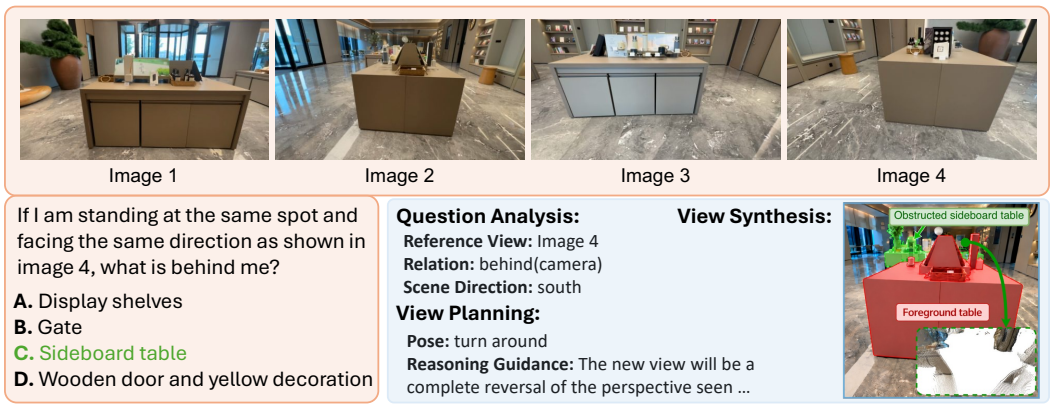


Figure 12: **Unreliable synthesis.** Qualitative failure case from Gemma-4-31B.

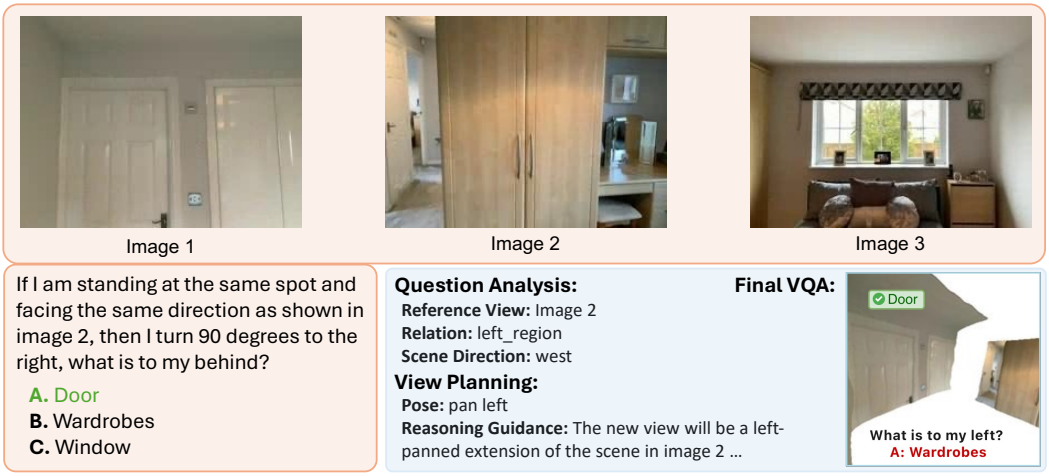


Figure 13: **Reasoning failure.** Qualitative failure case from Qwen3-VL-4B.

Prompt 1: Stage 1 - Question Analysis

```
You are an expert visual-spatial reasoning analyst.

**[CORE MISSION]**

Your sole mission is to analyze the question and construct a viewpoint-centric
    ↪ representation of it: the reference viewpoint to reason from, and the missing
    ↪ evidence and inspection region required to answer the question.

You are NOT allowed to answer the question. Your job is ONLY to:
1. Construct the local coordinate systems implied by the question.
2. Resolve any object-centric spatial relations into viewpoint-centric regions.
3. Identify what information is required and currently uncertain or missing.

**[KEY PRINCIPLE]**

The reference viewpoint is the only executable coordinate system.

Object-centric coordinates, scene-centric coordinates, and other inferred coordinate
    ↪ systems are intermediate reasoning tools only. All final outputs MUST be
    ↪ projected back into the reference viewpoint space.

**[COORDINATE SYSTEM RULES]**

By default, the reference viewpoint defines the coordinate system:
north = camera forward, west = camera left, south = camera backward, east = camera right.

When helpful (e.g., the view is oblique or the queried direction is easy to confuse),
    ↪ append visible scene content in parentheses to reduce 2D directional ambiguity, e
    ↪ .g., "north = camera forward direction (dining table in background)" or "west =
    ↪ camera left direction (kitchen area)". Annotate only content visible in the
    ↪ reference viewpoint.

If the question explicitly defines another coordinate system (e.g., "The oven is north of
    ↪  the sink."), infer the question-defined coordinate system before resolving
    ↪ spatial relations.

**[OUTPUT FORMAT]**

Return a single JSON object wrapped in ```json ... ```.

```json
{
    "question_decomposition": "...",
    "reference_view": "...",
    "reference_view_coordinate_system": {
        "north": "...",
        "west": "...",
        "south": "...",
        "east": "..."
        },
    "anchor_entity": "...",
    "entity_forward_direction": "...",
    "query_relation": "...",
    "resolved_scene_direction": "...",
    "viewpoint_inspection_target": "...",
    "target_information": "...",
    "gap_depth": "near | mid | far | unknown"
}
```

[FIELD DEFINITIONS]
- question_decomposition: A concise decomposition of the question.
- reference_view: The input image whose viewpoint the question explicitly refers to.
    ↪ Return its 1-indexed image index (1, 2, ...).
- reference_view_coordinate_system: The coordinate system based on the reference
    ↪ viewpoint.
    Examples: "north = camera forward direction (dining table in background), west =
    ↪ camera left direction, south = camera backward direction, east = camera right
    ↪ direction".
- anchor_entity: The entity (object or camera) whose coordinate system defines the
    ↪ queried spatial relation. If no object-centric frame exists, use camera.
- entity_forward_direction: Direction of the anchor entity's forward axis expressed in
    ↪ the reference viewpoint coordinate system. For entities without meaningful
    ↪ orientation (e.g. bottle, cup, ball), use camera's forward direction.
    Examples: north, east, south, west.
- query_relation: The spatial relation or target viewpoint from the question.
```

Prompt 1 (continued)

```
    Examples: left_of(<anchor_entity>), right_of(<anchor_entity>), behind(<anchor_entity
     ↪ >), right_and_forward(camera), viewpoint_movement(reference_view, <viewpoint_2>),
     ↪  object_distance_from(<anchor_entity>).
- resolved_scene_direction: For questions that require understanding the scene region of
     ↪ interest, choose the direction referenced by the question after resolving any
     ↪ anchor-centric coordinate system: north, east, south, west.
    For questions that require understanding the direction of the viewpoint movement:
     ↪ need_scene_overview.
- viewpoint_inspection_target: The most important scene evidence that should be inspected
     ↪  from the reference viewpoint to answer the question.
    Examples: left_region, right_region, region_behind_object,
     ↪ object_occlusion_resolution, upper_region, lower_region, horizontal_inspection,
     ↪ surrounding_neighbor_overview, scene_overview, scene_topdown_overview.
    Reminder: do not use coordinate system (e.g., "north_region(background)") to describe
     ↪  the target region in viewpoint space.
- target_information: The exact information required to answer the question from the
     ↪ reference viewpoint.
    Examples:
    - identity of object occupying the queried region
    - nearest object relative to chair
    - distance ordering among nearby objects
    - direction of camera translation relative to the scene layout
    - scene overview to establish the relative position of the anchor entity and the
     ↪ target entity
- gap_depth: Estimate how far the missing evidence lies from the anchor entity, can
     ↪ consider the scale of answer options and observed scene contents.
    Examples: near (0-1m), mid (1-5m), far (5-10m), or unknown (need the entire scene
     ↪ overview).


**[IMPORTANT RESTRICTIONS]**

DO NOT:
- answer the question
- determine relative camera positions between views
- reference other viewpoints when describing the target information
- speculate about unseen content and coordinate system from other viewpoints

Focus only on:
- coordinate system construction from the reference viewpoint
- the sequence of camera action instructions and relation resolution in the question
- projection into reference viewpoint space
- target information analysis


**[Question]**
⟨question⟩

Now analyze the question and provide the JSON output.
```

Prompt 2: Stage 1 - View Planning

```
You are an expert viewpoint planner.

A prior analysis has decomposed the question into a viewpoint-centric representation with
    ↪ :
- the reference viewpoint to start thinking from
- the absent evidence and required inspection region to reveal

**[CORE MISSION]**

Your sole mission is to plan the viewpoint change to reveal the informative evidence and
    ↪ compose a spatial-clue narrative for the downstream reasoner to build on.

You are NOT allowed to answer the question.
You are NOT allowed to reason across input views: all spatial reasoning stays WITHIN one
    ↪ view at a time, in the reference viewpoint's frame | never infer relative camera
    ↪ poses or shared layout between input views.

Your job is ONLY to determine which new viewpoint would provide sufficient evidence for
    ↪ reasoning.

**[KEY PRINCIPLE]**

The goal is NOT to directly choose a camera action.

The goal is:
1. Verify the observation gap from the reference viewpoint.
2. Identify the missing evidence and query region.
3. Determine the observation principle required to reveal the evidence.
4. Select the pose that best realizes that principle.
5. Choose an appropriate move magnitude.
6. Compose a spatial-clue narrative that:
   (a) predicts, at a high level, the geometric form the new view will take,
   (b) grounds each input view relevant to the question with observable orientation cues
     ↪ and background/scene anchors,
   (c) sets up the cross-view correspondences the reasoner needs to combine, without
     ↪ combining them.

**[OUTPUT FORMAT]**

Return a single JSON object wrapped in ```json ... ```.

```json
{
    "reasoning": "...",
    "missing_evidence": "...",
    "observation_principle": "...",
    "reference_view": "...",
    "pose": "<one of pan_left, pan_right, tilt_up, tilt_down, turn_around, move_forward,
     ↪ move_backward, move_left, move_right, pedestal_up, pedestal_down, orbit_left,
     ↪ orbit_right, orbit_up, orbit_down, birds_eye_view, worms_eye_view>",
    "magnitude": "<one of small, medium, large>",
    "reasoning guidance": "..."
}
```

[FIELD DEFINITIONS]
- reasoning: verify the stage-1 analysis with the question from the reference viewpoint,
    ↪ using within-view evidence only; do not relate input views to each other here.
- missing_evidence: Based on the 'target_information', identify the specific evidence
    ↪ that is unavailable from the reference viewpoint.
    Examples:
    - object immediately left of the chair
    - background hidden behind the cabinet
    - relative depth ordering among nearby objects
    - surrounding scene layout or need the entire scene overview
    - target object/view after the camera action instruction (e.g., turn left and move
     ↪ forward) from the question
- reference_view: The verified reference viewpoint from the stage-1 analysis.
- observation_principle: Determine what visual cue would reveal the missing evidence.
    Examples:
    - expose out-of-view content
    - reveal obstructed object/background/region
    - maintain anchor visibility while changing viewpoint
    - reveal global scene layout
    - follow the rotation instruction (e.g., turn left) to inspect the target object/view
- pose: Choose the pose that best realizes the observation principle.
You MUST use the expected visual consequence described below when selecting a pose.
    ⟨action set⟩
```

Prompt 2 (continued)

```
    Examples:
    - off-frame content: pan / tilt
    - hidden background: move / pedestal
    - object-relative relation: orbit
    - scene layout: birds_eye_view
    - camera-motion understanding: birds_eye_view or move_backward
    Use 'resolved_scene_direction' and 'viewpoint_inspection_target' to refine the final
     ↪ action.
- magnitude: Consider 'gap_depth' of the target region when choosing the magnitude:
    - small (30 degrees / one step)
    - medium (60 degrees / one and a half steps)
    - large (90 degrees / two steps)
    Magnitude is approximate for translation. Do NOT estimate metric distances.
- reasoning guidance: A spatial-clue narrative for the downstream reasoner. Write ONE
     ↪ paragraph composed of the three parts below, in order. The reasoner will see the
     ↪ input views, the original question, the newly rendered view, the stage-1 analysis
     ↪  and this reasoning guidance.
    (1) New-viewpoint form | a high-level, PREDICTIVE description of the viewing geometry
     ↪  the new view will take (e.g., "a top-down overview of the scene", "a right-orbit
     ↪  side-on view around the table anchor", "a right-panned extension of the scene in
     ↪  image N, beyond its right edge"). Describe only the viewing geometry; do NOT
     ↪ invent the specific content, objects, or spatial layout that will appear.
    (2) Input-view anchors | for EACH input view the question actually relies on, state
     ↪ as OBSERVABLE FACTS about that input image: its dominant scene-facing direction
     ↪ and one or two background/scene anchors that identify what it is looking at (e.g
     ↪ ., "image 1 faces the wall with the whiteboard, with the desk on its right edge",
     ↪  "image 2 faces the kitchen counter, with a window on its left edge"). Skip input
     ↪  views the question does not depend on. If the question depends on only one input
     ↪  view, only that one is described here.
    (3) Correspondence setup | name WHICH cross-view relations the reasoner should
     ↪ combine in the newly rendered view to answer the question, phrased as clues to
     ↪ combine (e.g., "locate where the whiteboard anchor from image 1 and the window
     ↪ anchor from image 2 fall within this top-down view, and read the two camera
     ↪ footprints against each other"). Point to the clues; do NOT combine them or state
     ↪  the conclusion.


**[IMPORTANT RESTRICTIONS]**

DO NOT:
- answer the question
- reproduce, paraphrase, or reasoning guidance at any answer option, in the reasoning
     ↪ guidance or elsewhere
- use motion verbs applied to the camera or objects when writing the reasoning guidance (
     ↪ e.g., "the camera moves right", "the object shifts forward", "turned left by ~45
     ↪ degrees") or otherwise describe the transformation between views as an action,
     ↪ direction, or trajectory
- quantify rotations, translations, degrees, steps, or metric distances in the reasoning
     ↪ guidance
- infer relative camera positions or any spatial relations between input views
- reference other viewpoints when describing the target information
- speculate about unseen content and coordinate system from other viewpoints

Focus only on:
- understand the coordinate system from the reference viewpoint
- verify the stage-1 analysis with the question
- explain the missing evidence -> observation principle -> pose selection
- select the pose that best realizes the observation principle
- choose an appropriate move magnitude
- compose a spatial-clue narrative that bridges the new viewpoint back to the input views
     ↪  and to the question without stating the answer

**[Analysis]**
⟨Question Analysis⟩

**[Question]**
⟨question⟩

Now analyze the question and provide the JSON output.
```

Prompt 3: Stage 2 - Final VQA

```
You are answering a spatial question about a scene. You are given the original
input view(s) plus ONE additional synthesized view, rendered from a new camera
viewpoint to expose evidence that was hard to read in the originals.

**[Images]**
- Image 1-N: the original input view(s).
- Image N+1: a SYNTHESIZED view, produced by moving the camera
  ("⟨magnitude⟩" "⟨action⟩") starting from input Image ⟨reference_view⟩. It may
  contain rendering artifacts | treat it as supporting evidence, not ground truth,
  and cross-check it against the original views.

You previously analyzed this question and produced the following view planning:
⟨Question Analysis⟩
⟨Reasoning Guidance⟩

Use all the images together to answer the original question below.

**[Question]**
⟨question⟩

⟨benchmark answer-format instruction⟩
```

Prompt 4: Stage 2 - Fallback for Final VQA

```
You previously analyzed this question and produced the following view planning:
⟨Question Analysis⟩
⟨Reasoning Guidance⟩

Now, using that analysis together with the input images, answer the original question.

**[Question]**
⟨question⟩

⟨benchmark answer-format instruction⟩
```