URUQI: LEARNING SPATIAL COGNITION FROM VISUAL EXPERIENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatial intelligence requires maintaining a coherent understanding of the world as the embodied agent moves. Like humans, the agent must use its own motion to interpret changes across observations and update object locations and spatial relations accordingly. Despite spatial post-training having substantially broadened the spatial intelligence of vision-language models (VLMs), they still struggle with two atomic spatial capabilities: tracking self-motion and mapping the surrounding world during motion. To address this gap, we provide dense multi-turn supervision over interleaved atomic capabilities within each training episode, mimicking the visual experience of a continuously moving agent that reasons as it observes. To scale this up, we synthesize 11,738 motif-driven camera trajectories over a broad range of 3D scenes, supporting self-motion tracking, persistent object mapping, and rich spatial operations within each visual experience. By training models to reason over these atomic questions, our URUQI$_{\text{Syn}}$-8B improves accuracy from 15.84% to 47.73% on our URUQI benchmark comprising 52k questions across 2.7k episodes. URUQI-SI-Mix-8B further reaches 50.41%, comparable to the 50.08% achieved by GPT-6 Astra. Trained solely on our synthesized data, URUQI$_{\text{Syn}}$-8B achieves an average relative accuracy improvement of 17.13% over its InternVL3-8B backbone across three external spatial benchmarks. These results highlight continuous visual experience as a scalable source of supervision for developing spatial cognition in VLMs.

1 INTRODUCTION

Spatial reasoning is becoming a core capability of general-purpose vision-language models (VLMs) that aim to understand the physical world. A VLM with native spatial intelligence is expected to ground open-vocabulary objects and integrate metric, relational, and reference-frame spatial information directly into open-ended visual-language reasoning. Recent work has shown that such capabilities can be substantially improved with large-scale spatial instruction tuning (Chen et al., 2024; Cheng et al., 2024; Cai et al., 2026).

However, it remains unclear whether success on diverse spatial questions reflects a coherent understanding of the scene across changing viewpoints. Figure 1 illustrates this issue with two examples from MMSI-Bench (Yang et al., 2026c) through self-motion tracking (M1) and object mapping (M2). Qwen3.8-27B (Qwen Team, 2026b) and SenseNova-SI-1.5 (Cai et al., 2026) make inaccurate motion and object-location predictions, while their estimates of the same objects are inconsistent across viewpoints. GPT-5.5 (OpenAI, 2026a) also predicts inaccurate object locations, but its estimates remain consistent across views. We swap the query order of M1 and M2 and find that its object predictions change accordingly. This dependency on query order suggests that GPT-5.5 computes object locations from earlier location estimates and its predicted camera motion. These results highlight a mismatch between task accuracy and coherent spatial understanding across observations.

To address this gap, prior work has explored supervising intermediate reasoning steps through chain-of-thought (Liu et al., 2025; Cai et al., 2026; Li et al., 2026a), cognitive maps (Wang et al., 2025; Yang et al., 2025), structured representations (Hua et al., 2026), and executable reasoning programs (Marsili et al., 2025). However, linguistic reasoning traces provide potentially ambiguous supervision for geometric state transitions (Kancheti et al., 2026), while explicit maps and programs rely on predefined representations and task-specific operations. This leaves a more general question:

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.

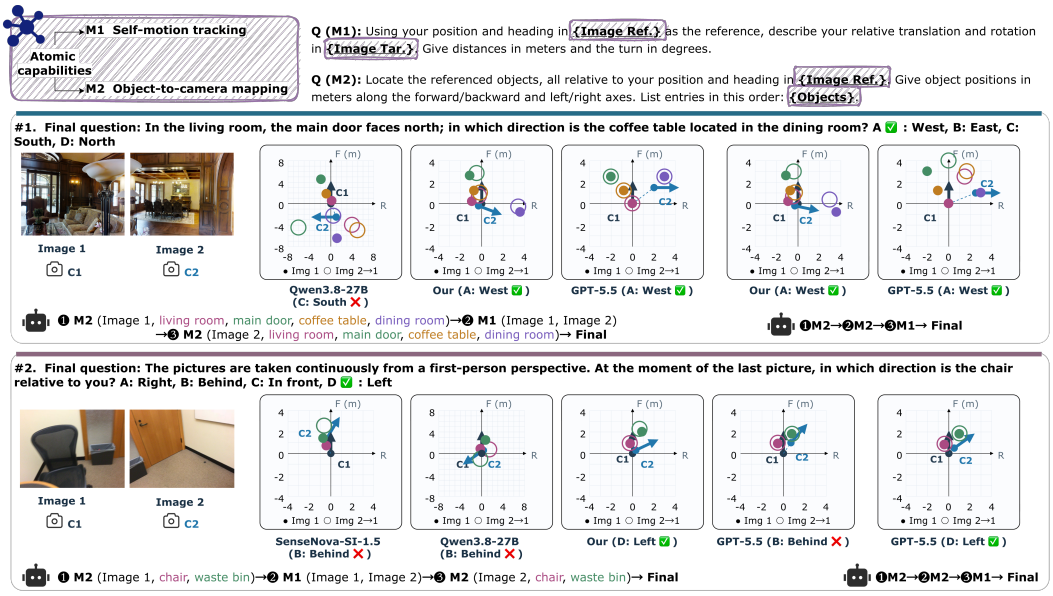


Figure 1: Visualization of two MMSI-Bench (Yang et al., 2026c) examples by querying M1 and M2 before asking the final question. The query order is shown below each plot. The coordinate frame is defined at C_1. The black and blue arrows mark the viewing directions at C_1 and C_2, respectively, while the blue dashed line denotes the displacement from C_1 to C_2 predicted by M1. For M2, filled markers show object locations predicted from Image 1, while hollow markers show Image 2 predictions transformed into the C_1 frame using the predicted self-motion. The overlap between filled and hollow markers reflects the cross-view consistency of object-location predictions.

What intermediate supervision can capture how spatial information evolves across observations and generalize to unseen spatial tasks?

We answer this question from a basic geometric property of *visual experience*: consecutive spatial observations are connected by the self-motion of the observer (Wang & Spelke, 2000; Wolbers et al., 2008). Unlike generic multi-image understanding, which mainly integrates semantic information across images (Jiang et al., 2024), spatial reasoning must also recover the geometric relationship between observations. Our key insight is that observer self-motion is more than a prediction target because it determines how spatial information should change from one observation to the next.

This suggests that motion estimation should be explicitly coupled with spatial state updates during learning. The model must first recover how the observer moves, use self-motion to update spatial relations established from previous observations, and then use the resulting state to answer the required query. Current spatial post-training often supervises these capabilities through separate task-specific questions, leaving the transition from observer motion to spatial state update largely implicit.

Learning this transition requires visual observations to be aligned with observer motion and the resulting changes in spatial relations. We therefore introduce URUQI, a scalable spatial compiler for synthesizing continuous visual experiences and compiling them into dense spatial supervision. URUQI generates motif-driven camera trajectories across diverse 3D scenes, where each motif controls how observer motion, visibility, and spatial evidence unfold over time. Using the underlying camera poses, scene geometry, and object identities, each trajectory is then compiled into multi-turn supervision over three linked capabilities: **self-motion tracking**, which captures changes in viewpoint; **persistent object mapping**, which tracks object locations as the observer moves; and **operations over state**, which supports downstream reasoning over the acquired spatial information. Questions derived from the same trajectory are further organized into shared episodes, so that motion, state updates, and spatial operations are learned as connected parts of the same experience rather than as isolated QA examples.

To the best of our knowledge, URUQI is the first to jointly supervise self-motion, persistent object mapping, and downstream spatial reasoning within continuous visual experiences. Scaling this

compilation across diverse trajectories, we construct URUQI-600k, a large-scale dataset with dense spatial supervision. URUQI$_{\text{Syn}}$-8B trained on this corpus achieves substantial gains in episodic spatial reasoning on URUQI benchmark in Section 4.2 and transfers effectively to external spatial benchmark without using any training data from these target benchmarks (Section 4.3). Moreover, combining our supervision with only 100K Sensenova-SI-8M examples further improves SenseNova-SI-1.5 (Cai et al., 2026), showing that our supervision complements existing spatial post-training. Beyond benchmark-level gains, dense trajectory evaluation in Section 4.4 further shows that URUQI$_{\text{Syn}}$-8B improves self-motion estimation and maintenance of previously observed object locations as the observer moves.

2 Related Work

Spatial Intelligence in Vision-Language Models. Recent work has improved spatial reasoning in VLMs by scaling spatially grounded instruction data for metric and relational reasoning (Chen et al., 2024; Cai et al., 2026), or by introducing stronger geometric priors such as depth and reconstructed 3D features (Cheng et al., 2024; Wu et al., 2026; Fan et al., 2026). Another line of work delegates geometric operations to specialized tools, with the VLM serving primarily as a semantic reasoner and planner (Dai et al., 2026; Chen et al., 2026a). While these approaches demonstrate the value of richer supervision, geometric representations, and external computation, the resulting progress motivates a fundamental question: can spatial intelligence be learned as a native capability of VLMs, enabling them to build and update spatial understanding from visual experience? We study this question by using privileged geometry to supervise how spatial information evolves during training.

Intermediate Supervision for Spatial Reasoning Recent work has complemented final-answer supervision by explicitly supervising intermediate spatial reasoning processes. Spatial CoT supervises linguistic reasoning traces (Li et al., 2026b). Cognitive-map methods construct explicit spatial scaffolds for subsequent reasoning or verification (Wang et al., 2025; Deng et al., 2026). Representation-based approaches instead supervise intermediate visual or geometric states, such as canonical views or latent 3D representations (Zhan et al., 2026; Chen et al., 2026b). These approaches demonstrate the benefit of exposing intermediate structure during spatial reasoning. However, their intermediate targets are typically defined by a particular reasoning format or spatial representation. URUQI instead derives supervision directly from scene geometry, capturing observer motion and spatial information updates as shared components of spatial reasoning.

Spatial Reasoning from Egocentric Visual Experience Recent work has extended spatial reasoning from images to egocentric videos, studying how models integrate spatial evidence across changing viewpoints and long temporal contexts (Yang et al., 2025; Ravi et al., 2025; Yang et al., 2026b). A growing line of work further maintains persistent spatial information through object-centric memories, learned spatial memories, or explicit 3D scene representations (Fan et al., 2025; Liu et al., 2026). From a complementary cognitive perspective, self-motion supports continuous spatial updating without requiring complete scene reconstruction (Wang & Spelke, 2000; Wolbers et al., 2008). Motivated by human spatial cognition, URUQI instead learns to maintain spatial understanding across observer motion, supporting persistent object mapping and subsequent spatial reasoning without complete scene reconstruction.

3 Methodology

In this section, we first formulate spatial supervision over visual experience in Section 3.1. We then introduce motif-driven trajectory rendering for generating visual experiences in Section 3.2. Finally, Section 3.3 describes how privileged scene geometry is used to derive spatial supervision from visual experiences and organize resulting targets into training episodes with shared context.

3.1 Problem formulation

We consider a static physical world $\mathcal{W}$ observed along a moving camera trajectory $\tau = (T_1, \ldots, T_L)$, where $T_t \in \mathrm{SE}(3)$ denotes the camera pose at step t. The trajectory yields an RGB sequence $I_{1:L}$, which constitutes the observer's visual experience. The VLM observes only $I_{1:L}$,

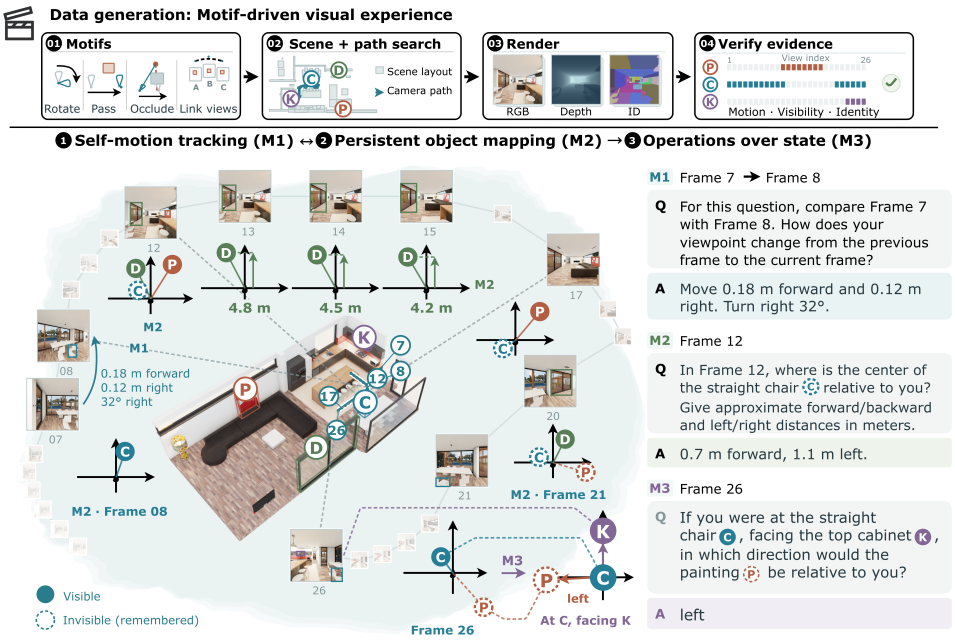


Figure 2: **Overview of URUQI. Top:** Visual experiences are generated by instantiating reusable motifs in 3D scenes, searching for feasible camera trajectories, and rendering RGB observations together with privileged depth and instance identities. **Bottom:** Each verified trajectory is compiled into supervision over three stages: self-motion tracking (M1), persistent object mapping (M2), and operations over spatial state (M3). The example illustrates spatial information evolving along a shared trajectory, with solid and dashed markers denoting visible and previously observed objects.

while the world geometry $\mathcal{W}$ and camera poses $T_{1:L}$ are privileged information used only to construct supervision.

For two consecutive observations, the relative observer motion is given by:

$$\Delta T_t = T_t^{-1} T_{t+1} \in \mathrm{SE}(3). \tag{1}$$

This motion changes the observer's reference frame and therefore constrains how previously acquired spatial information should be interpreted at the next viewpoint. Let S_t denote the supervision-level spatial state supported by observations up to step t. Upon receiving I_{t+1}, the state must be updated consistently with both the relative motion and the new visual evidence:

$$S_t \xrightarrow{\Delta T_t, I_{t+1}} S_{t+1}. \tag{2}$$

Here, S_t is an abstraction for supervision and does not assume that the VLM maintains an explicit map or a particular internal state representation. A spatial query q issued at step t_q specifies how the available spatial information should be used:

$$y_q = g_q(S_{t_q}), \tag{3}$$

where g_q denotes the query-dependent spatial operation and y_q is its target answer. This factorization exposes three supervision targets: observer motion ΔT_t, spatial-state transitions $S_t \to S_{t+1}$, and query-dependent operations over the resulting state.

3.2 Motif-driven Visual Experience Generation

The same physical world can present distinct visual experiences depending on how it is observed. Changes in visual experience reveal different geometric evidence and place distinct demands on spatial reasoning. For example, rotating in place changes the observer's orientation while keeping its position fixed, isolating changes in the reference frame. Exploring and revisiting a scene requires spatial information to persist across viewpoints and temporary loss of visibility. We aim to design a broad set of visual experiences to evaluate spatial reasoning under different conditions.

As shown in the top of Figure 2, we control the generation of visual experience through *spatial motifs*: reusable patterns of observer motion and visibility that determine how spatial evidence unfolds along a trajectory. Given a 3D scene, we instantiate a motif by selecting the relevant objects and searching for a feasible trajectory under the scene geometry. Each motif then defines how the observer should move and how spatial information should be revealed, such as rotating at a fixed location, moving until a previously seen object becomes occluded, or walking through landmarks.

The motif is instantiated through constrained trajectory search over the scene geometry, jointly determining camera positions, motion paths, and viewing directions. Detailed motif definitions and the corresponding search procedures are provided in Appendix B.1. Each candidate trajectory is then executed in OmniGibson (Li et al., 2022) to produce the RGB sequence together with privileged geometric signals, including camera poses, depth, and instance identities. We then verify that these observations and signals satisfy the motion, visibility, and object-identity requirements of the motif.

Using this procedure, we construct 11,738 visual trajectories, covering a broad range of ways in which spatial information can emerge, disappear, and reconnect over time. Detailed trajectory statistics are provided in Appendix B.3.

3.3 Compiling Spatial Supervision into Episodes

As shown in the bottom of Figure 2, for each visual trajectory, we use its privileged camera poses, object identities, and visibility annotations to instantiate the spatial supervision defined in Section 3.1. We first derive supervision targets from geometric quantities and then express them as natural-language QA pairs. These targets are subsequently placed along the trajectory according to the visual evidence available at each point to form a temporally grounded training episode.

Geometric supervision. The trajectory geometry provides three complementary forms of supervision. For **self-motion tracking** (M1), relative camera poses determine the translation and heading change between viewpoints, including both consecutive and longer-range motion along the trajectory. For **persistent object mapping** (M2), object identities associate entities across observations, while object geometry and camera poses determine their locations and relations in the reference frame required by each query. These targets cover both currently visible objects and previously observed objects that have moved outside the current field of view. For **operations over state** (M3), the query-dependent operator g_q acts on the spatial information represented by S_t to produce targets such as reference-frame transformations, hypothetical motion updates, metric or directional comparisons, and compositions of information acquired across observations.

Temporal grounding. We align each supervision target with the point in the trajectory at which its supporting visual evidence becomes available. At step t, a target may involve only entities and spatial information established by observations $I_{1:t}$. Once an entity has been observed, its spatial information remains available for subsequent targets as the observer moves, including after the entity becomes occluded or leaves the current view. Entities not yet encountered are introduced only after their first supporting observation. Accordingly, self-motion targets follow the observations defining the corresponding relative pose, object-mapping targets follow the observations establishing the relevant entities and spatial information, and operations targets are introduced once their required spatial information and reference frame are available.

Episode construction. Let $t_1 < \cdots < t_K$ denote the points at which the K supervision queries are inserted. We serialize the trajectory and its associated question–answer pairs into a single episode:

$$\mathcal{E} = \big(I_{1:t_1}, q_1, y_1;\ I_{t_1+1:t_2}, q_2, y_2;\ \ldots;\ I_{t_{K-1}+1:t_K}, q_K, y_K\big).$$

Observations retain their temporal order, while the context accumulates as the episode progresses. Thus, when answering q_k, the model has access to preceding visual observations and earlier QA exchanges. M1, M2, and M3 targets are interleaved along the visual trajectory, placing different forms of spatial supervision within the same evolving context.

Training objective. We train URUQI models using autoregressive next-token prediction on answer tokens. Let i index the N QA occurrences across all training episodes. For QA i, the context H_i contains the available images, the current question, and any preceding QA exchanges in the same episode. Its target answer is the token sequence $y_i = (y_{i,1}, \ldots, y_{i,n_i})$, where n_i is the number of

Table 1: **Evaluation on URUQI benchmark.** Each column reports task-specific accuracy (%). Subclasses are grouped by the three supervision stages. Overall is computed as the mean of the three stage scores, with subclasses weighted equally within each stage. Bold and underline denote the best and second-best scores, respectively.

Model	M1: self-motion			M2: object mapping				M3: state operations				**Overall**
	Mot.	Loc.	Yaw	ID	Pos.	Hist.	Inv.	Ref.	Upd.	Rel.	Evt.	↑
Proprietary models												
Gemini 3.1 Pro	13.64	17.96	43.58	29.41	11.84	25.15	48.65	25.40	34.82	27.37	25.56	27.37
Qwen3.8-Max	9.77	16.45	39.57	<u>32.10</u>	12.85	24.08	47.27	14.75	20.01	24.58	19.60	23.58
GPT-5.5	11.86	21.16	44.53	31.22	11.36	20.18	44.20	22.43	19.04	21.30	20.90	24.50
GPT-6 Astra	25.31	41.30	**89.70**	**52.53**	<u>29.72</u>	**54.91**	**61.38**	47.13	**63.25**	<u>45.89</u>	37.68	<u>50.08</u>
General-purpose open-source models												
InternVL3-8B	5.47	22.40	29.18	8.08	9.71	7.03	26.66	12.61	11.18	21.10	17.62	15.84
Qwen3-VL-8B	7.49	17.60	30.17	13.02	11.01	10.92	35.89	16.17	11.55	23.75	27.32	18.61
Qwen3.8-27B	7.80	13.48	33.75	19.83	9.37	15.01	37.76	9.15	14.90	16.53	7.30	16.94
Spatially specialized open-source models												
SenseNova-SI-1.5	2.09	12.99	35.34	6.81	10.82	10.87	24.14	22.92	11.25	24.71	25.47	17.02
Spatial-MLLM	0.00	1.50	6.85	0.09	2.93	1.51	3.93	5.99	4.94	11.13	18.25	4.99
SpaceR-7B	0.18	6.95	30.47	3.36	10.23	6.56	21.38	12.67	10.26	17.86	31.18	13.64
VST-7B-SFT	0.00	5.19	28.02	2.46	9.79	10.14	13.40	16.35	10.48	17.86	31.14	12.99
MindCube-RL	0.00	4.93	29.44	2.00	8.17	5.17	10.80	10.75	10.19	9.47	22.03	10.37
URUQI family												
URUQI$_{\text{Syn}}$-8B	**33.48**	<u>54.91</u>	71.08	24.07	28.03	37.68	52.82	59.49	44.08	41.07	72.93	47.73
URUQI-Mix-8B	<u>33.05</u>	54.88	69.27	25.49	27.85	39.05	<u>53.19</u>	<u>60.19</u>	43.56	44.60	<u>76.63</u>	48.35
URUQI-SI-Mix-8B	30.10	**56.45**	<u>71.68</u>	25.84	**30.27**	<u>42.13</u>	51.38	**66.31**	<u>47.25</u>	**46.47**	**84.31**	**50.41**

answer tokens. The training objective is:

$$\mathcal{L}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\frac{1}{n_i}\sum_{j=1}^{n_i}\log p_\theta\big(y_{i,j} \mid H_i, y_{i,<j}\big), \tag{4}$$

where $y_{i,<j}$ denotes the target answer tokens preceding position j.

4 EXPERIMENTS

In this section, we conduct experiments to address the following research questions:

- **RQ1**: Does post-training on continuous visual experience improve the underlying spatial capabilities targeted by our supervision?
- **RQ2**: Do these gains transfer to external spatial benchmarks and real-world visual observations?
- **RQ3**: Can VLMs maintain egocentric spatial understanding as the observer moves?

4.1 EXPERIMENTAL SETUP

Training data and evaluation data. The URUQI training corpus contains 611,348 QA instances in 455,863 episodes: 211,552 for self-motion tracking, 136,398 for persistent object mapping, and 263,398 for spatial-state operations. The benchmark contains 52,920 QA instances in 2,692 episodes from eight held-out scenes (Appendix B.3).

Training setting. We post-train all models in the URUQI family for one epoch on 16 NVIDIA H800 GPUs using AdamW with a learning rate of 10^{-5} and an effective batch size of 192 training episodes. We consider three URUQI-8B configurations.

- **URUQI$_{\text{Syn}}$-8B**: initialized from InternVL3-8B (Zhu et al., 2025) and post-trained only on our synthesized visual-experience supervision.

Table 2: **External spatial evaluation.** Num and MC denote numerical and multiple-choice tasks in VSI-Bench; Pos and MSR denote positional and multi-step spatial reasoning in MMSI-Bench; Rot, Amg, and Ard denote rotation, among, and around in MindCube-tiny. Avg reports each benchmark's overall score. Bold and underline indicate the best and second-best scores among open-source models, respectively. Superscripts indicate externally reported results: [a]SpatialAxiom (Lou et al., 2026); [b]Wang (Wang, 2026); [c]PhysBrain 1.5 (Team et al., 2026) .

	VSI-Bench			MMSI-Bench			MindCube-tiny			
Model	Num	MC	**Avg**	Pos.	MSR	**Avg**	Rot	Amg	Ard	**Avg**
Proprietary models (reference)										
Gemini 3.1 Pro	38.48	61.36	49.92	–	–	49.50[a]	90.50	71.33	83.20	77.81
Qwen3.7-Plus	61.56	68.51	65.04	–	–	45.00[a]	92.50	59.83	80.40	70.95
GPT-5.5	–	–	60.40[a]	–	–	42.20[a]	–	–	–	65.50[a]
GPT-6 Astra	64.57	81.53	73.05[b]	–	–	57.90[c]	–	–	–	78.80[c]
General-purpose open-source models										
InternVL3-8B	48.02	36.26	42.14	30.84	21.72	27.80	33.00	34.33	51.60	38.19
Qwen3-VL-8B	63.19	53.32	58.25	30.84	28.28	29.40	29.50	28.33	32.00	29.43
Qwen3.8-27B	48.40	53.10	50.75	40.80	**37.88**	**42.70**	**92.50**	67.33	83.60	76.00
Spatially specialized open-source models										
SenseNova-SI-1.5	65.85	68.57	67.21	45.40	28.28	39.10	90.50	**93.83**	88.80	92.00
Spatial-MLLM	50.96	41.49	46.33	27.16	29.80	26.10	39.00	30.51	36.00	33.46
SpaceR-7B	49.85	41.55	45.60	35.29	23.23	27.80	34.50	31.02	33.60	32.31
VST-7B-SFT	60.72	50.28	55.50	35.35	18.18	32.50	37.00	35.93	50.80	39.71
VST-7B-RL	61.21	51.10	56.15	36.13	19.19	32.50	37.00	37.80	45.60	39.52
MindCube-3B-SFT	15.87	18.62	17.24	2.15	2.02	1.70	34.00	51.02	67.60	51.73
MindCube-RL	25.85	37.09	31.47	27.39	27.78	27.70	30.50	51.17	61.60	49.71
Cambrian-S-7B	63.52	62.34	62.92	29.56	24.24	27.10	33.00	38.98	39.20	37.88
URUQI family										
URUQI$_{\text{Syn}}$-8B	39.28	55.23	47.26	34.67	22.22	33.10	44.00	50.33	36.80	45.90
URUQI-Mix-8B	55.38	59.73	57.56	45.02	21.72	39.80	49.00	57.83	57.20	56.00
URUQI-SI-Mix-8B	**66.39**	**69.12**	**67.76**	**50.19**	28.79	41.80	90.00	**93.83**	**91.60**	**92.57**

- **URUQI-Mix-8B**: initialized from InternVL3-8B and trained on a mixture of our visual-experience supervision and 100k examples randomly selected from SenseNova-SI-8M (Cai et al., 2026).
- **URUQI-SI-Mix-8B**: initialized from SenseNova-SI-1.5-InternVL3-8B (Cai et al., 2026) and further trained with the same mixed-data recipe as URUQI-MIX-8B.

Evaluation setting We evaluate spatial capabilities on URUQI benchmark in a multi-turn setting in Table 1. Observations and questions are presented in temporal order, and the model retains preceding observations and its own answers within each episode. We further evaluate models on VSI-Bench (Yang et al., 2025), MMSI-Bench (Yang et al., 2026c), and MindCube-tiny (Wang et al., 2025) using their respective evaluation protocols in Table 2. We compare URUQI model family against proprietary VLMs including Gemini 3.1 Pro (Google DeepMind, 2026), Qwen3.7-Plus (Qwen Team, 2026a), Qwen3.8-Max (Qwen Team, 2026c), GPT-5.5 (OpenAI, 2026a), and GPT-6 Astra (OpenAI, 2026b), general-purpose open-source VLMs including InternVL3-8B (Zhu et al., 2025), Qwen3-VL-8B (Bai et al., 2025), and Qwen3.8-27B (Qwen Team, 2026b); and spatially specialized models including SenseNova-SI-1.5 (Cai et al., 2026), Spatial-MLLM (Wu et al., 2026), SpaceR-7B (Ouyang et al., 2025), VST-7B-SFT and VST-7B-RL (Yang et al., 2026a), MindCube-3B-SFT and MindCube-RL (Wang et al., 2025), and Cambrian-S-7B (Yang et al., 2026b).

4.2 Evaluating Spatial Capabilities in Continuous Visual Experience (RQ1)

Metrics. Table 1 reports three groups of task-specific accuracies. M1 measures inter-view translation and rotation (Mot.), relative observer localization (Loc.), and heading changes (Yaw). M2

Table 3: **Dense evaluation of self-motion estimation and egocentric object localization.** Acc@0.5m measures the percentage of object predictions within 0.5 m of the ground-truth horizontal position. Bold and underline denote the best and second-best results per column, respectively.

Model	QA history	Self-motion		Object mapping				
		Translation error (m) ↓	Heading error (°) ↓	Acc@0.5m (%) ↑				Mean error (m) ↓
				All	Initial	Visible	Absent	
InternVL3-8B	None	0.397	93.57	1.00	0.00	2.42	0.00	3.004
	Model	0.341	16.57	0.67	0.00	1.45	0.13	3.040
	GT motion	0.189	12.72	0.78	0.00	1.82	0.09	3.052
Qwen3.8-27B	None	0.297	9.43	6.44	16.02	12.30	1.29	4.725
	Model	0.356	7.22	9.41	22.08	17.99	1.88	3.594
	GT motion	0.127	4.84	8.75	23.81	16.41	1.79	3.263
URUQI$_{\text{Syn}}$-8B	None	0.154	2.36	**49.56**	**75.32**	**66.14**	<u>35.13</u>	**0.743**
	Model	<u>0.090</u>	<u>0.97</u>	39.62	<u>74.89</u>	47.24	30.94	1.027
	GT motion	**0.031**	**0.56**	<u>47.04</u>	73.16	<u>55.97</u>	**38.30**	<u>0.832</u>

measures cross-view correspondence and referent resolution (ID), object positions, distances, and directions (Pos.), historical spatial states and visibility (Hist.), and distinct-object counts across views (Inv.). M3 measures reference-frame transformations (Ref.), hypothetical and composed motion updates (Upd.), metric relations and comparisons (Rel.), and counts of observed turns (Evt.). Overall averages the three stage scores, each computed as the mean of its subclasses.

Dense experience supervision improves the targeted capabilities. Table 1 evaluates self-motion estimation (M1), object mapping (M2), and operations over spatial state (M3). Post-training InternVL3-8B solely on our synthetic supervision raises its Overall score from 15.84% to 47.73%, with improvements in all 11 subclasses.

Supervision from experience complements existing spatial post-training. URUQI-Mix-8B reaches 48.35% Overall, compared with 47.73% for URUQI$_{\text{Syn}}$-8B. Starting from SenseNova-SI-1.5, URUQI-SI-Mix-8B increases Overall from 17.02% to 50.41%, improving all 11 subclasses. It also obtains the highest Overall score among the evaluated models, closely followed by GPT-6 Astra at 50.08%. GPT-6 Astra retains a higher M2 average (49.64% vs 37.41%).

4.3 Evaluating Transfer to External Spatial Benchmarks (RQ2)

Benchmarks and metrics. We evaluate transfer on three external spatial benchmarks: VSI-Bench (Yang et al., 2025), MMSI-Bench (Yang et al., 2026c), and MindCube-tiny (Wang et al., 2025). Table 2 reports their overall scores together with task-group results.

Synthetic supervision transfers to external spatial tasks. Compared with InternVL3-8B, URUQI$_{\text{Syn}}$-8B improves the overall scores from 42.14% to 47.26% on VSI-Bench, from 27.80% to 33.10% on MMSI-Bench, and from 38.19% to 45.90% on MindCube-tiny. These gains show that the benefits of our synthetic supervision extend beyond the tasks in URUQI benchmark.

Combining synthetic and existing spatial data improves transfer. With the addition of 100K examples from SenseNova-SI-8M (1.25% of the corpus), URUQI-Mix-8B reaches 57.56%, 39.80%, and 56.00% on VSI-Bench, MMSI-Bench, and MindCube-tiny, respectively, improving over the synthetic-only variant on all three benchmarks. VSI-Bench Num also increases to 55.38%, exceeding the InternVL3-8B baseline. Starting from SenseNova-SI-1.5, URUQI-SI-Mix-8B achieves the highest overall scores on VSI-Bench and MindCube-tiny among the open-source models in Table 2, while ranking second on MMSI-Bench, behind only Qwen3.8-27B.

4.4 Dense Evaluation of Egocentric Spatial Tracking (RQ3)

We evaluate self-motion estimation and egocentric object localization on 100 trajectories in URUQI benchmark, comprising 2,312 frames and 231 registered objects.

Table 4: **Localization from visual estimates and camera poses.** Acc@0.5m (%) is averaged over 2,240 absent-object queries. (a) Direct prediction uses each current VLM answer; pose-based propagation stores the first accepted object estimate and computes subsequent locations from camera poses. Both use No QA history, with GT poses supplied only to the geometry module. (b) Pose-based propagation uses Model-history object predictions; Predicted denotes poses integrated from the model's motion estimates. Bold marks the best result within each model and panel. Complete metrics are in Table 7.

(a) Localization method

No QA history; GT poses

Model	Method	Absent Acc@0.5m ↑
Qwen3.8-27B	Direct prediction	1.29
	Pose-based propagation	**16.88**
URUQI$_{\text{Syn}}$-8B	Direct prediction	35.13
	Pose-based propagation	**71.61**

(b) Camera-pose source

Model history; pose-based propagation

Model	Pose	Absent Acc@0.5m ↑
Qwen3.8-27B	GT	**18.48**
	Predicted	3.97
URUQI$_{\text{Syn}}$-8B	GT	**72.86**
	Predicted	34.02

To examine whether previous answers help subsequent localization, we compare three history conditions, all of which provide the full observed image history. *No QA history* excludes previous questions and answers; *Model history* includes the model's earlier motion and location answers; *GT motion history* replaces earlier motion answers with ground truth.

Table 3 reports motion errors and localization metrics, including Acc@0.5m, the percentage of predictions within 0.5 m of the true horizontal position. Construction rules, visibility stages, and scoring details are in Appendix D.

Spatial tracking from visual history. Under No QA history, URUQI$_{\text{Syn}}$-8B improves both motion estimation and object localization over InternVL3-8B and Qwen3.8-27B. Its overall localization accuracy reaches 49.56%, compared with 1.00% and 6.44%, respectively. For absent objects, it achieves 35.13%, versus 0.00% and 1.29%, showing improved localization out of view.

Effect of answer history. For URUQI$_{\text{Syn}}$-8B, Model history improves motion estimation but reduces overall object-localization accuracy from 49.56% to 39.62%. GT motion history partially recovers overall object-localization accuracy and yields the highest absent-object accuracy among the three conditions (38.30%). The improvement over Model history suggests that accurate motion history provides useful context for absent-object localization.

Pose-assisted localization. We test whether an external geometry module can use camera poses to propagate each object's first accepted location estimate. Under *No QA history*, supplying GT poses only to this module improves URUQI$_{\text{Syn}}$-8B's absent-object accuracy from 35.13% with direct prediction to 71.61% with propagation (Table 4a).

In a separate comparison, we fix the *Model history* object predictions and propagation method, replacing GT poses with poses obtained by integrating model-predicted self-motion. Accuracy decreases from 72.86% to 34.02%, compared with 3.97% for Qwen3.8-27B using predicted poses (Table 4b). Thus, the initial object estimates support pose-assisted localization, but its accuracy remains sensitive to errors in the estimated camera trajectory.

5 Conclusion

We study how spatial cognition in VLMs can be learned from continuous visual experience. URUQI couples consecutive observations through observer self-motion and jointly supervises motion tracking, persistent object mapping, and spatial operations. Our 8B model achieves the best overall performance among evaluated open-source models, reaching 50.41% on URUQI benchmark and leading open-source results on two of three external benchmarks. It matches GPT-6 Astra on URUQI benchmark and surpasses several proprietary models on external benchmarks, highlighting continuous visual experience as an effective, scalable supervision paradigm for spatially capable VLMs.

AI USE STATEMENT

Generative AI tools were used to assist with implementing the trajectory generation and spatial-supervision compilation pipeline, as well as language polishing, manuscript organization and the presentation of experimental results. All codes, references, experimental results, and content were reviewed and verified by the authors, who take responsibility for the final text, methods, and claims.

ETHICS STATEMENT

Our spatial supervision is generated from simulated scenes and does not involve human participants or personally identifiable information. All experiments are conducted in simulated or benchmark environments, and any released data, models, and derived assets will follow applicable licensing and usage requirements. The resulting models are developed for research purposes and are not evaluated for safety-critical deployment.

REPRODUCIBILITY STATEMENT

We provide details of data construction, model training, evaluation protocols, and implementation settings in the main text and appendix. The appendix further documents the procedures used for trajectory generation, spatial supervision, and evaluation. Where permitted, we will release the configurations, checkpoints, and evaluation pipeline.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

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.

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*, pp. 14455–14465, 2024.

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, 2026a.

Zhangquan Chen, Manyuan Zhang, Xinlei Yu, Xufang Luo, Mingze Sun, Zihao Pan, Xiang An, Yan Feng, Peng Pei, Xunliang Cai, et al. Think with 3d: Geometric imagination grounded spatial reasoning from limited views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2613–2624, 2026b.

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. *Advances in Neural Information Processing Systems*, 37:135062–135093, 2024.

Yalun Dai, Hao Li, Shulin Tian, Runmao Yao, Yuhao Dong, Fangzhou Hong, Zhaoxi Chen, Fangfu Liu, Baoliang Tian, Dingwen Zhang, et al. S-agent: Spatial tool-use elicits reasoning for spatial intelligence. *arXiv preprint arXiv:2606.20515*, 2026.

Wei Deng, Xianlin Zhang, and Mengshi Qi. Active exploring like a pigeon: Reinforcing spatial reasoning via agentic vision-language models. *arXiv preprint arXiv:2606.02459*, 2026.

Yue Fan, Xiaojian Ma, Rongpeng Su, Jun Guo, Rujie Wu, Xi Chen, and Qing Li. Embodied videoagent: Persistent memory from egocentric videos and embodied sensors enables dynamic scene understanding. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6342–6352. IEEE, 2025.

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Peihao Wang, Huaizhi Qu, Shijie Zhou, et al. Vlm-3r: Vision-language models augmented with instruction-aligned 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 31054–31065, 2026.

Google DeepMind. Gemini 3.1 pro. Model Card, February 2026. URL `https://deepmind.google/models/model-cards/gemini-3-1-pro/`. Published February 19, 2026.

Jiacheng Hua, Yishu Yin, Yuhang Wu, Tai Wang, Yifei Huang, and Miao Liu. Unleashing spatial reasoning in multimodal large language models via textual representation guided reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13616–13637, 2026.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. Mantis: Interleaved multi-image instruction tuning. *arXiv preprint arXiv:2405.01483*, 2024.

Sai Srinivas Kancheti, Aditya Sanjiv Kanade, Vineeth N Balasubramanian, and Tanuja Ganu. Chain-of-thought degrades visual spatial reasoning capabilities of multimodal llms. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 862–876, 2026.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, Mona Anvari, Minjune Hwang, Manasi Sharma, Arman Aydin, Dhruva Bansal, Samuel Hunter, Kyu-Young Kim, Alan Lou, Caleb R Matthews, Ivan Villa-Renteria, Jerry Huayang Tang, Claire Tang, Fei Xia, Silvio Savarese, Hyowon Gweon, Karen Liu, Jiajun Wu, and Li Fei-Fei. BEHAVIOR-1k: A benchmark for embodied AI with 1,000 everyday activities and realistic simulation. In *6th Annual Conference on Robot Learning*, 2022. URL `https://openreview.net/forum?id=_8DoIe8G3t`.

Hongxing Li, Dingming Li, Zixuan Wang, Yuchen Yan, Hang Wu, Wenqi Zhang, Yongliang Shen, Weiming Lu, Jun Xiao, and Yueting Zhuang. Spatialladder: Progressive training for spatial reasoning in vision-language models. In *International Conference on Learning Representations*, volume 2026, pp. 76566–76592, 2026a.

Zongzhao Li, Zongyang Ma, Mingze Li, Songyou Li, Yu Rong, Tingyang Xu, Ziqi Zhang, Deli Zhao, and Wenbing Huang. Star-r1: Multi-view spatial transformation reasoning by reinforcing multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12041–12051, 2026b.

Fangfu Liu, Diankun Wu, Jiawei Chi, Yimo Cai, Yi-Hsin Hung, Xumin Yu, Hao Li, Han Hu, Yongming Rao, and Yueqi Duan. Spatial-ttt: Streaming visual-based spatial intelligence with test-time training. In *European Conference on Computer Vision*, pp. 339–357. Springer, 2026.

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

Yujing Lou, Pingyi Chen, Shen Cao, Jiaqi Gu, Jinhui Guo, Jintao Tong, Yunzhuo Hao, Yao Liu, Yue Wu, Lubin Fan, and Jieping Ye. Spatialaxiom: An open spatial intelligence model for general spatial reasoning, July 2026. URL `https://d2i-ai.github.io/SpatialAxiom`.

Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic ai for spatial reasoning with a dynamic api. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19446–19455. IEEE, 2025.

OpenAI. GPT-5.5 System Card. https://openai.com/index/gpt-5-5-system-card/, 2026a.

OpenAI. GPT-6 Astra system card. System Card, September 2026b. URL https://deploymentsafety.openai.com/gpt-6-astra. Published September 3, 2026.

Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. Spacer: Reinforcing mllms in video spatial reasoning. *arXiv preprint arXiv:2504.01805*, 2025.

Qwen Team. Qwen3.7-plus. Model Card, 2026a. URL https://www.qwencloud.com/models/qwen3.7-plus.

Qwen Team. Qwen3.8-27B. Hugging Face Model Card, August 2026b. URL https://huggingface.co/Qwen/Qwen3.8-27B. Released August 14, 2026.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026c. URL https://qwen.ai/blog?id=qwen3.8.

Sahithya Ravi, Gabriel Herbert Sarch, Vibhav Vineet, Andrew D Wilson, and Balasaravanan Thoravi Kumaravel. Out of sight, not out of context? egocentric spatial reasoning in vlms across disjoint frames. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 16146–16161, 2025.

DeepCybo Team, Yu Bin, Haipeng Cao, Zheng Chang, Kai Chen, Youning Chen, Kailin Deng, Yichao Du, Xiaotong Fu, Haoyang Ge, Yunlong Guo, Chenliu Hao, Jiyan He, Xuguo He, Yakun Hou, Kai Hu, Cong Huang, Tuopusen Huang, Yu Huang, Hong Li, Peize Li, Shijie Lian, Xiaopeng Lin, Yun Lin, Haibao Liu, Haochen Liu, Qiuzhi Liu, Shengcai Liu, Zhiqiang Liu, Tao Luo, Peng Ren, Shuo Ren, Chaoyi Ruan, Zhaolong Shen, Yukun Shi, Qiyuan Su, Yuxuan Tian, Yining Wang, Changti Wu, Hao Wu, Xueyin Xu, Ruoqi Yang, Zhaoyang Yang, Hang Yuan, Zhaoyang Zeng, Hanwen Zhang, Ruimeng Zhang, Yao Zhang, Yibo Zhang, Yuxiang Zhang, Zhirui Zhang, Ziyi Zhang, Zubin Zheng, and Zishen Zhuang. Physbrain 1.5: From vision-language models to physical foundation models, 2026. URL https://arxiv.org/abs/2609.14973.

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, 2025. URL https://arxiv.org/abs/2506.21458.

Ranxiao Frances Wang and Elizabeth S Spelke. Updating egocentric representations in human navigation. *Cognition*, 77(3):215–250, 2000.

Yipeng Wang. How well does astra understand real-world space? https://www.yipeng.dev/blog/astra-spatial-intelligence, September 2026. Independent evaluation. Published September 22, 2026. Accessed September 26, 2026.

Thomas Wolbers, Mary Hegarty, Christian Büchel, and Jack M Loomis. Spatial updating: how the brain keeps track of changing object locations during observer motion. *Nature neuroscience*, 11 (10):1223–1230, 2008.

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence. *Advances in neural information processing systems*, 38: 13569–13597, 2026.

Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643. IEEE, 2025.

Rui Yang, Ziyu Zhu, Yanwei Li, Jingjia Huang, Shen Yan, Siyuan Zhou, Zhe Liu, Xiangtai Li, Shuangye Li, Wenqian Wang, et al. Visual spatial tuning. In *European Conference on Computer Vision*, pp. 192–211. Springer, 2026a.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, et al. Cambrian-s: Towards spatial supersensing in video. In *International Conference on Learning Representations*, volume 2026, pp. 78185–78225, 2026b.

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, 2026c.

Shaoxiong Zhan, Yanlin Lai, Zheng Liu, Hai Lin, Shen Li, Xiaodong Cai, Zijian Lin, Wen Huang, and Hai-Tao Zheng. 3viewsense: Spatial and mental perspective reasoning from orthographic views in vision-language models. *arXiv preprint arXiv:2603.07751*, 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.

A Qualitative Examples of Egocentric Spatial Tracking

We compare URUQI$_{\text{Syn}}$-8B, Qwen3.8-27B, and InternVL3-8B on five selected trajectories. The examples illustrate localization during visibility, absence, and reappearance, followed by matched comparisons of *No QA history* and *Model history*. Numbers report horizontal localization errors; dashed circles mark the 0.5 m threshold. GT scene geometry and poses are used only for visualization. Only selected frames are shown and each prediction uses the full image history available at that step.

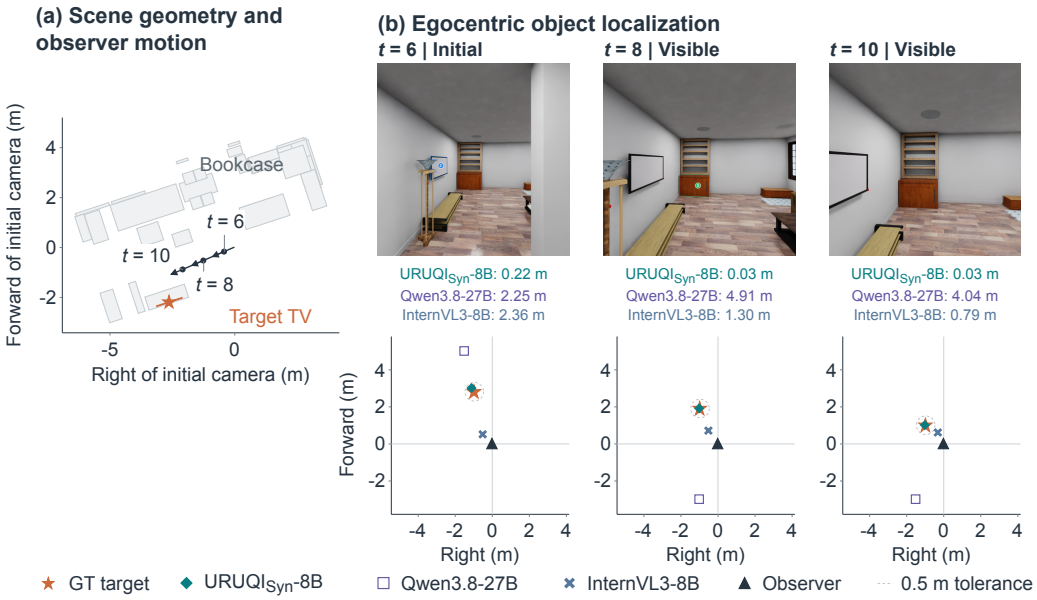


Figure 3: **Localization while the target remains visible.** Under *No QA history*, URUQI$_{\text{Syn}}$-8B localizes the TV across changing viewpoints, with errors of 0.22, 0.03, and 0.03 m.

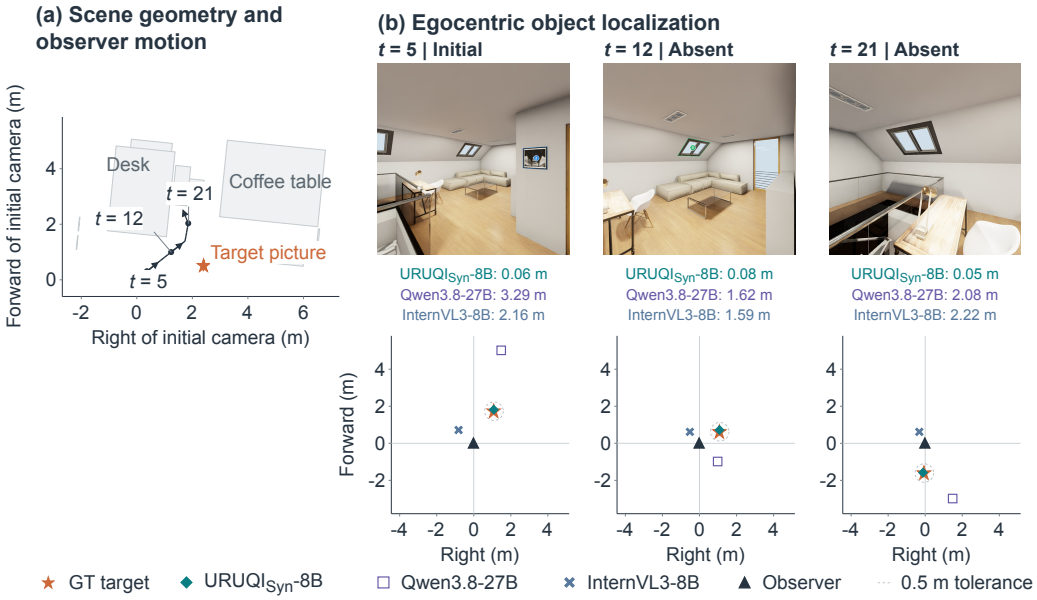


Figure 4: **Localization after the target leaves view.** Under *No QA history*, the picture is absent at the two later observations; URUQI$_{\text{Syn}}$-8B maintains low errors of 0.08 and 0.05 m in this selected trajectory.

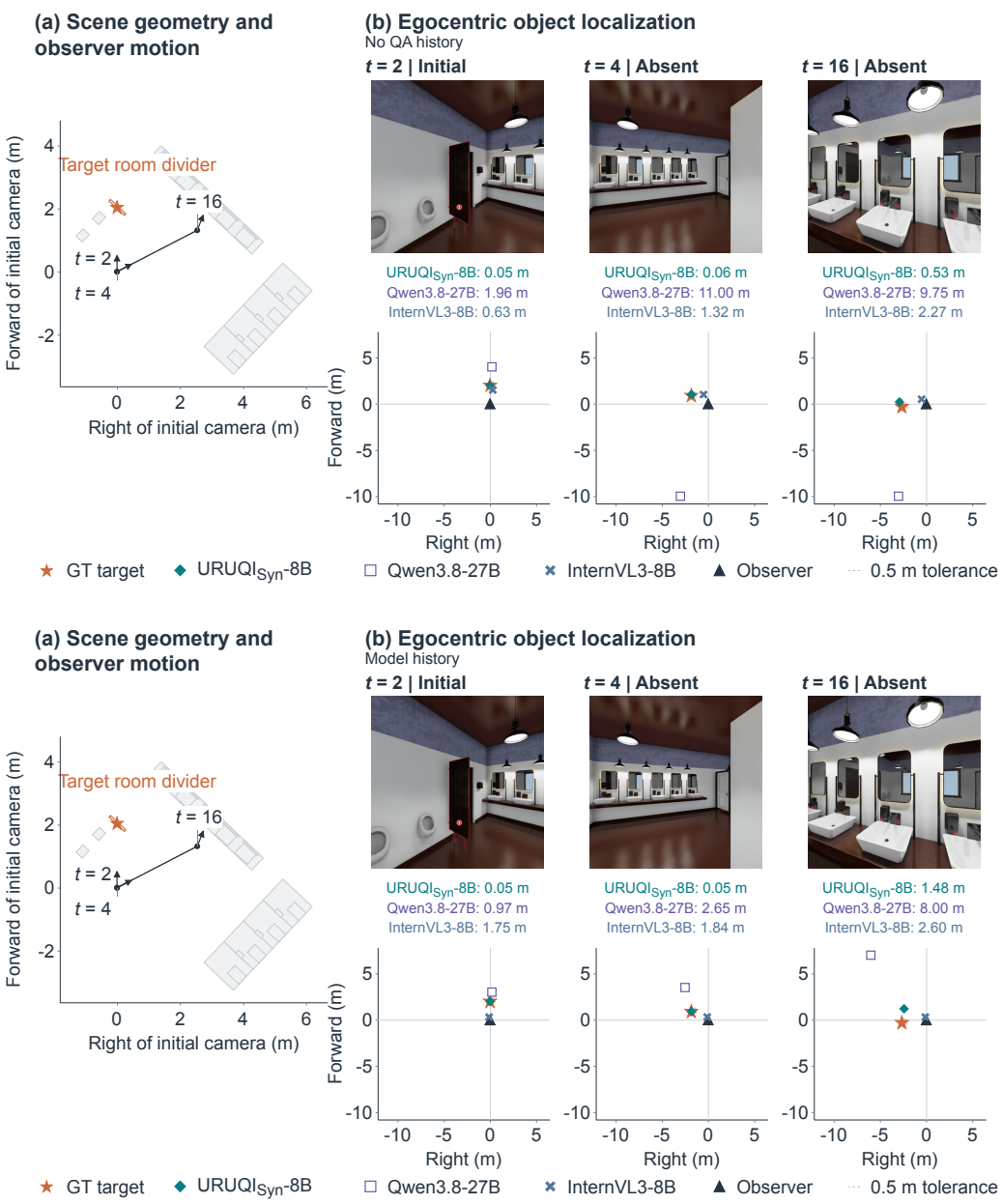


Figure 5: **Localization error during object absence.** The room divider is absent at $t = 4$ and $t = 16$. At $t = 16$, URUQI$_{\text{Syn}}$-8B has an error of 0.53 m under No QA history and 1.48 m under Model history. Top: *No QA history*; bottom: *Model history*. Both conditions use the same object and observation steps, with identical coordinate limits across all six localization plots.

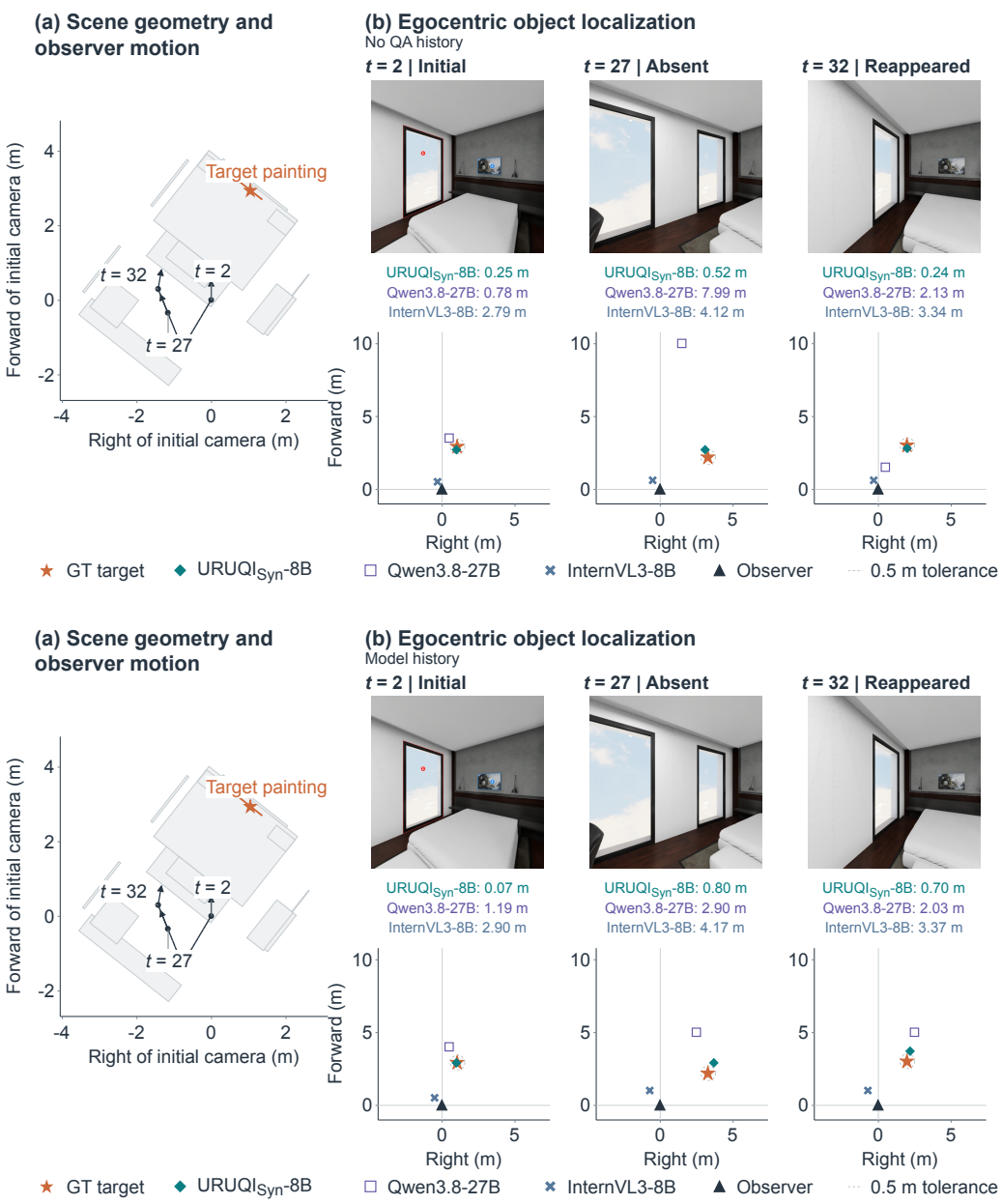


Figure 6: **Localization after object reappearance.** The painting is absent at $t = 27$ and reappears at $t = 32$. For URUQI$_{\text{Syn}}$-8B, error decreases from 0.52 to 0.24 m under No QA history and from 0.80 to 0.70 m under Model history. Top: *No QA history*; bottom: *Model history*. Both conditions use the same object and observation steps, with identical coordinate limits across all six localization plots.

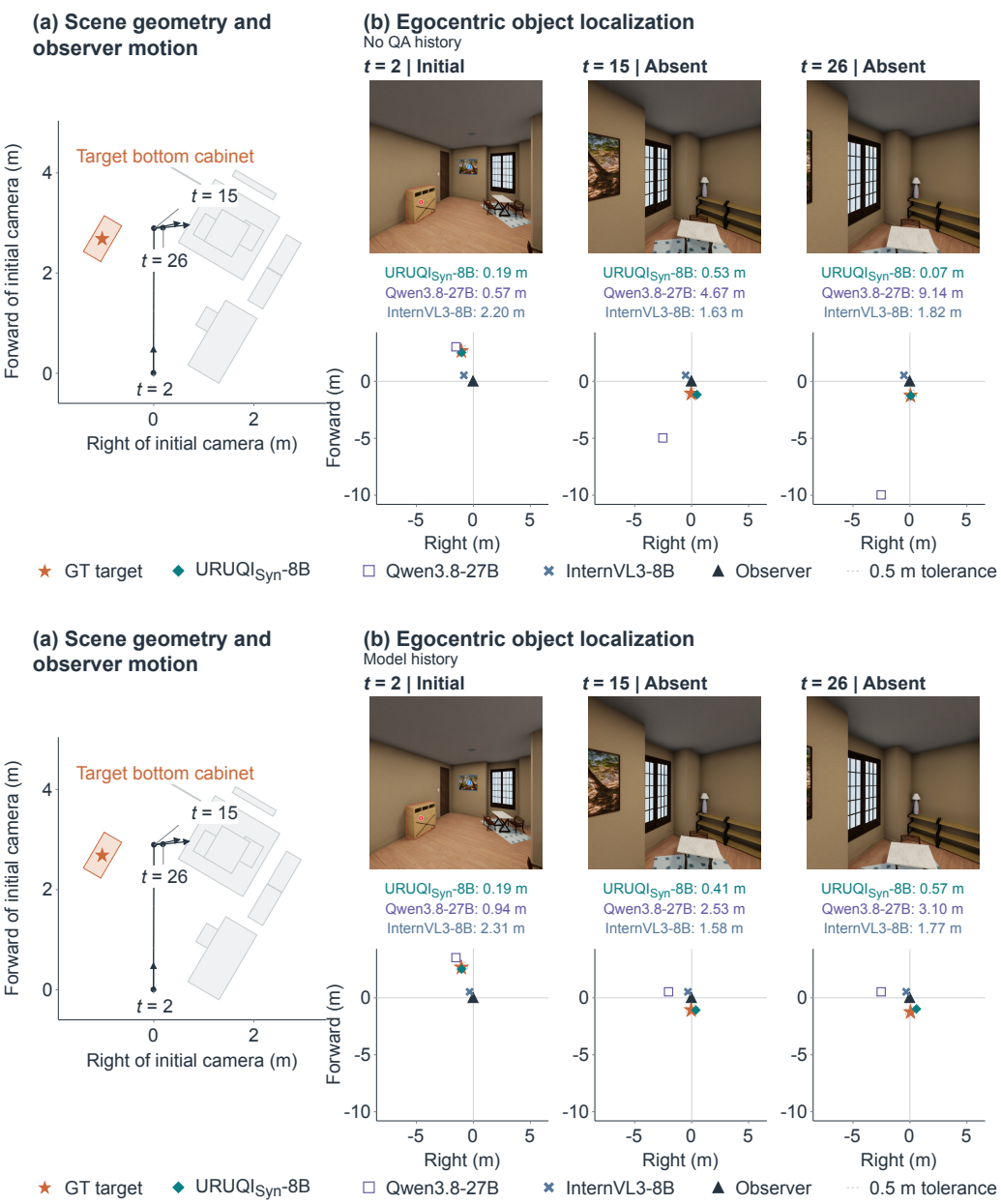


Figure 7: **Variation in the effect of answer history.** For the absent cabinet, URUQI$_{\text{Syn}}$-8B has lower error under Model history at $t = 15$ (0.41 versus 0.53 m), but higher error at $t = 26$ (0.57 versus 0.07 m). Top: *No QA history*; bottom: *Model history*. Both conditions use the same object and observation steps, with identical coordinate limits across all six localization plots.

B SPATIAL MOTIFS AND TRAJECTORY ACQUISITION

B.1 MOTIF INSTANTIATION

Motif definitions. We instantiate each motif by selecting scene objects and searching for a camera trajectory that satisfies its motion and visibility constraints. *(1) Rotate in place* bounds displacement from the initial camera position while changing heading. *(2) Straight pass* translates past the object with a bound on cumulative rotation. *(3) Walk and turn* combines translation and heading changes after an initial observation of the target. *(4) Multi-turn walk* separates the motion into translational and turning segments, with constraints on both the number of turns and accumulated rotation. *(5) Occlusion traversal* moves the observer toward or through an occluding region, with constraints that distinguish physical occlusion from disappearance outside the field of view. *(6) Reference survey* binds an origin object, a facing anchor, and a target, then scans their bearings from a fixed station. Each object must be observed, while no single frame provides a clear view of all three. *(7) Landmark chain* connects two endpoint objects through intermediate anchors. Each adjacent pair must have clear co-visible observations, while the endpoints remain non-co-visible throughout the sequence.

Trajectory search. To search for these trajectories, we project scene geometry onto the ground plane to construct an occupancy map and retain object centers, extents, categories, and instance identities for role binding. Camera stations are sampled in free space. Unobstructed station pairs are connected directly; other pairs use grid-based A* search. Routes are simplified and interpolated into camera poses with bounded inter-frame translation and rotation. Candidate trajectories are screened for camera clearance, path validity, and the constraints of the selected motif.

Acquisition settings. The default trajectory-planning configuration uses a camera height of 1.5 m and a horizontal field of view of 90° for geometric planning. Consecutive planned observations are constrained to at most 1.2 m of translation and 40° of rotation. Camera clearance is checked with a 0.30 m horizontal radius over heights of 0.10–1.70 m. The planned poses are exported as an ordered camera schedule, with one observation associated with each scheduled viewpoint. Sequence length is therefore measured in observation steps.

B.2 TRAJECTORY RENDERING AND VALIDATION

Candidate trajectories are rendered in OmniGibson (Li et al., 2022), recording RGB, camera pose, metric depth, and instance and semantic identity channels at each observation. Geometric visibility estimates guide trajectory search; rendered instance masks establish object support in the resulting images. The evidence checks evaluate the required visibility transitions and co-visibility relations. For occlusion-traversal candidates, depth and instance evidence are used to check whether disappearance is caused by a foreground blocker. Object bindings, poses, and supporting observations are retained for compilation.

B.3 TRAJECTORY STATISTICS

The source pool for URUQI contains 11,738 distinct trajectories across 50 BEHAVIOR scenes (Li et al., 2022), with 212,209 rendered observations in total. Trajectories contain 6–52 observations, with a mean of 18.08 and a median of 15. Each source trajectory yields multiple QA episodes by varying the selected observations, target objects, and queries.

Data splits. We partition the source trajectories into 8,818 training, 2,696 test, and 224 validation trajectories. The 12 test scenes are disjoint from the 38 training scenes. The final benchmark uses eight of the 12 held-out test scenes.

C Spatial Supervision Compilation

C.1 Coordinate Conventions and Motion Targets

Reference frames. Positions are expressed in meters and reported angles in degrees. The world frame is right-handed with $+Z$ upward. The pose T_t maps observer-frame coordinates to world coordinates; therefore, $T_i^{-1}T_j$ maps coordinates from frame j to frame i.

For planar motion targets and horizontal object-location targets, we use forward–left coordinates: forward and left are positive, while backward and right are negative. At zero yaw, the observer faces world $+Y$, with world $+X$ to its right. For yaw ψ_t, the forward and left basis vectors form the matrix

$$B_t = \begin{bmatrix} -\sin\psi_t & -\cos\psi_t \\ \cos\psi_t & -\sin\psi_t \end{bmatrix}, \tag{5}$$

which maps observer-relative horizontal coordinates to world-plane coordinates.

Self-motion targets. Let $o_t \in \mathbb{R}^2$ denote the observer's horizontal position. For observations $i < j$, the displacement and heading targets are

$$d_{ij} = B_i^{\mathsf{T}}(o_j - o_i), \qquad \delta\psi_{ij} = \mathrm{wrap}(\psi_j - \psi_i), \tag{6}$$

where wrap returns the signed principal angle. The two components of d_{ij} give displacement along the forward and left axes of the earlier observation. Positive heading change denotes a left turn. These targets describe the net change between the named viewpoints. Traveled distance and accumulated rotation are computed separately by summing translation magnitudes and absolute heading changes over the intervening observed steps.

Object coordinates under observer motion. For a static object with world bounding-box center c, let c_{xy} denote its horizontal projection. Its horizontal location at step t is

$$z_t = B_t^{\mathsf{T}}(c_{xy} - o_t). \tag{7}$$

The same location expressed at a later viewpoint satisfies

$$z_j = B_j^{\mathsf{T}} B_i(z_i - d_{ij}). \tag{8}$$

This relation expresses the same static object in the observer frames before and after motion. Object-location targets are computed directly from scene geometry.

C.2 Object Identity and Persistent Localization

Establishing object identity. Persistent object-localization queries refer to objects clearly observed in the available image history. Each question identifies its target by an unambiguous category name or by a marker in an earlier image. Scene instance identities associate these references across viewpoints, while rendered visibility evidence determines whether the object is currently visible or supported only by earlier observations. This ensures that queries about objects outside the current view refer to entities already encountered in the episode.

Localization across viewpoints. We construct sequences of location queries for the same object as the observer moves, including periods when the object is no longer visible and, when available, its subsequent reappearance. Each target is computed from the object's fixed world center and the camera pose at the queried viewpoint, following Eq. 8. These sequences supervise how an established object's observer-relative location changes with self-motion, even when no new visual observation of that object is available.

C.3 Operations over Spatial State

Operation families. M3 supervision applies spatial operations to information established along a trajectory. We organize these operations into six broad families, described in Table 5.

Table 5: Examples of spatial operations used in M3 supervision.

Operation family	Operation	Examples
Reference-frame reasoning	Express relations in a specified frame	Camera-relative direction; object-centered reference
Hypothetical motion	Update relations after imagined motion	Rotation; translation; turn–move sequences
Inverse spatial inference	Solve for an unknown from spatial constraints	Target identification from directional constraints
Measurement and comparison	Measure, compare, or rank geometry	Distance; size; height; area
Temporal and set operations	Query events and combine observations	Appearance order; reappearance; counts; set overlap
Route reasoning	Evaluate paths between locations	Route description; traversable-length comparison

D Dense Trajectory Evaluation Details

D.1 Evaluation Protocol

Object registration. An object is registered after two consecutive observations satisfying the following conditions: its center is 0.5–5 m from the camera horizontally, its instance mask contains at least 4,000 pixels and lies at least 16 pixels from the image boundary, the mask covers at least 60% of the projected bounding-box area, and at most 5% of the projected box lies outside the image. Pixel thresholds are normalized to a 1024×1024 image. Mask coverage serves as a visibility proxy.

We register at most three objects per trajectory in chronological order. Registration uses only current and past observations. A numbered marker identifies each object in its registration image, which subsequent questions reference. Once registered, an object remains eligible for queries even when it becomes invisible or leaves the registration distance range.

Queries and visibility stages. At each frame after the first, we query motion from the preceding frame, followed by the current observer-relative locations of all registered objects. Location targets are the horizontal camera-frame coordinates of the scene object bounding-box centers. Of the 100 trajectories, 90 contain registered objects and 87 include absent-object queries; the other ten contribute only self-motion queries.

Table 6 defines the five mutually exclusive stages used to group localization queries. Overall localization scores include all five stages.

Table 6: Localization query stages in the dense evaluation. Pixel counts are normalized to 1024×1024 resolution; stages other than Initial apply after registration.

Stage	Definition	Queries
Initial	Query at the object's registration frame	231
Absent	Zero visible instance pixels	2,240
Reappeared	First observation with at least 256 instance pixels after an absence	79
Visible	Other observations with at least 256 instance pixels after registration	1,651
Weak visibility	Positive instance-pixel count below 256	6
Total		4,207

D.2 Baseline Prompts

For InternVL3-8B and Qwen3.8-27B, we append answer-format instructions to each motion and object-localization question.

Self-motion estimation.

> **User.** For this question, compare Image 1 with the current image.
> How do your position and heading change from Image 1 to the current image?
> Answer briefly with estimated numeric distances in meters and the heading change in degrees, relative to the earlier camera. Use forward/backward, left/right, and turn left/right. Format example only: Move 0.4 m forward and 0.2 m right. Turn left 15 degrees. For no change, answer: Stay still. Give your best estimate; output only the short answer, no explanation.
> **InternVL3-8B:** Move 0.3 m forward. Turn right 10 degrees.
> **Qwen3.8-27B:** Stay still.

Object localization.

> **User.** Using observations through Image 2, where is the object introduced as Marker 1 in Image 2 relative to your position and heading in Image 2? Give forward/backward and left/right distances in meters.
> Answer briefly with two estimated distances in meters in the current camera frame, using forward/backward and left/right. Format example only: 2.0 m forward, 0.5 m left. Give your best estimate; output only the short answer, no explanation.
> **InternVL3-8B:** 1.5 m forward, 0.5 m right.
> **Qwen3.8-27B:** 3.0 m forward, 1.5 m right

D.3 Metrics and Geometric Propagation

Scoring. Object locations use the horizontal observer frame defined in Appendix C.1. For a query set Q, let $V \subseteq Q$ contain the queries with valid location estimates. Position error and correctness are

$$e_i = \|\hat{z}_i - z_i\|_2, \qquad c_i = \begin{cases} \mathbf{1}[e_i \le 0.5\,\mathrm{m}], & i \in V, \\ 0, & i \notin V. \end{cases} \tag{9}$$

Missing or unparseable predictions, including uninitialized propagated states, count as incorrect. Accuracy and mean position error are

$$\mathrm{Acc}_{\mathrm{query}} = \frac{100}{|Q|} \sum_{i \in Q} c_i, \qquad \overline{e} = \frac{1}{|V|} \sum_{i \in V} e_i. \tag{10}$$

All localization results weight queries equally. Accuracy includes every query, while mean error is computed over valid estimates. The geometric conditions in Table 7 provide estimates for all queries.

For self-motion, translation error is the Euclidean difference between predicted and ground-truth horizontal displacements in the earlier camera frame. For motion query i, the predicted and ground-truth heading changes are $\widehat{\delta\psi}_i$ and $\delta\psi_i$; their error is $|\,\mathrm{wrap}_{[-180^\circ,180^\circ)}(\widehat{\delta\psi}_i - \delta\psi_i)|$. Motion errors are averaged over valid outputs.

Geometric propagation. An object is initialized at the first step t_0 with a positive model visibility response and a valid location estimate $\hat{z}_{t_0}$. Using the horizontal basis B_t and observer position o_t from Appendix C.1, we store a fixed world-plane point and compute its subsequent observer-relative locations as

$$\hat{c}_{xy} = B_{t_0}\hat{z}_{t_0} + o_{t_0}, \qquad \hat{z}_t^{\mathrm{prop}} = B_t^{\mathsf{T}}(\hat{c}_{xy} - o_t). \tag{11}$$

Later VLM location estimates do not update the stored point. Initialization uses model predictions without ground-truth instance masks; direct prediction requires no visibility gate. The pose-source comparison holds object-location and auxiliary visibility predictions fixed and uses the same propagation rule, varying only whether poses are ground truth or integrated from predicted motion. Ground-truth poses are supplied only to the geometry module.

D.4 Complete Results

The following table supplements Table 4 with overall accuracy and mean position errors under the same evaluation conditions.

Table 7: **Complete pose-assisted object-localization results.** (a) Direct prediction and pose-based propagation use the same object predictions under No QA history; GT poses are used for propagation. (b) With Model-history object predictions and pose-based propagation fixed, we vary the source of camera poses. GT poses are supplied only to the geometry module, not to the VLM. Acc@0.5m uses a 0.5 m horizontal-error threshold in the current camera frame; all metrics weight queries equally. All and Absent contain 4,207 and 2,240 queries, respectively. Bold indicates the best result within each model and panel.

(a) Direct prediction versus pose-based propagation

Fixed: No QA history; GT poses for propagation.

Model	Localization method	Acc@0.5m (%) ↑		Mean error (m) ↓
		All	Absent	All
Qwen3.8-27B	Direct prediction	6.44	1.29	4.725
	Pose-based propagation	**16.81**	**16.88**	**1.322**
URUQI$_{\text{Syn}}$-8B	Direct prediction	49.56	35.13	0.743
	Pose-based propagation	**69.65**	**71.61**	**0.412**

(b) Effect of camera-pose source

Fixed: Model history; pose-based propagation. Evaluated on absent objects.

Model	Camera-pose source	Acc@0.5m (%) ↑ Absent	Mean error (m) ↓ Absent
Qwen3.8-27B	GT poses	**18.48**	**0.961**
	Integrated predicted motion	3.97	3.737
URUQI$_{\text{Syn}}$-8B	GT poses	**72.86**	**0.397**
	Integrated predicted motion	34.02	1.084

D.5 Auxiliary Visibility and Object Correspondence

Diagnostic protocol. For each localization query, we separately ask whether the registered object is currently visible and, if so, request a pixel on that object. The model receives all images up to the current frame, including the marked registration image, without QA history or privileged geometry. The same predictions are reused across the model's history and pose conditions. A valid positive response includes an in-bounds pixel coordinate; invalid or uncertain responses cannot initialize propagation. Instance masks are used only to score pixel correspondence, not to accept an initialization. This auxiliary query does not affect the native localization scores in Table 3.

Metrics. Ground-truth visibility requires at least one target-instance pixel, giving 1,967 visible and 2,240 absent queries. Precision and recall measure valid positive visibility responses against these labels. False-visible rate is the fraction of absent queries receiving a valid positive response. Pixel Hit is the fraction of all ground-truth-visible queries with a valid positive response and a predicted pixel on the correct instance; missed detections and invalid responses receive no credit.

Table 8: Auxiliary visibility and object correspondence on 4,207 queries. All values are percentages; bold marks the best value in each column.

Model	Visibility precision ↑	Visibility recall ↑	Pixel Hit ↑	False-visible rate ↓
InternVL3-8B	54.13	88.87	7.52	66.12
Qwen3.8-27B	**93.51**	78.39	50.64	**4.78**
URUQI$_{\text{Syn}}$-8B	68.31	**99.49**	**71.68**	40.54

Results. Table 8 shows that URUQI$_{\text{Syn}}$-8B achieves the highest Pixel Hit, but has a higher false-visible rate than Qwen3.8-27B (40.54% versus 4.78%). These metrics are computed over their

respective query sets, not only at initialization. As described in Appendix D.3, visibility responses after initialization do not change the stored object location.