MUSHROOM: BENCHMARKING LONG-TERM SPATIAL REASONING IN MULTI-ROOM ENVIRONMENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term spatial reasoning requires vision-language models (VLMs) to consolidate partial observations into spatial knowledge and update it as spatial states change. We introduce MUSHROOM, a benchmark for long-term spatial reasoning in multi-room environments, comprising 230 egocentric videos that traverse multiple rooms, where room boundaries disrupt spatial continuity. We design tasks along two complementary axes: Static Spatial Reasoning within a stationary configuration, and Dynamic Spatial Reasoning over state changes. We further propose SPORE, a training-free pipeline that organizes a single video into a structured representation: an object-centric spatial graph. Our evaluation demonstrates that contemporary VLMs have fundamental limitations in cross-region reasoning and updating spatial states, while SPORE substantially improves both static and dynamic reasoning, with consistent gains on real-world indoor benchmarks. Further analysis shows that the spatial graph links temporally distant observations and stabilizes geometric estimation, yet cross-region reasoning falters under frequent transitions, a pattern reminiscent of the doorway effect.[1]

1 INTRODUCTION

> *"One remarkable feat of the human mind is to conceive of some large spaces as integrated wholes rather than piecemeal as they are experienced."*
>
> — Barbara Tversky (Tversky, 2003, p. 72)

When exploring a multi-room environment, can embodied agents develop spatial awareness from fragmented observations? Recent advances in vision-language models (VLMs) have established them as a core component of embodied agents that interact with the physical environment (Driess et al., 2023; Kim et al., 2025; Black et al., 2025; Bu et al., 2025). During multi-region exploration, agents must integrate sequential observations into a coherent representation, preserving object–region associations across visits and updating spatial knowledge as the environment changes (Hirtle & Jonides, 1985; Montello & Pick, 1993; Tversky, 2003; McNamara et al., 2008). This motivates the evaluation of long-term spatial reasoning in complex spaces, which consist of multiple regions (e.g., rooms) defined by physical and perceptual boundaries (Brunec et al., 2018). Existing benchmarks have extended the scope of spatial understanding from images to videos (Yang et al., 2025; Li et al., 2025b; Lin et al., 2025b; Zhang et al., 2026d), yet long-term spatial reasoning across multiple regions and environmental changes remains underexplored.

To bridge this gap, we introduce MUSHROOM, a benchmark for long-term spatial reasoning in multi-room environments, featuring 230 egocentric videos of complex indoor scenes. As illustrated in Figure 1, each video traverses multiple rooms with frequent region transitions, capturing environmental changes induced by object relocations. We design nine multiple-choice tasks along two complementary axes: Static Spatial Reasoning (4 tasks; 2,121 questions) examines spatial understanding within a stationary configuration, while Dynamic Spatial Reasoning (5 tasks; 3,136 questions) assesses the ability to track and update spatial states.

A key aspect of long-term spatial reasoning is constructing a structured representation with object locations and region membership—which region each object belongs to (Hirtle & Jonides, 1985;

[1] The dataset and code will be publicly released.

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.

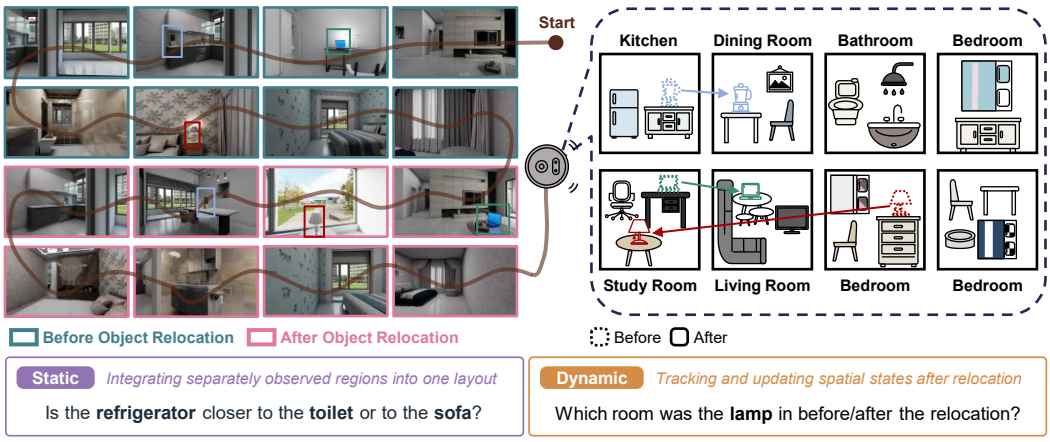

Figure 1: From fragmented observations during multi-room exploration (left), a model must construct an abstract spatial representation (right) for static and dynamic spatial reasoning (below).

McNamara et al., 1989; 2008). In this light, we propose SPORE, a training-free pipeline that organizes observations from a single video into an object-centric spatial graph. Each node encodes an object's position, region assignment, category, and semantic description. Together, these capture both its spatial state and semantic identity. SPORE proceeds in three stages: (1) extracting 3D spatial information from frame-level detections, (2) constructing spatial nodes by aggregating consistent object tracks, and (3) retrieving query-relevant nodes as candidate spatial states.

Our experiments reveal that contemporary VLMs struggle to integrate spatial information across regions and track state changes during long-term exploration—limitations that persist even with spatial supervision or frame scaling. Leveraging the object-centric graph, SPORE takes a step forward in long-term spatial reasoning, with improvements on both static and dynamic tasks. Augmenting visual observations with an oracle graph, we underscore the importance of structured representations: the graph improves the linking of temporally distant observations and stabilizes geometric estimation in multi-region environments. Nevertheless, the spatial graph falls short in cross-region reasoning with denser room transitions, a pattern reminiscent of the "doorway effect" (Radvansky & Copeland, 2006; Radvansky et al., 2011) that exposes the critical hurdle of maintaining spatial continuity in multi-region exploration. We summarize our contributions as follows:

- We introduce MUSHROOM, a benchmark comprising 230 egocentric videos and approximately 5K questions for evaluating static and dynamic spatial reasoning during multi-region exploration.
- We propose SPORE, a training-free pipeline that constructs object-centric spatial graphs from a single video, improving long-term spatial reasoning on MUSHROOM and real-world indoor benchmarks.
- We analyze how spatial representations support temporal integration and geometric estimation, while revealing persistent challenges in maintaining spatial continuity for cross-region reasoning through frequent room transitions.

2 Related Works

Spatial Language Understanding. Translating spatial scenes into linguistic expressions is central to how VLMs interpret visual observations. Previous work focuses on spatial relations in the image domain, under single-view or multi-view settings (Johnson et al., 2017; Hudson & Manning, 2019; Yang et al., 2019; Goyal et al., 2020; Liu et al., 2023a; Kamath et al., 2023; Du et al., 2024; Wang et al., 2024; Fu et al., 2024; Ma et al., 2025; Zhang et al., 2025a). With advances in video understanding (Maaz et al., 2024; Lin et al., 2024; Wu et al., 2024; Fu et al., 2025; Zhou et al., 2025; Chen et al., 2026), recent work pushes this further into continuous visual sequences, assessing visuospatial capabilities of VLMs within a stable configuration of environments (Yang et al., 2025; Li et al., 2025b; Wang et al., 2025; Guo et al., 2025; Lin et al., 2025b; Song et al., 2025; Lin et al., 2025a; Wu et al., 2026; Zhang et al., 2026a; Cheng et al., 2026; Zhang et al., 2026d; Lang et al., 2026; Zhang

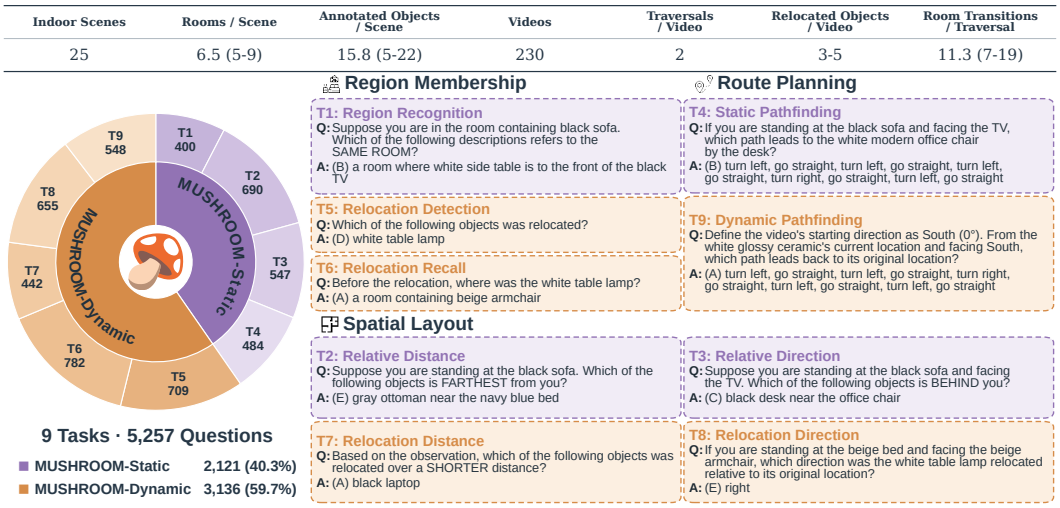

Figure 2: Overview of MUSHROOM. Top: The video statistics. Bottom: The distribution and examples of tasks for static and dynamic spatial reasoning, grouped into three categories. We formulate tasks in a multiple-choice format with five options.

et al., 2026b). In this light, we extend this scope to multi-region exploration, requiring VLMs to integrate spatial information across regions and reason about spatial state changes over time.

Spatial State Tracking in VLMs. Tracking spatial states is a fundamental component of long-context spatial reasoning (Wang & Spelke, 2000; Merhav & Wolbers, 2019). Recent work examines state tracking in linguistic contexts, asking whether LMs can maintain entity states over long contexts and sequences of state-changing operations (Merrill & Sabharwal, 2023; Bhattamishra et al., 2020; Strobl et al., 2024; Kim & Schuster, 2023; Tang et al., 2026). We extend state tracking to visual contexts, where VLMs face a distinct challenge: in long videos, objects and previously visited regions may return to view only after intervening observations, requiring object re-identification between non-contiguous views (Li & Loy, 2018; Bansal et al., 2019; Wald et al., 2019) and visual place recognition under changing viewpoints (Arandjelović et al., 2016; Berton et al., 2023; Garg et al., 2024; Yao et al., 2025). In this work, we investigate whether VLMs can establish object–place correspondences to maintain and update spatial states in complex environments.

Spatial Reasoning with VLMs. Recent VLMs follow the prevailing paradigm of connecting visual encoders to language models, enabling powerful visual semantic understanding (Liu et al., 2023b; Li et al., 2023). Under language-modeling objectives, VLMs face a fundamental limitation in spatial reasoning, struggling to infer 3D structure from 2D inputs (Chen et al., 2024; Daxberger et al., 2025). Prior studies address this limitation through supervised fine-tuning (SFT) and reinforcement learning (RL), training VLMs to infer geometric information from visual inputs (Ray et al., 2025; Ouyang et al., 2025; Liu et al., 2026; Yang et al., 2026a; Yin et al., 2026). Geometry-aware architectures incorporate 3D features, extracted from point encoders or reconstruction modules (Hong et al., 2023; Mao et al., 2025; Wu et al., 2025; Fan et al., 2026; Hu et al., 2026). Another line of work translates outputs from external modules into structured inputs, providing VLMs with object coordinates, scene abstractions, or topological maps (Ma et al., 2024; Lee et al., 2025; Gu et al., 2024; Zhang et al., 2026c; Honerkamp et al., 2024; Yan et al., 2025; Ju et al., 2026). Building on this direction, we abstract visual observations into an object-centric graph for long-term spatial memory.

3 MUSHROOM: Benchmarking Spatial Reasoning over Multi-Region Exploration

We introduce **MUSHROOM**, a benchmark for **M**emory **U**pdating across **S**patial c**H**anges in multi-**ROOM** environments. During exploration of complex environments, agents must maintain long-term spatial memory by integrating observations across regions and preserving the resulting spatial knowledge over successive visits. This requires reasoning about spatial relationships across regions

in a stationary configuration, and updating object states when the environment changes. Building on the requirements of long-term spatial memory, we establish a two-axis benchmark with multi-traversal egocentric videos. See Appendix A for more details on the data construction.

3.1 Long-Term Video Construction

Our dataset consists of egocentric videos in which a robot navigates furnished indoor environments with multiple interconnected rooms where crossing a doorway often replaces the set of visible objects. We use 25 synthetic scenes from InteriorAgent (SpatialVerse Research Team, 2025) and render the videos using NVIDIA Isaac Sim (NVIDIA, 2025). Each scene contains two waypoint routes that traverse its rooms in different orders. Each video follows one route twice: **Traversal 1** records the environment before object relocation, while **Traversal 2** repeats the same camera trajectory after selected objects are relocated to different rooms. We render each video as a continuous sequence covering both traversals without explicit boundary markers, recording the source and destination of every relocation. Multiple relocation configurations yield 230 videos. We also select 396 salient objects and annotate each object with a distinctive referring expression (e.g., "white office chair") to support unambiguous spatial queries. The top of Figure 2 summarizes the video statistics.

3.2 Task Formulation

We formulate nine multiple-choice question (MCQ) tasks along three dimensions of long-term spatial reasoning: region membership, spatial layout, and route planning. Region membership captures which bounded region contains each object through object–region associations. With its location, an object's region membership defines its spatial state. Spatial layout captures the global configuration of objects and regions, supporting distance and direction judgments. We describe relative directions using eight fine-grained terms. Route planning translates this spatial knowledge into a sequence of four navigation actions: go straight, turn left, turn right, and turn around. We organize the tasks into two complementary groups: MUSHROOM-Static (MRS) for reasoning within stable spatial configurations and MUSHROOM-Dynamic (MRD) for reasoning over changes in object spatial states. The bottom of Figure 2 presents statistics and an example for each task.

MRS: Static Spatial Reasoning. It comprises four tasks covering spatial knowledge within a static configuration. Region Recognition (T1) identifies a room from the configuration of its stationary objects. Relative Distance (T2) compares distances between objects situated in different regions, while Relative Direction (T3) determines the direction of a target object from an arbitrary reference pose. Both tasks require integrating separately observed regions into a coherent spatial layout. Static Pathfinding (T4) requires determining a sequence of navigation actions connecting two locations.

MRD: Dynamic Spatial Reasoning. Five tasks examine how VLMs update spatial memory after object relocation and reason over the resulting spatial states. Relocation Detection (T5) identifies the relocated objects, while Relocation Recall (T6) recovers the object–region association from either the prior or updated spatial state. Relocation Distance (T7) compares which relocated objects traveled farther, while Relocation Direction (T8) determines the direction of relocation from an arbitrary pose. Dynamic Pathfinding (T9) requires determining a sequence of navigation actions to return a relocated object from its updated location to its original location.

4 SPORE: Spatial Parsing for Object-Centric Representation Encoding

We propose SPORE, a training-free pipeline that organizes observations from a single video into an object-centric spatial graph. As illustrated in Figure 3, SPORE proceeds in three stages: (1) extracting 3D spatial information from frame-level detections, (2) constructing spatial nodes by aggregating consistent tracklets, and (3) retrieving query-relevant nodes as candidate spatial states. We provide implementation details and additional experiments in Appendix B.

Object-Centric Representation. In complex environments, central to long-term spatial reasoning is forming an abstract representation of objects' spatial configuration—their locations and region associations (McNamara et al., 1989; Huttenlocher et al., 1994). We formulate this representation as the object-centric graph, where each node encodes an object state. Each node is denoted as

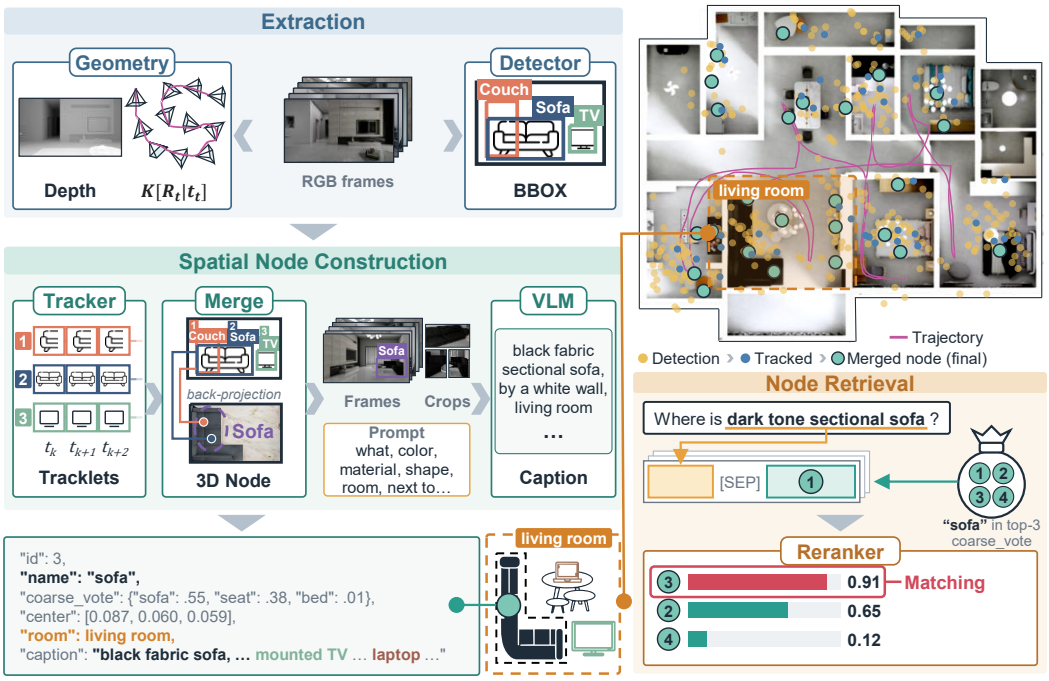

Figure 3: From a single video, SPORE constructs and retrieves a node for the "dark tone sectional sofa" through three stages: (1) predicting scene geometry and detecting objects, (2) merging consistent tracklets into a spatial node with a caption, and (3) reranking candidate nodes against the query.

$N_k = (\mathbf{c}_k, \omega_k, \kappa_k, a_k)$, where $\mathbf{c}_k \in \mathbb{R}^3$ is the position, ω_k the region assignment, κ_k the category set, and a_k the textual description. The position and region assignment define the spatial state, while the categories and description characterize the semantic identity. Without explicit edges for spatial relations or navigation paths, the graph provides spatial cues for long-term spatial reasoning.

Stage 1: Spatial Information Extraction. Given a video $V = (I_1, \ldots, I_T)$ of T frames, a feed-forward geometry model jointly estimates the camera intrinsics K and frame-wise scene geometry. For each frame t, it predicts a depth map D_t and a camera pose $[R_t \mid \mathbf{t}_t]$, parameterized by rotation R_t and translation $\mathbf{t}_t$. An open-vocabulary detector identifies objects using predefined category-level labels (e.g., "vase"), without access to fine-grained query descriptions (e.g., "white glossy ceramic"). We extract a 2D object coordinate from each tracked box b in frame t, which is back-projected to a 3D point $\mathbf{p}_{b,t} \in \mathbb{R}^3$ using the predicted depth, camera intrinsics, and camera pose.

Stage 2: Spatial Node Construction. With global motion compensation, a multi-object tracker associates detections between consecutive frames, without resetting between traversals; a tracklet $\mathcal{T}_i$ is the set of detections sharing the same track ID. For each tracklet i, we compute the centroid $\bar{\mathbf{p}}_i$ as the coordinate-wise median of the detection points, and assign the dominant category $\hat{\kappa}_i$ by confidence-weighted voting over detection labels. Since the tracker may fragment the same object into multiple tracklets between intermittent observations, SPORE merges tracklets into clusters: each cluster C_k grows from a seed tracklet i and takes every unassigned tracklet j with $\hat{\kappa}_j = \hat{\kappa}_i$ and $\|\bar{\mathbf{p}}_j - \bar{\mathbf{p}}_i\|_2 \leq \rho$, where ρ is the merge radius. The aggregated detections of C_k form the observation set $\mathcal{O}_k = \bigcup_{i \in C_k} \mathcal{T}_i$, from which the spatial node N_k is constructed. $\mathbf{c}_k$ is their coordinate-wise median point, and κ_k the top-3 coarse categories from confidence-weighted voting. A VLM-based captioner generates a_k from full frames with the object's bounding box and from tight crops, describing its appearance, surroundings, and room. SPORE infers ω_k by combining region cues from a_k and κ_k with distance-weighted evidence from nearby nodes. A relocated object yields separate nodes for each spatial state, since its two positions lie farther apart than ρ despite sharing $\hat{\kappa}_i$.

Stage 3: Query-Conditioned Node Retrieval. Given a query q, SPORE maps the queried object to a predefined coarse category and keeps the nodes whose category sets κ_k contain it. A cross-encoder reranker R scores every candidate against the query, selecting the top-10 candidate set $\mathcal{L}(q)$. The first returned node is $k_1 = \arg\max_{k \in \mathcal{L}(q)} \big(R(q, a_k) + \beta\, \mathbf{1}[\omega_k = \omega(q)]\big)$, where $\omega(q)$

Table 1: Quantitative comparisons on MUSHROOM under the default (64-frame) setting, with VQA *Accuracy* (*Acc.*, %) as the evaluation metric. We report Qwen3-VL-8B's text-only performance as a blind baseline to assess reliance on language priors. Excluding proprietary models, the best result within each model group is shown in **bold**.

Model	Size	MRS				MRD					Overall
		T1	T2	T3	T4	T5	T6	T7	T8	T9	
Blind Baseline											
Qwen3-VL (Bai et al., 2025a)	8B	34.75	18.84	12.07	16.53	35.40	25.58	18.33	16.18	16.42	21.74
Proprietary VLMs											
GPT-5.2 (OpenAI, 2025)	–	59.60	24.10	19.00	31.80	54.30	39.40	29.40	15.60	33.80	33.73
Gemini-3.5-Flash (Google DeepMind, 2026)	–	51.00	28.99	21.39	31.20	49.08	40.66	33.48	21.83	35.58	34.70
General VLMs											
Qwen2.5-VL (Bai et al., 2025b)	7B	49.50	17.97	21.39	21.28	33.00	28.39	15.84	23.36	18.07	25.11
Qwen2.5-VL	72B	50.75	24.20	25.59	36.16	25.67	29.41	27.38	19.39	**32.30**	28.95
Qwen3-VL	2B	48.50	27.97	28.34	28.51	13.96	30.56	18.33	**29.92**	21.17	26.84
Qwen3-VL	8B	61.50	27.54	**29.43**	25.62	**56.84**	**36.70**	**31.22**	19.39	29.74	**34.98**
LLaVA-OneVision (Li et al., 2025a)	7B	61.25	28.84	24.68	20.87	35.26	24.81	24.21	20.76	23.36	28.44
LLaVA-OneVision	72B	**62.50**	**30.00**	24.86	42.36	44.15	29.28	27.60	20.00	29.74	33.40
InternVL3 (Zhu et al., 2025)	8B	57.25	28.70	27.06	25.41	43.44	24.68	23.98	19.85	25.18	29.92
InternVL3	78B	59.25	26.38	24.86	**43.38**	42.88	26.60	23.76	18.02	32.11	31.87
Spatial VLMs											
Qwen2.5-VL-SAT (Ray et al., 2025)	7B	**57.50**	**30.87**	24.31	**28.72**	37.09	16.88	26.02	21.53	25.55	28.65
SpaceR (Ouyang et al., 2025)	7B	53.50	20.72	21.39	16.94	35.12	**28.26**	13.57	**27.02**	17.70	25.87
Video-R1 (Feng et al., 2025)	7B	54.75	24.49	**28.34**	22.31	35.97	24.94	26.02	18.63	23.18	27.87
VST-SFT (Yang et al., 2026a)	7B	45.75	15.22	11.33	20.45	24.26	23.02	20.14	25.80	16.61	21.88
VST-RL	7B	48.50	23.77	23.95	26.24	24.54	26.47	16.06	20.61	22.45	25.22
Cambrian-S (Yang et al., 2026b)	7B	57.25	27.25	25.41	20.25	**45.42**	24.81	**27.60**	22.75	18.43	**29.33**
Spatial-MLLM (Wu et al., 2025)	8B	54.50	15.94	18.65	14.67	14.95	15.09	19.68	21.37	13.87	19.55
Spatial Prompting (Grid Map)											
Qwen3-VL	8B	52.50	24.93	27.61	21.69	40.62	34.14	24.66	22.29	24.64	30.11
Cambrian-S	7B	50.50	31.01	32.72	21.69	28.49	36.06	27.83	27.48	21.72	30.55
SPORE											
Qwen3-VL	8B	72.75	**51.88**	**37.29**	**27.07**	**61.35**	**54.09**	**47.06**	**33.28**	**32.30**	**46.51**
Cambrian-S	7B	**74.75**	36.96	36.75	22.11	52.33	42.84	35.07	32.82	24.27	39.40

is the region the queried category implies and $\beta > 0$ adds a fixed bonus to nodes in that region. Starting from k_1, SPORE returns up to m nodes from $\mathcal{L}(q)$, each lying farther than δ from the previously returned ones in Euclidean distance on the floor plan. These same-category nodes at distinct locations represent candidate spatial states of a relocated object.

5 Experiments

With MUSHROOM, we investigate the central question: can VLMs retain and update spatial memory during multi-region exploration? We begin with the comparison between baselines and SPORE. Then, we examine the roles of visual observations and spatial graphs, followed by an error analysis. Details on evaluation setup are provided in Appendix C.

5.1 Main Results

Under the default 64-frame setting, we evaluate long-term spatial reasoning in general and spatial VLMs, including proprietary models. For Qwen3-VL-8B and Cambrian-S-7B, we compare SPORE with grid-map prompting (Yang et al., 2025; Yin et al., 2026), which predicts object locations on a coordinate grid. With evaluation on VSI-Bench (Yang et al., 2025) and ReVSI (Zhang et al., 2026d), we examine the broader applicability of our pipeline for real indoor scenes.

Limits of Visual Observation. As shown in Table 1, with visual observation alone, contemporary VLMs face fundamental limitations in integrating observations across regions and updating spatial states during long-term exploration. Most VLMs perform well on T1, indicating that they can associate objects with the corresponding regions. However, performance drops substantially on T6, revealing a shared difficulty in tracking spatial state changes. Notably, general VLMs show pronounced gains in route planning as model scale increases, with InternVL3-78B reaching 43.38% on

Table 2: Results on MCQ tasks from VSI-Bench (Yang et al., 2025) and ReVSI (Zhang et al., 2026d) (*Acc.*, %): Relative Distance (Rel. Distance), Relative Direction (Rel. Direction), and Route Planning. Overall reflects performance across both benchmarks combined.

Model	VSI-Bench			ReVSI			Overall
	Rel. Distance	Rel. Direction	Route Planning	Rel. Distance	Rel. Direction	Route Planning	
Qwen3-VL-8B	53.24	48.24	32.99	57.34	40.75	37.26	48.03
+ Grid Map	51.69	46.18	32.47	48.10	36.54	40.57	44.44
+ SPORE	52.68	62.09	36.60	52.15	47.00	42.45	**52.32**
Cambrian-S-7B	62.82	50.21	34.54	38.23	48.56	26.89	47.54
+ Grid Map	60.70	50.93	35.05	41.39	37.98	27.83	45.71
+ SPORE	59.86	61.36	36.60	53.54	51.56	29.72	**54.10**

T4. Meanwhile, spatial VLMs show limited transfer to multi-room environments, yielding modest improvements over their backbone (Qwen2.5-VL-7B) despite explicit supervision on video spatial reasoning. Furthermore, explicit 3D geometric cues provide limited support for long-term spatial reasoning: Spatial-MLLM scores below blind baseline (Overall: –2.19%p).

Improvement with SPORE. With structured spatial graph, SPORE substantially outperforms video-only baselines across general and spatial VLMs (Qwen3-VL-8B: +11.53%p; Cambrian-S-7B: +10.07%p; Table 1). On static tasks, it drives notable gains in cross-region spatial understanding. Qwen3-VL-8B shows a substantial improvement on relative distance (T2: +24.34%p), while Cambrian-S-7B benefits more on relative direction (T3: +11.34%p), suggesting that explicit positions and region assignments facilitate understanding of the global layout across separated rooms. On dynamic tasks, the spatial graph enables VLMs to reason about spatial changes in complex environments. Qwen3-VL-8B shows the larger advance, with better recall of prior and updated spatial states from the two traversals (T6: +11.25%p over Cambrian-S-7B). We further investigate whether the benefits of SPORE transfer to real indoor scenes, extending this comparison to VSI-Bench and ReVSI. As shown in Table 2, it improves overall accuracy for both models (Qwen3-VL-8B: +4.29%p; Cambrian-S-7B: +6.56%p), with the broader applicability for real-world environments.

Comparison with Spatial Prompting. Table 1 shows that grid-map prompting falls substantially short of SPORE. For Qwen3-VL-8B, it even falls below the video-only baseline (Overall: -4.87%p), indicating that spatial prompting cannot elicit abstract representations from long-term videos. As shown in Table 2, the same gap appears on VSI-Bench and ReVSI, where SPORE outperforms grid-map prompting. Notably, it achieves the largest gains on Rel. Direction (VSI-Bench: Qwen3-VL-8B +15.91%p, Cambrian-S-7B +10.43%p; ReVSI: Qwen3-VL-8B +10.46%p, Cambrian-S-7B +13.58%p). Compared to spatial prompting, our graph provides more reliable support for long-term spatial reasoning in complex environments.

5.2 Analysis I: The Roles of Visual Observation and Spatial Representation

Building on the improvement with SPORE, we move to the next step: the respective roles of visual observation and structured spatial representation. For comparison with SPORE, we scale the number of input frames to 512 and evaluate our predicted graphs without video frames. We then test whether VLMs reason well over the oracle representation; the ground-truth spatial graph (GT graph) is constructed from each traversal's object spatial states, including positions and region assignments, with object identity corresponding to the same description in MUSHROOM. We conduct an ablation study on two components: region assignment and spatial states from each traversal.

Synergy of Visual Observation and Spatial Graph. As shown in Table 3, frame scaling yields saturated performance, indicating that additional visual observation is insufficient to resolve bottlenecks in cross-region spatial reasoning and spatial state updating; results across additional frame settings are reported in Appendix D. However, without visual observation, the predicted graph alone falls short on overall score, particularly for Cambrian-S-7B. This demonstrates the complementary effect: the spatial graph organizes information from unstructured observations, while the video retains visual evidence beyond what the graph encodes.

Closing the Gap with Oracle Graph. Table 3 highlights the importance of complete spatial representations for long-term, multi-region exploration. Supplementing visual observations with GT graphs substantially narrows the gap in long-term spatial reasoning. For Qwen3-VL-8B, the GT

Table 3: Results with frame scaling and spatial graphs. Trav. 1/2 denote prior/updated states. Δ is the Overall change from SPORE in Table 1 (%p); the largest improvement is in **bold**.

Input	MRS				MRD					Overall	Δ
	T1	T2	T3	T4	T5	T6	T7	T8	T9		
Qwen3-VL-8B											
Video Only (512 Frames)	58.50	28.70	30.71	30.17	58.39	35.55	28.73	20.15	30.29	35.44	–11.07
Graph Only (Pred.)	71.25	69.71	59.23	20.04	41.04	42.84	43.89	37.40	20.99	45.03	–01.48
Graph Only (GT)	85.50	79.86	65.45	22.11	98.03	92.71	93.44	42.90	17.70	67.89	**+21.38**
Video + GT Graph	80.50	63.04	41.32	30.17	98.03	93.48	95.93	39.24	32.66	64.96	+18.45
w/o Region Assignment	79.50	66.09	38.21	21.49	98.17	60.36	96.38	34.50	25.91	58.00	+11.49
w/o Prior State (Trav. 1)	80.25	68.99	36.75	30.37	59.66	43.73	26.47	22.60	22.99	43.77	–02.74
w/o Updated State (Trav. 2)	81.50	69.57	44.97	27.07	39.63	49.87	23.30	22.90	18.61	42.02	–04.49
Cambrian-S-7B											
Video Only (512 Frames)	58.50	27.39	23.03	18.39	47.67	22.76	22.62	21.53	19.71	28.59	–10.81
Graph Only (Pred.)	71.00	39.71	29.25	18.18	50.63	34.14	29.41	28.70	18.43	35.21	–04.19
Graph Only (GT)	79.00	37.54	34.92	18.80	96.47	33.50	35.07	30.84	18.80	43.05	+03.65
Video + GT Graph	76.00	36.81	36.93	22.31	96.19	54.35	36.20	34.35	28.83	47.90	**+08.50**
w/o Region Assignment	75.75	37.10	37.48	20.66	94.36	35.42	34.62	31.30	22.63	43.60	+04.20
w/o Prior State (Trav. 1)	77.00	36.81	36.93	20.87	30.89	29.80	31.67	32.82	22.45	34.14	–05.26
w/o Updated State (Trav. 2)	76.25	36.81	36.93	19.21	42.03	43.86	33.26	34.05	22.26	37.80	–01.60

graph alone achieves the highest score (67.89%), showing that general VLMs can reason about spatial configurations given structured inputs. Visual inputs still provide complementary information for pathfinding, improving accuracy over the GT graph alone (T4: +8.06%p, T9: +14.96%p). Cambrian-S-7B exhibits a different pattern: retaining video improves overall score (+4.85%p), indicating that models may differ in how effectively they exploit visual and structured information.

Ablation of Graph Components. We ablate region assignment and state history to measure their individual contributions. Without region assignment, despite intact object coordinates and observation history, performance drops sharply on relocation tracking (T6: Qwen3-VL-8B, –33.12%p; Cambrian-S-7B, –18.93%p): VLMs cannot bind the updated state of objects to rooms. This deficiency further degrades cross-region pathfinding (T4/T9). Ablating either the prior or updated state, performance declines across all dynamic tasks, despite access to the remaining state and both video traversals. Relocation detection drops substantially for both models; without each state, the models fail to identify object state changes during exploration. While static tasks remain robust across both ablations, dynamic task depends on both object–region associations and access to each spatial state.

5.3 Analysis II: Temporal and Spatial Dimensions of Long-Term Spatial Reasoning

With Qwen3-VL-8B, the strongest model on MUSHROOM, we frame the analysis along two dimensions: for the temporal dimension, we examine how error scales with the frame distance between observations and how state recall degrades during exploration; for the spatial dimension, we analyze geometric estimation errors in the static configuration of objects. Bringing the two together, we assess the effect of region boundaries through room transition density.

Difficulty in Temporal Integration. Figure 4 (a) demonstrates that, for static states, the spatial graph bridges temporally distant observations into the coherent representation: video-only errors increase sharply once relevant frames are farther apart, while errors with the spatial graph stay modest across frame-distance quartiles. However, for dynamic states, Figure 4 (b) reveals the difficulty of tracking spatial state during long-term exploration. In the video-only setting, recall declines the further into the exploration a state is observed. Since the updated state appears later in the video, Qwen3-VL-8B recalls the prior state better, reflecting an asymmetry in processing temporal order rather than a preference for prior states. As visual observations accumulate throughout long-term exploration, the model sustains robust recall with the GT graph; SPORE mitigates the degradation without closing the gap.

Bottlenecks in Geometric Grounding. As illustrated in Figures 5 (a, b), the model exhibits limitations in geometric estimation from visual observation, in contrast to the GT graph alone. As the

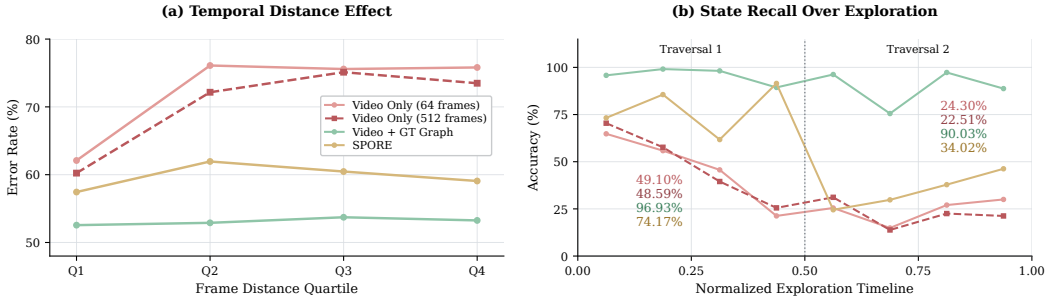

Figure 4: Analysis along the temporal dimension. (a) Error rates on MRS (T2–T4) by quartiles of the frame distance between anchor and target observations. (b) Accuracy on T6 over the normalized exploration timeline. We report average scores for each state.

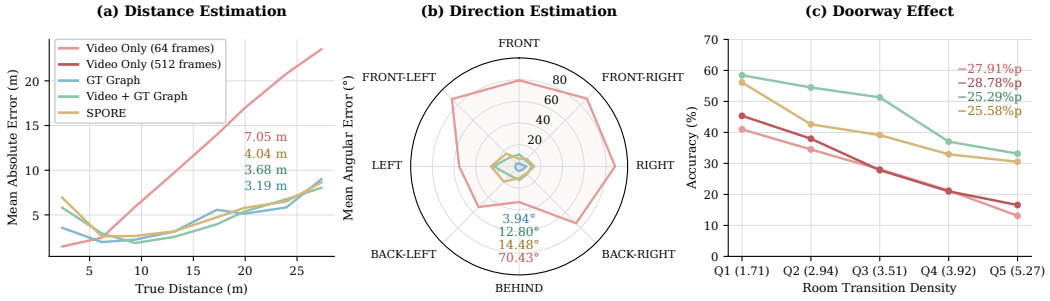

Figure 5: Analysis along the spatial dimension. (a) Mean absolute error (T2) by true distance between the anchor and target objects. (b) Mean angular error (T3) by true direction relative to the reference pose. (c) MRS accuracy (T2–T4) by quintiles of room transition density. We report the mean transitions per quintile and the Q1–Q5 accuracy change.

true distance increases, distance error scales up in video-only, while remaining modest with the spatial graph. Angular error exposes blind spots in directional reasoning: the model struggles to infer relative angles from video alone, whereas the spatial graph supports reliable estimation. However, the model still confuses fine-grained directional terms: angular errors stay low with both Video + GT Graph and SPORE, yet T3 accuracy remains below 50%, indicating difficulty in translating geometric information into linguistic form.

Doorway Effect. Bringing the temporal and spatial dimensions together, we examine the effect of region boundaries through room transition density—the number of room transitions divided by the frame gap between reference and target observations. Figure 5 (c) shows a pattern reminiscent of the "doorway effect", where spatial shifts across room boundaries reduce access to information from the preceding context (Radvansky & Copeland, 2006; Radvansky et al., 2011). As room transitions occur frequently, cross-region spatial reasoning falters even with the spatial graph, which lacks explicitly encoded boundaries or paths. This highlights the fundamental challenge of preserving spatial continuity across distinct boundaries in multi-region environments.

6 Conclusion

We introduced MUSHROOM, a benchmark for long-term spatial reasoning in multi-region environments, requiring VLMs to integrate spatial observations across regions and reason over state changes during exploration. Given the importance of structured representations, we proposed SPORE, a training-free pipeline that constructs an object-centric spatial graph from a single video. Our evaluation reveals that current VLMs have fundamental limitations in long-term spatial reasoning, while SPORE substantially improves both static and dynamic reasoning. Further analysis demonstrates that the spatial graph links temporally distant observations and stabilizes geometric estimation, while pointing to spatial continuity across region boundaries as a remaining challenge for long-term spatial reasoning.

AI USE STATEMENT

In this work, we used generative AI tools for editing the manuscript to improve readability. We have not used generative AI tools to propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, generate synthetic datasets, develop theoretical models or conceptual frameworks, clean or reformat datasets, assist with translation, support qualitative or thematic data analysis, or interpret results; formulating mathematical claims, providing critical ingredients for proofs, and assisting in the writing of proofs are not applicable to this work. Additionally, we used generative AI tools for identifying relevant literature. We have reviewed all AI-assisted work: all text was verified and revised by the authors, and all cited works were read and verified 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.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce our results. The construction of MUSHROOM, including scene selection, traversal recording, object relocation, and question generation, is described in Appendix A. Implementation details of SPORE appear in Appendix B, covering the specific models used for geometry estimation, detection, tracking, captioning, and reranking, together with the key hyperparameters; further settings of SPORE are included in the supplementary code. Evaluation protocols, prompts, and inference settings for all baselines and models are reported in Appendix C. All components of SPORE are off-the-shelf and publicly available, and the pipeline is training-free.

REFERENCES

Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. BoT-SORT: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022.

Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5297–5307, 2016.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Vaibhav Bansal, Stuart James, and Alessio Del Bue. re-obj: Jointly learning the foreground and background for object instance re-identification. In *Image Analysis and Processing – ICIAP 2019*, volume 11752 of *Lecture Notes in Computer Science*, pp. 402–413. Springer, 2019.

Gabriele Berton, Gabriele Trivigno, Barbara Caputo, and Carlo Masone. EigenPlaces: Training viewpoint robust models for visual place recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Satwik Bhattamishra, Kabir Ahuja, and Navin Goyal. On the ability and limitations of transformers to recognize formal languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7096–7116. Association for Computational Linguistics, 2020.

Kevin Black, Noah Brown, James Darpinian, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 2025.

Iva K. Brunec, Morris Moscovitch, and Morgan D. Barense. Boundaries shape cognitive representations of spaces and events. *Trends in Cognitive Sciences*, 22(7):637–650, 2018.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Learning to act anywhere with task-centric latent actions. In *Proceedings of Robotics: Science and Systems*, 2025.

Jinkun Cao, Jiangmiao Pang, Xinshuo Weng, Rawal Khirodkar, and Kris Kitani. Observation-centric SORT: Rethinking SORT for robust multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Nicolas Carion et al. SAM 3: Segment anything with concepts. In *International Conference on Learning Representations (ICLR)*, 2026.

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

Tao Chen, Kun Zhang, Qiong Wu, Xiao Chen, Chao Chang, Xiaoshuai Sun, Yiyi Zhou, and Rongrong Ji. Scaling the long video understanding of multimodal large language models via visual memory mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 31877–31888, 2026.

Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. YOLO-World: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16901–16911, 2024.

Zixu Cheng, Jian Hu, Ziquan Liu, Chenyang Si, Wei Li, and Shaogang Gong. V-STaR: Benchmarking video-LLMs on video spatio-temporal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 9155–9164, 2026.

Erik Daxberger, Nina Wenzel, David Griffiths, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, and Peter Grasch. MM-Spatial: Exploring 3D spatial understanding in multimodal LLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7395–7408, 2025.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. PaLM-E: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8469–8488. PMLR, 2023.

Mengfei Du, Binhao Wu, Zejun Li, Xuanjing Huang, and Zhongyu Wei. EmbSpatial-Bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 346–355. Association for Computational Linguistics, 2024.

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Peihao Wang, Huaizhi Qu, Shijie Zhou, Dilin Wang, Zhicheng Yan, Hongyu Xu, Justin Theiss, Tianlong Chen, Jiachen Li, Zhengzhong Tu, Zhangyang Wang, and Rakesh Ranjan. VLM-3R: Vision-language models augmented with instruction-aligned 3D reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31054–31065, 2026.

Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Junfei Wu, Xiaoying Zhang, Benyou Wang, and Xiangyu Yue. Video-R1: Reinforcing video reasoning in MLLMs. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24108–24118, 2025.

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

Kartik Garg, Sai Shubodh Puligilla, Shishir N. Y. Kolathaya, Madhava Krishna, and Sourav Garg. Revisit anything: Visual place recognition via image segment retrieval. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

Google DeepMind. Gemini 3.5 Flash model card. `https://deepmind.google/models/model-cards/gemini-3-5-flash/`, 2026.

Ankit Goyal, Kaiyu Yang, Dawei Yang, and Jia Deng. Rel3D: A minimally contrastive benchmark for grounding spatial relations in 3D. In *Advances in Neural Information Processing Systems*, volume 33, pp. 10514–10525, 2020.

Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5021–5028, 2024.

Xianda Guo, Ruijun Zhang, Yiqun Duan, Yuhang He, Dujun Nie, Wenke Huang, Chenming Zhang, Shuai Liu, Hao Zhao, and Long Chen. SURDS: Benchmarking spatial understanding and reasoning in driving scenarios with vision language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Agrim Gupta, Piotr Dollár, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

Stephen C. Hirtle and John Jonides. Evidence of hierarchies in cognitive maps. *Memory & Cognition*, 13(3):208–217, 1985.

Daniel Honerkamp, Martin Büchner, Fabien Despinoy, Tim Welschehold, and Abhinav Valada. Language-grounded dynamic scene graphs for interactive object search with mobile manipulation. *IEEE Robotics and Automation Letters*, 9(8):6902–6909, 2024.

Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3D-LLM: Injecting the 3D world into large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Wenbo Hu, Jingli Lin, Yilin Long, Yunlong Ran, Lihan Jiang, Yifan Wang, Chenming Zhu, Runsen Xu, Tai Wang, and Jiangmiao Pang. G^2VLM: 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, 2026.

Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985. doi: 10.1007/BF01908075.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6700–6709, 2019.

Janellen Huttenlocher, Nora Newcombe, and Elisabeth H. Sandberg. The coding of spatial location in young children. *Cognitive Psychology*, 27(2):115–147, 1994.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1988–1997, 2017.

Yuanchen Ju, Yongyuan Liang, Yen-Jen Wang, Nandiraju Gireesh, Yuanliang Ju, Seungjae Lee, Qiao Gu, Elvis Hsieh, Furong Huang, and Koushil Sreenath. MomaGraph: State-aware unified scene graphs with vision-language model for embodied task planning. In *The Fourteenth International Conference on Learning Representations*, 2026. Oral presentation.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9161–9175. Association for Computational Linguistics, 2023.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P. Foster, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2025.

Najoung Kim and Sebastian Schuster. Entity tracking in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3835–3855. Association for Computational Linguistics, 2023.

Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955. doi: 10.1002/nav.3800020109.

Shiqiang Lang, Jing Liu, Haoyang He, Peiwen Sun, Yuanteng Chen, Tao Liu, Lan Yang, Longteng Guo, and Honggang Zhang. LongSpace: Exploring long-horizon spatial memory from perception to recall in video. *arXiv preprint arXiv:2606.05677*, 2026.

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 (ICCV)*, pp. 9241–9251, 2025.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19730–19742. PMLR, 2023.

Xiaoxiao Li and Chen Change Loy. Video object segmentation with joint re-identification and attention-aware mask propagation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 90–105, 2018.

Yun Li, Yiming Zhang, Tao Lin, Xiangrui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. STI-Bench: Are MLLMs ready for precise spatial-temporal world understanding? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5622–5632, 2025b.

Long Lian, Yifan Ding, Yunhao Ge, Sifei Liu, Hanzi Mao, Boyi Li, Marco Pavone, Ming-Yu Liu, Trevor Darrell, Adam Yala, and Yin Cui. Describe anything: Detailed localized image and video captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 21766–21777, 2025.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-LLaVA: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5971–5984, 2024.

Jingli Lin, Runsen Xu, Shaohao Zhu, Sihan Yang, Peizhou Cao, Yunlong Ran, Miao Hu, Chenming Zhu, Yiman Xie, Yilin Long, Wenbo Hu, Dahua Lin, Tai Wang, and Jiangmiao Pang. MMSI-Video-Bench: A holistic benchmark for video-based spatial intelligence. *arXiv preprint arXiv:2512.10863*, 2025a.

Jingli Lin, Chenming Zhu, Runsen Xu, Xiaohan Mao, Xihui Liu, Tai Wang, and Jiangmiao Pang. OST-Bench: Evaluating the capabilities of MLLMs in online spatio-temporal scene understanding. In *Advances in Neural Information Processing Systems*, volume 38, 2025b.

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

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023b.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

Yuhong Liu, Beichen Zhang, Yuhang Zang, Yuhang Cao, Long Xing, Xiaoyi Dong, Haodong Duan, Dahua Lin, and Jiaqi Wang. Spatial-SSRL: Enhancing spatial understanding via self-supervised reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9570–9581, 2026.

Chenyang Ma, Kai Lu, Ta-Ying Cheng, Niki Trigoni, and Andrew Markham. SpatialPIN: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3D priors. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3DSRBench: A comprehensive 3D spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6924–6934, 2025.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Khan. Video-ChatGPT: Towards detailed video understanding via large vision and language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 12585–12602, 2024.

Steve Macenski, Tom Moore, David V. Lu, Alexey Merzlyakov, and Michael Ferguson. From the desks of ROS maintainers: A survey of modern & capable mobile robotics algorithms in the robot operating system 2. *Robotics and Autonomous Systems*, 2023.

Steven Macenski, Francisco Martín, Ruffin White, and Jonatan Ginés Clavero. The marathon 2: A navigation system. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

Steven Macenski, Tully Foote, Brian Gerkey, Chris Lalancette, and William Woodall. Robot operating system 2: Design, architecture, and uses in the wild. *Science Robotics*, 7(66):eabm6074, 2022.

Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. SpatialLM: Training large language models for structured indoor modeling. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Timothy P. McNamara, Jennifer K. Hardy, and Stephen C. Hirtle. Subjective hierarchies in spatial memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15(2):211–227, 1989.

Timothy P. McNamara, J. Sluzenski, and B. Rump. Human spatial memory and navigation. In *Learning and Memory: A Comprehensive Reference*, pp. 157–178. Elsevier, 2008.

Maayan Merhav and Thomas Wolbers. Aging and spatial cues influence the updating of navigational memories. *Scientific Reports*, 9:11469, 2019.

William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023.

Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. In *Advances in Neural Information Processing Systems*, 2023.

Daniel R. Montello and Herbert L. Pick, Jr. Integrating knowledge of vertically aligned large-scale spaces. *Environment and Behavior*, 25(4):457–484, 1993.

NVIDIA. Isaac Sim, 2025. URL https://developer.nvidia.com/isaac/sim/.

OpenAI. Update to GPT-5 system card: GPT-5.2. https://openai.com/index/gpt-5-system-card-update-gpt-5-2/, 2025.

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.

Gabriel A. Radvansky and David E. Copeland. Walking through doorways causes forgetting: Situation models and experienced space. *Memory & Cognition*, 34(5):1150–1156, 2006.

Gabriel A. Radvansky, Sabine A. Krawietz, and Andrea K. Tamplin. Walking through doorways causes forgetting: Further explorations. *Quarterly Journal of Experimental Psychology*, 64(8): 1632–1645, 2011.

Arijit Ray, Jiafei Duan, Ellis Brown, Reuben Tan, Dina Bashkirova, Rose Hendrix, Kiana Ehsani, Aniruddha Kembhavi, Bryan A. Plummer, Ranjay Krishna, Kuo-Hao Zeng, and Kate Saenko. SAT: Dynamic spatial aptitude training for multimodal language models. In *Conference on Language Modeling*, 2025.

Kyujin Shim, Kangwook Ko, Yujin Yang, and Changick Kim. Focusing on tracks for online multi-object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11687–11696, 2025.

Zijian Song, Xiaoxin Lin, Qiuming Huang, Guangrun Wang, and Liang Lin. SIRI-Bench: Challenging VLMs' spatial intelligence through complex reasoning tasks. *arXiv preprint arXiv:2506.14512*, 2025.

SpatialVerse Research Team. InteriorAgent: Interactive USD interior scenes for Isaac Sim-based simulation, 2025. URL https://huggingface.co/datasets/spatialverse/InteriorAgent. Manycore Tech Inc.

Lena Strobl, William Merrill, Gail Weiss, David Chiang, and Dana Angluin. What formal languages can transformers express? a survey. *Transactions of the Association for Computational Linguistics*, 12:543–561, 2024.

Zilu Tang, Qiao Zhao, Gabriel Franco, Derry Wijaya, Aaron Mueller, Sebastian Schuster, and Najoung Kim. Do language models track entities across state changes? In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

Mehmet Kerem Turkcan. Detect anything in real time: From single-prompt segmentation to multi-class detection. *arXiv preprint arXiv:2603.11441*, 2026.

Barbara Tversky. Structures of mental spaces: How people think about space. *Environment and Behavior*, 35(1):66–80, 2003.

Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380, 1991.

Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Niessner. RIO: 3D object instance re-localization in changing indoor environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7658–7667, 2019.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. VGGT-Ω. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Yixuan Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

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

Wenqi Wang, Reuben Tan, Pengyue Zhu, Jianwei Yang, Zhengyuan Yang, Lijuan Wang, Andrey Kolobov, Jianfeng Gao, and Boqing Gong. SITE: towards spatial intelligence thorough evaluation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9058–9069, 2025.

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-MLLM: Boosting MLLM capabilities in visual-based spatial intelligence. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A benchmark for long-context interleaved video-language understanding. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Mingrui Wu, Zhaozhi Wang, Fangjinhua Wang, Jiaolong Yang, Marc Pollefeys, and Tong Zhang. From indoor to open world: Revealing the spatial reasoning gap in MLLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16789–16799, 2026.

Zhijie Yan, Shufei Li, Zuoxu Wang, Lixiu Wu, Han Wang, Jun Zhu, Lijiang Chen, and Jihong Liu. Dynamic open-vocabulary 3D scene graphs for long-term language-guided mobile manipulation. *IEEE Robotics and Automation Letters*, 10(5):4252–4259, 2025.

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

Kaiyu Yang, Olga Russakovsky, and Jia Deng. SpatialSense: An adversarially crowdsourced benchmark for spatial relation recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2051–2060, 2019.

Rui Yang, Ziyu Zhu, Yanwei Li, Jingjia Huang, Shen Yan, Siyuan Zhou, Zhe Liu, Xiangtai Li, Shuangye Li, Wenqian Wang, Yi Lin, and Hengshuang Zhao. Visual spatial tuning. In *European Conference on Computer Vision*, 2026a.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Rob Fergus, Yann LeCun, Fei-Fei Li, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. In *International Conference on Learning Representations (ICLR)*, 2026b.

Runmao Yao, Yi Du, Zhuoqun Chen, Haoze Zheng, and Chen Wang. AirRoom: Objects matter in room reidentification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1385–1394, 2025.

Baiqiao Yin, Qineng Wang, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, Saining Xie, Manling Li, Jiajun Wu, and Li Fei-Fei. Spatial mental modeling from limited views. In *The Fourteenth International Conference on Learning Representations*, 2026.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Jilin Mei, Chunhui Chen, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. From flatland to space: Teaching vision-language models to perceive and reason in 3D. In *Advances in Neural Information Processing Systems*, volume 38, 2025a.

Le Zhang, Jihan Yang, Soundarya Krishnan, Jimit Majmudar, Xiou Ge, Prasoon Puri, Prathamesh Saraf, Shruti Bhargava, Dhivya Piraviperumal, Yinan Ling, Cindy Pan, Hong Yu, Aishwarya Agrawal, and Bo-Hsiang Tseng. From where things are to what they are for: Benchmarking spatial-functional intelligence in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12052–12063, 2026a.

Qifeng Zhang, Kaixiang Huang, Heng Dong, Huang Fang, Junting Chen, Junjie Zhu, Yonghang Chen, Zhiyu Zhang, and Wei Li. GST-Bench: Can VLMs develop global spatial awareness from video? *arXiv preprint arXiv:2608.05747*, 2026b.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025b.

Yi Zhang, Youya Xia, Yong Wang, Meng Song, Xin Wu, Wenjun Wan, Bingbing Liu, Aixue Ye, Hongbo Zhang, and Feng Wen. SSR: Pushing the limit of spatial intelligence with structured scene reasoning. *arXiv preprint arXiv:2603.00409*, 2026c.

Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. ByteTrack: Multi-object tracking by associating every detection box. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

Yiming Zhang, Jiacheng Chen, Jiaqi Tan, Yongsen Mao, Wenhu Chen, and Angel X. Chang. ReVSI: Rebuilding visual spatial intelligence evaluation for accurate assessment of VLM 3D reasoning. In *International Conference on Machine Learning (ICML)*, 2026d.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Zhengyang Liang, Shitao Xiao, Minghao Qin, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. MLVU: Benchmarking multi-task long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13691–13701, 2025.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, 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.

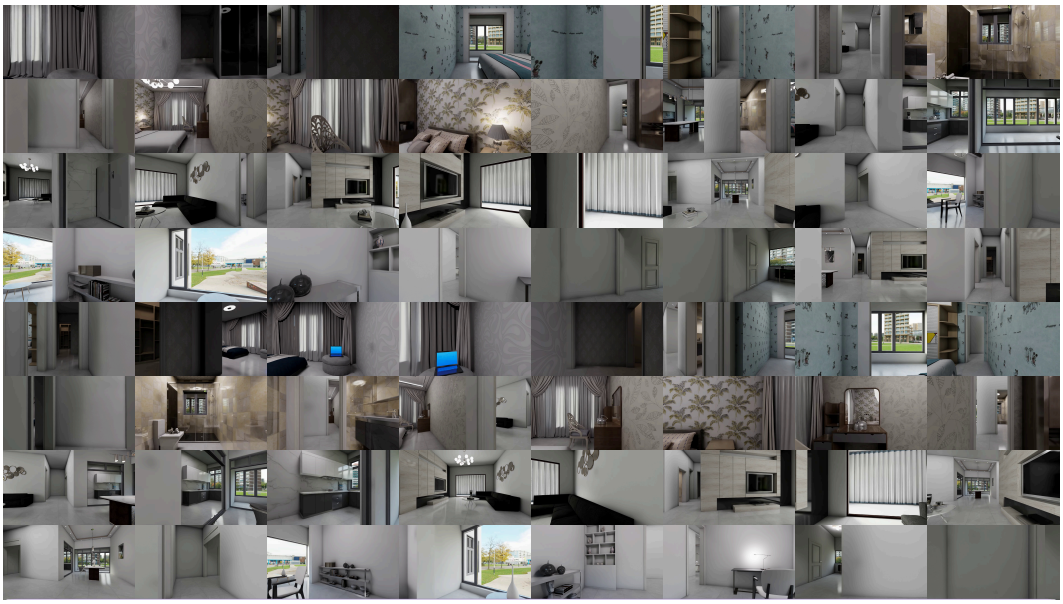

Figure 6: 64 uniformly sampled frames from a representative video in MUSHROOM.

APPENDIX

This appendix expands on the construction of MUSHROOM, the implementation of SPORE, and the analyses underlying our findings:

- § A details benchmark construction, including video generation, question design and validation, and representative examples.
- § B provides implementation details and evaluates spatial graph quality, compares pipeline components, and examines transfer to VSI-Bench and the effect of detection vocabularies.
- § C describes the evaluation protocols, model configurations, and baseline settings.
- § D presents frame-scaling experiments and internal analyses of region representations and spatial state recall in VLMs.

A BENCHMARK DETAILS

A.1 VIDEO CONSTRUCTION DETAILS

Table 4 summarizes the scale and composition of MUSHROOM, including scenes, routes, object relocations, and room transitions. Below, we detail the video capture process, camera configuration, and navigation settings. Figure 6 shows 64 uniformly sampled frames from a representative video.

Capture. A mobile robot drives each scene in NVIDIA Isaac Sim (NVIDIA, 2025) while we render its egocentric camera. The waypoints of a route are fixed in advance, and Nav2 (Macenski et al., 2020) on ROS 2 (Macenski et al., 2022) carries the robot from one to the next. Once the route is finished we relocate the selected objects and send the robot around it again, rendering throughout, so the two traversals arrive as one video. Scenes are rendered at twice their real-world size to give the robot clearance, and the camera and navigation values below are in these simulator units.

Camera. The camera sits 1.95 m above the floor and renders 960×540 frames at 2 fps, without distortion. The intrinsics it renders with,

$$K_{\text{render}} = \begin{bmatrix} 458.12 & 0 & 480.0 \\ 0 & 458.12 & 270.0 \\ 0 & 0 & 1 \end{bmatrix},$$

give fields of view of 92.7° horizontally and 61.0° vertically. SPORE never reads them and estimates its own K from the video instead (Appendix B.1).

Table 4: Summary of video statistics. Each video traverses one route twice, before and after one relocation configuration. Quantities shown with a range are means with min–max. Room and waypoint transitions exclude corridors and grazing contacts.

Item	Value
Indoor scenes	25
Routes per scene	2
Videos	230
Relocated objects per video	3 / 4 / 5
Videos per relocation count	198 / 16 / 16
Relocated objects (total)	738
Annotated objects (total)	396
Rooms (total)	163
Rooms per scene	6.5 (5–9)
Room transitions per traversal	11.3 (7–19)
Waypoint transitions per traversal	12.0 (9–17)
Annotated objects per video	16.4

Navigation. Paths come from NavFn and are followed by DWB (Macenski et al., 2023): NavFn propagates a cost potential from the goal with an A* heuristic (Hart et al., 1968) and descends its gradient, and DWB scores, at 20 Hz, 20 linear and 20 angular velocities simulated 1.7 s ahead over a dynamic window. The costmap is the static occupancy grid alone, at 0.05 m/px, inflated by 0.5 m around a robot radius of 0.05 m. We plan once per waypoint, never replan, and take the pose from the simulator, so a scene rendered twice follows the same trajectory and the camera extrinsics of every frame match the scene coordinates.

Object Relocation. Between the two traversals, we relocate 3–5 selected objects to different rooms while keeping the remaining objects in place. Of the 230 videos, 198 contain three relocated objects, 16 contain four, and 16 contain five, yielding 738 object relocations in total. For each relocated object, we record its position and region assignment before and after relocation. These paired annotations provide the ground truth for identifying relocated objects, recalling their prior and updated states, and reasoning about their displacement and return paths.

A.2 QA-Pair Generation Details

We describe how QA pairs are generated and validated to ensure referent visibility and unambiguous answers. Figures 7 and 8 illustrate representative questions for MRS and MRD, respectively.

Referent Visibility under Frame Sampling We verify that the standard 64-frame setting provides visual access to every object referenced by the evaluation questions. All videos have a resolution of 960×540. Following the visibility criterion used in ReVSI (Zhang et al., 2026d), we determine object visibility from its segmentation-mask coverage. Let F_K denote the set of K uniformly sampled frames and $M_{o,f}$ the visible segmentation mask of object o in frame f. We define

$$\mathrm{Visible}_K(o) = \mathbf{1}\left[\max_{f \in F_K} \frac{\mathrm{Area}(M_{o,f})}{\mathrm{Area}(f)} > 0.03\right]. \tag{1}$$

Thus, an object is considered visible if its segmentation mask occupies more than 3% of at least one sampled frame. Under this criterion, every query object and answer-option object is visible in the standard 64-frame setting, providing complete referent coverage for all evaluation questions.

Validating Distance Queries. For distance queries (T2 and T7), we determine the correct answer through Euclidean distance comparisons using ground-truth object positions. In T2, we measure the distance from the reference object to each answer-option object and select the one that satisfies the queried comparison. The reference object and all answer-option objects belong to mutually distinct regions, ensuring that each question requires cross-region reasoning. In T7, we compare the displacements of relocated objects between their prior and updated positions. When only three or four objects are relocated, we include non-relocated objects as distractors. The prompt explicitly instructs models to compare only relocated objects and disregard those that did not move. For both tasks, we check all options to ensure that only the designated answer satisfies the comparison.

TASK 1: Region Recognition

Question:
Suppose you are in the room containing black sofa. Which of the following descriptions refers to the SAME ROOM?

(A) a room containing white desk chair
(B) a room where white side table is to the front of the black TV
(C) a room containing gray ottoman
(D) a room containing wooden dressing table
(E) a room containing silver fridge

TASK 2: Relative Distance

Question:
Suppose you are standing at the black sofa. Which of the following objects is FARTHEST from you?

(A) white office chair near the black desk
(B) white dining table
(C) toilet
(D) silver fridge near the cutting board
(E) gray ottoman near the navy blue bed

TASK 3: Relative Direction

Question:
Suppose you are standing at the black sofa and facing the TV. Which of the following objects is BEHIND you?

(A) gray ottoman
(B) white dining table
(C) black desk near the office chair
(D) toilet
(E) silver fridge near the cutting board

TASK 4: Static Pathfinding

Question:
If you are standing at the black sofa and facing the TV, which path leads to the white modern office chair by the desk?

(A) turn right, go straight, turn left, go straight, turn left, go straight, turn right, go straight, turn left, go straight
(B) turn left, go straight, turn left, go straight, turn left, go straight, turn right, go straight, turn left, go straight
(C) turn left, go straight, turn right, go straight, turn left, go straight, turn right, go straight, turn left, go straight
(D) turn left, go straight, turn left, go straight, turn right, go straight, turn right, go straight, turn left, go straight
(E) turn left, go straight, turn left, go straight, turn left, go straight, turn around, go straight

Figure 7: Representative questions for MRS: Static Spatial Reasoning (T1–T4), based on the video in Figure 6. Correct answers are shown in **bold**.

Validating Directional Queries. Directional queries (T3 and T8) require interpreting object positions or displacements relative to a specified reference pose. We define this pose by an anchor object's position and a heading toward another object in the same room or nearby. Eight directional categories divide the horizontal plane at 45° intervals: front, front-right, right, back-right, back, back-left, left, and front-left. In T3, the anchor and all answer-option objects belong to mutually distinct regions. We select a target as the correct answer only when its relative angle lies within 15° of the queried direction, excluding targets near category boundaries. For example, a target behind the observer lies within 165°–195° relative to the reference heading. All distractors lie outside the queried directional category. In T8, we compute the displacement vector from the object's prior position to its updated position and assign its direction relative to the reference heading.

Validating Pathfinding Queries. We validate each pathfinding query (T4 and T9) by executing its instruction sequence on the floor plan. Each sequence uses four actions: go straight, turn left, turn right, and turn around. Left and right turns rotate the heading by 90° counterclockwise and clockwise, respectively, while turning around rotates it by 180°. For T9, the route starts at the relocated object's updated position and leads back to its prior position. We use the video's initial heading as a shared directional reference to specify the starting orientation consistently across locations and traversals. Starting from the specified position and heading, we execute each candidate sequence and propagate the reachable area step by step (Figure 9). The correct sequence must reach the target location, while all distractor sequences must fail to reach it.

TASK 5: Relocation Detection

Question:
Which of the following objects was relocated?

(A) gray ottoman
(B) black TV
(C) black shower booth
(D) white table lamp
(E) light gray bed

TASK 6: Relocation Recall

Question:
Before the relocation, where was the white table lamp?

(A) a room containing beige armchair
(B) a room where black shower screen is by the toilet
(C) a room containing white office chair
(D) a room containing light gray bed and shelving unit
(E) a room containing black sofa

TASK 7: Relocation Distance

Question:
Based on the observation, which of the following objects was relocated over a LONGER distance?

(A) white glossy ceramic (B) white table lamp **(C) silver kettle**
(D) blender (E) black laptop

TASK 8: Relocation Direction

Question:
If you are standing at the beige bed and facing the beige armchair, which direction was the white table lamp relocated relative to its original location?

(A) back (B) back-left **(C) right**
(D) front-left (E) front

TASK 9: Dynamic Pathfinding

Question:
Define the video's starting direction as South (0°). From the white glossy ceramic's current location and facing South, which path leads back to its original location?

(A) turn left, go straight, turn left, go straight, turn right, go straight, turn left, go straight, turn left, go straight
(B) turn right, go straight, turn left, go straight, turn right, go straight, turn left, go straight, turn left, go straight
(C) turn left, go straight, turn right, go straight, turn right, go straight, turn left, go straight, turn left, go straight
(D) turn left, go straight, turn left, go straight, turn left, go straight, turn left, go straight, turn left, go straight
(E) turn left, go straight, turn left, go straight, turn right, go straight, turn left, go straight, turn right, go straight

Figure 8: Representative questions for MRD: Dynamic Spatial Reasoning (T5–T9), based on the video in Figure 6. Correct answers are shown in **bold**.

B SPORE

B.1 Implementation Details

Models and Settings. We predict geometry with VGGT-Ω (Wang et al., 2026) and detect objects with OWLv2 (Minderer et al., 2023) at a confidence threshold of 0.20, over the vocabularies in Figure 11: 163 indoor categories on MUSHROOM and 69 on VSI-Bench, each combined with the same 45 categories that belong to neither. TrackTrack (Shim et al., 2025) associates the detections, compensating camera motion with sparse optical flow, Qwen3-VL (Bai et al., 2025a) captions the nodes, and retrieval uses the Qwen3 reranking model (Zhang et al., 2025b). Both distance thresholds are in units of s defined below: the merge radius $\rho = 0.03$ bounds the 3D Euclidean distance between tracklet centroids, and the separation $\delta = 0.13$ the Euclidean distance between the returned nodes on the floor plan, of which at most $m = 2$ are returned per query, one for each position a relocated object can occupy. The region bonus of retrieval is $\beta = 0.20$.

Categories. We group detection labels into coarse categories (Table 5). Tracklets merge only when their dominant coarse categories agree, and retrieval keeps a node when its top-3 coarse categories contain that of the query. We set no minimum vote: the label the detector returns varies from frame to frame, 1.7 categories per tracklet on average and three or more for 14% of them, so dropping the low-ranked categories would drop the correct one from the candidate set. A node whose detections carry fewer than three categories keeps those it has.

Geometry and Lifting. When a video exceeds the input capacity of the geometry model, we apply it over windows of 200 frames that overlap by 120 and bring each window into the coordinates of the first with a Umeyama similarity transform (Umeyama, 1991) between the camera centers of the overlapping frames. We then smooth the camera centers with a moving average over nine frames, leaving the rotations unchanged. The predicted geometry has no metric scale, so we normalize

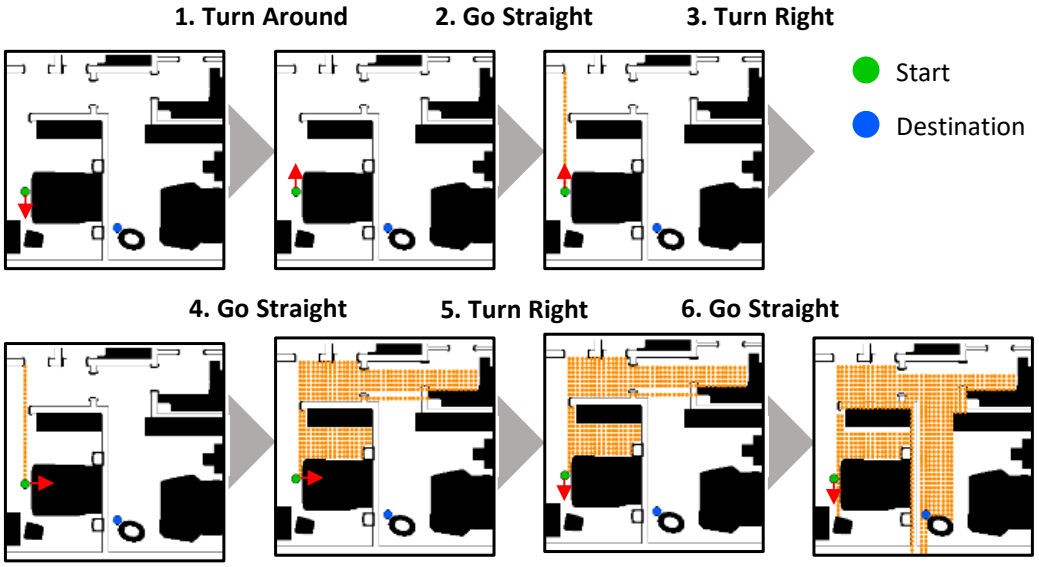

Figure 9: Validating a pathfinding query (T4 and T9). Each panel applies one instruction and plots in orange the points the route can then reach; the arrow is the current heading. An option is kept as the answer only when the final points cover it.

depth and camera translation by s, the diagonal length of the bounding box of the estimated camera centers, writing $D'_t = D_t/s$ and $\mathbf{t}'_t = \mathbf{t}_t/s$; all distances in SPORE, including ρ, are in these units. For a detection box b in frame t, let $W(b)$ be a square window at the box center whose half-width is 15% of the shorter box side; taking the median depth within it limits background contamination. With $[R_t \mid \mathbf{t}_t]$ mapping world coordinates to camera coordinates, the box center $\mathbf{u}_b = [\bar{u}, \bar{v}, 1]^\top$ is back-projected into world coordinates as

$$d = \underset{(u,v)\in W(b)}{\text{median}} D'_t(u,v), \qquad \mathbf{p}_{b,t} = R_t^\top \left(d\, K^{-1} \mathbf{u}_b - \mathbf{t}'_t \right). \tag{2}$$

Clustering and Node Position. The tracklet with the most detections seeds a cluster and absorbs every unassigned tracklet of the same dominant category within ρ; clusters whose centers then remain within ρ are merged. The position of node k is the coordinate-wise median of all detection points in its cluster,

$$\mathbf{c}_k = \underset{(b,t)\in\mathcal{O}_k}{\text{median}}\, \mathbf{p}_{b,t}. \tag{3}$$

Merging compares tracklet centroids in 3D, whereas the relocation rule and evaluation use the top-down projection of $\mathbf{c}_k$ onto the floor plan.

Captioning. We select sixteen observations of the node evenly in time and give the captioner four of them as full frames, with the object's bounding box drawn in orange, together with two tight crops and the prompt below.

Captioning prompt

The object of interest is marked with an orange rectangle in the first images; the last images are close-up crops of that same object. Describe the boxed object concisely: what it is, its color, material, shape, where it sits in the room, and what is next to it. Answer in English, in one paragraph, no preamble.

Retrieval. The cross-encoder scores every candidate the category filter leaves, and the ten highest form $\mathcal{L}$. Since the filter matches the coarse category alone, a modifier such as a color first acts here, where the reranker reads it against the caption. The first returned node is the one of $\mathcal{L}$ with the highest score once a bonus β is added to a node whose region matches the region the queried category implies; a region named in the query itself carries no bonus. The categories that imply a region are toilet and shower for the bathroom; stove, fridge, range hood, and microwave for the kitchen; bed for the bedroom; and sofa for the living room. The second is the first node of $\mathcal{L}$, in

Table 5: Coarse categories and the detection labels mapped to each, for the 163 labels of the MUSHROOM vocabulary; the 45 labels added to every run are grouped by the same table (e.g., couch into sofa). Categories group labels that the detector can confuse, such as laptop and tv, rather than strict semantic classes. Labels in the ungrouped row use the label itself as their coarse category.

Coarse category	#	Detection labels
sculpture	12	sculpture, statue, figurine, bust, sculpture bust, figure, ornament, cross, stuffed animal, star, doll, toy
painting	10	painting, poster, picture frame, wall art, artwork, photograph, photo, print, wall decoration, canvas
tv	9	television, tv, flat-screen tv, screen, monitor, computer, desktop computer, desktop, laptop
cabinet	8	cabinet, sideboard, dresser, wardrobe, nightstand, tv stand, vanity, drawer
seat	8	chair, office chair, lounge chair, armchair, bar stool, stool, ottoman, bench
shower	6	shower head, shower screen, shower door, shower booth, shower panel, bathtub
table	6	table, coffee table, side table, dining table, dressing table, console table
vase	6	vase, jar, jug, pitcher, pot, ceramics
plant	5	potted plant, plant, flower, tree, planter
stove	4	stove, range, cooker, oven
bed	3	bed, mattress, bedding
curtain	3	curtain, blind, partition
lamp	3	lamp, table lamp, floor lamp
cup	3	cup, mug, teacup
toilet	3	toilet, closestool, bidet
washer	3	washing machine, washer, dryer
candle	3	candle, candlestick, menorah
shelf	2	shelf, bookshelf
sofa	2	sofa, sectional sofa
ceiling light	2	ceiling light, chandelier
sink	2	sink, basin
fridge	2	refrigerator, fridge
handbag	2	handbag, backpack
clock	2	clock, alarm clock
range hood	2	range hood, hood
rug	2	rug, carpet
pillow	2	pillow, cushion
blanket	2	blanket, duvet
bottle	2	bottle, wine bottle
plate	2	plate, saucer
teapot	2	teapot, kettle
box	2	box, tissue box
basket	2	basket, trash can
(ungrouped)	36	bath mat, blender, book, bowl, camera, chopstick, countertop, cutting board, desk, detergent, dishwasher, door, door handle, doorway, fork, headboard, keyboard, kitchen counter, knife, laundry basket, magazine, microwave, mirror, pan, railing, remote control, soap dispenser, speaker, spoon, staircase, telephone, toilet paper, towel, tray, wall light, window

reranker order, that lies farther than δ from it. SPORE does not decide whether an object has moved: every query runs the same procedure, and the separation preserves both locations of a relocated object while skipping duplicate fragments near one position.

Algorithm. Algorithm 1 collects these steps, with the notation of Section 4. BUILDGRAPH runs once per video and returns the graph: it estimates the geometry, detects and tracks the objects, back-projects the tracked boxes, clusters the tracklets, and summarizes each cluster as a node. RETRIEVE runs once per query and returns up to m nodes: it filters the nodes by category, ranks them by reranker score plus the region bonus, and adds a second node only when one lies farther than δ from the first.

Algorithm 1 SPORE: spatial graph construction from a single video and query-conditioned retrieval

Require: video $V = (I_1, \dots, I_T)$, detection vocabulary with coarse-category map g, query q
Ensure: up to $m = 2$ nodes as candidate spatial states of the queried object

function BUILDGRAPH(V)
  $K, \{D_t, [R_t \mid \mathbf{t}_t]\}_{t=1}^{T} \leftarrow$ GEOMETRY(V)
  $s \leftarrow$ diagonal of the bounding box of the camera centers; $D'_t \leftarrow D_t/s$, $\mathbf{t}'_t \leftarrow \mathbf{t}_t/s$
  for $t = 1, \dots, T$ **do**
    $\mathcal{B}_t \leftarrow$ DETECT(I_t)
  end for
  $\{\mathcal{T}_i\} \leftarrow$ TRACK($\{\mathcal{B}_t\}_{t=1}^{T}$)
  for all tracklets i **do**
    for all $(b, t) \in \mathcal{T}_i$ **do**
      $\mathbf{p}_{b,t} \leftarrow$ back-project the center of b with D'_t, K, R_t, $\mathbf{t}'_t$ ▷ Eq. (2)
    end for
    $\bar{\mathbf{p}}_i \leftarrow$ coordinate-wise median of $\{\mathbf{p}_{b,t}\}_{(b,t)\in\mathcal{T}_i}$
    $\hat{\kappa}_i \leftarrow$ coarse category $g(\cdot)$ with the highest confidence-weighted vote over $\mathcal{T}_i$
    $w_i \leftarrow$ number of detections in $\mathcal{T}_i$
  end for
  $\{C_k\} \leftarrow$ CLUSTER($\{(\bar{\mathbf{p}}_i, \hat{\kappa}_i, w_i)\}, \rho$) ▷ greedy, heaviest tracklet seeds first
  $\{C_k\} \leftarrow$ MERGECLOSE($\{C_k\}, \rho$) ▷ clusters whose centers stay within ρ
  for all clusters C_k **do**
    $\mathcal{O}_k \leftarrow \bigcup_{i\in C_k} \mathcal{T}_i$
    $\mathbf{c}_k \leftarrow$ coordinate-wise median of $\{\mathbf{p}_{b,t}\}_{(b,t)\in\mathcal{O}_k}$ ▷ Eq. (3)
    $\kappa_k \leftarrow$ top-3 coarse categories by confidence-weighted vote over $\mathcal{O}_k$
    $a_k \leftarrow$ CAPTION(frames and crops of $\mathcal{O}_k$)
    $\omega_k \leftarrow$ INFERREGION(a_k, κ_k, nearby nodes)
  end for
  return $\mathcal{G} = \{N_k = (\mathbf{c}_k, \omega_k, \kappa_k, a_k)\}$
end function

function RETRIEVE($q, \mathcal{G}$)
  $\mathcal{Q}(q) \leftarrow \{k : g(q) \in \kappa_k\}$ ▷ category filter
  $\mathcal{L} \leftarrow$ top-10 of $\mathcal{Q}(q)$ by $R(q, a_k)$ ▷ reranker over every candidate
  $k_1 \leftarrow \arg\max_{k\in\mathcal{L}} R(q, a_k) + \beta\, \mathbf{1}[\omega_k = \omega(q)]$ ▷ region bonus
  $k_2 \leftarrow$ first $k \in \mathcal{L} \setminus \{k_1\}$, in R order, farther than δ from k_1
  return $\{k_1, k_2\}$ ▷ k_2 may not exist
end function

B.2 CONSTRUCTED SPATIAL GRAPH RESULTS

We evaluate the constructed graph directly, independently of any model that reads it, to see how accurately a spatial graph can be built from video alone. We measure where its nodes are placed, how well they match the object descriptions used in queries, and how well they capture relocated objects.

Experimental Setup. A query names an object using its referring expression, and retrieval returns up to two nodes. We count the first as matched when it lies within 1.5 m of the object's ground-truth position on the floor plan, after aligning the graph to the ground truth with a Umeyama similarity transform per traversal (Umeyama, 1991), estimated between the predicted and ground-truth camera centers, which absorbs the arbitrary scale and orientation of a reconstruction built without supervision. The transform is used only for scoring, and graph construction never reads the ground truth. The scenes are rendered at twice their real-world size to give the robot clearance, so we report all distances at half their rendered value. To assess relative configuration, we align only a global scale and rotation, which gives each object a predicted position $\hat{\mathbf{p}}_i$, the horizontal projection of its node position (Eq. 3), to compare with its ground-truth position $\mathbf{g}_i$ on the same floor plan. Writing $\mathbf{v}_{ij} = \mathbf{g}_j - \mathbf{g}_i$ and $\hat{\mathbf{v}}_{ij} = \hat{\mathbf{p}}_j - \hat{\mathbf{p}}_i$, a pair of objects (i, j) yields a distance error and a bearing error,

$$e_{ij}^{\text{dist}} = \frac{\big|\, \|\hat{\mathbf{v}}_{ij}\|_2 - \|\mathbf{v}_{ij}\|_2 \,\big|}{\|\mathbf{v}_{ij}\|_2}, \qquad e_{ij}^{\text{bear}} = \angle\big(\hat{\mathbf{v}}_{ij}, \mathbf{v}_{ij}\big), \tag{4}$$

Table 6: Stage ablation of SPORE over all 230 videos. Detection keeps every detection as a node, tracking keeps one node per tracklet, and merging is the full pipeline. Tokens and time are per video, and duplicates are averaged over annotated objects; relative configuration follows Eqs. (4) and (5). Reloc. counts a query matched only when two returned nodes cover both positions of the relocated object.

	Matching (%)			Relative configuration (%)				Graph			
Stage	**1.5m↑**	**Reloc.↑**	**Oracle↑**	**L–R↑**	**F–B↑**	**Dist.↓**	**Bear. (°)↓**	**Nodes**	**Dup.↓**	**Tokens↓**	**Min↓**
Detection only	68.4	31.9	**98.0**	**75.4**	**80.7**	17.2	**11.4**	22,579	121.85	1.75M	193
+ Tracking	**68.5**	41.0	96.5	74.7	80.3	**17.0**	11.9	1,197	9.04	94.3K	25.1
+ Merging	68.4	**41.8**	96.6	74.7	80.5	17.3	12.0	696	**5.63**	**55.0K**	**17.4**

Table 7: Replacing components of SPORE with ground truth on MUSHROOM over all 230 videos. Box replaces the detected boxes and pose the predicted camera poses.

Ground truth		**Matching (%)**				**Relative configuration (%)**			
Box	**Pose**	**Static↑**	**Reloc.↑**	**Overall↑**	**Oracle↑**	**L–R↑**	**F–B↑**	**Dist.↓**	**Bear. (°)↓**
		74.8	41.8	68.4	96.6	74.7	80.5	17.3	12.0
✓		78.7	53.3	73.7	97.9	77.3	82.0	15.7	10.3
	✓	82.3	52.1	76.4	97.3	75.4	80.9	15.7	11.1
✓	✓	**87.0**	**64.9**	**82.7**	**98.7**	**78.0**	**82.4**	**14.0**	**8.9**

and a triple (i, j, k), viewed from i toward j, is judged correct when the predicted and ground-truth signs agree,

$$\underbrace{\operatorname{sgn}\big(\hat{\mathbf{v}}_{ij} \times \hat{\mathbf{v}}_{ik}\big) = \operatorname{sgn}\big(\mathbf{v}_{ij} \times \mathbf{v}_{ik}\big)}_{\text{left–right: which side of } i \to j}, \qquad \underbrace{\operatorname{sgn}\big(\hat{\mathbf{v}}_{ij} \cdot \hat{\mathbf{v}}_{ik}\big) = \operatorname{sgn}\big(\mathbf{v}_{ij} \cdot \mathbf{v}_{ik}\big)}_{\text{front–back: ahead of or behind } i}, \tag{5}$$

where $\times$ denotes the scalar cross product of two floor-plan vectors; we report the median error over pairs and the accuracy over triples. As a reference, an oracle returns the node closest to the ground-truth position among the nodes of the queried category, and we score that single node by the 1.5 m criterion, for relocated objects as well. A relocated object occupies two positions, and we count such a query as matched only when two distinct returned nodes cover both. We also count the nodes that lie within the same 1.5 m radius of an annotated object and whose top-3 categories contain its category. This duplicate count is the number of nodes retrieval could return and still be scored as matched, and it is ideally one. We restrict it to the annotated targets, 16.4 per video, since the ground-truth graph holds only those; counting every other node as a false positive would penalize every object the ground truth leaves unannotated, favoring a pipeline that detects less.

Construction Accuracy. As shown in Table 6, matching within 1.5 m barely moves across the three stages (68.4%, 68.5%, and 68.4%). This tie says more about the measure than about the stages: a query counts as matched when the returned node is close enough, and detection alone leaves 121.85 nodes of the queried object inside that radius, against 5.63 after merging, so the criterion is easy to satisfy before any aggregation. Tracking and merging instead change the rest of the table. Relocation coverage rises from 31.9% to 41.8% (+9.9%p). The graph also becomes small enough to be read: a video is summarized in 696 nodes and 55.0K caption tokens rather than 22,579 nodes and 1.75M tokens, several times the context window of the model that reads them, and is built in 17.4 minutes rather than 193. Relative configuration is largely unchanged across the three stages. Under scale- and rotation-only alignment, the resulting graph places objects with median errors of 12.0° in bearing and 17.3% in inter-object distance (Eq. 4), and its categorical judgments (Eq. 5) reach 74.7% on left–right and 80.5% on front–back.

Ground-Truth Components. The geometry model and the detector are off-the-shelf components, and improving them is outside the scope of this work. To measure how much each limits the graph, and how much SPORE stands to gain as these components improve, we replace their estimates, the boxes and the camera poses, with ground truth (Table 7). On overall matching, the camera pose is the larger limit: ground-truth poses raise overall matching from 68.4% to 76.4% (+8.0%p) and

Table 8: Detectors on MUSHROOM with TrackTrack and Qwen3-VL captions. DART (Turkcan, 2026) detects several classes per pass with SAM 3 (Carion et al., 2026). (b) Detection speed on one NVIDIA RTX A6000 with FP16 PyTorch; min/video averages over all 230 videos.

(a) Matching

Detector	Thr.	Matching (%) Overall↑	Static↑	Reloc.↑	Oracle↑
OWLv2 (Minderer et al., 2023)	0.20	68.4	74.8	41.8	96.6
	0.15	67.1	73.3	38.0	96.6
	0.10	66.0	71.9	36.0	**96.9**
YOLO-World (Cheng et al., 2024)	0.10	67.1	73.3	**43.4**	92.6
DART (Turkcan, 2026)	0.80	**68.8**	**75.5**	43.3	92.8

(b) Speed

Detector	s/frame	Relative	Boxes/frame	After NMS	min/video
YOLO-World	0.013	1.0×	6.7	5.3	0.2
OWLv2	0.134	10.3×	19.8	18.3	2.1
DART	2.251	173.2×	149.8	68.4	35.4

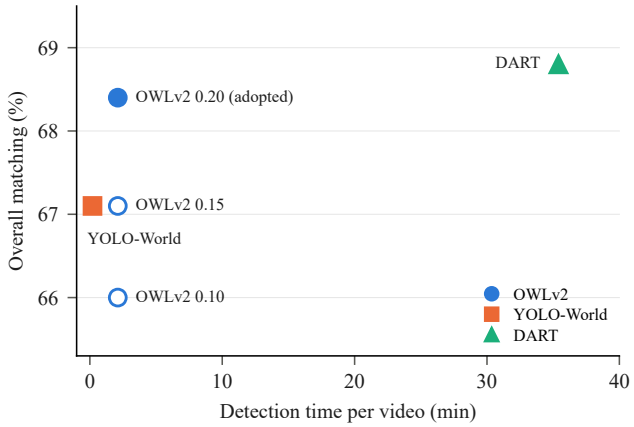

Figure 10: Detection time per video against overall matching for the detectors of Table 8. The filled marker is the adopted setting: OWLv2 at 0.20 matches nearly as many queries as DART in a seventeenth of the time.

ground-truth boxes to 73.7% (+5.3%p), while the two together reach 82.7% and raise relocation coverage from 41.8% to 64.9% (+23.1%p).

B.3 Component Choices

SPORE is assembled from off-the-shelf components for geometry estimation, detection, tracking, and captioning, and none of these tasks is a problem this work sets out to solve. Their accuracy still matters, since the graph inherits the estimates each one supplies (Table 7). We therefore compare the alternatives for each component by how accurate a graph they yield, keeping the other components fixed. These comparisons come from a separate evaluation run, so their numbers are comparable within each of Tables 8–10 but not with Tables 6, 7, and 12.

Detector. As shown in Table 8a, DART (Turkcan, 2026) matches the most queries, but its lead comes from density rather than coverage: it keeps 68.4 boxes per frame after suppression, against 18.3 for OWLv2 (Table 8b), while its lower oracle indicates that fewer of the queried objects enter the graph at all. It is also 17 times slower than OWLv2 per frame, taking 35.4 minutes per video for detection alone instead of 2.1. This does not contradict the real-time rate reported for DART, which is measured with four classes: the cost of its class-conditioned encoder–decoder rises with

Table 9: Trackers on MUSHROOM with YOLO-World detections and Qwen3-VL captions. TrackTrack uses global motion compensation.

	Matching (%)			
Tracker	**Overall↑**	**Static↑**	**Reloc.↑**	**Oracle↑**
TrackTrack (Shim et al., 2025)	**67.1**	**73.3**	**43.4**	**92.6**
BoT-SORT (Aharon et al., 2022)	66.7	73.2	34.1	90.2
ByteTrack (Zhang et al., 2022)	56.4	61.3	19.9	88.4
OC-SORT (Cao et al., 2023)	50.5	55.0	13.7	85.5

Table 10: Captioners on MUSHROOM, on graphs built with YOLO-World. DAM is the Describe Anything Model (Lian et al., 2025), and the last row uses no captioner, describing each node by its label alone.

	Matching (%)		
Captioner	**Overall↑**	**Static↑**	**Reloc.↑**
Qwen3-VL	**67.1**	**73.3**	**43.4**
DAM	57.1	61.7	34.1
None (label only)	41.8	46.6	31.8

the number of classes, and its released code processes a large vocabulary in several passes, five passes of 16 classes for the 80 COCO classes. We process our vocabulary in six passes of about 35 classes. Figure 10 places each detector by detection time and overall matching. YOLO-World is the fastest, but its lower oracle again indicates that more of the queried objects never enter the graph. Within OWLv2, raising the confidence threshold from 0.10 to 0.20 improves relocation coverage from 36.0% to 41.8% ($+5.8$%p), and we adopt OWLv2 at 0.20, which trails the other two detectors on relocation by about 1.5%p but has a higher oracle than both, placing more of the queried objects in the graph, at a seventeenth of DART's detection time.

Tracker. As shown in Table 9, with detections and captions fixed, TrackTrack leads on overall matching, relocation coverage, and the oracle. BoT-SORT comes close on overall matching but covers far fewer relocations (34.1% against 43.4%, -9.3%p), and ByteTrack and OC-SORT fall behind on every measure.

Captioner. The captions carry much of what retrieval relies on: on the same graph, describing each node by its category label alone lowers overall matching from 67.1% to 41.8% (-25.3%p; Table 10). Among the captioners, Qwen3-VL exceeds DAM by 10.0%p overall and 9.3%p on relocation, and we adopt it.

B.4 Evaluation on VSI-Bench

To test whether SPORE relies on properties specific to MUSHROOM, we apply it to VSI-Bench (Yang et al., 2025) and answer its relative distance and relative direction tasks by arithmetic over node coordinates, without a network at answer time. We keep the pipeline and its settings unchanged, replacing only the vocabulary of the detector (Minderer et al., 2023) with the 69 categories of VSI-Bench and the same 45 as before; as on our videos, we estimate the intrinsics rather than calibrate them. As shown in Table 11, the resulting graphs reach 51.4% on relative distance and 77.1% on relative direction, exceeding the majority baseline by 33.8%p and 29.7%p.

B.5 Vocabulary Settings

Each dataset uses its own detection vocabulary (Figure 11), drawn from three sources: the MUSHROOM vocabulary covers every ground-truth label of the simulator and splits some of them into finer categories a detector can separate, the VSI-Bench vocabulary is the one the benchmark itself uses, and the remaining indoor objects come from public vocabularies, mainly LVIS (Gupta et al.,

Table 11: Relative distance and relative direction on VSI-Bench, over the 152 scenes that contain these two tasks. Accuracies use all questions of a task as the denominator, counting as incorrect any that cannot be grounded in the graph. The majority baseline always selects the most frequent answer of a task; for relative distance this is an object name that need not appear among a question's options, so the baseline can fall below random.

Task	Options	Questions	Random	Majority	Ours
Relative distance	4	710	25.0	17.6	51.4
Relative direction	2–4	968	33.9	47.4	77.1
easy	2	217	50.0	51.2	75.6
medium	3	378	33.3	49.7	79.1
hard	4	373	25.0	42.9	75.9
All		1,678	30.1	34.8	66.2

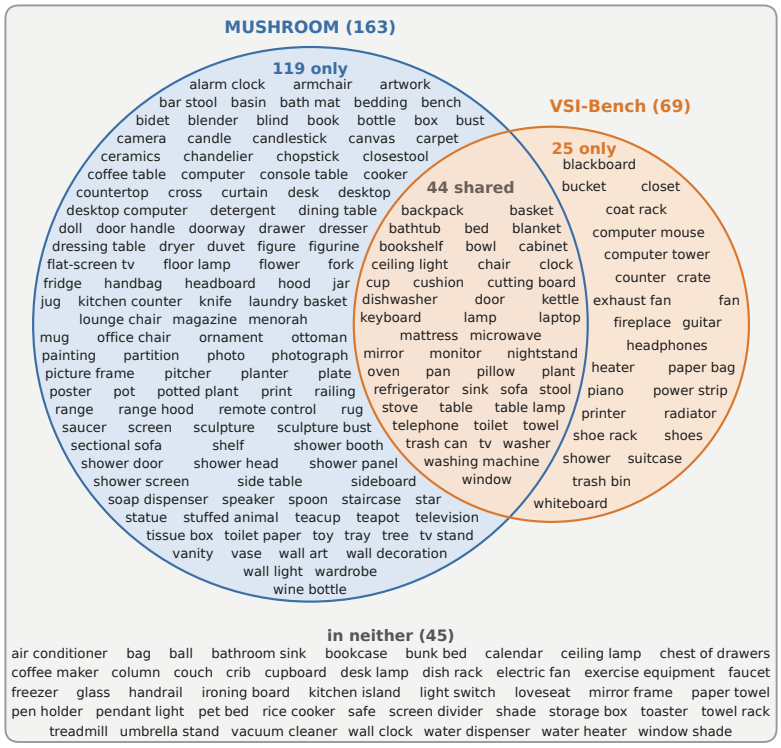

Figure 11: Detection vocabularies of MUSHROOM (163 categories) and VSI-Bench (69), which share 44. Every run also detects the 45 categories that belong to neither, for 208 and 114 labels, and the unified vocabulary of Table 12 adds them to the union of 188, for 233.

2019). Merging the two into a single 233-category vocabulary lets the same pipeline run on both datasets without change, and the graphs remain usable: overall matching on MUSHROOM falls from 68.4% to 66.7% (-1.7%p; Table 12a), and overall accuracy on VSI-Bench from 66.2% to 59.8% (-6.4%p; Table 12b). The other experiments therefore keep the per-dataset vocabularies.

C Evaluation Details

C.1 Experimental Setup

Models. For general VLMs, we benchmark four open-source model families: Qwen2.5-VL (Bai et al., 2025b), Qwen3-VL (Bai et al., 2025a), LLaVA-OneVision (Li et al., 2025a), and InternVL3 (Zhu et al., 2025). For spatial VLMs, we consider two training paradigms: SFT with

Table 12: Per-dataset and unified detection vocabularies (Figure 11) on MUSHROOM (all 230 videos) and VSI-Bench (152 scenes). VSI-Bench accuracies are scored as in Table 11.

(a) MUSHROOM

	Matching (%)				Relative configuration (%)			
Vocabulary	**Static↑**	**Reloc.↑**	**Overall↑**	**Oracle↑**	**L–R↑**	**F–B↑**	**Dist.↓**	**Bear. (°)↓**
Per-dataset (208)	**74.8**	**41.8**	**68.4**	**96.6**	**74.7**	**80.5**	**17.3**	**12.0**
Unified (233)	73.1	40.2	66.7	95.0	73.8	79.6	18.4	13.1

(b) VSI-Bench

	Relative direction (%)				
Vocabulary	**Easy**	**Medium**	**Hard**	**Rel. distance (%)**	**Overall (%)**
Per-dataset (114)	**75.6**	**79.1**	**75.9**	**51.4**	**66.2**
Unified (233)	72.4	70.4	70.0	44.9	59.8

Qwen2.5-VL-SAT (Ray et al., 2025), Cambrian-S (Yang et al., 2026b), and VST-SFT (Yang et al., 2026a); and RL with SpaceR (Ouyang et al., 2025), Video-R1 (Feng et al., 2025), and VST-RL (Yang et al., 2026a). Additionally, we include Spatial-MLLM (Wu et al., 2025), which incorporates a geometry encoder for 3D grounding, along with proprietary models GPT-5.2 (OpenAI, 2025) and Gemini-3.5-Flash (Google DeepMind, 2026).

Input Configuration. We set 64 uniformly sampled frames as the default configuration for all models, since fewer frames risk omitting observations of the queried objects. For the frame scaling analysis, we evaluate at 64, 128, 256, and 512 frames. Proprietary models are evaluated only at 64 frames due to API constraints. Larger models (>70B) and Spatial-MLLM (Wu et al., 2025) are limited to 64 frames due to GPU memory constraints.

Inference Settings. We run inference with a batch size of 1 and a maximum of 4,096 new tokens. Open-source models use greedy decoding (`do_sample=False`, `temperature=0`, `top_p=None`, `num_beams=1`). For proprietary models, we query the official APIs with temperature set to 0, and set `reasoning_effort` to minimal for GPT-5.2 so that all models reason within the same prompt format. We report results from a single run, as decoding is deterministic. Cambrian-S, Spatial-MLLM, and larger models (>70B) are evaluated on two NVIDIA RTX PRO 6000 Blackwell GPUs, and all other models on a single NVIDIA RTX A6000.

Prompts. All settings share the same prompt structure: the task instruction, the spatial graph when provided, a task-specific computation guide, the question with its options, and the answer format. Nodes are provided in JSON format, as shown in Figure 3. The computation guide is included in every setting, including video-only, so that models reason through the same procedure regardless of whether explicit coordinates are available.

Prompt structure

`[Task]`
You are given a video of an indoor environment. Answer the question based on the information presented in the video.

`[Graph]`

`[How to Compute]`

`[Question]`

`[Answer Format]`
Use the following format:
`<think>`(your reasoning here)`</think>`
`<answer>`(your answer here)`</answer>`
Inside `<answer>`, provide only the option letter (A, B, C, D, or E).

Relative Distance (T2)

[Your Position]
Identify the reference object specified in the question.
You are standing at its position (x_ref, y_ref).

[How to Compute]
1. For each candidate object at (x, y), compute:
   distance = sqrt((x − x_ref)^2 + (y − y_ref)^2).
2. FARTHEST = the candidate with the largest distance;
   CLOSEST = the candidate with the smallest distance.

Relative Direction (T3)

[Your Position and Heading]
Identify the reference object at (x_ref, y_ref) and the object you are facing at (x_face, y_face).
Your position is (x_ref, y_ref), and your heading vector is:
   (dx_heading, dy_heading) = (x_face − x_ref, y_face − y_ref).

[How to Compute]
1. Compute your heading:
   theta = (degrees(atan2(dy_heading, dx_heading)) + 180) mod 360.
2. For each candidate object at (x_target, y_target), compute:
   target_bearing = (degrees(atan2(y_target − y_ref, x_target − x_ref)) + 180) mod 360.
3. Compute rel = theta − target_bearing, normalized to (−180, 180].
4. Map rel to a direction using the ranges below.
5. Select the candidate in the direction specified by the question.

[Direction Ranges]
front: −22.5 <= rel < 22.5
front-right: 22.5 <= rel < 67.5
right: 67.5 <= rel < 112.5
back-right: 112.5 <= rel < 157.5
back: rel >= 157.5 or rel < −157.5
back-left: −157.5 <= rel < −112.5
left: −112.5 <= rel < −67.5
front-left: −67.5 <= rel < −22.5

Relocation Distance (T7)

[Object Positions]
Identify each candidate object's prior position (x_before, y_before) and updated position (x_after, y_after).

[How to Compute]
1. Consider only relocated objects; disregard objects that did not move.
2. For each relocated object, compute:
   distance = sqrt((x_after − x_before)^2 + (y_after − y_before)^2).
3. LONGER = the object with the largest relocation distance;
   SHORTER = the object with the smallest relocation distance.

Relocation Direction (T8)

[Your Position and Heading]
Identify the reference object at (x_ref, y_ref) and the object you are facing at (x_face, y_face).
Your position is (x_ref, y_ref), and your heading vector is:
   (dx_heading, dy_heading) = (x_face − x_ref, y_face − y_ref).

[Object Positions]
Identify the queried object's prior position (x_before, y_before) and updated position (x_after, y_after).

[How to Compute]
1. Compute your heading:
   theta = (degrees(atan2(dy_heading, dx_heading)) + 180) mod 360.
2. Compute the object's movement vector:
   (dx, dy) = (x_after - x_before, y_after - y_before).
3. Compute its movement bearing:
   movement_bearing = (degrees(atan2(dy, dx)) + 180) mod 360.
4. Compute rel = theta - movement_bearing, normalized to (-180, 180].
5. Map rel to a direction using the ranges below and select the corresponding option.

[Direction Ranges]
identical to T3

Static Pathfinding (T4)

[Your Position and Heading]
Identify the reference object at (x_ref, y_ref) and the object you are facing at (x_face, y_face).
Your position is (x_ref, y_ref), and your heading vector is:
(dx_heading, dy_heading) = (x_face - x_ref, y_face - y_ref).

[Path Instruction Format]
Each option is a sequence of moves, performed in order, relative to your CURRENT facing direction at that point in the path (not a fixed compass direction):
"turn left" = rotate 90 degrees counterclockwise
"turn right" = rotate 90 degrees clockwise
"turn around" = rotate 180 degrees (reverse direction)
"go straight" = keep moving forward without changing direction

Dynamic Pathfinding (T9)

[Object Positions]
Identify the queried object's prior position (x_before, y_before) and updated position (x_after, y_after).
Start at its updated position with the heading specified in the question; the destination is its prior position.

[Path Instruction Format]
identical to T4

C.2 Additional Evaluation Settings

Model Selection. We select Qwen3-VL-8B and Cambrian-S-7B as the primary baselines for evaluating spatial graphs, based on their leading performance on MUSHROOM within the general and spatial VLM groups, respectively. For detailed analysis, we focus on Qwen3-VL-8B, whose responses more explicitly describe how the supplied graph informs its reasoning, making it easier to examine the role of structured spatial information.

Grid-Map Prompting. We follow the cognitive map prompt of VSI-Bench (Yang et al., 2025), which asks the model to estimate the center position of each object on a 10×10 grid and to output the result in JSON format before answering. As in the original setup, the provided categories are restricted to those mentioned in the question.

Evaluation on VSI-Bench and ReVSI. We evaluate on both benchmarks under the same 64-frame configuration used for MUSHROOM, with identical prompts and inference settings. Unlike Appendix B.4, which assesses the constructed graphs through arithmetic over node coordinates, here we supply the graphs to VLMs and measure how they reason over them. We evaluate the MCQ tasks of both benchmarks: Relative Distance, Relative Direction, and Route Planning. These tasks ask models to express spatial relations in linguistic form, which is the focus of our evaluation.

Table 13: Quantitative result of comparably sized (7B–8B) VLMs on MUSHROOM with 128, 256, and 512 input frames; results at 64 frames are reported in Table 1

Model	Frames	Static Spatial Reasoning				Dynamic Spatial Reasoning					Overall
		T1	T2	T3	T4	T5	T6	T7	T8	T9	
General VLMs											
Qwen2.5-VL-7B	128	51.75	19.57	22.67	15.70	38.50	31.46	16.74	22.75	17.15	26.21
	256	49.50	18.70	23.58	17.77	40.62	32.23	14.71	24.27	16.79	26.59
	512	50.75	19.13	23.22	21.28	42.03	33.38	12.90	21.37	18.80	27.09
Qwen3-VL-8B	128	61.00	26.96	29.43	27.27	61.50	37.72	30.77	17.86	29.74	35.57
	256	57.75	26.38	28.34	29.34	58.11	35.81	31.67	19.08	31.75	35.02
	512	58.50	28.70	30.71	30.17	58.39	35.55	28.73	20.15	30.29	35.44
LLaVA-OneVision-7B	128	60.25	27.54	25.05	21.49	38.08	21.87	23.30	19.54	22.81	27.94
	256	55.00	26.67	24.13	20.45	41.33	19.31	23.08	19.24	21.72	27.13
	512	55.50	27.68	24.68	22.73	36.53	22.38	22.85	18.78	22.45	27.37
InternVL3-8B	128	59.25	27.25	25.96	23.35	40.48	23.91	22.62	20.92	27.37	29.31
	256	58.75	28.41	28.15	21.69	38.93	20.33	21.95	20.92	24.09	28.36
	512	58.50	26.67	26.33	20.87	38.65	21.23	21.27	18.63	23.72	27.56
Spatial VLMs											
Qwen2.5-VL-SAT-7B	128	58.75	30.00	25.41	30.37	37.94	19.95	25.34	21.53	23.54	29.20
	256	59.25	28.99	24.50	28.72	36.95	17.14	24.21	21.98	24.45	28.36
	512	59.75	30.00	23.58	28.72	37.38	15.86	22.17	21.07	23.72	27.94
SpaceR-7B	128	54.25	21.30	19.93	19.21	35.12	30.05	11.31	29.31	15.88	26.23
	256	59.25	18.70	18.10	18.80	38.50	28.77	12.44	28.70	16.24	26.36
	512	60.25	18.70	18.65	17.77	33.57	26.21	10.41	27.02	16.42	25.00
Video-R1-7B	128	57.50	23.62	29.07	22.73	39.49	26.34	26.24	18.32	24.64	28.89
	256	57.00	20.87	28.70	19.63	41.04	30.82	25.57	17.25	24.27	28.82
	512	57.50	20.43	28.34	21.28	43.16	30.95	24.89	16.79	24.27	29.10
VST-SFT-7B	128	53.75	23.04	17.73	24.38	26.23	14.19	21.72	26.87	16.24	23.72
	256	56.00	23.33	14.44	24.38	25.67	11.25	19.68	28.70	18.80	23.40
	512	55.00	21.45	13.16	23.76	25.67	9.08	21.27	27.33	20.99	22.75
VST-RL-7B	128	53.25	21.30	21.02	25.83	23.55	25.96	16.74	20.76	20.44	24.58
	256	52.75	18.70	22.30	22.31	28.77	28.77	15.38	20.61	22.26	25.19
	512	53.25	16.38	22.49	22.31	26.23	21.61	17.42	22.44	25.18	24.23
Cambrian-S-7B	128	57.50	24.93	24.31	17.77	46.69	22.38	23.30	21.37	19.53	28.10
	256	58.50	25.80	23.03	17.98	47.53	23.66	23.98	21.98	19.53	28.61
	512	58.50	27.39	23.03	18.39	47.67	22.76	22.62	21.53	19.71	28.59

D Extended Experiments on MUSHROOM

D.1 Frame Scaling

A critical question is whether observed bottlenecks stem primarily from a scarcity of accumulated visual evidence. Evaluating comparably sized (7B–8B) models across visual inputs from 64 to 512 uniformly sampled frames, Figure 12 illustrates that frame scaling yields saturated performance across general and spatial VLMs; full results are reported in Table 13. On static tasks, denser sampling provides more continuous observations around room transitions, yet fails to consistently improve cross-region spatial reasoning. Although Qwen2.5-VL-SAT, explicitly trained with scene-transition supervision, achieves the highest score among spatial VLMs, frame scaling yields negligible improvements, indicating that additional observations are insufficient. On dynamic tasks, denser sampling cannot overcome limitations in spatial state tracking; under frame scaling, no spatial VLM surpasses Qwen3-VL-8B under the default setting. These results indicate that frame scaling alone is insufficient to resolve bottlenecks in cross-region spatial reasoning and spatial state updating.

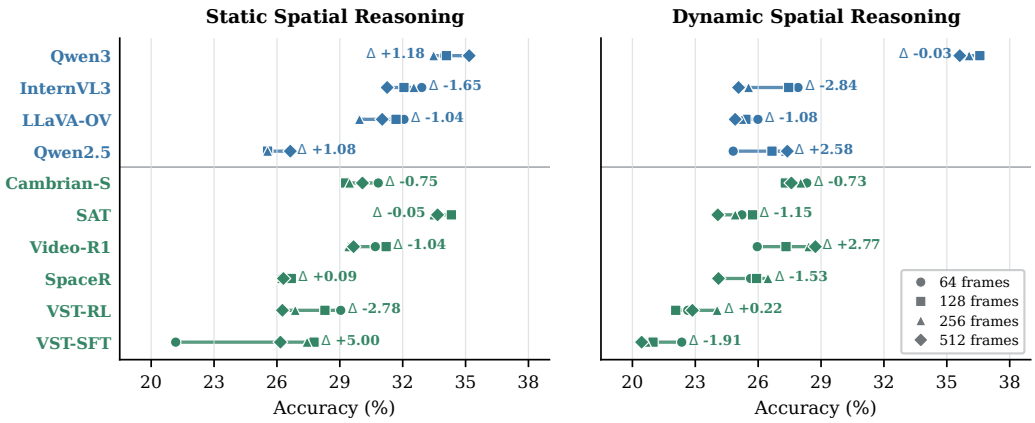

Figure 12: Frame scaling across comparably sized (7B–8B) VLMs. The performance gap Δ denotes the change from 64 to 512 frames.

Table 14: Accuracy (%) with the GT graph. Δ denotes the Overall change from the corresponding video-only 64-frame baseline in Table 1 (%p).

Model	MRS				MRD					Overall	Δ
	T1	T2	T3	T4	T5	T6	T7	T8	T9		
General VLMs											
Qwen2.5-VL-7B	76.50	55.65	31.26	20.45	97.60	76.34	82.58	17.71	20.44	54.06	+28.95
LLaVA-OneVision-7B	75.00	33.62	28.88	22.11	97.46	45.78	42.76	18.93	20.80	43.24	+14.80
InternVL3-8B	70.75	43.91	27.42	21.07	84.77	61.00	50.45	21.68	21.17	45.60	+15.68
Spatial VLMs											
Qwen2.5-VL-SAT-7B	79.50	31.74	37.66	24.38	97.46	45.78	38.69	23.05	28.10	45.39	+16.74
SpaceR-7B	82.00	44.93	25.78	18.18	97.04	74.04	58.37	21.37	12.23	49.44	+23.57
Video-R1-7B	71.25	49.57	31.63	25.21	96.33	74.81	78.28	16.34	15.15	51.85	+23.98
VST-SFT-7B	68.00	32.32	21.57	23.14	94.50	43.86	46.15	41.68	15.88	43.79	+21.91
VST-RL-7B	64.75	34.93	22.30	21.90	97.46	48.34	45.48	21.37	24.64	43.24	+18.02

D.2 The Effect of the Oracle Graph

In the main text, we analyze the GT graph with Qwen3-VL-8B and Cambrian-S-7B. Here we extend this setting to the remaining comparably sized models to test whether the benefit of structured representations holds across VLMs. As shown in Table 14, every VLM improves over its video-only baseline, with gains ranging from 14.80 to 28.95%p. The improvements concentrate on dynamic tasks, where relocation detection (T5) exceeds 84% for all models, whereas cross-region pathfinding (T4, T9) remains near the video-only level. This pattern mirrors our findings in the main text: spatial graph resolve what the video leaves implicit, while paths between regions remain outside what the graph encodes.

D.3 Prior–Updated Recall Gap

Figure 13 shows that the prior–updated gap observed with Qwen3-VL-8B holds across models: all but one favor the prior state, with gaps reaching 43.48%p for Cambrian-S-7B. GPT-5.2 is the sole exception, recalling the updated state more accurately (−12.20%p). The gap appears in both general and spatial VLMs, indicating that the difficulty of tracking updated states is shared across model scales and training approaches.

D.4 Internal Analysis of Qwen3-VL-8B

We analyze Qwen3-VL-8B, the strongest model in video-only evaluation, to better understand its limitations in long-term spatial reasoning. We focus on T1 and T6 because they directly probe

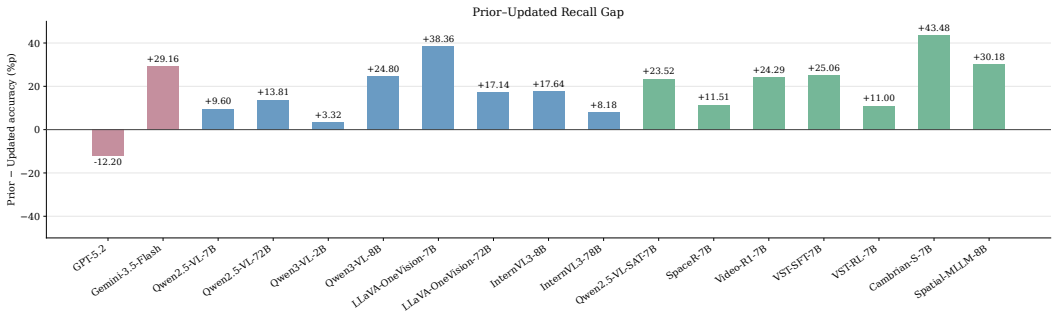

Figure 13: Prior–updated recall gap on T6, measured as the accuracy difference between queries about the prior and updated states.

the retrieval of object–region associations from stable and changing spatial states, respectively. We first test whether the model preserves region identity in its frame representations, a prerequisite for integrating observations across regions, and then examine how it retrieves observations from the two traversals.

Region Clustering. We evaluate whether Qwen3-VL-8B frame embeddings preserve region identity. For a scene containing n frames and K ground-truth regions, let z_i denote the embedding of frame i and $y_i \in \{1, \ldots, K\}$ its region label. We apply K-means with oracle K:

$$\min_{\mu, c} \sum_{i=1}^{n} \|z_i - \mu(c_i)\|_2^2 , \tag{6}$$

where c_i is the predicted cluster assignment. The *Full* setting pools frames from both traversals, whereas *Traversal 1* and *Traversal 2* are clustered separately. Because cluster indices are permutation invariant, we align predicted clusters with region labels using Hungarian matching (Kuhn, 1955) and report accuracy as

$$\text{Acc.} = \frac{1}{n} \max_{\pi \in \mathcal{S}} \sum_{a=1}^{K} N_{a,\pi(a)}, \tag{7}$$

where $N_{ab} = \sum_i \mathrm{I}[y_i = a \wedge c_i = b]$ and $\mathcal{S}$ is the set of permutations over the K labels. We further report normalized mutual information (NMI) and the adjusted Rand index (ARI) (Hubert & Arabie, 1985). NMI measures the information shared between region labels and cluster assignments, while ARI measures pairwise agreement corrected for chance.

Region Retrieval. We examine final-token attention to frames associated with the region queried in each question. Let $\mathcal{Q}_g$ denote the queries in group $g \in \{\mathrm{T1}, \mathrm{T6\text{-}prior}, \mathrm{T6\text{-}updated}\}$, $F_{\text{region}}(q)$ the frames associated with the queried region for query q, and $\alpha_f(q)$ the final-token attention assigned to frame f. We normalize each frame position within the two-traversal video to $x_f \in [0, 1]$, with the midpoint separating the traversals, and divide this timeline into equal intervals I_j. For each interval, we compute

$$A_{g,j} = \text{mean}\{\alpha_f(q) : q \in \mathcal{Q}_g,\ f \in F_{\text{region}}(q),\ x_f \in I_j\}. \tag{8}$$

We compute the resulting attention profiles separately for T1 queries and for T6 queries about the prior and updated states. This comparison shows whether retrieval from the queried region changes with the requested spatial state or remains dominated by its position in the video.

Region Identity and State Retrieval. Figure 14 (left) shows that frame embeddings contain region-discriminative information. Clustering accuracy remains similar across the full video and individual traversals, whereas NMI and ARI are consistently higher when each traversal is clustered separately. Region identity is therefore distinguishable within each traversal, but its representation becomes less consistent when observations from repeated visits are pooled. Figure 14 (right) illustrates that the model retrieves spatial information well when it appears near the beginning or end of the video. This pattern is consistent with the "lost-in-the-middle" (Liu et al., 2024). In T1, the same static state is observed in both traversals. When one observation appears near the middle of the sequence, the corresponding observation from the other traversal can compensate for its limited attention. In T6, each traversal captures a different spatial state: Traversal 1 contains the prior state, whereas

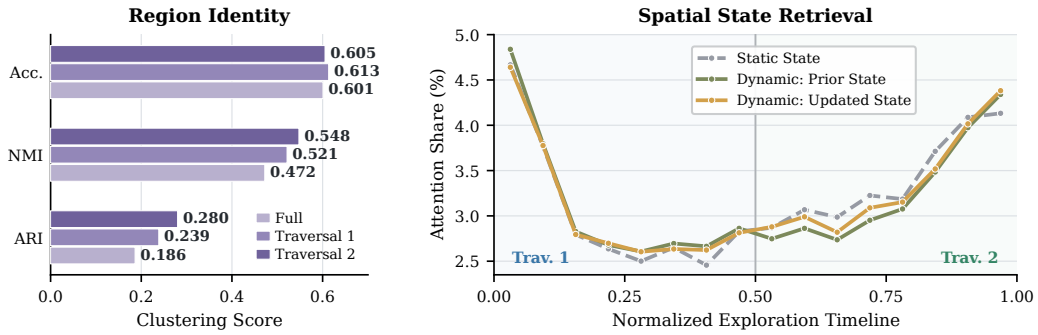

Figure 14: Internal analysis of Qwen3-VL-8B. Left: Clustering of frame embeddings with oracle K evaluates whether region identity remains distinguishable across the full video and within each traversal. Right: Final-token attention to frames associated with the queried region over the normalized exploration timeline for T1 and T6 queries; the vertical line separates the two traversals.

Traversal 2 contains the updated state. This prevents information from one traversal from substituting for the other. Similar patterns for prior- and updated-state queries indicate limited retrieval of the requested state. Together, the analyses reveal a gap between representation and retrieval: region identity remains partially separable in frame embeddings, yet access to the corresponding observations remains sensitive to their temporal position.

D.5 Limitations and Future Work

MUSHROOM is built in a simulated environment, which allows precise control over object relocations and ground-truth annotations but does not capture the full variability of real-world recordings; we address this by evaluating on VSI-Bench and ReVSI. SPORE relies on off-the-shelf components for geometry estimation and detection, and its graph quality is bounded by their accuracy: replacing predicted camera poses and boxes with ground truth raises overall node matching from 68.4% to 82.7% (Appendix B). Finally, the object-centric graph encodes object positions and region assignments but not boundaries or traversable paths, which limits cross-region pathfinding as room transitions grow denser. Extending the representation to preserve spatial continuity across region boundaries, rather than only where objects lie within them, is a natural next step.