SPATIALMEM: OBJECT-LEVEL 3D MEMORY BOOSTS SPATIAL UNDERSTANDING FROM MONOCULAR VIDEOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatial understanding from monocular video requires locating objects in 3D and recognizing the same object across different views. Predicted depth and camera poses place observations in a common 3D coordinate system, but connected regions in the reconstruction do not always correspond to individual objects. Reconstruction artifacts may join geometry from neighboring objects, while occlusion may leave a single object represented by disconnected fragments. These ambiguities can compromise the semantic and geometric consistency of scene representations, undermining spatial understanding. To this end, we introduce SPATIALMEM, a framework that constructs reusable object-level 3D memory from monocular video by combining predicted geometry with cross-view visual evidence. The system associates semantic masks across views through the reconstructed geometry and uses visual evidence to determine which observations belong to the same object. It stores each instance with its observed geometry and links to supporting views. The memory supports both read-only spatial programs and a lightweight latent-injection interface to a frozen vision-language model. Using up to 300 sampled RGB frames per video for memory construction, SPATIALMEM achieves a seven-task average of 65.3% on ReVSI, the highest among the compared methods. In exploratory evaluations, latent features extracted from this memory also improve exact-match accuracy from 53.2% to 58.2% on SQA3D and from 28.6% to 30.6% on ScanQA. Our code will be released at `https://anonymous.4open.science/r/SpatialMem-7E3D/`.

1 INTRODUCTION

Spatial understanding from monocular video is a core capability for multimodal large language models (MLLMs) that reason about the physical world (Yang et al., 2025; Fan et al., 2026). It requires recognizing objects, retaining their locations, and understanding their dimensions and spatial relationships across viewpoints. For embodied agents, such knowledge supports target localization, navigation planning, and reasoning about object interactions (Gu et al., 2024). Monocular RGB video provides an accessible source of these observations, making spatial understanding from ordinary cameras valuable for both multimodal assistants and embodied systems.

Recent reconstruction models recover shared 3D geometry directly from RGB observations (Wang et al., 2024; 2025b; Lin et al., 2026). This progress supports geometry-aware vision-language models (VLMs) that incorporate spatial information into learned representations (Wu et al., 2025; Fan et al., 2026; Wu et al., 2026). A complementary approach organizes observations into explicit scene memory, making objects and geometry available for repeated spatial queries (Gu et al., 2024; Bharadwaj et al., 2026). However, recovering a shared coordinate system does not by itself specify which observations belong to each physical object.

This distinction matters when memory is constructed from predicted geometry. Reconstruction artifacts may connect the reconstructed geometry of neighboring chairs, while occlusion may leave one chair represented by disconnected fragments. Treating these connected or fragmented regions as objects can corrupt both enumeration and geometric measurement. Our goal is to build memory

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.

that distinguishes individual objects despite these reconstruction errors and retains each object's observed geometry for spatial measurement.

We introduce SPATIALMEM, a reusable object-level 3D memory constructed from monocular video without measured depth or reference camera poses. A **Lifter** associates predicted geometry with semantic masks (Lin et al., 2026; Carion et al., 2026), and an evidence-guided **Refiner** separates merged candidates, verifies categories, and consolidates fragmented identities. The memory retains persistent instances, observed geometry, and supporting views (Figure 1). Read-only spatial programs reuse these records across questions, while a complementary learned interface supplies object summaries to a frozen VLM.

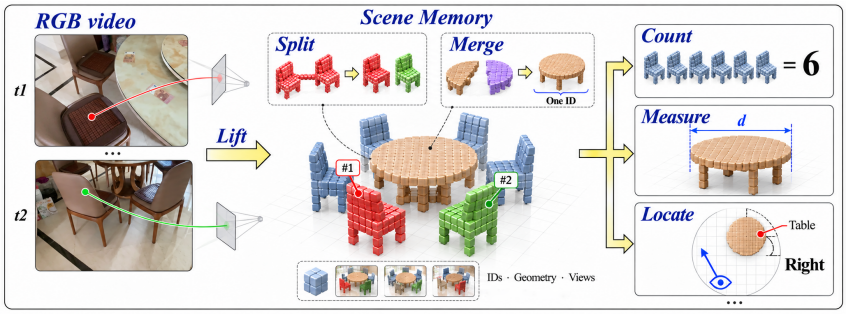

Figure 1: **From monocular observations to reusable object-level 3D memory. Left:** Predicted geometry associates observations across views. **Center:** Cross-view evidence separates different objects and consolidates fragments of the same object. Each instance retains its observed geometry and supporting views. **Right:** The memory supports counting, measurement, and relative-position queries. RGB frames, occupancy, and count are from Figure 5(a); metric and directional readouts are schematic.

Our contributions are threefold. **(1) Reusable object-level 3D memory.** We expose persistent objects, geometry, and visual evidence for explicit spatial computation and learned question answering. **(2) Evidence-guided identity refinement.** We separate candidate partitioning, category verification, and identity consolidation, and analyze their effects on enumeration and retained geometric evidence. **(3) Leading spatial understanding performance.** Using up to 300 sampled RGB frames per video for memory construction, SPATIALMEM achieves a seven-task average of 65.3% on ReVSI (Zhang et al., 2026a), the highest among the compared methods. Its object summaries also improve exact-match accuracy on SQA3D and ScanQA (Ma et al., 2023; Azuma et al., 2022) over the memory-free backbone.

2 RELATED WORK

We organize prior work into three broad, complementary routes according to how spatial information is made available to the reasoning model.

Spatial understanding directly from video frames. The first route uses video MLLMs to answer spatial questions directly from encoded RGB frames. General-purpose models such as LLaVA-Video and Qwen3-VL encode video observations and generate textual answers through multimodal instruction following (Zhang et al., 2025; Bai et al., 2025). Building on such backbones, Visual Spatial Tuning (VST) improves spatial perception and reasoning through spatial instruction tuning and reinforcement learning while retaining the standard VLM architecture (Yang et al., 2026a). SpaceR combines spatially guided reinforcement learning with cognitive-map generation within the reasoning trace to strengthen video spatial reasoning (Ouyang et al., 2025). VSI-Bench highlights the difficulty of recovering spatial relationships from sequential observations (Yang et al., 2025). These approaches infer scene structure within video-to-text prediction, without maintaining an externally reconstructed 3D scene memory for geometric queries.

Injecting 3D information into model representations. The second route strengthens spatial representations through geometric features or 3D supervision. Spatial-MLLM fuses semantic features

with features from a geometry encoder, while VLM-3R aligns reconstruction-derived 3D tokens with language instructions (Wu et al., 2025; Fan et al., 2026). SpaceMind++ builds a voxelized cognitive map and fuses its spatial information back into visual features (Gu et al., 2026). SpatioLM uses pseudo-depth and camera supervision to train a spatial module without requiring extra 3D inputs at inference (Wu et al., 2026). These methods primarily use geometry to improve the model's internal computation; even when an intermediate map is constructed, the model consumes its information through learned representations.

Reasoning over explicitly constructed spatial structures. The third route constructs explicit scene representations for language-model reasoning. Language-aligned 3D fields (Peng et al., 2023; Jatavallabhula et al., 2023) and cross-view mask propagation (Cen et al., 2023) connect semantics with geometry. Multi-view feature aggregation and mask fusion (Takmaz et al., 2023; Lu et al., 2023; Nguyen et al., 2024), together with graph-based view consensus (Yan et al., 2024), organize observations into object-level representations. ConceptGraphs further encodes object relations in open-vocabulary 3D scene graphs (Gu et al., 2024).

Such structures support visual prompting and tool-based reasoning. GPT4Scene uses global–local correspondence and consistent object markers (Qi et al., 2026), while Think3D and R3D provide tools for 3D exploration and spatial computation (Zhang et al., 2026b; Horton et al., 2026). Following the broader principle of executable visual reasoning (Gupta & Kembhavi, 2023; Surís et al., 2023), Flame3D and S-Agent ground tool-based reasoning in scene memory (Bharadwaj et al., 2026; Dai et al., 2026). Within this route, SPATIALMEM constructs reusable object-level memory from predicted monocular geometry, refining identities through cross-view visual evidence while retaining observed surfaces and supporting views. The memory supports both read-only spatial programs and learned readout, linking construction-time identity refinement to downstream spatial reasoning.

3 METHOD

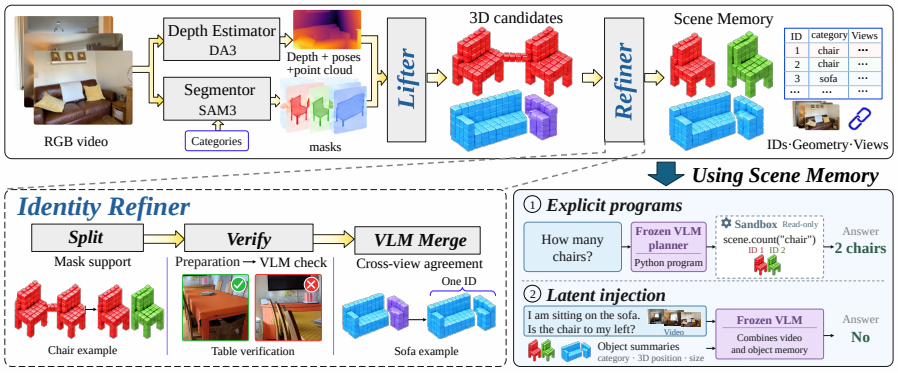

Figure 2: **Overview of SPATIALMEM. (1)** The **Lifter** associates predicted geometry and semantic masks to form 3D candidates. **(2)** The **Refiner** splits candidates, verifies their categories, and merges candidates representing the same object. Each instance retains its observed geometry and supporting views. **(3)** For each question, the system reuses cached object records and constructs records for newly requested categories before readout. **Explicit programs** query the memory through read-only spatial operations, while **latent injection** supplies object summaries to a VLM. Intermediate 3D scenes are schematic.

Motivation. A reusable scene memory should record both which objects are present and the spatial evidence associated with them. Predicted geometry places observations in a common 3D coordinate system, but geometric connectivity alone does not reliably establish object identity. Our design therefore uses geometry to associate observations and cross-view visual evidence to refine their object assignments, retaining the observed surfaces and supporting views of the resulting instances.

Overview. Given sampled RGB frames $\mathcal{V} = \{\mathbf{I}_t\}_{t=1}^{T}$ of a static scene, SPATIALMEM follows the **Lifter**–**Refiner**–Scene Memory pipeline in Figure 2. Shared geometry is reconstructed once.

For each question, we derive coarse categories from its object descriptions, reuse cached object records, and construct records only for categories not yet processed. The **Lifter** associates their image masks with the shared geometry, and the **Refiner** splits, verifies, and merges the resulting candidates into object instances. These records are cached before readout and reused by subsequent questions. Explicit spatial programs query the resulting memory, while latent injection uses its object summaries.

3.1 LIFTER: FROM OBSERVATIONS TO 3D CANDIDATES

DA3 predicts depth, confidence, camera intrinsics, and poses from the RGB frames (Lin et al., 2026), using its metric branch for scale without measured depth, scanned geometry, or reference trajectories. After aligning reconstruction chunks, we fuse reliable observations into voxel occupancy $\mathcal{O}$ and preserve frame–pixel–voxel associations $\mathcal{A}$, linking the fused geometry back to its image evidence. For each requested category c, SAM3 supplies masks independently in each frame (Carion et al., 2026). A frame may contain multiple masks for the same category; their frame-local IDs do not establish identities across views. The associations $\mathcal{A}$ map each mask to its voxel support. A voxel is assigned to c when target votes outnumber non-target votes among its tracked observations. Spatial connectivity groups these voxels into initial 3D candidates. Appendix A provides reconstruction, association, and sampling details.

3.2 REFINER: RESOLVING OBJECT IDENTITY

The initial candidates group nearby geometry, but need not correspond to individual objects. The **Refiner** separates candidates using cross-view mask support (**Split**), checks their categories in selected images (construction-time **Verify**), and consolidates surviving fragments that depict the same object (post-verification **VLM Merge**).

Mask-support graph splitting (Split). Two adjacent chairs may become connected in 3D while remaining separable in image masks. **Split** therefore groups mask observations by shared 3D support rather than treating geometric connectivity as object identity. Within each candidate G, a node $n = (t, j)$ represents mask j in frame t, with voxel support $S_n \subseteq G$ retrieved through $\mathcal{A}$. Their agreement is measured by

$$w(n, n') = \frac{|S_n \cap S_{n'}|}{\min\big(|S_n|, |S_{n'}|\big)}. \tag{1}$$

Normalizing by the smaller support allows a partial observation to agree strongly with a more complete one. Nodes are connected when $|S_n \cap S_{n'}| \geq m_\cap$ and $w(n, n') \geq \tau_w$. Graph components define sub-instance candidates, and cross-frame voting assigns voxels to them. Geometrically connected regions can therefore be separated when their mask observations form distinct connected components in the support graph. Graph settings and limitations are given in Appendices A.3 and A.5.

Construction-time visual verification (Verify). **Split** can leave fragmented or noisy candidates. The **Verify** stage prepares these candidates before a VLM category check. During preparation, early symbolic merging consolidates adjacent candidates from different initial components when their valid shared views show strong same-mask agreement. This restricted step prepares local fragments for the category check. Geometric cleanup then deduplicates overlapping support and filters small components and noise slivers (Appendix A.3).

For each prepared candidate, retained associations identify the most-overlapping mask in each frame. We select at most three representative views using candidate support, mask confidence, and viewpoint diversity, and provide the VLM with original and candidate-highlighted images. The VLM judges whether the highlighted candidate represents the intended coarse category, rather than whether that category appears elsewhere in the image. An explicit *No* removes the entire 3D candidate.

Post-verification identity merging (VLM Merge). Category-valid candidates may still depict the same object. Whereas **Split** links mask observations by voxel overlap, **VLM Merge** checks whether distinct candidates associate with the same image mask, even when their voxel supports do

not overlap. Unlike the early merge, this stage is not restricted by adjacency or initial-component origin. For a pair (a, b), shared-mask agreement is $\mathrm{agree}(a, b) = N_{ab}^{\mathrm{same}}/N_{ab}^{\mathrm{candidate}}$, where the denominator counts frames with valid mask associations for both candidates, and the numerator counts the subset in which both candidates are associated with the same mask. We accept a merge at agreement ≥ 0.8 and reject it at agreement ≤ 0.2. For intermediate or unavailable agreement, a VLM uses at most three diverse views to judge whether the candidates represent the same object. Union–find consolidates accepted pairs and unions their existing voxel support without adding new observations. **VLM Merge** and view-selection settings are detailed in Appendix A.3.

Resulting scene memory. The scene memory contains the shared geometry and object records for all categories processed so far:

$$\mathcal{M} = \big(\mathcal{O}, \mathcal{A}, \{(i, c_i, G_i)\}_{i=1}^{N}, \mathcal{S}\big), \tag{2}$$

where each instance has an ID i, category c_i, and observed voxel support G_i. The scene table $\mathcal{S} = \{(i, c_i, \mathbf{p}_i, \mathbf{s}_i, \mathbf{r}_i)\}_{i=1}^{N}$ summarizes its center, axis-aligned size, and observation support (Appendix A.1). The table and voxel records share instance IDs, providing compact summaries and detailed geometric support for the same stored objects. This memory augments VLM spatial understanding in two complementary ways: generated programs that call its spatial interfaces (Section 3.3) and latent injection of its object summaries (Section 3.4) for questions that are difficult to decompose into explicit spatial operations.

3.3 Explicit Spatial Programs

A frozen Qwen3-VL-32B (Bai et al., 2025) planner translates each question into a Python program, executed in a restricted sandbox with read-only spatial operations. The planner interprets object descriptions and relations; the executor resolves the corresponding candidates and computes quantities from stored identities, geometry, and linked views without modifying persistent memory.

Counting enumerates stored instances $\mathcal{I}_c = \{i : c_i = c\}$. For finer descriptions, query-time **Fine Verify** uses the linked views to screen candidate instances. Only candidates receiving an explicit *No* are excluded from the current count; stored instances remain unchanged. Distance queries read selected support sets P_A, P_B and compute $d(P_A, P_B) = \min_{\mathbf{x}\in P_A,\, \mathbf{y}\in P_B} \|\mathbf{x} - \mathbf{y}\|_2$; relative distance ranks candidate references by this quantity. Direction and landmark-sequence queries establish the question's reference frame from stored positions. Details are provided in Appendix A.6.

3.4 Latent Injection

Explicit programs express a question as a sequence of spatial operations, but questions requiring joint semantic and spatial reasoning can be difficult to decompose this way. We therefore introduce a complementary interface that injects object summaries from $\mathcal{S}$, rather than raw voxel geometry, into a VLM whose backbone is frozen by default.

A trainable row encoder maps category, center, size, and observation-support fields to memory vectors, stacked as $\mathbf{M} = \mathrm{Enc}_{\boldsymbol{\theta}}(\mathcal{S})$. The VLM retains its video and question inputs, including a situation description when provided. After decoder layer ℓ, its hidden states $\mathbf{H}^{(\ell)}$ read the memory through gated cross-attention (Alayrac et al., 2022):

$$\begin{aligned} \mathbf{A}^{(\ell)} &= \mathrm{softmax}_{\mathrm{row}}\left(\frac{(\mathbf{H}^{(\ell)}\mathbf{W}_Q)(\mathbf{M}\mathbf{W}_K)^\top}{\sqrt{d}} + \mathbf{B}\right), \\ \widetilde{\mathbf{H}}^{(\ell)} &= \mathbf{H}^{(\ell)} + \tanh(g)\, \mathbf{A}^{(\ell)}(\mathbf{M}\mathbf{W}_V)\mathbf{W}_O. \end{aligned} \tag{3}$$

Here, decoder states supply queries, while memory rows supply keys and values; d is the attention dimension and $\mathbf{B}$ masks padded entries. The projection matrices and gate g are shared across decoder layers. Initializing $g = 0$ leaves the hidden states unchanged at initialization; training then learns how strongly the retrieved scene information contributes to the decoder states.

We train only the row encoder and shared readout, using answer supervision and tables synthesized by corrupting training-scene GT object boxes. To encourage use of scene content, a targeted row-intervention objective favors the original training table over one with altered target rows when predicting the same answer. At inference, the readout uses tables constructed from predicted video geometry. Appendix B details the row encoding and training protocol.

4 EXPERIMENTS

We evaluate SPATIALMEM primarily on ReVSI through explicit spatial programs, examining how identity refinement affects object enumeration and geometric queries. SQA3D and ScanQA provide complementary evaluations of learned scene-table reading.

4.1 EXPERIMENTAL SETUP

Benchmarks and metrics. ReVSI (Zhang et al., 2026a) revises VSI-Bench (Yang et al., 2025) with video-verified annotations for evaluating object counting, geometric measurement, and spatial relations. We use its all_frame evaluation subset, comprising 6,808 questions across 380 scenes and seven spatial tasks, while constructing memory from at most 300 sampled RGB frames per video. We report mean relative accuracy (MRA) for numerical tasks and accuracy for multiple-choice tasks, following official task-macro aggregation with failed answers assigned zero credit (Appendix A.7). For situated question answering on SQA3D (Ma et al., 2023), we report exact match (EM) and refined exact match (ER1) on 3,519 test questions. For scene-level question answering on ScanQA (Azuma et al., 2022), we report generative multi-reference EM and CIDEr (Vedantam et al., 2015) on 4,675 validation questions.

Implementation and baselines. Memory construction uses DA3 geometry, SAM3 masks, and Qwen3-VL-32B visual verification. On ReVSI, we use up to 300 sampled RGB frames per video. For each question, we derive coarse-category prompts from the mentioned object types, construct and cache instances for missing categories using the shared geometry, and then execute the read-only spatial program. Explicit queries use a frozen Qwen3-VL-32B planner. We use reported results for external baselines (Zhang et al., 2026a; Dai et al., 2026), except for SpatioLM, which we reevaluate using its default 32-frame input. By default, latent injection uses frozen Qwen3-VL-8B (Bai et al., 2025) with a trainable row encoder and shared readout. Appendices A and B provide implementation details.

4.2 MAIN RESULTS ON REVSI

Task-dependent strengths. SPATIALMEM achieves a seven-task average of 65.3%, the highest among the methods in Table 1, exceeding the strongest open-source baseline, SpatioLM, by 9.4 percentage points and even the strongest proprietary model, Gemini 3 Pro, by 4.4 percentage points. These results highlight the effectiveness of augmenting a frozen open-source VLM with explicit object-level 3D memory for spatial understanding. SPATIALMEM leads in counting, absolute distance, relative distance, and room area. Counting is particularly strong, reaching 70.1% MRA—21.7 percentage points above the best open-source counting model, Cambrian-S-7B, and 16.1 percentage points above S-Agent. Rather than inferring a scene-level count directly from video representations, SPATIALMEM enumerates persistent object identities refined through cross-view evidence. This provides an explicit mechanism for separating adjacent objects and consolidating repeated observations of the same instance, addressing ambiguities that methods without persistent object records must resolve implicitly. The qualitative examples in Appendix A.9 illustrate how these operations prevent merged objects and duplicated fragments from distorting enumeration. Beyond counting, retained object surfaces support distance measurements in a common 3D coordinate system, while scene occupancy supports area estimation. The corresponding gains of 6.3 and 4.3 percentage points over the strongest absolute-distance and room-area baselines demonstrate that the advantages extend beyond object enumeration to geometric measurement.

Remaining challenges. SPATIALMEM remains below the strongest reported results on object size, relative direction, and route planning. These tasks require more than consistent object identities: size estimation depends on observations spanning the object's extent, while direction and route queries additionally require reliable grounding of object descriptions and spatial references. Incomplete observations and unresolved reference ambiguity therefore remain potential bottlenecks.

We next examine how identity refinement affects both enumeration and the evidence retained for geometric queries. Table 2 compares five construction configurations using fixed programs generated by the frozen 32B planner.

Table 1: ReVSI results (%; numerical tasks: MRA, otherwise accuracy). Bold indicates SPATIALMEM scores that are the highest in their respective columns. †: default-setting reevaluation; ‡: five-task mean; ∗: recomputed seven-task mean.

Method	Frames	Cnt	Dist	Size	Room	R.Dist	R.Dir	Route	Avg
Frequency baseline									
Chance (Zhang et al., 2026a)	ALL	52.2	40.1	17.4	20.9	25.8	31.9	30.2	31.2*
Proprietary models									
GPT-5.2 (OpenAI, 2025)	64	56.2	41.5	73.9	63.0	48.4	34.9	38.2	50.9
Gemini 3 Flash (Doshi, 2025)	1 FPS	65.7	53.1	77.6	52.8	64.6	47.9	41.8	57.6
Gemini 3 Pro (Woodward, 2025)	1 FPS	60.1	54.7	79.3	51.9	68.1	56.0	56.4	60.9
Open-source, zero-shot									
Qwen3-VL-8B (Bai et al., 2025)	64	40.4	52.3	69.0	45.1	57.1	39.5	40.5	49.1
Qwen3-VL-32B (Bai et al., 2025)	64	46.9	65.0	70.4	55.8	53.8	34.0	47.3	53.3
InternVL3.5-8B (Wang et al., 2025c)	64	43.3	54.6	64.2	47.6	45.0	36.3	44.4	47.9
InternVL3.5-38B (Wang et al., 2025c)	64	43.8	60.6	70.2	58.4	57.4	45.9	42.7	54.1
LLaVA-Video-72B (Zhang et al., 2025)	64	40.1	29.6	59.3	27.9	39.6	24.8	43.0	37.8
Specialized, trained on VSI-style data									
Cambrian-S-7B (Yang et al., 2026b)	128	48.4	60.5	65.5	46.7	37.1	48.5	37.0	49.1
VST-7B-SFT (Yang et al., 2026a)	4 FPS	35.4	52.6	67.9	47.2	49.2	36.9	35.4	46.4
VLM-3R-7B (Fan et al., 2026)	32	41.6	61.6	64.8	52.5	46.5	49.5	34.1	50.1
Spatial-MLLM-4B (820k) (Wu et al., 2025)	16	41.5	40.0	53.1	–	30.7	39.2	–	40.9‡
SpaceR-7B (Ouyang et al., 2025)	32	30.7	34.5	52.0	18.6	22.8	34.5	20.2	30.5
SpatioLM-8B† (Wu et al., 2026)	32	47.4	64.0	68.8	57.4	44.6	51.8	57.4	55.9
Agentic, per-question tool use									
S-Agent (GPT-5.4) (Dai et al., 2026)	64	54.0	45.6	62.6	53.4	63.6	66.4	66.1	58.8
Ours: object-level memory									
SPATIALMEM (Qwen3-VL-32B)	≤300	**70.1**	**71.3**	68.1	**67.3**	**69.4**	62.1	48.9	**65.3**

All retain common geometric processing; we interpret their differences as configuration contrasts rather than isolated module effects (Appendix A.4). Figure 3 illustrates these complementary refinement decisions within one scene.

Table 2: Identity-repair comparisons on ReVSI (%). **S/V/M**: **Split/Verify/VLM Merge** in Figure 2. **V** includes early symbolic merging and the VLM category check; ✓/–: on/off. Count/Dist.: MRA; Dir.: accuracy. Protocol: Appendix A.4.

Config.	S	V	M	Count	Abs. dist.	Rel. dir.
(a) Full pipeline	✓	✓	✓	**70.1**	**71.3**	**62.1**
(b) **V+M**	–	✓	✓	48.9	61.1	46.3
(c) **S+V**	✓	✓	–	55.6	68.3	58.2
(d) **S** only	✓	–	–	46.2	65.1	53.9
(e) Baseline	–	–	–	49.2	64.8	53.9

Split provides finer candidates, not final identities. The full pipeline (a) exceeds no-**Split** (b) by 21.2 percentage points in counting, yet **Split** only (d) scores below the candidate baseline (e). In the latter contrast, undercounts fall from 539 to 154 questions, while overcounts rise from 168 to 600. This shift is consistent with separating connected objects while exposing fragments as independent candidates. The benefit of splitting therefore depends on how these finer candidates are subsequently verified and consolidated, rather than on treating every resulting component as a distinct object.

Verify affects which spatial references survive. With **Split** enabled and **VLM Merge** disabled, enabling the **Verify** stage improves counting MRA from 46.2% (d) to 55.6% (c), with gains on absolute distance and relative direction as well. The comparison between no-**Split** (b) and baseline (e) highlights a risk of applying category verification to unsplit candidates. Although their counting scores are similar (48.9% versus 49.2%), (b) leaves 100 more distance questions and 156 more direction questions unanswered. In this diagnostic, missing geometric references cause the executor to return None, which receives zero credit and is distinguished from a valid but incorrect answer. Construction and execution logs trace these additional failures to whole-candidate rejection, which removes the geometric references required by the queries; they are not program errors (Appendix A.4). Thus, a category decision can affect not only which objects are counted, but also whether a later spatial computation has the evidence it needs.

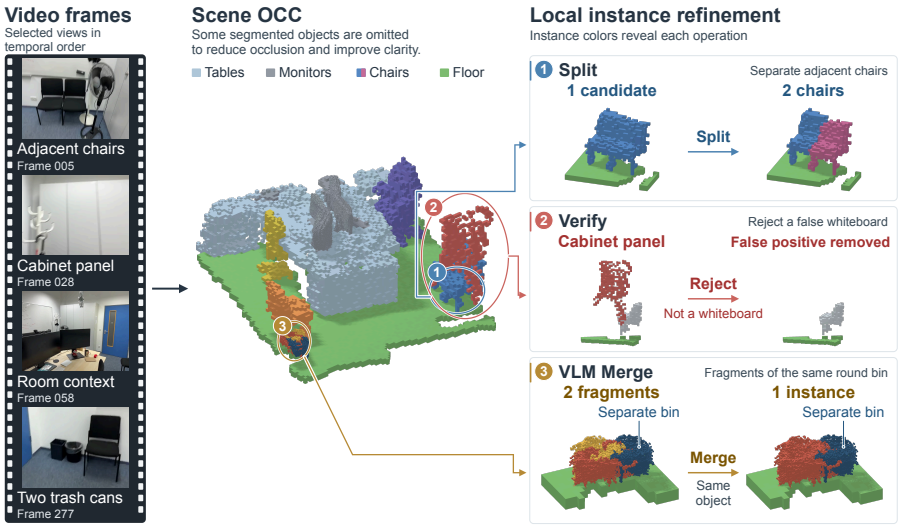

Figure 3: **Object identity refinement within one scene.** Sampled RGB frames (left) provide context; scene occupancy (center) marks the locations of the three refinement examples (right). (1) **Split** separates one connected candidate into two adjacent chairs. (2) **Verify** removes a cabinet panel misclassified as a whiteboard. (3) **VLM Merge** consolidates two fragments of one round bin into a single instance while retaining the neighboring bin. Colors in the local views distinguish instance identities.

VLM Merge addresses duplication after category verification. Passing a category check does not establish whether two candidates depict different objects. Compared with no-**VLM Merge** (c), the full pipeline (a) improves counting by 14.5 percentage points and reduces overcounts from 458 to 248 questions, consistent with consolidating repeated identities among category-valid candidates. Undercounts nevertheless increase from 226 to 288, so this is a net improvement rather than a reduction in every error type.

Together, these contrasts show why memory quality depends on both coherent object identities and the spatial references retained for later queries.

4.3 Spatial question answering on SQA3D and ScanQA

Having examined the memory through explicit computation, we next test whether its object summaries also improve learned question answering (Table 3).

Evaluation protocol. We compare the memory-augmented model with its frozen Qwen3-VL-8B backbone using identical 24-frame VLM inputs, prompts, and short-answer constraints. The memory branch additionally reads scene tables constructed from 300 frames per SQA3D scene and 295–300 per ScanQA scene. Neither condition receives GT object geometry or GT observer poses at inference. Readout training uses answer supervision and corrupted training-scene GT boxes, with training observer poses additionally used for SQA3D. Appendix B.3 records training details and calibration-scene overlap. External methods in Table 3 are grouped by primary scene inputs and differ in auxiliary processing, visual budgets, and supervision.

Performance and external comparisons. Under this exploratory protocol, SpatialMem improves exact-match accuracy over the memory-free backbone by 5.0 percentage points on SQA3D and 2.0 on ScanQA, with gains also in SQA3D ER1 and ScanQA CIDEr. On ScanQA, SpatialMem achieves 30.6% EM-1 and 101.9 CIDEr, surpassing the best available baseline scores in the RGB-video-based group of Table 3. These results support the use of object-level memory for learned spatial question answering, extending its utility beyond explicit spatial computation.

Table 3: **Spatial QA on SQA3D test and ScanQA validation.** EM, ER1, and EM-1 are percentages. Bold marks our scores exceeding all available external results in the RGB-video-based group.

Method	Input	Frames	SQA3D		ScanQA	
			EM	ER1	EM-1	CIDEr
Blind (official) (Ma et al., 2023)	Text	0	43.7	–	–	–
Depth/pose-augmented methods						
LLaVA-3D (v1) (Zhu et al., 2025)	RGB-D+C	20	55.6	57.6	27.0	91.7
Video-3D-LLM (Zheng et al., 2025)	RGB-D+C	32	58.6	–	30.1	102.1
Ross3D (Wang et al., 2025a)	RGB-D+C	32	63.0	65.7	30.8	107.0
SpatialPrompting (Taguchi et al., 2026)	RGB-D+C[a]	30	52.7	–	27.3	88.6
RGB-video-based methods						
GPT4Scene (HDM) (Qi et al., 2026)	RGB[b]	32	59.4	62.4	28.2	96.3
Spatial-MLLM-4B (Wu et al., 2025)	RGB	16[c]	55.9	58.7	26.3	91.8
SpatioLM-8B (Wu et al., 2026)	RGB	48	56.5	58.6	–	101.8
Loc3R-VLM (Qu et al., 2026)	RGB	32	62.8	65.0	28.2	100.4
Qwen3-VL-8B† (Bai et al., 2025)	RGB	24	53.2	56.6	28.6	82.3
SPATIALMEM	RGB	24[d]	58.2	61.0	**30.6**	**101.9**
SPATIALMEM‡	RGB	24[d]	**63.4**	**66.5**	**32.3**	**106.2**

Notes: Input lists primary scene inputs; auxiliary inputs are detailed below. C: scanning-camera intrinsics and poses; Frames: RGB views in the QA branch. †: our baseline evaluations; –: unavailable for the listed configuration. [a] SpatialPrompting selects keyframes using depth; GPT-4o receives RGB keyframes and camera-pose text. [b] GPT4Scene (Qwen2-VL-7B; HDM, v1) adds a BEV from ScanNet reconstruction and uses camera parameters in preprocessing. [c] Spatial-MLLM processes 128 candidate frames and selects 16. Video-3D-LLM samples uniformly.
[d] SPATIALMEM constructs memory from up to 300 RGB frames; the unmarked row trains only the row encoder and shared readout, while the ‡ row also trains LoRA adapters (Hu et al., 2022).

Current-scene content matters. The augmented model adds both scene-table information and a trained readout to the frozen backbone. Table 4 tests the contribution of scene content by fixing the readout and replacing only its scene-table input.

With normal video, the current table outperforms all replacements. Its EM gains over the constant table are 6.0 percentage points on SQA3D and 1.6 on ScanQA, supporting a contribution from scene-specific content.

With black VLM frames and video-derived memory unchanged, the current table again outperforms every replacement. However, scores remain below those with normal video (Appendix C.1).

Table 4: **Scene-input interventions (EM, %).** The first row uses the memory-free backbone; all other rows use the same trained readout as the unmarked SPATIALMEM row in Table 3. Video/Black denotes normal/black VLM frames; the video-derived memory is unchanged by the black-frame intervention.

	SQA3D		ScanQA	
Scene-table input	Video	Black	Video	Black
No memory (backbone)	53.2	36.7	28.6	15.4
Current scene table	**58.2**	**44.2**	**30.6**	**18.4**
Constant table	52.2	40.0	29.0	16.2
Incorrect scene table (1 class)	51.5	39.8	28.8	15.7
Incorrect scene table (2 classes)	51.8	39.8	27.8	14.7
Incorrect scene table (5 classes)	52.2	39.8	27.9	14.8

5 CONCLUSION

In this work, we study the construction and use of 3D memory for spatial understanding from monocular video. SPATIALMEM builds reusable object-level memory with persistent object identities, observed geometry, and supporting views. The system constructs object records for newly requested categories before readout and reuses cached records across questions. Using up to 300 sampled RGB frames per video for memory construction, SPATIALMEM achieves a seven-task average of 65.3% on ReVSI, the highest among the compared methods. The ablations show how splitting changes the candidates checked by the verifier, category rejection removes geometry needed for queries, and merging reduces duplicate counting. In exploratory evaluations under the reported training and calibration protocol, the memory-augmented model improves exact-match accuracy over its frozen backbone on both SQA3D and ScanQA.

AI USE STATEMENT

Generative AI tools were used to assist with literature retrieval and organization, manuscript editing, and LaTeX formatting. The authors take responsibility for the final content of this paper, including its text, claims, and reported results.

ETHICS STATEMENT

To the best of our knowledge, this work does not raise any specific ethical concerns.

REPRODUCIBILITY STATEMENT

The main text specifies the scene-memory representation, identity-repair algorithm, and read-only query interfaces. Appendix A details implementation settings and execution semantics; Appendix A.7 records the ReVSI evaluator and dataset versions, scoring rules, task-macro aggregation, and per-subtype sample counts and results. Error-type and query-coverage diagnostics use paired per-question predictions, execution traces, and construction-time **Verify** logs (Appendix A.4). Appendices B and C document the learned readout's training and inference protocol and input interventions, respectively.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikołaj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: A visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 23716–23736, 2022. doi: 10.52202/068431-1723. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/960a172bc7fbf0177ccccbb411a7d800-Abstract-Conference.html.

Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. ScanQA: 3D question answering for spatial scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19129–19139, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Azuma_ScanQA_3D_Question_Answering_for_Spatial_Scene_Understanding_CVPR_2022_paper.html.

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

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitScenes: A diverse real-world dataset for 3D indoor scene understanding using mobile RGB-D data. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/hash/66f041e16a60928b05a7e228a89c3799-Abstract-round1.html.

Sagar Bharadwaj, Ziyong Ma, Anurag Ghosh, Srinivasan Seshan, and Anthony Rowe. Flame3D: Zero-shot compositional reasoning of 3D scenes with agentic language models. *arXiv preprint arXiv:2605.09218*, 2026. URL https://arxiv.org/abs/2605.09218.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. In *International Conference on Learning Representations*, pp. 138846–138923, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/e0982cbc81401df3430ee1ff780dc7a2-Abstract-Conference.html.

Jiazhong Cen, Zanwei Zhou, Jiemin Fang, Chen Yang, Wei Shen, Lingxi Xie, Dongsheng Jiang, Xiaopeng Zhang, and Qi Tian. Segment anything in 3D with NeRFs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 25971–25990, 2023. doi: 10.52202/075280-1130. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/525d24400247f884c3419b0b7b1c4829-Abstract-Conference.html.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5828–5839, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Dai_ScanNet_Richly-Annotated_3D_CVPR_2017_paper.html.

Yalun Dai, Hao Li, Shulin Tian, Runmao Yao, Yuhao Dong, Fangzhou Hong, Zhaoxi Chen, Fangfu Liu, Tao Wang, Kim-Hui Yap, and Ziwei Liu. S-Agent: Spatial tool-use elicits reasoning for spatial intelligence. *arXiv preprint arXiv:2606.20515*, 2026. URL https://arxiv.org/abs/2606.20515.

Tulsee Doshi. Gemini 3 Flash: Frontier intelligence built for speed. Google Blog, December 2025. URL https://blog.google/products-and-platforms/products/gemini/gemini-3-flash/.

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*, pp. 31054–31065, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Fan_VLM-3R_Vision-Language_Models_Augmented_with_Instruction-Aligned_3D_Reconstruction_CVPR_2026_paper.html.

Bo Gu, Zhikang Zhang, Zizhuang Wei, Zhenyuan Chen, Lingyun Li, and Zhuoyi Song. SpaceMind++: Toward allocentric cognitive maps for spatially grounded video MLLMs. *arXiv preprint arXiv:2605.09449*, 2026. URL https://arxiv.org/abs/2605.09449.

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 *IEEE International Conference on Robotics and Automation*, pp. 5021–5028, 2024. doi: 10.1109/ICRA57147.2024.10610243. URL https://ieeexplore.ieee.org/document/10610243.

Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14953–14962, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Gupta_Visual_Programming_Compositional_Visual_Reasoning_Without_Training_CVPR_2023_paper.html.

Maxwell Horton, Wei Lu, Quan Tran, Yury Astashonok, Kirmani Ahmed, Babak Damavandi, Anuj Kumar, Xiao Zhang, and Seungwhan Moon. R3D: Quantitative 3D spatial reasoning for egocentric wearables. *arXiv preprint arXiv:2607.02921*, 2026. URL https://arxiv.org/abs/2607.02921.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Alaa Maalouf, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, K. Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. ConceptFusion: Open-set multimodal 3D mapping. In *Robotics: Science and Systems*, 2023. doi: 10.15607/RSS.2023.XIX.066. URL https://www.roboticsproceedings.org/rss19/p066.html.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Yang Zhao, Sida Peng, Hengkai Guo, Xiaowei Zhou, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth Anything 3: Recovering the visual space from any views. In *International Conference on Learning Representations*, pp. 141261–141285, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/e4cd50120b6d7e8daff1749d6bbaa889-Abstract-Conference.html.

Shiyang Lu, Haonan Chang, Eric Pu Jing, Abdeslam Boularias, and Kostas Bekris. OVIR-3D: Open-vocabulary 3D instance retrieval without training on 3D data. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1610–1620, 2023. URL https://proceedings.mlr.press/v229/lu23a.html.

Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. SQA3D: Situated question answering in 3D scenes. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=IDJx97BC38.

Yongsen Mao, Yiming Zhang, Hanxiao Jiang, Angel X. Chang, and Manolis Savva. MultiScan: Scalable RGBD scanning for 3D environments with articulated objects. In *Advances in Neural Information Processing Systems*, volume 35, pp. 9058–9071, 2022. doi: 10.52202/068431-0658. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/3b3a83a5d86e1d424daefed43d998079-Abstract-Conference.html.

Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3DIS: Open-vocabulary 3D instance segmentation with 2D mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4018–4028, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Nguyen_Open3DIS_Open-Vocabulary_3D_Instance_Segmentation_with_2D_Mask_Guidance_CVPR_2024_paper.html.

OpenAI. Introducing GPT-5.2. OpenAI, December 2025. URL https://openai.com/index/introducing-gpt-5-2/.

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. URL https://arxiv.org/abs/2504.01805.

Songyou Peng, Kyle Genova, Chiyu "Max" Jiang, Andrea Tagliasacchi, Marc Pollefeys, and Thomas Funkhouser. OpenScene: 3D scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 815–824, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Peng_OpenScene_3D_Scene_Understanding_With_Open_Vocabularies_CVPR_2023_paper.html.

Zhangyang Qi, Zhixiong Zhang, Ye Fang, Jiaqi Wang, and Hengshuang Zhao. GPT4Scene: Understand 3D scenes from videos with vision-language models. In

International Conference on Learning Representations, pp. 157277–157302, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/fef126561bbf9d4467dbb8d27334b8fe-Abstract-Conference.html.

Kevin Qu, Haozhe Qi, Mihai Dusmanu, Mahdi Rad, Rui Wang, and Marc Pollefeys. Loc3R-VLM: Language-based localization and 3D reasoning with vision-language models. *arXiv preprint arXiv:2603.18002*, 2026. URL https://arxiv.org/abs/2603.18002.

Dídac Surís, Sachit Menon, and Carl Vondrick. ViperGPT: Visual inference via Python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11888–11898, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Suris_ViperGPT_Visual_Inference_via_Python_Execution_for_Reasoning_ICCV_2023_paper.html.

Shun Taguchi, Hideki Deguchi, Takumi Hamazaki, and Hiroyuki Sakai. SpatialPrompting: Pose-aware keyframe prompting for 3D spatial QA toward smart indoor environments. *Frontiers in Robotics and AI*, 13:1756174, 2026. doi: 10.3389/frobt.2026.1756174. URL https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2026.1756174/full.

Ayça Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-vocabulary 3D instance segmentation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 68367–68390, 2023. doi: 10.52202/075280-2989. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d77b5482e38339a8068791d939126be2-Abstract-Conference.html.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. URL https://arxiv.org/abs/2502.14786.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4566–4575, 2015. URL https://openaccess.thecvf.com/content_cvpr_2015/html/Vedantam_CIDEr_Consensus-Based_Image_2015_CVPR_paper.html.

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*, pp. 7658–7667, 2019. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Wald_RIO_3D_Object_Instance_Re-Localization_in_Changing_Indoor_Environments_ICCV_2019_paper.html.

Haochen Wang, Yucheng Zhao, Tiancai Wang, Haoqiang Fan, Xiangyu Zhang, and Zhaoxiang Zhang. Ross3D: Reconstructive visual instruction tuning with 3D-awareness. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9275–9286, 2025a. URL https://openaccess.thecvf.com/content/ICCV2025/html/Wang_Ross3D_Reconstructive_Visual_Instruction_Tuning_with_3D-Awareness_ICCV_2025_paper.html.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5294–5306, 2025b. URL https://openaccess.thecvf.com/content/CVPR2025/html/Wang_VGGT_Visual_Geometry_Grounded_Transformer_CVPR_2025_paper.html.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jérôme Revaud. DUSt3R: Geometric 3D vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20697–20709, 2024. URL

https://openaccess.thecvf.com/content/CVPR2024/html/Wang_DUSt3R_Geometric_3D_Vision_Made_Easy_CVPR_2024_paper.html.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, Jingjing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Zhi Hou, Haoran Hao, Tianyi Zhang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingtong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biqing Qi, Jiaye Ge, Qipeng Guo, Wenwei Zhang, Songyang Zhang, Maosong Cao, Junyao Lin, Kexian Tang, Jianfei Gao, Haian Huang, Yuzhe Gu, Chengqi Lyu, Huanze Tang, Rui Wang, Haijun Lv, Wanli Ouyang, Limin Wang, Min Dou, Xizhou Zhu, Tong Lu, Dahua Lin, Jifeng Dai, Weijie Su, Bowen Zhou, Kai Chen, Yu Qiao, Wenhai Wang, and Gen Luo. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025c. URL https://arxiv.org/abs/2508.18265.

Josh Woodward. Gemini 3 brings upgraded smarts and new capabilities to the Gemini app. Google Blog, November 2025. URL https://blog.google/products-and-platforms/products/gemini/gemini-3-gemini-app/.

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, pp. 13569–13597, 2025. doi: 10.52202/085713-0457. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/142b166c8b18b43774c4f0e0c1e9948e-Abstract-Conference.html.

Jing Wu, Jianhua Wu, Jiayi Guan, Jiahong Chen, Jinghui Lu, Hangjun Ye, Bingzhao Gao, and Long Chen. SpatioLM: Towards general physical spatial intelligence in vision-language models. In *International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=CHavqrN1X9.

Mi Yan, Jiazhao Zhang, Yan Zhu, and He Wang. MaskClustering: View consensus based mask graph clustering for open-vocabulary 3D instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28274–28284, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Yan_MaskClustering_View_Consensus_based_Mask_Graph_Clustering_for_Open-Vocabulary_3D_CVPR_2024_paper.html.

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*, pp. 10632–10643, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yang_Thinking_in_Space_How_Multimodal_Large_Language_Models_See_Remember_CVPR_2025_paper.html.

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. URL https://eccv.ecva.net/virtual/2026/poster/4687.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. In *International Conference on Learning Representations*, pp. 78185–78225, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/7e3dcf772fa2e8d9087b599c7c07d4cf-Abstract-Conference.html.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A high-fidelity dataset of 3D indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12–22, 2023. URL https://openaccess.thecvf.com/

content/ICCV2023/html/Yeshwanth_ScanNet_A_High-Fidelity_Dataset_of_3D_Indoor_Scenes_ICCV_2023_paper.html.

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*, 2026a. URL https://openreview.net/forum?id=FTC7najzZT.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. LLaVA-Video: Video instruction tuning with synthetic data. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=EElFGvt39K.

Zaibin Zhang, Yuhan Wu, Lianjie Jia, Yifan Wang, Zhongbo Zhang, Yijiang Li, Binghao Ran, Fuxi Zhang, Zhuohan Sun, Yizhuang Peng, Zhenfei Yin, Lijun Wang, and Huchuan Lu. Think3D: Thinking with space for spatial reasoning. *arXiv preprint arXiv:2601.13029*, 2026b. URL https://arxiv.org/abs/2601.13029.

Duo Zheng, Shijia Huang, and Liwei Wang. Video-3D LLM: Learning position-aware video representation for 3D scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8995–9006, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zheng_Video-3D_LLM_Learning_Position-Aware_Video_Representation_for_3D_Scene_Understanding_CVPR_2025_paper.html.

Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. LLaVA-3D: A simple yet effective pathway to empowering LMMs with 3D capabilities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4295–4305, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhu_LLaVA-3D_A_Simple_yet_Effective_Pathway_to_Empowering_LMMs_with_ICCV_2025_paper.html.

SUPPLEMENTARY MATERIAL

The supplementary material provides construction and query details (Appendix A), the latent-injection architecture and training protocol (Appendix B), and additional scene-input diagnostics (Appendix C).

A IMPLEMENTATION DETAILS AND EXECUTION SEMANTICS

A.1 RECONSTRUCTION AND SCENE-TABLE FIELDS

Back-projection and coordinate frame. Depth Anything 3 (DA3) estimates depth maps $\mathbf{D}_t$, confidence maps $\mathbf{C}_t$, camera intrinsics $\mathbf{K}_t$, and poses. For frame t in chunk k, let $(\mathbf{R}_t^{\mathrm{loc}}, \mathbf{t}_t^{\mathrm{loc}})$ be its local camera-to-world pose and $(s_k, \mathbf{R}_k, \mathbf{t}_k)$ the similarity transform from the chunk to the global reconstruction frame. For pixel depth $d_t(u,v) = (\mathbf{D}_t)_{u,v}$, back-projection gives

$$\mathbf{x}_t(u,v) = s_k \mathbf{R}_k \big(\mathbf{R}_t^{\mathrm{loc}} d_t(u,v) \mathbf{K}_t^{-1} [u, v, 1]^\top + \mathbf{t}_t^{\mathrm{loc}}\big) + \mathbf{t}_k. \tag{4}$$

Low-confidence observations are filtered before voxel fusion, which retains the frame–pixel–voxel associations used by identity repair. Position and direction queries on ReVSI operate in the reconstruction frame, with the horizontal plane estimated from geometry. Table 5 records model, alignment, confidence, resolution, and storage settings.

Scene-table summaries. For each instance i with voxel support G_i and voxel centers $\mathbf{x}_v$, we compute the center and axis-aligned size as

$$\mathbf{p}_i = \frac{1}{|G_i|} \sum_{v \in G_i} \mathbf{x}_v, \qquad \mathbf{s}_i = \max_{v \in G_i} \mathbf{x}_v - \min_{v \in G_i} \mathbf{x}_v. \tag{5}$$

The observation-support feature vector is

$$\mathbf{r}_i = \big(n_i^{\mathrm{obs}}, |G_i|, \bar{C}_i, \mathbb{1}\big[n_i^{\mathrm{obs}} > 0\big]\big), \tag{6}$$

where n_i^{obs} is the number of distinct supporting frames and $\bar{C}_i$ is the mean depth confidence. The remaining fields record voxel count and observation-record availability.

A.2 CONSTRUCTION AND IDENTITY REPAIR

Figure 2 summarizes perception, instance processing, and memory storage. The geometric and semantic streams share observation frames and exchange evidence through frame–pixel–voxel associations. The preparation phase of **Verify** applies early symbolic merging and geometric cleanup before the VLM category check. Early merging requires candidates to come from different initial components and to have sufficiently close bounding boxes. Post-verification **VLM Merge** considers candidates retained by the category filter without either restriction.

A.3 CONSTRUCTION SETTINGS AND MEMORY REUSE

Voxel-resolution ablation. Table 6 compares the default 5 cm + 1 cm mixed-resolution memory with a 5 cm-only configuration. Adding the finer representation primarily benefits counting, raising MRA from 67.30% to 70.12% (+2.82 percentage points) and the seven-task average from 64.99% to 65.31% (+0.32 percentage points). Relative-distance accuracy improves by 0.31 percentage points; changes in absolute distance, object size, room area, and relative direction are at most 0.11 percentage points in magnitude. Route-planning accuracy decreases by 0.96 percentage points.

Multiple masks and invalid associations. Valid mask IDs start at 0; background is encoded as -1. A voxel may associate with multiple pixels, and thus multiple masks, in one frame. For mask $m \geq 0$ in frame t, the corresponding node has voxel support $S_{t,m} = \{v : \exists (t,u,w) \in \mathcal{A}(v),\ (\mathbf{M}_t)_{u,w} = m\}$, where $\mathcal{A}(v)$ denotes the observation records of voxel v and $\mathbf{M}_t$ is the mask-label map for frame t. The auxiliary early symbolic merge uses each fragment's dominant mask satisfying its share threshold. If none is valid, the association is $\perp$ and the frame is excluded from that early merge's shared-frame set. These dominant-mask support thresholds specify the early step, rather than the post-verification rule in Section 3.2.

Table 5: Construction settings for 5 cm + 1 cm mixed-resolution memory. Observation counts denote frame–pixel records, not distinct frames.

Stage	Settings and execution rules
Frames and resolution	Dynamic sampling across the full video, targeting at most 300 frames. The 380 scene caches contain 56–300 frames each; 350 have 300, totaling 110,713 frames. RGB width is 1280 and height is 960, 1707, or 2276; the depth-processing resolution parameter is 756.
DA3 and point clouds (Lin et al., 2026)	DA3NESTED-GIANT-LARGE-1.1 with its metric branch; chunk length 80, overlap 40, dense Sim(3) alignment, and no loop closure. Export retains points with confidence $\geq 0.5\bar{c}$ and $> 10^{-5}$, where $\bar{c}$ is the scene mean; 25.0% of valid points are randomly retained per frame.
SAM3 (Carion et al., 2026)	Image mode with text segmentation on each sampled frame, without video tracking; cache score threshold 0.35. The frozen index contains 3,588 scene–category entries across 302 scenes; each stores SAM3 outputs for all DA3 frames in that scene.
Voxels and candidates	Counting and object-geometry queries share 5 cm + 1 cm mixed-resolution voxel memory. Initial connectivity uses a Chebyshev radius of 10 cells and a 20-voxel minimum. Framework length/volume thresholds scale with grid resolution; internal split-support counts do not scale uniformly.
Category votes and nodes	Target votes must strictly exceed non-target votes among tracked observations. A component needs at least 50 observations in a frame; a mask needs at least 30 observations and a share ≥ 0.05. After two erosions, at least 30 observations and five distinct voxels must remain.
Graph and assignment	Connect nodes with intersection ≥ 20 voxels and overlap coefficient ≥ 0.5. Assign voxels when the leading group has vote share ≥ 0.60; attach unresolved voxels through 26-neighborhoods and apply two mode-smoothing passes. Groups may merge again if support overlap is ≥ 0.30 and they are co-visible in at most two frames.
Early symbolic merge	Preparation step within **Verify**, before the VLM category check. Compare fragments from different initial components with bbox gaps ≤ 2 cells on each axis. A valid shared frame requires at least 20 observations and dominant-mask share ≥ 0.5 per fragment. Accept a merge at $r = n_{\text{same}}/(n_{\text{same}} + n_{\text{diff}}) \geq 0.8$; reject at $r \leq 0.2$ and $n_{\text{diff}} \geq 5$; otherwise retain separation. No additional minimum shared-frame count is imposed.
Geometric cleanup	Deduplicate when a one-cell dilation of the larger fragment covers ≥ 0.60 of the smaller. Relative voxel-count threshold: 0.08, with category-specific exemptions. Delete a noise sliver only if its gap is > 0.10 m, thin dimension ≤ 0.10 m, count ≤ 30 voxels, and count below 5.0% of the largest component.
Verify: category check	Qwen3-VL-32B-Instruct (Bai et al., 2025); at most three views, each provided as original and highlighted images. Per frame, select the mask with maximum overlap with candidate support. Rank views by candidate support multiplied by SAM3 confidence, prefer optical-axis separation $\geq 45°$, and fill unused slots in rank order. An explicit No removes the entire candidate.
Post-verification **VLM Merge**	Compute same-mask agreement over frames with valid mask associations for both candidates: accept at ≥ 0.8 and reject at ≤ 0.2. For intermediate or unavailable agreement, ask the VLM to judge the pair using at most three views. Rank views by the smaller candidate support, then total support; prefer optical-axis separation $\geq 45°$ without filling unused slots. Union–find combines accepted identities and existing voxel support. No initial-component or bbox-gap restrictions apply.
Stored observations	Retain the four highest-confidence observations per physical voxel, with frame, pixel, and confidence. This storage cap does not truncate the original tracked observations used for splitting.

Table 6: Voxel-resolution ablation on ReVSI (%; numerical tasks: MRA, otherwise accuracy). Δ: 5 cm + 1 cm minus 5 cm only, in percentage points. Scores and differences are rounded separately.

Task	5 cm only	5 cm + 1 cm	Δ (pp)
Object counting	67.30	**70.12**	+2.82
Absolute distance	**71.37**	71.31	−0.06
Object size	67.95	**68.06**	+0.11
Room area	67.32	67.32	0
Relative distance	69.04	**69.35**	+0.31
Relative direction	62.13	62.13	≈ 0
Route planning	**49.84**	48.87	−0.96
Seven-task average	64.99	**65.31**	+0.32

A.4 Three-Task Ablation: Configurations and Evaluation Protocol

Stage switches and common processing. In Table 2, **V** denotes the **Verify** stage, comprising early symbolic merging and the VLM category check. Configuration (b) skips only **Split**. Configuration (c) disables all same-mask agreement and VLM pair decisions in **VLM Merge**, retaining **Split** and the complete **Verify** stage. **Split** only (d) also disables early symbolic merging and the VLM category check; the candidate baseline (e) additionally disables **Split**. For substage diagnostics, (f) **S**+aux. retains early symbolic merging but disables the VLM category check and **VLM Merge**. Its Count/Dist./Dir. scores are 46.0/65.1/53.9%, respectively. Thus, (f)→(c) tests the category check with early merging enabled, whereas (d)→(c) compares the complete **Verify** stage. All configurations retain their configured counting **Fine Verify** and common processing, including shared pre-merging, cross-instance geometric deduplication, small-component filtering, relative voxel-size filtering, and category-specific minimum-component filtering. These common operations are retained when **V** is disabled.

Table 7: Detailed construction switches and unanswered questions. (a)–(e) match Table 2; (f) retains only the early merge within **Verify**. **S/M**: **Split/VLM Merge**; ✓/–: on/off.

	Construction switches				Unanswered questions		
Configuration	**S**	Early merge	VLM check	**M**	Count	Dist.	Dir.
(a) Full pipeline	✓	✓	✓	✓	8	136	202
(b) **V+M**	–	✓	✓	✓	14	318	505
(c) **S+V**	✓	✓	✓	–	8	184	275
(d) **S** only	✓	–	–	–	9	218	349
(e) Baseline	–	–	–	–	9	218	349
(f) **S**+aux.	✓	✓	–	–	9	218	349

Evaluation protocol and comparison scope. The evaluation sets contain 1,091 counting, 1,232 absolute-distance, and 1,387 relative-direction questions, following the scoring rules in Appendix A.7 with failed answers assigned zero credit within their subtypes. The geometric tools in this comparison use category support; generated programs are held fixed, with no replanning or video fallback in the coverage diagnostic.

Error types and paired statistics. Table 7 counts unanswered questions, separately from valid but incorrect answers. Execution diagnostics distinguish missing references yielding answer_none, absent programs, program exceptions, and invalid answer formats. For counting, under/overcount proportions use the complete 1,091-question denominator; MAE uses valid numerical predictions and never substitutes zero for None. Paired MAE compares the two configurations on the same questions, restricted to those with valid numerical predictions from both. Per-question score changes use the official subtype weights. These subset analyses assess answer quality when both configurations

produce an answer; overall task scores still include all questions and assign zero credit to failed answers.

Coverage decomposition. Table 8 decomposes score changes on the complete evaluation set into contributions from questions answered by only one configuration (Coverage) and questions answered by both configurations (Joint). Totals use the reported task scores; Coverage and Joint contributions are computed from paired predictions before rounding to one decimal place.

Table 8: Paired geometric-score decomposition (percentage points). Configurations match Table 7; ΔN: recovered (+) or lost (−) answers; –: unreported component.

Configuration change	Task	ΔN	Coverage	Joint	Total
(e) → (b): refinement without **Split**	Distance	-100	-5.2	$+1.5$	-3.7
	Direction	-156	-7.4	-0.2	-7.6
(f) → (c): enable VLM category check	Distance	$+34$	$+1.8$	$+1.4$	$+3.2$
	Direction	$+74$	$+3.9$	$+0.4$	$+4.3$
(c) → (a): enable **VLM Merge**	Distance	$+48$	$+2.8$	–	$+3.0$
	Direction	$+73$	$+3.1$	–	$+3.9$

Counting alone does not diagnose memory quality. The no-**Split** configuration and Candidate baseline both omit **Split**, but only the former retains subsequent **Verify** and merging. Their similar counting MRA (48.9% versus 49.2%) masks a shift in error structure: from Candidate baseline to no-**Split**, overcounts fall from 15.4% to 6.5%, while undercounts rise from 49.4% to 57.8%. Scores improve on 102 questions and worsen on 142, contributing approximately $+6.0$ and -6.3 percentage points to macro-averaged MRA; the net change is -0.3 percentage points. MAE on the 1,077 jointly valid predictions rounds to 2.6 for both configurations.

The spatial deficits mainly reflect coverage loss. On the complete distance evaluation set, no-**Split** scores 3.7 percentage points below Candidate baseline. The 100 additional unanswered questions contribute a 5.2-point loss, while the 914 questions answered by both configurations contribute a 1.5-point gain; both contributions are normalized over the complete evaluation set. On the 914-question subset alone, the scores are 82.3% for no-**Split** and 80.3% for Candidate baseline. For direction, 156 additional unanswered questions contribute -7.4 percentage points to the -7.6 percentage-point gap (97.4%), with -0.2 percentage points from jointly answered questions. These results distinguish lost references from the accuracy of measurements that remain available. MRA uses subtype-macro averaging, under/overcount rates use all 1,091 questions, and MAE excludes invalid numerical predictions rather than treating them as zero.

Tracing missing references to candidate deletion. The 100 additional distance failures and 156 additional direction failures in no-**Split** are all answer_none, linked to 69 distinct scene–category pairs. Construction-time **Verify** explicitly rejected all 82 candidates belonging to these pairs, leaving no category geometry. Every visual check returned normally, without timeout or parsing failure; query programs were available and the additional failures were not syntax exceptions. The resulting chain is candidate rejection, empty category support, unavailable geometric reference, None, and zero evaluation credit. With replanning and video fallback disabled in this diagnostic, the executor preserves these unanswered cases. The positive coverage contrasts for other configurations involve different candidates and construction records.

A.5 Limitations of the Mask-Support Graph

The overlap coefficient in Equation (1) allows a small mask to associate strongly with a larger support set, but connected-component grouping is transitive. Consider disjoint object supports A and B and a mixed support $U = A \cup B$ from another view. If both A and B satisfy the minimum intersection requirement, then

$$\frac{|A \cap U|}{\min(|A|, |U|)} = 1, \qquad \frac{|B \cap U|}{\min(|B|, |U|)} = 1. \tag{7}$$

Thus, A and B can belong to the same connected component through U. A mixed mask can consequently override evidence that separates objects in other views.

A.6 Query Assets and Execution Rules

The following rules specify the explicit spatial programs used for the ReVSI main results. A frozen Qwen3-VL-32B planner generates Python programs that call a read-only `scene` object and write the result to `answer`. Programs execute with restricted builtins; numerical tasks return spatial quantities, and multiple-choice tasks map operation outputs to answer options. Table 9 summarizes the queried assets and fallbacks. Instance-based computation and category fallback are distinct policies; the latter does not guarantee a uniquely identified object.

Table 9: Default explicit-query assets and fallbacks for the ReVSI main results. All object-based queries use 5 cm + 1 cm mixed-resolution memory without modifying it.

Query	Primary asset and operation	Ambiguity or fallback policy
Count	All surviving instances in the 5 cm + 1 cm mixed-resolution memory.	**Fine Verify** for fine descriptions; retain *Yes* and *Unsure*.
Absolute distance	Largest connected cluster within each bound instance; minimum surface distance.	Fall back for the entire pair to the largest retained cluster of each category.
Relative distance	One bound anchor shared across bound candidate instances; rank minimum surface distances.	Fall back for the entire ranking to category distances; never mix instance and category distances.
Object size	Extent estimation within a sole candidate instance.	Use the category-level procedure for ambiguity or invalid instance measurements; eight-frame VLM fallback for missing geometry.
Relative direction	Instance-support centroids; require unanimous directions across all valid candidate configurations.	Use centroids of all retained category support if any configuration is invalid or directions disagree.
Landmark sequence	Instance centroids with repeated references bound consistently; require a unanimous complete turn sequence.	Use category-centroid routes for invalid or disagreeing configurations.
Room area	Precomputed scene-footprint cache, or the tracked scene point cloud.	Estimate the footprint when uncached; use occupied-grid area if the concave hull fails or is empty.

Name resolution and geometric preprocessing. We resolve object names by first matching them against the original map labels, then applying fine-to-coarse normalization and alias matching. Label lookup spans the shared 5 cm + 1 cm mixed-resolution memory. Counting, distance, size, direction, and route queries use the corresponding instances and geometric support from this memory. Geometric preprocessing uses a 0.15 m connectivity radius, removes clusters smaller than 50 voxels, and resolves cross-category overlap by removing the smaller cluster when $|A \cap B| / \min(|A|, |B|) \geq 0.30$. If filtering empties a category, its original point set is restored. Category-size prior filtering is disabled.

Distance definition and fallback geometry. For nonempty support sets P_A, P_B, the distance is $d(A, B) = \min_{\mathbf{p} \in P_A, \mathbf{q} \in P_B} \|\mathbf{p} - \mathbf{q}\|_2$. The points are voxel centers, so this is a discrete approximation to the nearest distance between observed surfaces, not a center-to-center distance. Instance-based distance uses the largest connected cluster *within* each selected instance, rather than selecting the largest object instance. Category fallback instead selects the largest retained cluster of each category, which need not correspond to a single persistent instance. Direction and route fallbacks use the centroid of *all* retained support of each category, not just its largest cluster.

Visual calls and read-only reuse. **Fine Verify** is query-time screening, distinct from construction-time coarse-category **Verify**. It reads original and target-highlighted support images and returns *Yes*, *No*, or *Unsure*. Count invokes it internally; the planner has no independent

`scene.fine_verify(...)` interface. An instance below 800 voxels uses the frame containing its highest-confidence observation. Larger instances use at most three frames selected by voxel coverage and 45° orientation diversity, stopping at 60.0% cumulative coverage. **Fine Verify** parsing may be retried twice, and decisions are cached by dataset, scene, instance, and description. Screening changes only the current answer. Distance disambiguation requires a unique *Yes* and all other responses to be *No*, whereas counting retains *Unsure*. Construction-time **Verify** instead deletes persistent candidates on explicit *No*. The runner supports up to two replans after initial generation, modifying only the program. Read-only execution does not rebuild or modify persistent maps, while permitting support-image access and visual-model calls. Failed answers receive zero evaluation credit.

A.7 Metrics and Evaluation Sets

We use the all_frame test split from the official Hugging Face release and follow its ReVSIMetric evaluator. The evaluator is pinned to torchmetrics-ext==0.3.2, and the dataset to Hugging Face revision e80b1cde.

Per-question scoring. All numerical targets in the 6,808-question evaluation set are positive. For a numerical question q, the score is

$$s_q = \frac{1}{10} \sum_{\theta \in \{0.50, 0.55, \ldots, 0.95\}} \mathbb{1}\left[\frac{|\hat{y}_q - y_q|}{y_q} \leq 1 - \theta \right]. \tag{8}$$

The evaluator generates the ten thresholds with torch.linspace in torch.float64 and uses the inclusive comparison shown above. Thus, $s_q \in \{0, 0.1, \ldots, 1\}$; numerical parsing failures receive zero credit. For a fixed target, equal under- and overestimates receive the same score. Relative error strictly above 50.0% scores zero, whereas exactly 50.0% retains 10.0% credit; threshold intervals create score plateaus. An MRA of 100.0% does not imply an exact count because nonzero errors within 5.0% can receive full credit. Invalid answers and valid but sufficiently inaccurate numerical answers can both score zero and are recorded separately in diagnostics. Multiple-choice questions use case-insensitive exact matching of the parsed option letter, with $s_q \in \{0, 1\}$.

Task-macro aggregation. Let $\mathcal{B}_t$ denote the official subtypes of task t, and $\mathcal{D}_{t,b}$ the complete question set for subtype b, including failed answers. Task scores and the seven-task average are

$$S_t = \frac{100}{|\mathcal{B}_t|} \sum_{b \in \mathcal{B}_t} \frac{1}{|\mathcal{D}_{t,b}|} \sum_{q \in \mathcal{D}_{t,b}} s_q, \qquad \text{Avg} = \frac{1}{7} \sum_{t=1}^{7} S_t. \tag{9}$$

Counting and room size each average single and multiple subtypes equally; relative distance averages closest and farthest equally. Relative direction averages the four forward/backward $\times$ easy/hard subtypes equally. Absolute distance, object size, and route each have one subtype.

Table 10 lists the 13 official subtypes and their question counts, covering 6,808 questions. Task-level results are reported in Table 1.

Answer parsing and units. The evaluator trims surrounding whitespace, retains the first space-delimited token, and removes trailing periods before parsing a numerical answer or matching an option letter. Number words, comma-formatted numbers, units, and synonyms receive no additional normalization. Counting uses unitless counts; absolute distance uses meters; object size uses the longest side in centimeters; and room size uses square meters. Relative distance, relative direction, and route return option letters.

Why we use ReVSI. VSI-Bench constructs video-based spatial questions from indoor-scan 3D annotations (Yang et al., 2025). The subsequent ReVSI audit identifies missing objects, incorrect labels, and inaccurate geometric answers caused by reconstruction and annotation errors. Sparse frame sampling may omit queried objects, so questions based on the full scene may no longer be answerable from the sampled frames. For example, among the 565 VSI-Bench object-counting questions audited, 27.0% had incorrect answers and 11.0% were ambiguous (Zhang et al., 2026a).

Table 10: Official ReVSI subtypes, question counts, and evaluation metrics. N: questions; EM: exact-match accuracy.

Task	Official question_type	N	Metric
Object counting	object_counting_single	695	MRA
	object_counting_multiple	396	MRA
Absolute distance	object_abs_distance	1,232	MRA
Object size	object_size_estimation	1,134	MRA
Room size	room_size_estimation_single	174	MRA
	room_size_estimation_multiple	128	MRA
Relative distance	object_rel_distance_closest	801	EM
	object_rel_distance_farthest	550	EM
Relative direction	object_rel_direction_forward_easy	416	EM
	object_rel_direction_backward_easy	271	EM
	object_rel_direction_forward_hard	371	EM
	object_rel_direction_backward_hard	329	EM
Route planning	route_planning	311	EM
Total		6,808	

We therefore use the official ReVSI release. It reannotates objects and geometry against raw videos, regenerates and manually verifies questions, controls answer-distribution bias, and provides visibility-matched evaluation at 16, 32, 64, and all frames. We evaluate its all_frame seven-task set of 6,808 questions across 380 scenes (Zhang et al., 2026a), drawn from ScanNet v2 (Dai et al., 2017), ScanNet++ v2 (Yeshwanth et al., 2023), ARKitScenes (Baruch et al., 2021), 3RScan (Wald et al., 2019), and MultiScan (Mao et al., 2022).

A.8 Scene Memory Construction Results

Figure 4 shows the constructed scene memory for seven scenes and a close-up of adjacent chairs, complementing the instance-level refinement examples in Appendix A.9.

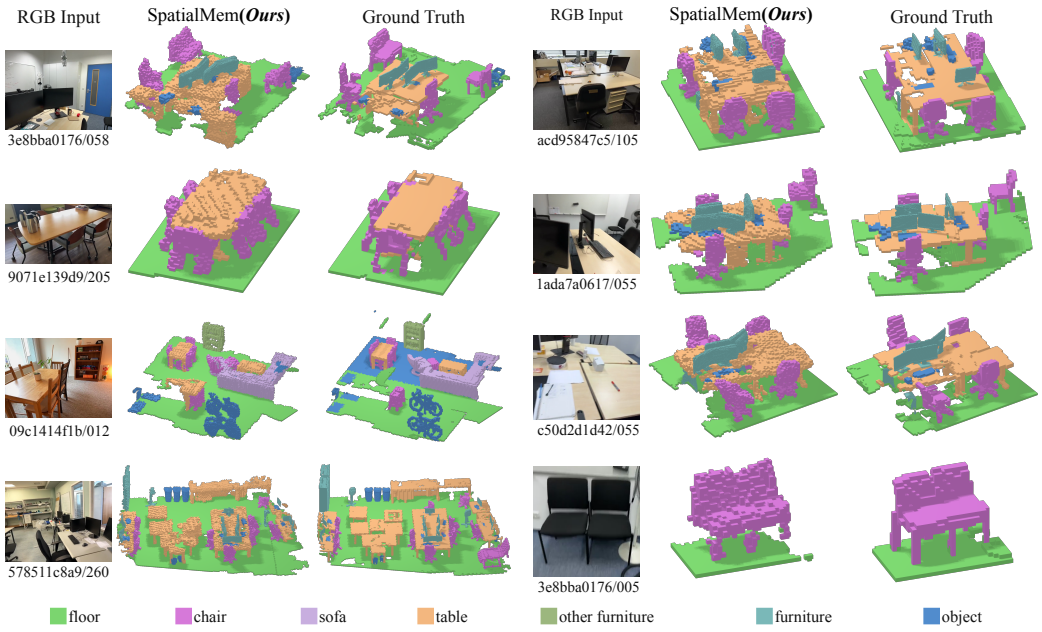

Figure 4: **Scene memory construction results.** Each group shows a representative RGB frame, the scene memory constructed by SpatialMem, and the corresponding ground-truth geometry. Colors indicate semantic categories, using the same color scheme for our results and ground truth.

A.9 Additional Qualitative Evidence

Visual evidence. The **Split**, **Verify**, and **VLM Merge** examples in Figure 5 show paired final outputs from two configurations, with context frames, masks, verification decisions, and voxel geometry. They come from the exported qualitative construction runs and illustrate the construction mechanisms. For the **VLM Merge** example, both configurations construct memory from the same VSI-Bench video cache, although the associated question records reference ReVSI videos. In the **Split** example, both configurations retain chair geometry, but no-**Split** represents it as one instance whereas Full SpatialMem separates it into six. This example illustrates object separation rather than recovery from whole-candidate rejection. The table example illustrates category validity, and **VLM Merge** illustrates duplicate-identity consolidation.

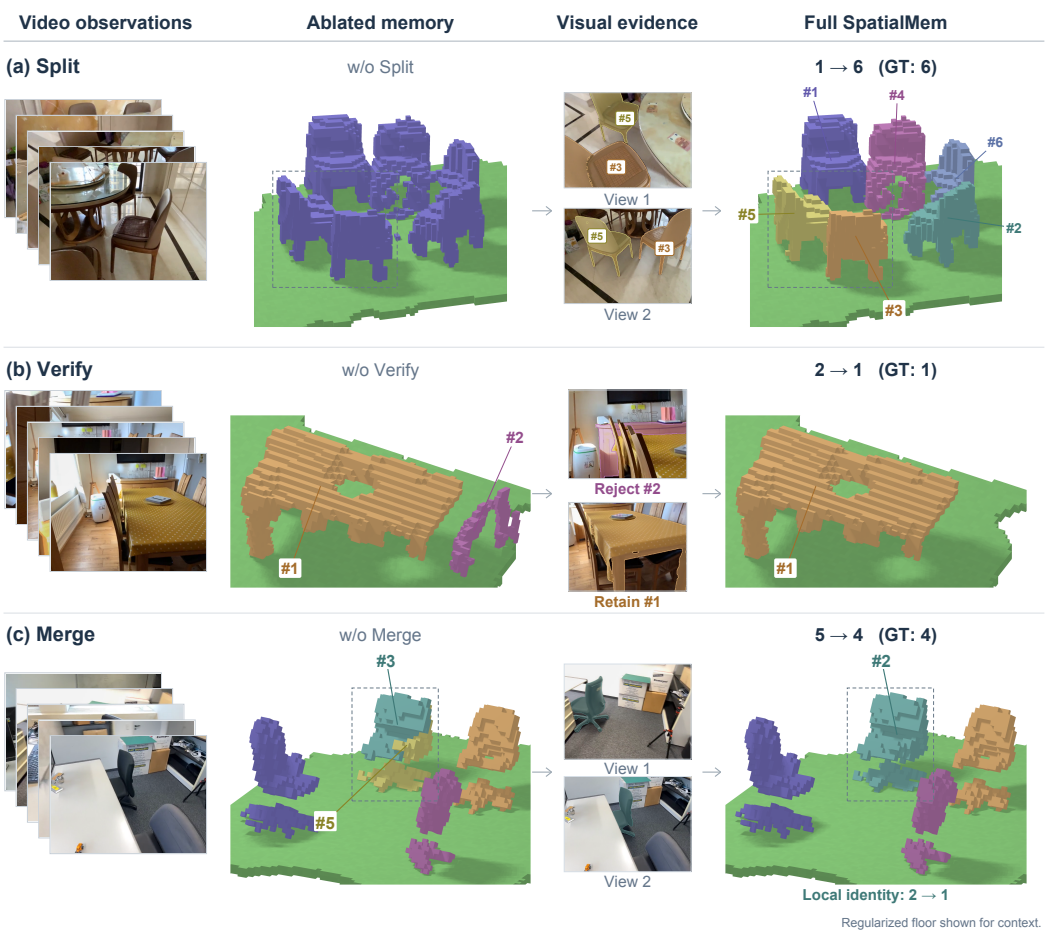

Figure 5: **Instance refinement in paired configuration comparisons.** Each row compares final instances from an ablation and Full SpatialMem; sampled RGB frames provide scene context and masked views provide object-level evidence. (a) **Split** separates adjacent chairs ($1 \rightarrow 6$). (b) Construction-time **Verify** rejects table candidate #2 and retains #1 ($2 \rightarrow 1$). (c) Post-verification **VLM Merge** consolidates two fragments of one chair (local identities: $2 \rightarrow 1$; scene count: $5 \rightarrow 4$). GT denotes ground truth.

B Latent Injection: Architecture and Training Protocol

B.1 Row Encoding and Shared Readout

Figure 6 summarizes the row encoder and shared memory readout.

Scene-table row encoding. Each scene-table row $(i, c_i, \mathbf{p}_i, \mathbf{s}_i, \mathbf{r}_i)$ is encoded as

$$\begin{aligned} \mathbf{h}_i^{\text{row}} &= \text{MLP}([\mathbf{W}_c\phi(c_i) \,\|\, \gamma(\mathbf{p}_i) \,\|\, \gamma(\mathbf{e}_i) \,\|\, \pi_i \,\|\, \gamma(\mathbf{s}_i) \,\|\, \text{MLP}_r(\tilde{\mathbf{r}}_i)]) \,, \\ \mathbf{m}_i &= \mathbf{W}_o\mathbf{h}_i^{\text{row}}. \end{aligned} \tag{10}$$

Here, ϕ is the category embedding function of a frozen SigLIP2 text encoder (Tschannen et al., 2025), and $\tilde{\mathbf{r}}_i$ contains log-normalized observation-support features. The geometric positional encoding is

$$\gamma(\mathbf{x}) = \left[\sin(2^k\pi\mathbf{x}/R), \cos(2^k\pi\mathbf{x}/R)\right]_{k=0}^{5} \,\|\, \mathbf{x}/R, \qquad R = 8\,\text{m}. \tag{11}$$

The fields $\mathbf{e}_i$ and π_i encode observer-relative geometry and pose confidence. They are set to zero in the current paired evaluation, which does not receive GT observer position or orientation. Learnable placeholder vectors represent coarse categories absent from the scene table; they do not distinguish unobserved objects from true absence. The row encoder has approximately 5.74M parameters.

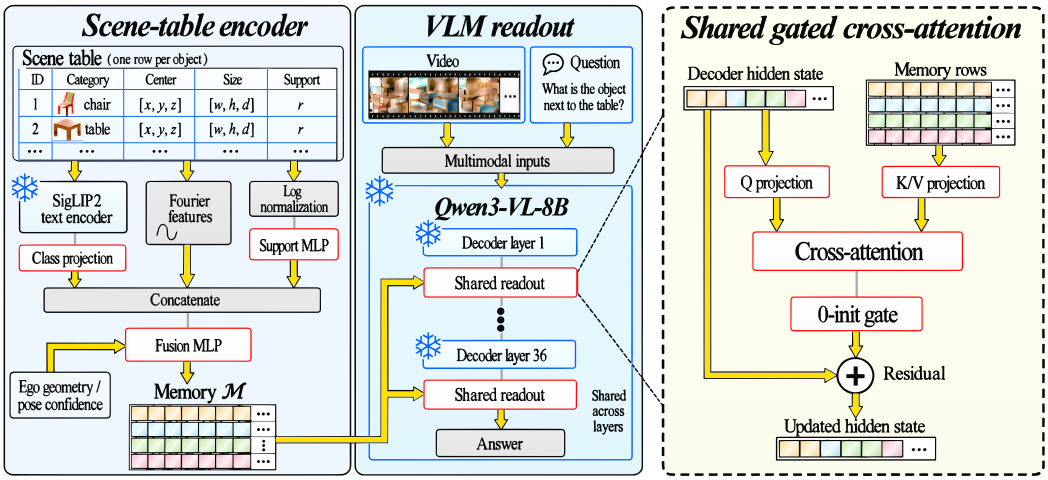

Figure 6: **Learned scene-table readout.** A trainable row encoder converts category, geometry, and observation-support fields into memory vectors. Frozen Qwen3-VL-8B retains its video and question inputs and reads the memory after each decoder layer through shared gated cross-attention. Decoder states supply queries; memory rows supply keys and values. The projections and zero-initialized gate are shared across 36 layers.

Shared gated cross-attention. In Equation (3), $\mathbf{H}^{(\ell)} \in \mathbb{R}^{L\times D}$ contains decoder hidden states and $\mathbf{M} \in \mathbb{R}^{N\times D}$ contains the encoded memory rows, with $D = 4096$. The attention dimension is $d = 512$, with $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{D\times d}$ and $\mathbf{W}_O \in \mathbb{R}^{d\times D}$. The padding mask $\mathbf{B} \in \mathbb{R}^{L\times N}$ is zero for valid entries and masks padded entries. The shared readout is applied after each of the 36 decoder layers and has approximately 8.4M parameters.

B.2 Training with Targeted Row Interventions

We match category names in the training answer first, followed by the question text alone, accounting for singular and plural forms and preferring longer category names. In this order, we select the first matched category with instance rows in the current scene table and include all of its rows in the target set $\mathcal{T}$. Donor rows must come from a different physical scene within the mini-batch on the same device. We therefore exclude other scans of the target ScanNet scene, even when their scan-ID suffixes differ. Only target-row categories and centers are replaced to form $\mathcal{S}^-$; other fields remain unchanged. Let ℓ^+ and ℓ^- be positive- and negative-branch NLLs averaged over each question's answer tokens. For samples with valid interventions, the objective is

$$\mathcal{L} = \ell^+ + \lambda\,\mathrm{ReLU}\big(1 - [\ell^- - \mathrm{sg}(\ell^+)]\big), \qquad \lambda = 0.2. \tag{12}$$

Stop-gradient applies only to the positive branch inside the margin. Samples without target rows, valid donors, or an actual change are excluded from the margin term. The objective encourages the model to assign higher likelihood to the original training answer under the original table than under the altered table. It does not specify or guarantee a correct alternative answer for the altered table. Inference does not use this answer-based target selection.

B.3 Training and Inference Protocol

Training supervision. Training uses the SQA3D and ScanQA training sets. The available manifest contains 51,351 questions across 518 full scene IDs, with 24 frames per question: 26,623 questions include a situation description and 24,728 do not. Training scene tables are synthesized by corrupting training-scene GT object boxes. SQA3D training examples additionally use training observer poses to construct ego fields; ScanQA sets ego and pose-confidence fields to zero. Both answer supervision and 3D annotation supervision are part of training.

The readout is trained for two epochs with batch size 2 per device, 16 ranks, and gradient accumulation 2, yielding 64 positive examples per optimization step and 1,606 steps. The default config-

uration freezes Qwen3-VL-8B-Instruct. The readout parameters are shared across decoder layers, and no weights are loaded from a previous row-encoder checkpoint. The learning rate is 5×10^{-5} with 0.03 warmup and cosine decay. We use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay 0.01. The gate is optimized by AdamW but is exempt from weight decay. Training uses bf16 and gradient checkpointing, without gradient clipping.

Synthesizing noisy training tables. Predicted object memories are unavailable for the training scenes, so we synthesize object tables directly from their GT object boxes. The following corruptions simulate perception errors and supply the observation-support fields required by the row encoder.

Missed detections and geometric errors. Instances are randomly removed according to calibrated category-specific recall rates. Retained box centers receive isotropic random displacements: with probability 0.989, the displacement magnitude is resampled from the empirical distribution of matched-instance position errors; with probability 0.011, it is sampled uniformly from $[0.75, 2.0]$ m to simulate large errors. Categories with insufficient matched samples use the global displacement statistics. Each of the three box dimensions is independently multiplied by a log-normal factor with log-standard deviation 0.364.

Spurious instances and scene scale. For each category c, we sample a spurious-instance count from $\mathrm{Poisson}(\rho_c n_c)$, where ρ_c is the calibrated spurious-instance rate and n_c is the number of retained instances of that category. Each inserted instance is anchored to a retained instance of the same category, with its displacement magnitude sampled from the empirical false-positive distance distribution and its position constrained to the scene bounding box. Its dimensions are sampled from retained boxes of the same category. We then apply a single scene-wide scale jitter about the center of the scene bounding box's bottom face; the scale factor has log-standard deviation $\ln(1.04)$.

Observation-support fields. GT boxes do not provide supporting-frame counts, voxel counts, mean depth confidence, or coverage indicators. For retained instances outside the large-error displacement branch, we jointly sample supporting-frame counts, mean depth confidence, and coverage indicators from the empirical pool of matched instances. Voxel counts are generated by a log-space regression on box volume plus a sampled residual. For inserted spurious instances and instances assigned to the large-error displacement branch, the complete observation-support vector is sampled jointly from the false-positive pool instead.

Normalization and reproducibility. After corruption, we normalize coordinates by centering the horizontal bounding box of instance centers and subtracting the lowest instance-center height. Random yaw augmentation is applied to the center coordinates during training; size fields are not recomputed after this rotation. The random seed depends jointly on the scene, training epoch, and sample index, making the perturbations reproducible while allowing them to vary across epochs. Inference instead reads object tables constructed from video, without this GT-box synthesis procedure.

Calibration scenes. The statistics and empirical pools used to synthesize training tables are estimated from 31 calibration scenes, of which 11 overlap with SQA3D test and 18 with ScanQA validation. Although the base GT boxes used for table synthesis belong to training scenes, calibration is therefore not restricted to the training splits. The learned-readout results are reported as exploratory under this calibration protocol. At inference, the readout uses video-derived predicted tables rather than GT object geometry.

Inference inputs and memory construction. At inference, the VLM receives 24 RGB frames and the question, together with a situation description when available and an instruction to give a short answer. The learned readout receives a scene table constructed from predicted video geometry. Neither the VLM nor the readout receives GT object geometry or GT observer poses at inference; ego and pose-confidence fields are zero on both datasets. The table covers 14 training coarse categories and missing-category placeholders, with a total budget of 96 rows. If the table exceeds the 96-row budget, we iteratively remove instance rows in ascending order of voxel count until the final table, including missing-category placeholders, contains at most 96 rows. The relative order of retained rows is preserved. Coordinate normalization centers the horizontal AABB of instance centers and

subtracts the lowest instance-center height. It uses no GT registration; the vertical origin does not represent the true floor. Random yaw augmentation is applied during training only.

SQA3D test has 3,519 questions and ScanQA val has 4,675; ScanQA uses generative multi-reference EM. Both arms share the 24-frame VLM inputs, prompts, and short-answer constraints.

External comparison conventions. In Table 3, ER1 denotes refined exact match (also EM-R1 or EM-R), and EM-1 denotes top-1 exact match. In Table 3, the Input column reports primary scene inputs, with auxiliary processing specified in the table notes; the RGB-video-based group does not imply RGB-only preprocessing. Shared task text is omitted from the column, not from the model inputs. SpatialPrompting uses depth for keyframe selection and supplies RGB keyframes and scanning-camera poses to the VLM. SpatioLM denotes the SenseNovaSI-8B configuration with 48 RGB frames (Wu et al., 2026). Its SQA3D EM and ER1 values and ScanQA CIDEr in Table 3 are taken from the paper. ScanQA EM-1 is not reported for the listed configuration.

C Latent Injection: Additional Diagnostics

C.1 Black Frames and Incorrect Scene Inputs

Table 4 evaluates the learned readout of the unmarked SpatialMem row in Table 3 with normal video or black frames, replacing the current scene table with two types of incorrect scene input. A **constant table** uses identical fixed object information in every row. An **incorrect scene table** is constructed synthetically and does not describe the current scene. We use versions with one, two, or five object categories to test whether category count alone can account for the gains.

Separating a trained-branch effect from a scene-content effect. The no-memory row and the table-input rows differ in whether they use the trained readout, so their score differences do not isolate the information in the current table. The more direct content comparison holds the trained readout fixed and replaces its input. Under normal video, the current table outperforms the constant table by 6.0 and 1.6 percentage points on SQA3D and ScanQA, respectively (Table 4). The current table also exceeds every tested synthetic incorrect table. These results show that the observed benefit is not explained solely by keeping the learned branch active: its supplied content matters in the tested conditions.

Memory remains informative without direct scene frames. With black frames, the current-versus-constant advantage remains 4.2 and 2.2 percentage points on SQA3D and ScanQA, respectively. The table can thus provide useful information even when the direct video input no longer depicts the scene. However, replacing the VLM's input frames with black frames while retaining the current scene table lowers EM by 14.0 percentage points on SQA3D and 12.2 percentage points on ScanQA. Retaining the object summaries is therefore insufficient to maintain the accuracy achieved with normal video. The current-table advantage over the constant table becomes smaller without video on SQA3D, but larger on ScanQA. Accordingly, these comparisons support the usefulness of both inputs without establishing a uniform interaction pattern between them across datasets.

Incorrect-table scores do not demonstrate that memory is ignored. Some incorrect-input conditions outperform the memory-free backbone, most visibly for SQA3D with black frames. Those gains cannot be credited to correct scene information, since the supplied tables do not describe the current scene. They also do not show that the branch is ignored: the current table still performs better when the same trained readout is used. The no-memory comparison combines removal of scene information with removal of the trained branch, whereas the table replacements test different inputs to that branch. Keeping these comparisons separate avoids conflating learned answer tendencies with a benefit from the current scene.

Scope of the input interventions. We construct the incorrect tables synthetically rather than taking real scene tables from other scenes. Replacing the current scene table with these synthetic tables changes category information and other table fields together. They therefore do not isolate the contribution of metric geometry, category identity, or observation-support features, nor establish robustness to plausible wrong-scene memories. Black frames additionally change the visual input

distribution, so the resulting score drops should not be interpreted as an additive decomposition of visual and memory information. Finally, sensitivity to table replacement is consistent with the motivation for targeted row-intervention training, but these input tests do not isolate the contribution of that training objective.