FROM REASONING FAILURES TO COMPOSABLE VIDEO SPATIAL INTELLIGENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatial reasoning benchmarks evaluate vision-language models across diverse tasks, but task-level scores do not reveal which underlying capabilities account for success or failure. Each task requires recovering spatial evidence, representing geometry, and reasoning over it. We disentangle these capabilities by comparing predicted and ground-truth spatial context under a shared schema and coordinate contract. This comparison reveals four recurring sources of error: inaccurate perception, missing information in the spatial context, selection of the wrong measurement, and errors in reference frames or in tracking position and orientation. Guided by this diagnosis, we develop CROSS, a training-free library of typed geometric operators and spatial skills that function over available evidence to support reliable video spatial reasoning. The resulting library supplies verified context to non-coding VLMs or callable skills to a SpatialClaw agent. We evaluate CROSS on five benchmarks. CROSS raises the average score from 55.9% to 60.2% on ReVSI and improves the SpatialClaw result from 62.8% to 66.3% on DSI-Bench. These gains demonstrate that explicit handling of spatial conventions can repair systematic reasoning failures without additional training.

1 INTRODUCTION

Video and multi-view spatial reasoning benchmarks (Yang et al., 2025a; Li et al., 2025d; Lin et al., 2025a; Zhang et al., 2025b; Wang et al., 2026; Zhang et al., 2026b; Wang et al., 2025b; Zhou et al., 2025b; Lin et al., 2025b; Wei et al., 2026) evaluate models on tasks such as object geometry, distance comparison, egocentric direction, and route planning. Each task requires models to recover spatial evidence and use appropriate quantities, reference frames, and temporal states. Task-level scores, however, do not reveal which capabilities account for success or failure. A wrong answer may result from missing or noisy perception, a representation that does not support the requested quantity, or misuse of otherwise sufficient evidence. Treating these causes as a single "spatial reasoning error" obscures both where models struggle and which interventions could help.

Existing approaches support video spatial reasoning through large-scale training with tailored data (Chen et al., 2024; Yang et al., 2026; Cai et al., 2025; Yang et al., 2025b; Liao et al., 2025; Huang et al., 2025; Ouyang et al., 2025; Wang & Ling, 2025; Li et al., 2025c; Zhao et al., 2025; Zhou et al., 2025a; Wang et al., 2025a; Lu et al., 2026; Chen et al., 2025), structured memory (Wu et al., 2024; Yang et al., 2025a; Chen et al., 2026; Ma et al., 2026), scene traces (Wang et al., 2026; He et al., 2026; Linghu et al., 2025), and executable code (Cho et al., 2026), though improved task performance does not establish which underlying capabilities these approaches address. More recent studies (Ropero et al., 2026; Yeung et al., 2026; Li & Yu, 2026) explicitly decouple visual perception from reasoning, and demonstrate that substantial gains can be achieved by supplying perfect perceptual results. However, perception and reasoning each encompass several distinct capabilities, and attributing failure to either alone offers limited guidance on which evidence or computation needs to improve.

We therefore give a closer examination. We investigate the failure cases along different reasoning dimensions across both static and dynamic video spatial reasoning tasks, linking errors to specific evidence requirements and computations, and explaining why better perception resolves some tasks but leaves others unsolvable. Specifically, by comparing video-only inference with inference augmented by spatial context, we reveal four recurring sources of error: First, inaccurate perception limits the recovery of relevant scene information. Second, even accurate spatial context can omit

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.

necessary information, such as room footprints or motion between stored trajectory poses. Third, models select the wrong measurement, such as computing center-to-center instead of the requested closest-surface gap. Fourth, models use the wrong reference frame or state, leading to left–right reversals, incorrect coordinate transformations, or reuse of the outdated pose.

Guided by these diagnoses, we develop CROSS, a library of **C**omposable **R**easoning **O**perators and **S**patial **S**kills for video spatial reasoning. Its operators perform individual computations, such as measuring a surface distance, transforming coordinates, or updating heading. We further compose these operators into task-specific skills. For example, a relative-direction skill constructs the observer's reference frame, projects the target into that frame, and determines its direction. A route planning skill also uses these computations, while updating position and heading after each step. Explicit input requirements and validity checks help detect when the available evidence cannot support a reliable result. Explicit geometric operations already support spatial reasoning in Map2Thought (Gao et al., 2026). Yet tool access and code execution do not eliminate geometric errors: SpatialClaw reports failures in coordinate, distance, and angle computations even with sound perceptual evidence (Cho et al., 2026). CROSS targets these recurring computations, guided by the wrong-measurement and frame/state errors in our diagnosis (Section 3.3).

We evaluate CROSS through two interfaces: derived spatial contexts for non-coding VLMs (*e.g.*, Qwen3-VL (Bai et al., 2025) and Gemma-4 (Team et al., 2026)), and callable spatial skills for code agent SpatialClaw (Cho et al., 2026). Our experiments span five benchmarks, with controlled diagnosis covering 13 benchmark-defined subtasks in REVSI-TINY and RESTI-TINY. Beyond these diagnostic subsets, CROSS raises the ReVSI average from 55.9% to 60.2% over direct video inference (Zhang et al., 2026b). Crucially, we also test cross-benchmark transfer: with the operator library frozen, adding CROSS raises matched SpatialClaw accuracy from 62.8% to 66.3% on DSI-Bench (Zhang et al., 2025b). This supports reuse of the operator basis through task-specific compositions, even in an agent already equipped with perception tools and code execution.

Our primary contributions are summarized below:

- A controlled diagnosis of video spatial reasoning across 13 dimensions that identifies four recurring sources of error: inaccurate perception, missing information in spatial context, selection of the wrong measurement, and errors in reference frames or spatial state.
- CROSS, a training-free library that turns these diagnoses into reusable geometric operators and task-specific skills.
- CROSS raises the ReVSI average from 55.9% to 60.2%, and the same operators raises SpatialClaw accuracy from 62.8% to 66.3% on DSI-Bench.

2 RELATED WORK

Spatial Benchmarks and Diagnostic Studies Spatial benchmarks span image perception in BLINK (Fu et al., 2024), viewpoint-dependent localization in ViewSpatial-Bench (Li et al., 2025a), and video-based scene reasoning in VSI-Bench (Yang et al., 2025a). SIBench organizes a broader evaluation around spatial perception, understanding, and planning (Yu et al., 2025). ReVSI (Zhang et al., 2026b) further examines annotation validity and frame-dependent answerability. STI-Bench (Li et al., 2025d), DSI-Bench (Zhang et al., 2025b), and UCS-Bench (Wang et al., 2026) extend evaluation to motion and persistent spatial memory. End-to-end results alone do not locate failures, motivating interventions that separate perception from reasoning. RESTI (Sun et al., 2026) audits STI-Bench against source annotations and repairs coordinate, timestamp, and target-definition errors. RieMind (Ropero et al., 2026) measures reasoning with a ground-truth (GT) 3D scene graph and geometric tools on static VSI-Bench; CRISP (Li & Yu, 2026) contrasts predicted and GT scene-graph inputs to diagnose grounding and reasoning in static images. SPLIT (Yeung et al., 2026) replaces estimated perception-tool outputs with GT measurements while retaining a planner that composes them, revealing substantial perceptual headroom. These studies already combine diagnostic interventions with reasoning mechanisms. Our emphasis is on identifying recurring wrong-measurement, frame/state, and missing-information failures under a shared video-context schema across static and dynamic tasks, then testing reusable repairs through both context augmentation and code-writing agents.

Methods for Video Spatial Reasoning Beyond the training approaches discussed in Section 1, explicit representations expose spatial evidence to the reasoner. SpatialMind (Zhang et al., 2025a),

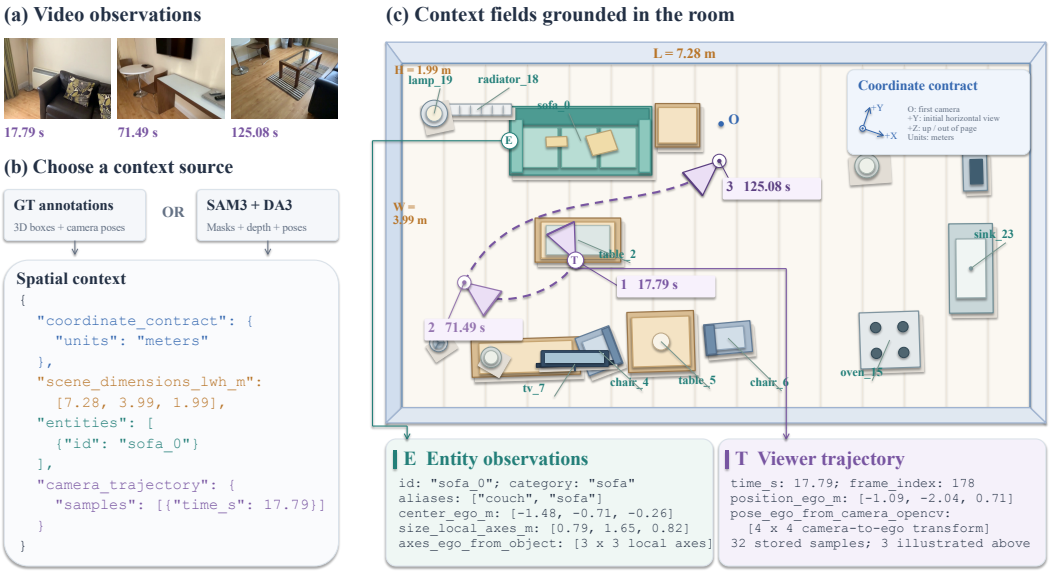


Figure 1: Illustration of spatial context. For each video, either ground-truth or tool-derived spatial clues populate the coordinate contract, scene geometry, entity observations, and viewer trajectory.

See&Trek (Li et al., 2025b), and TRACE (Hua et al., 2026) use structured prompts to organize scene or trajectory information. EgoMind builds linguistic scene graphs across frames (Chen et al., 2026), while GR3D connects image-marked object identities to textual 3D geometry (Yuan et al., 2026). Cog3DMap maintains geometrically grounded memory tokens (Gwak et al., 2026), whereas CoCoSI uses collaborating agents to construct cognitive maps (Zhang et al., 2026a). Map2Thought (Gao et al., 2026) combines metric maps with deterministic geometric operations. Agentic methods integrate specialist tools: ViSRA (Mou et al., 2026) and S-Agent (Dai et al., 2026) organize spatial tools, VADAR synthesizes a dynamic Python API (Marsili et al., 2025), and SpatialClaw supplies a stateful Python execution interface (Cho et al., 2026). In contrast, CROSS derives its operators from the failures diagnosed in Section 3.3: each targets a recurring wrong-measurement or frame/state error and makes the corresponding computation explicit, reusable, and verifiable.

3 Diagnosing Reasoning Failures

3.1 Problem Formulation

Let x denote a sampled video, q a spatial question, c a structured spatial context, and y the benchmark answer. We refer to supplying c alongside the video and question as *spatial context augmentation*. Ordinary end-to-end evaluation observes only $f(x, q)$, so an incorrect answer does not identify whether the relevant evidence was never recovered or was used incorrectly. We intervene on the available evidence by comparing video-only inference $f(x, q)$ with $f(x, q, c_{\text{pred}})$ and $f(x, q, c_{\text{gt}})$, where c_{pred} and c_{gt} are predicted and ground-truth instances of the same structured spatial context. They share the same schema, semantic field definitions, coordinate contract, and serialization; only the source and accuracy of their contents differ. Comparing performance under c_{gt} and c_{pred} measures the penalty associated with estimating the same context from video. We call this difference the *perception penalty*. On examples that pass the sufficiency and validity checks in Appendix A.4, failures under c_{gt} define the *reasoning residual*.

3.2 Structured Spatial Context

We call the injected artifact *structured spatial context* because it contains externally constructed scene evidence rather than model-generated reasoning steps. For a video-question pair, we represent it as $c = \{\kappa, g, p_{1:T}, \mathcal{E}\}$. The **coordinate contract** κ declares the world and camera frames, axis directions, handedness, units, and pose-transform direction. The **global geometry** g summarizes

the metric extent of the observed scene or queried region. The **viewer trajectory** $p_{1:T}$ records timestamped camera positions and poses, including the endpoints of the queried interval. The **entity observations** $\mathcal{E} = \{e_i\}$ assign each persistent entity a category, 3D position, and physical extent. All geometric fields use a common ego-anchored, gravity-aligned coordinate frame; its axis and transformation conventions are detailed in the supplementary material. Figure 1 illustrates how the two evidence sources populate this shared representation.

GT context populates this schema from source annotations, whereas predicted context uses SAM3 entity masks and tracks together with DA3 metric geometry and camera poses. We retain only information types available in both branches and apply the same coordinate normalization and timestamp schedule. At inference, c is serialized as a separate context block alongside the video and question; it contains observations and their conventions, but no model-generated chain of thought, candidate-dependent calculation, or final answer. All arms share the question, answer parser, and frame budget; the augmented arms share the context renderer, and provenance fields are stripped.

3.3 Failure taxonomy

Our controlled comparison identifies four recurring failure types. First, comparing predicted and GT context in Table 2 reveals a **perception gap**. Beyond perception, almost all absolute-distance failures under GT context match center-to-center distance within 5%, indicating use of the **wrong measurement** instead of the requested closest-surface distance. A third failure type is using the **wrong frame or state** when interpreting spatial relations or motion. This manifests as observer-frame errors in relative direction, left–right turn reversals in route planning, and incorrect frame composition or reuse of the initial pose in pose estimation. Finally, **missing information** limits what can be inferred even from accurate context. The omitted information includes room footprints for room-size estimation, motion between stored poses for path-length estimation, attributes that distinguish target instances for 3D grounding, and target objects or sufficiently precise measurements for dimensional measurement.

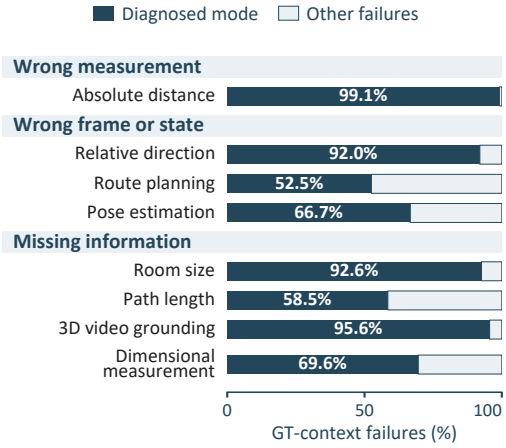


Figure 2: Share of GT-context failures matching the diagnosed mode (dark); pale segments show other failures. Rules and counts: Appendix A.5.

Figure 2 shows that these modes recur across eight subtasks, with 52.5–99.1% of their GT-context failures matching the diagnostic rules. Section 5.1 examines the task-level results and Appendix F provides the detailed examples.

4 CROSS: Composable Reasoning Operators and Spatial Skills

The controlled diagnosis reveals that models repeatedly reconstruct the same spatial procedures and repeatedly make the same mistakes: they select the wrong measurement, mix coordinate frames, reverse directional conventions, or fail to update state. CROSS avoids this inefficient and fragile rediscovery by assigning these recurring computations to verified atomic operators and composing them into task-specific spatial skills. Figure 3 gives an overview.

4.1 Applying CROSS to augmented context

Applying spatial skills and operators. Let c denote the structured spatial context and let $z_0 = (q, c)$ be the initial typed state for question q. A lightweight router selects the corresponding spatial skill, whose i-th atomic operator o_i updates the state as $z_i = o_i(z_{i-1})$ for $i = 1, \ldots, n$. The final state z_n contains the derived result, its supporting evidence, and validity information. This division lets the pipeline reuse a convention-specified procedure instead of asking the answering model to reconstruct

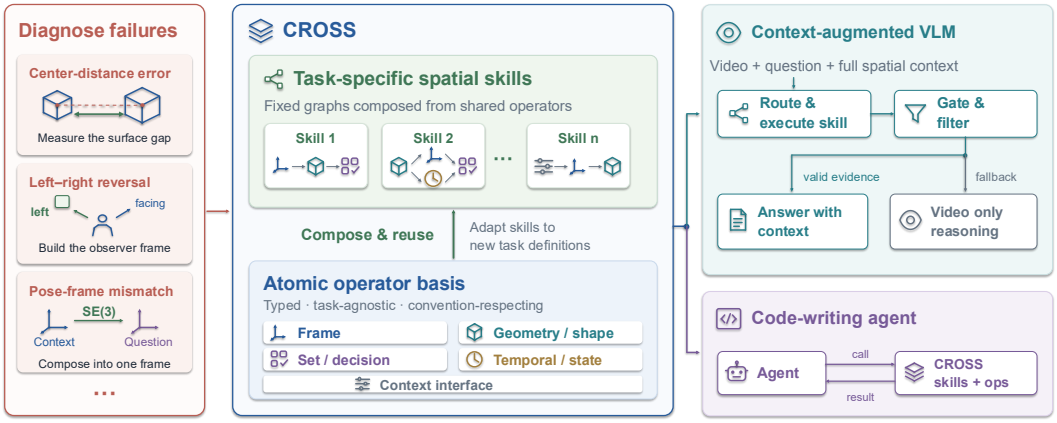


Figure 3: Overview of CROSS. Overview of CROSS. Operators derived from diagnosed failures (left) are composed into task-specific skills (middle) that serve two interfaces (right): context-augmented VLMs receive gated and filtered skill outputs or fall back to video-only reasoning, and code-writing agents call the skills and operators directly.

a spatial algorithm for every question; Sections 4.2 and 4.3 detail the operator basis and its skill compositions.

Gating and filtering the augmented context. Because perception tools may fail to reconstruct the entities or geometry required by a question, we apply a shared deterministic safeguard g after operator execution:

$$\widetilde{c} = g(c, z_n) = \begin{cases} \mathrm{filter}(c, z_n), & \text{if the derivation is valid,} \\ \varnothing, & \text{otherwise.} \end{cases} \tag{1}$$

For a valid derivation, the filter retains the derived result and its question-relevant support while removing unrelated entities and geometry. If the derivation is rejected, $\widetilde{c} = \varnothing$ and the model falls back to video-only reasoning. Thus g prevents detectably unreliable results from entering the prompt and compresses accepted context without changing the preceding computation.

4.2 Diagnosis-derived atomic operator basis

The systematic failure modes in Section 3 determine the operator basis in Table 1. Wrong-measurement errors motivate geometry and temporal operators; coordinate-frame and handedness errors motivate frame operators; and repeated left–right and route-state errors motivate local-sector and state-update operators. The resulting operators have single responsibilities and documented conventions, rather than encoding one patch per benchmark question. The full operator inventory and family responsibilities appear in Appendix C.1.

Table 1: Atomic operator families.

Family	Role
Frame	Align bases and poses.
Geometry/ shape	Measure surfaces, extents, and floor area.
Set/decision	Deduplicate, count, and match with margins.
Temporal/ state	Integrate paths; update position and heading.

4.3 Composing operators into spatial skills

For task k, we define its spatial skill as the ordered operator sequence $s_k = (o_{k,1}, \ldots, o_{k,n_k})$. Context-augmented inference appends the shared safeguard g to this sequence:

$$s_k^{\mathrm{ctx}} : \quad z_0 \xrightarrow{o_{k,1}} z_1 \xrightarrow{o_{k,2}} \cdots \xrightarrow{o_{k,n_k}} z_{n_k} \xrightarrow{g} \widetilde{c}. \tag{2}$$

The operators encode reusable spatial primitives, whereas s_k fixes their task-specific roles and execution order; code-writing deployment reuses this operator sequence without the context-specific

Table 2: Diagnosis and CROSS gains on REVSI-TINY and RESTI-TINY. Both context and code-writing arms use Qwen3-VL-32B-Thinking. Green parentheses show gains over the paired baseline.

		Video	Predicted context		GT context		Code writing	
Subtask	n	CoT	Base	+CROSS	Base	+CROSS	SpatialClaw	+CROSS
REVSI-TINY								
Object counting	118	58.2	46.2	58.2 (+12.0)	93.1	94.8 (+1.7)	42.4	45.4 (+3.0)
Absolute distance	239	54.5	48.1	63.5 (+15.4)	58.4	100.0 (+41.6)	44.8	49.6 (+4.8)
Object size	215	67.9	54.6	67.9 (+13.3)	100.0	100.0	37.1	37.2 (+0.1)
Room size	30	55.0	62.0	55.0	56.7	56.0	31.3	33.3 (+2.0)
Relative distance	111	53.2	56.8	52.3	91.0	86.5	49.6	55.0 (+5.4)
Relative direction	53	35.9	39.6	62.3 (+22.7)	52.8	100.0 (+47.2)	28.3	62.3 (+34.0)
Route planning	57	36.8	38.6	40.4 (+1.8)	29.8	49.1 (+19.3)	33.3	38.6 (+5.3)
Average	823	55.9	50.0	60.4 (+10.4)	76.2	92.3 (+16.1)	40.7	45.9 (+5.2)
RESTI-TINY								
Speed & acceleration	115	20.9	59.1	61.7 (+2.6)	97.4	100.0 (+2.6)	56.5	55.7
Pose estimation	142	24.7	27.5	70.4 (+42.9)	66.2	94.4 (+28.2)	66.2	74.7 (+8.5)
Displacement & path length	187	22.5	35.8	39.6 (+3.8)	71.1	75.4 (+4.3)	31.0	38.0 (+7.0)
Egocentric orientation	102	17.7	40.2	45.1 (+4.9)	97.1	100.0 (+2.9)	37.3	37.3
3D video grounding	177	17.5	20.3	20.3	74.6	75.1 (+0.5)	16.4	20.3 (+3.9)
Dimensional measurement	52	28.9	28.9	25.0	55.8	59.6 (+3.8)	40.4	42.3 (+1.9)
Average	775	21.3	34.3	43.9 (+9.6)	77.3	84.7 (+7.4)	39.4	43.5 (+4.1)

safeguard. Adapting CROSS to a new task therefore requires mapping its definition to an existing skill or recombining the same operator basis.

For example, the surface-distance skill aggregates reliable entity geometry before measuring the closest boundary gap, directly addressing center-distance substitution. The egocentric-direction skill constructs an observer-centered basis and projects the target into it, making perspective and handedness explicit. For rigid-pose alignment, the skill composes camera motion and the question's initial pose in a common $SE(3)$ frame before matching answer candidates. The complete skill graphs and fallback conditions appear in supplementary Table 10.

Code-writing deployment. We expose the operators as callable routines and each s_k as a callable skill that an agent selects and invokes within its native execution loop, with units, frames, and task semantics fixed by the operator contracts.

5 EXPERIMENTS

We first diagnose failures under controlled evidence and test their repair on REVSI-TINY and RESTI-TINY. We then compare with prior work on ReVSI, VSI-Bench, and STI-Bench and test transfer to DSI-Bench. Per-subtask effects, ablations, and limitations appear in Appendices D and E.

5.1 DIAGNOSTIC EXPERIMENTS

Scope. We conduct controlled diagnosis on two compact benchmark subsets using the evidence and target validity checks in Appendix A.4. Our REVSI-TINY evaluation uses 823 questions from ReVSI's released tiny split (Zhang et al., 2026b). For dynamic spatial reasoning, we construct RESTI-TINY, a 775-question diagnostic subset of RESTI (Sun et al., 2026) covering six tasks involving camera pose, motion, time-indexed quantities, and metric measurement. The diagnostic runs use dataset-specific frame budgets of 32 for REVSI-TINY and 30 for RESTI-TINY, held fixed across matched conditions. Table 2 reports the results; subset construction and exclusion details are provided in Appendix A.

Models and metrics. All context-based diagnostic experiments use Qwen3-VL-32B-Thinking with a shared decoding configuration; the code-writing arms evaluate SpatialClaw. The four numerical REVSI-TINY tasks use the benchmark's mean relative accuracy (MRA); its categorical tasks and

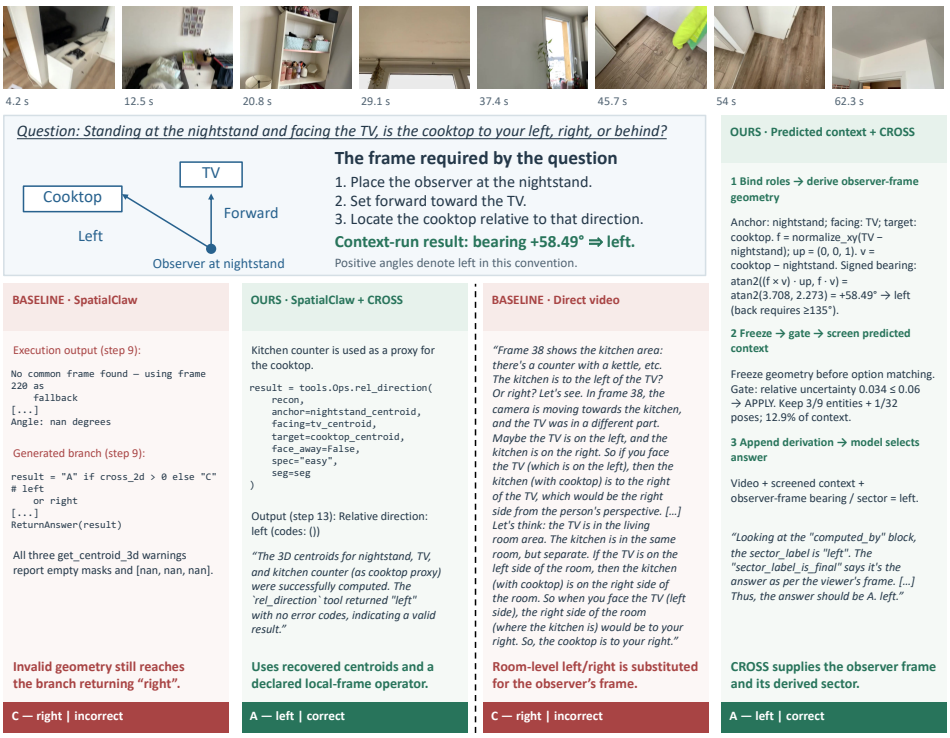


Figure 4: Relative-direction correction through code-writing (left pair) and context augmentation (right pair): both baselines answer right, whereas both CROSS runs answer left.

all RESTI-TINY tasks use multiple-choice accuracy. We report both in percent and differences in percentage points.

Predicted context exposes different perception regimes. On REVSI-TINY, predicted context lowers the average from 55.9% to 50.0%, chiefly on counting, absolute distance, and object size; on RESTI-TINY, it raises the average from 21.3% to 34.3% and improves five of six subtasks. This contrast suggests that the current predictor recovers trajectory structure more reliably than object identity and object-level metric geometry.

Does GT context suffice for reasoning? GT context largely resolves tasks that require access to accurate entities or camera geometry, but leaves errors in operation choice, frame conventions, and evidence sufficiency. Counting rises from 58.2% to 93.1% and object size from 67.9% to 100.0%; relative distance reaches 91.0%, whereas absolute distance remains at 58.4% because the model often substitutes center-to-center distance for the closest surface gap.

Frame-sensitive tasks remain harder: relative direction reaches 52.8%, with left–right reversals and misinterpretations of "facing away," which requires reversing the anchor-to-reference vector by 180°. Route planning falls from 36.8% to 29.8%, with left–right swaps accounting for 43 of 64 incorrect turn tokens. This recurring signature does not by itself establish a failure to update heading. Pose estimation reaches only 66.2%, with errors matching omitted or incorrect ego-to-world transformations. By contrast, speed and egocentric orientation reach 97.4% and 97.1%.

The combined displacement-and-path-length row in Table 2 hides a further distinction: endpoint displacement reaches 99.2% accuracy, but path length reaches only 19.7%, because sparse poses retain the endpoints without preserving the motion between them. Room size has an analogous representation limit: multiplying the dimensions of a scene envelope, such as an axis-aligned bounding box (AABB), gives the area of an enclosing rectangle rather than the queried floor polygon. This rectangle can coincide with an axis-aligned rectangular room, but counts the missing corner of an L-shaped room and can span multiple rooms. Thus, accurate context can still be insufficient, even when the arithmetic is correct.

Table 3: Detailed comparison on ReVSI. Numerical questions use MRA and multiple-choice questions use accuracy. Green parentheses show positive score-point gains over the corresponding SpatialClaw or direct-video baseline.

System	Frames	Numerical questions (MRA)				Multiple-choice questions (Acc.)			Avg. ↑
		Obj. cnt.	Abs. dist.	Obj. size	Room size	Rel. dist.	Rel. dir.	Route plan	
Baselines									
Chance (random)	All	–	–	–	–	23.7	26.8	26.0	–
Chance (frequency)	All	52.2	40.1	17.4	20.9	25.8	31.9	30.2	31.4
Commercial models									
GPT-5.2	64	56.2	41.5	73.9	63.0	48.4	34.9	38.2	50.9
Gemini 3 Pro	1 FPS	60.1	54.7	79.3	51.9	68.1	56.0	56.4	60.9
Open-source general models									
InternVL3.5-38B	64	43.8	60.6	70.2	58.4	57.4	45.9	42.7	54.1
Qwen3-VL-32B-Instruct	64	46.9	65.0	70.4	55.8	53.8	34.0	47.3	53.3
Spatially specialized models									
SpaceR-7B (SG-RLVR)	32	30.7	34.5	52.0	18.6	22.8	34.5	20.2	30.5
VST-7B-SFT	4 FPS	35.4	52.6	67.9	47.2	49.2	36.9	35.4	46.4
Cambrian-S-7B	128	48.4	60.5	65.5	46.7	37.1	48.5	37.0	49.1
VLM3R-7B	32	41.6	61.6	64.8	52.5	46.5	49.5	34.1	50.1
Published agentic methods									
S-Agent	64	54.0	45.6	62.6	53.4	63.6	66.4	66.1	58.8
Our code-agent runs (Gemma-4-31B-FP8)									
SpatialClaw (DA3 + SAM3)	64	52.9	68.4	61.3	47.6	59.0	60.9	39.4	55.6
SpatialClaw + CROSS	64	53.4 (+0.5)	73.2 (+4.8)	65.1 (+3.8)	49.2 (+1.6)	62.2 (+3.2)	62.7 (+1.8)	46.5 (+7.1)	58.9 (+3.3)
Our context runs (Qwen3-VL-32B-Thinking)									
Direct (video only)	64	63.5	58.6	68.7	50.6	56.6	41.6	51.6	55.9
w/ pred. context	64	45.5	50.1	55.8	69.2	59.2	42.9	42.3	52.2
w/ pred. context, CROSS	64	64.5 (+1.0)	63.6 (+5.0)	67.8	52.6 (+2.0)	59.3 (+2.7)	64.3 (+22.7)	49.1	**60.2** (+4.3)

5.2 Effectiveness of CROSS on diagnosed failures

GT context. CROSS improves reasoning over the same GT context, as shown by the paired scores and gain annotations in Table 2. On ReVSI-Tiny, it raises the average from 76.2% to 92.3% (+16.1), with the largest gains on absolute distance (+41.6) and relative direction (+47.2)— the tasks most affected by the diagnosed wrong-measurement and frame errors. The gains are not uniform: object size, relative distance, and room size do not improve. On ReSTI-Tiny, the average rises from 77.3% to 84.7% (+7.4), led by pose estimation (+28.2). These results show that accurate evidence alone does not remove the need for convention-correct reasoning.

Predicted evidence across inference interfaces. CROSS also improves inference when the spatial context is predicted rather than supplied by GT annotations. On ReVSI-Tiny, noisy predicted context lowers the average from 55.9% for direct inference to 50.0%, but adding CROSS raises it to 60.4% (+10.4 over predicted context), exceeding both baselines. On ReSTI-Tiny, predicted context raises the average from 21.3% to 34.3%, and CROSS further improves it to 43.9% (+9.6). The benefit extends beyond GT context to imperfect predicted evidence.

For code-writing, we expand SpatialClaw's toolbox with CROSS tools rather than supplying an augmented context. This raises SpatialClaw's average from 40.7% to 45.9% (+5.2) on ReVSI-Tiny and from 39.4% to 43.5% (+4.1) on ReSTI-Tiny. The improvements in both interfaces support the reuse of the same reasoning operators through either context augmentation or agent tool calls.

Correcting the same question through both interfaces. Figure 4 asks whether the cooktop is left, right, or behind when facing the TV from the nightstand. Left pair: empty masks produce invalid centroids and `Angle: nan degrees`, yet SpatialClaw returns C (right). The CROSS run recovers centroids and calls `rel_direction` to obtain left, with a kitchen-counter proxy for the cooktop. Right pair: direct video reasoning confuses room-level and observer-relative directions; predicted context with CROSS supplies a $+58.49°$ bearing and yields A (left). The agent prompts and geometry differ, so this selected success does not isolate the operator's causal effect.

5.3 Main evaluation on ReVSI, VSI-Bench, and STI-Bench

ReVSI. ReVSI is our primary large-scale test because it covers the same failure taxonomy as the diagnostic subset while using the 64-frame setting. Table 3 presents all seven subtask columns

Table 4: VSI-Bench results.

Method	Avg. ↑
Baselines	
Chance (frequency)	34.0
Commercial models	
GPT-5.2	49.2
Gemini 3 Pro	60.5
Open-source general models	
InternVL3.5-38B	60.8
Qwen3-VL-32B-Instruct	61.8
Spatially specialized models	
SpaceR-7B (SG-RLVR)	43.5
VLM3R-7B	60.9
VST-7B-SFT	65.2
Cambrian-S-7B	67.5
Training-free (ours)	
Qwen3-VL-32B-Thinking	59.8
w/ pred. context, CROSS	**61.4** (+1.6)

Table 5: STI-Bench results.

Method	Acc. ↑
Baselines	
Chance (random)	20.0
Commercial models	
GPT-4o	35.2
Gemini 2.5 Pro	41.4
Open-source general models	
Qwen2.5-VL-7B-Instruct	32.3
Qwen2.5-VL-72B-Instruct	40.7
InternVL2.5-78B	38.5
VideoChat-Flash	36.3
Spatially specialized models	
SpaceR-7B (SG-RLVR)	38.7
Training-free (ours)	
Qwen3-VL-32B-Thinking	41.6
w/ pred. context, CROSS	**45.3** (+3.7)

Table 6: DSI-Bench results.

Method	Acc. ↑
Full benchmark	
Gemini 2.5 Pro	46.9
SpatialTrackerV2	42.4
PolyV	58.7
SpatialClaw (1,000 examples)	
No tools	45.3
SpaceTools	43.7
Toolshed	43.0
Single-pass code	57.9
Structured tool-call	58.4
SpatialClaw	62.9
Our runs (full benchmark)	
SpatialClaw reproduction	62.8
SpatialClaw + CROSS	**66.3** (+3.5)

alongside prior work (Zhang et al., 2026b; Dai et al., 2026). CROSS improves the ReVSI average from 55.9% for direct inference and 52.2% for predicted context to 60.2% with Qwen3-VL-32B-Thinking fixed. Our score is numerically 1.4 points above S-Agent's reported 58.8% and 0.7 points below Gemini 3 Pro's 60.9% (Dai et al., 2026; Zhang et al., 2026b). For code-based inference with Gemma-4-31B-FP8 (Team et al., 2026), adding CROSS raises SpatialClaw's average from 55.6% to 58.9% (+3.3). CROSS exceeds the spatially specialized models listed in Table 3, without adding a spatial fine-tuning stage. Cross-system protocol differences are discussed in Appendix E.

VSI-Bench and STI-Bench. Tables 4 and 5 compare our training-free method with published systems on the released benchmarks (Yang et al., 2025a; Zhang et al., 2026b; Li et al., 2025d; Ouyang et al., 2025). We use 64 video frames for VSI-Bench and at most 30 uniformly sampled frames for STI-Bench. On VSI-Bench, CROSS raises the average over eight subtasks from 59.8% to 61.4% (+1.6). On STI-Bench, the average rises from 41.6% to 45.3% (+3.7), exceeding the listed Gemini 2.5 Pro (41.4%) and SpaceR (38.7%) scores.

5.4 Cross-benchmark transfer on DSI-Bench

Table 6 tests transfer through the code-writing interface on the full DSI-Bench. Published rows, including SpatialClaw's results on a fixed-seed subset of 1,000 questions, are external context only (Zhang et al., 2025b; Wu et al., 2026; Cho et al., 2026). Our full-set SpatialClaw reproduction obtains 62.8%, close to the 62.9% reported on that subset. With the atomic operators frozen and composed into skills for DSI-Bench tasks, SpatialClaw+CROSS reaches 66.3%, a 3.5-point gain over our reproduction. This result supports transfer of the operator basis through the code-writing interface; it does not establish transfer through context augmentation on DSI-Bench.

6 Conclusion

In this paper, we show that accurate spatial evidence alone does not make video spatial reasoning reliable: with ground-truth context, models still select the wrong measurement or reference frame, while other failures stem from perception or missing information. This diagnosis motivates CROSS, a training-free library of composable geometric operators and task-specific spatial skills that make these computations explicit and verifiable. Across five benchmarks and two inference interfaces, CROSS consistently improves spatial reasoning, including transfer to unseen tasks through a shared operator basis. The results suggest a composable path toward spatial intelligence: improve perception and spatial representations to recover sufficient evidence, while using explicit, reusable operators to reason over that evidence correctly. More broadly, separating *what spatial evidence is recovered* from *how it is computed* offers a practical framework for building more reliable and transferable spatial reasoning systems, which we hope will inspire future research.

AI USE STATEMENT

We used generative AI assistants in four ways. For writing, they drafted parts of sections, which the authors then revised, and polished wording and LaTeX formatting throughout the manuscript. For retrieval and discovery, they searched for and summarized related work, and the authors checked the cited papers and bibliography entries against their primary sources. For research ideation and execution, they helped discuss experimental designs and wrote and debugged code for experiments, analyses, and figures; the authors verified this code and the resulting numbers against the logged model outputs. The vision-language models evaluated in this paper are objects of study, not writing aids. The authors take full responsibility for the content of this paper, including text, claims, and artifacts produced with AI assistance.

REFERENCES

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

Zhongang Cai, Ruisi Wang, Chenyang Gu, Fanyi Pu, Junxiang Xu, Yubo Wang, Wanqi Yin, Zhitao Yang, Chen Wei, Qingping Sun, Tongxi Zhou, Jiaqi Li, Hui En Pang, Oscar Qian, Yukun Wei, Zhiqian Lin, Xuanke Shi, Kewang Deng, Xiaoyang Han, Zukai Chen, Xiangyu Fan, Hanming Deng, Lewei Lu, Liang Pan, Bo Li, Ziwei Liu, Quan Wang, Dahua Lin, and Lei Yang. Scaling spatial intelligence with multimodal foundation models. *arXiv preprint arXiv:2511.13719*, 2025. doi: 10.48550/arXiv.2511.13719. URL https://arxiv.org/abs/2511.13719.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. *arXiv preprint arXiv:2401.12168*, 2024.

Siyi Chen, Mikaela Angelina Uy, Chan Hee Song, Faisal Ladhak, Adithyavairavan Murali, Qing Qu, Stan Birchfield, Valts Blukis, and Jonathan Tremblay. SpaceTools: Tool-augmented spatial reasoning via double interactive RL. *arXiv preprint arXiv:2512.04069*, 2025. doi: 10.48550/arXiv.2512.04069. URL https://arxiv.org/abs/2512.04069.

Zhenghao Chen, Huiqun Wang, and Di Huang. EgoMind: Activating spatial cognition through linguistic reasoning in MLLMs. *arXiv preprint arXiv:2604.03318*, 2026.

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

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.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024.

Xiangjun Gao, Zhensong Zhang, Dave Zhenyu Chen, Songcen Xu, Long Quan, Eduardo Pérez-Pellitero, and Youngkyoon Jang. Map2Thought: Explicit 3d spatial reasoning via metric cognitive maps. *arXiv preprint arXiv:2601.11442*, 2026.

Chanyoung Gwak, Yoonwoo Jeong, Byungwoo Jeon, Hyunseok Lee, Jinwoo Shin, and Minsu Cho. Cog3DMap: Multi-view vision-language reasoning with 3d cognitive maps. *arXiv preprint arXiv:2603.23023*, 2026.

Siming He, Leo Huang, Adam Lilja, Fabio Huebel, Jonas Frey, Marco Pavone, S Shankar Sastry, Jitendra Malik, and Claire Tomlin. Farm: Find anything using relational spatial memory. *arXiv preprint arXiv:2606.15476*, 2026.

Jiacheng Hua, Yishu Yin, Yuhang Wu, Tai Wang, Yifei Huang, and Miao Liu. Unleashing spatial reasoning in multimodal large language models via textual representation guided reasoning. *arXiv preprint arXiv:2603.23404*, 2026.

Ting Huang, Zeyu Zhang, and Hao Tang. 3D-R1: Enhancing reasoning in 3D VLMs for unified scene understanding. *arXiv preprint arXiv:2507.23478*, 2025.

Dingming Li, Hongxing Li, Zixuan Wang, Yuchen Yan, Hang Zhang, Siqi Chen, Guiyang Hou, Shengpei Jiang, Wenqi Zhang, Yongliang Shen, Weiming Lu, and Yueting Zhuang. ViewSpatial-Bench: Evaluating multi-perspective spatial localization in vision-language models. *arXiv preprint arXiv:2505.21500*, 2025a.

Pengteng Li, Pinhao Song, Wuyang Li, Weiyu Guo, Huizai Yao, Yijie Xu, Dugang Liu, and Hui Xiong. See&Trek: Training-free spatial prompting for multimodal large language model. *arXiv preprint arXiv:2509.16087*, 2025b.

Xinhao Li, Ziang Yan, Desen Meng, Lu Dong, Xiangyu Zeng, Yinan He, Yali Wang, Yu Qiao, Yi Wang, and Limin Wang. VideoChat-R1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*, 2025c. doi: 10.48550/arXiv.2504.06958. URL https://arxiv.org/abs/2504.06958.

Yun Li, Yiming Zhang, Tao Lin, XiangRui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. STI-Bench: Are MLLMs ready for precise spatial-temporal world understanding? *arXiv preprint arXiv:2503.23765*, 2025d.

Zhixing Li and Yinan Yu. From hallucination to grounding: Diagnosing visual spatial intelligence via CRISP. *arXiv preprint arXiv:2606.26535*, 2026.

Zhenyi Liao, Qingsong Xie, Yanhao Zhang, Zijian Kong, Haonan Lu, Zhenyu Yang, and Zhijie Deng. Improved visual-spatial reasoning via R1-Zero-like training. *arXiv preprint arXiv:2504.00883*, 2025.

Jingli Lin, Runsen Xu, Shaohao Zhu, Sihan Yang, Peizhou Cao, Yunlong Ran, Miao Hu, Chenming Zhu, Yiman Xie, Yilin Long, et al. Mmsi-video-bench: A holistic benchmark for video-based spatial intelligence. *arXiv preprint arXiv:2512.10863*, 2025a.

Jingli Lin, Chenming Zhu, Runsen Xu, Xiaohan Mao, Xihui Liu, Tai Wang, and Jiangmiao Pang. OST-Bench: Evaluating the capabilities of MLLMs in online spatio-temporal scene understanding. *arXiv preprint arXiv:2507.07984*, 2025b. doi: 10.48550/arXiv.2507.07984. URL https://arxiv.org/abs/2507.07984.

Xiongkun Linghu, Jiangyong Huang, Ziyu Zhu, Baoxiong Jia, and Siyuan Huang. SceneCOT: Eliciting grounded chain-of-thought reasoning in 3d scenes. *arXiv preprint arXiv:2510.16714*, 2025.

Meng Lu, Ran Xu, Yi Fang, Wenxuan Zhang, Yue Yu, Gaurav Srivastava, Yuchen Zhuang, Mohamed Elhoseiny, Charles Fleming, Carl Yang, Zhengzhong Tu, Yang Xie, Guanghua Xiao, Di Jin, Wenqi Shi, and Xuan Wang. Scaling agentic reinforcement learning for tool-integrated reasoning in VLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26518–26529, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Lu_Scaling_Agentic_Reinforcement_Learning_for_Tool-Integrated_Reasoning_in_VLMs_CVPR_2026_paper.html.

Weijian Ma, Shizhao Sun, Tianyu Yu, Ruiyu Wang, Tat-Seng Chua, and Jiang Bian. Thinking with blueprints: Assisting vision-language models in spatial reasoning via structured object representation. *arXiv preprint arXiv:2601.01984*, 2026.

Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic AI for spatial reasoning with a dynamic API. *arXiv preprint arXiv:2502.06787*, 2025.

Tingshu Mou, Jiabo He, Renying Wang, Ce Liu, Hao Yang, Tiehua Zhang, Jingjing Chen, and Xingjun Ma. ViSRA: A video-based spatial reasoning agent for multi-modal large language models. *arXiv preprint arXiv:2605.10106*, 2026.

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

Fernando Ropero, Erkin Turkoz, Daniel Matos, Junqing Du, Antonio Ruiz, Yanfeng Zhang, Lu Liu, Mingwei Sun, and Yongliang Wang. RieMind: Geometry-grounded spatial agent for scene understanding. *arXiv preprint arXiv:2603.15386*, 2026.

Pengzhan Sun, Ramanathan Rajaraman, Shiu-Hong Kao, Junbin Xiao, and Angela Yao. ReSTI: A source-grounded audit and repair of STI-Bench. *arXiv preprint arXiv:2609.24727*, 2026.

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

Kangrui Wang, Pingyue Zhang, Zihan Wang, Yaning Gao, Linjie Li, Qineng Wang, Hanyang Chen, Chi Wan, Yiping Lu, Zhengyuan Yang, Lijuan Wang, Ranjay Krishna, Jiajun Wu, Li Fei-Fei, Yejin Choi, and Manling Li. VAGEN: Reinforcing world model reasoning for multi-turn VLM agents. *arXiv preprint arXiv:2510.16907*, 2025a. doi: 10.48550/arXiv.2510.16907. URL https://arxiv.org/abs/2510.16907.

Peiyao Wang and Haibin Ling. SVQA-R1: Reinforcing spatial reasoning in MLLMs via view-consistent reward optimization. *arXiv preprint arXiv:2506.01371*, 2025. doi: 10.48550/arXiv.2506.01371. URL https://arxiv.org/abs/2506.01371.

Qineng Wang, Baiqiao Yin, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, Saining Xie, Jiajun Wu, Li Fei-Fei, and Manling Li. MindCube: Spatial mental modeling from limited views. *arXiv preprint arXiv:2506.21458*, 2025b.

Yun Wang, Junbin Xiao, Han Lyu, Yifan Wang, Jing Zuo, Zhanjie Zhang, Hong Huang, Dapeng Wu, and Angela Yao. Keep it in mind: User centric continual spatial intelligence reasoning in egocentric video streams. *arXiv preprint arXiv:2606.15200*, 2026.

Yuxi Wei, Wei Huang, Qirui Chen, Lu Hou, and Xiaojuan Qi. See, remember, explore: A benchmark and baselines for streaming spatial reasoning. *arXiv preprint arXiv:2603.23864*, 2026.

Shengqiong Wu, Lanhu Wu, Mingyang Bao, Wenhao Xu, Hanwang Zhang, Shuicheng Yan, Hao Fei, and Tat-Seng Chua. Modeling cross-vision synergy for unified large vision model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Wenshan Wu, Shaoguang Mao, Yadong Zhang, Yan Xia, Li Dong, Lei Cui, and Furu Wei. Mind's eye of LLMs: Visualization-of-thought elicits spatial reasoning in large language models. *arXiv preprint arXiv:2404.03622*, 2024. doi: 10.48550/arXiv.2404.03622. URL https://arxiv.org/abs/2404.03622.

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

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. *arXiv preprint arXiv:2511.05491*, 2025b.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Daohan Lu, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=rBFDvZu6pb.

Jacob Yeung, Mohit Goyal, Florian Dubost, Gary L. Vondran Jr., Federico Tombari, Deva Ramanan, and Michael J. Tarr. Perception, not reasoning, limits video spatial understanding, 2026. URL https://split-spatial.github.io/.

Songsong Yu, Yuxin Chen, Hao Ju, Lianjie Jia, Fuxi Zhang, Shaofei Huang, Yuhan Wu, Rundi Cui, Binghao Ran, Zaibin Zhang, Zhedong Zheng, Zhipeng Zhang, Yifan Wang, Lin Song, Lijun Wang, Yanwei Li, Ying Shan, and Huchuan Lu. How far are VLMs from visual spatial intelligence? a benchmark-driven perspective. *arXiv preprint arXiv:2509.18905*, 2025.

Jiangye Yuan, Gowri Kumar, and Baoyuan Wang. Boosting MLLM spatial reasoning with geometrically referenced 3d scene representations. *arXiv preprint arXiv:2603.08592*, 2026.

Haoyu Zhang, Meng Liu, Zaijing Li, Haokun Wen, Weili Guan, Yaowei Wang, and Liqiang Nie. Spatial understanding from videos: Structured prompts meet simulation data. *arXiv preprint arXiv:2506.03642*, 2025a. doi: 10.48550/arXiv.2506.03642. URL https://arxiv.org/abs/2506.03642.

Yiming Zhang, Ruoxuan Cao, and Zhihang Zhong. CoCoSI: Collaborative cognitive map construction for spatial intelligence. *arXiv preprint arXiv:2606.10401*, 2026a.

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. *arXiv preprint arXiv:2604.24300*, 2026b.

Ziang Zhang, Zehan Wang, Guanghao Zhang, Weilong Dai, Yan Xia, Ziang Yan, Minjie Hong, and Zhou Zhao. DSI-Bench: A benchmark for dynamic spatial intelligence. *arXiv preprint arXiv:2510.18873*, 2025b.

Baining Zhao, Ziyou Wang, Jianjie Fang, Chen Gao, Fanhang Man, Jinqiang Cui, Xin Wang, Xinlei Chen, Yong Li, and Wenwu Zhu. Embodied-R: Collaborative framework for activating embodied spatial reasoning in foundation models via reinforcement learning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 11071–11080, 2025. doi: 10.1145/3746027.3755703. URL https://arxiv.org/abs/2504.12680.

Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Wang, Tiejun Huang, Lu Sheng, and Shanghang Zhang. RoboRefer: Towards spatial referring with reasoning in vision-language models for robotics. *arXiv preprint arXiv:2506.04308*, 2025a. doi: 10.48550/arXiv.2506.04308. URL https://arxiv.org/abs/2506.04308.

Shengchao Zhou, Yuxin Chen, Yuying Ge, Wei Huang, Jiehong Lin, Ying Shan, and Xiaojuan Qi. Learning to reason in 4D: Dynamic spatial understanding for vision language models. *arXiv preprint arXiv:2512.20557*, 2025b. doi: 10.48550/arXiv.2512.20557. URL https://arxiv.org/abs/2512.20557.

OVERVIEW OF THE APPENDIX

- Appendix A: diagnostic subsets, implementation details, prompts, evidence and target checks, and the GT-context failure audit.
- Appendix B: coordinate contract, construction, and an example of the structured spatial context.
- Appendix C: the operator inventory, skill graphs, routing and gating, and SpatialClaw integration of CROSS.
- Appendix D: per-subtask effects of CROSS and ablations on VSI-Bench.
- Appendix E: limitations.
- Appendix F: diagnostic and correction examples.

A EVALUATION SETUP AND FAILURE AUDIT

A.1 DIAGNOSTIC SUBSETS

REVSI-TINY. We draw from the 1,093-question tiny split released by ReVSI (Zhang et al., 2026b) and evaluate the 823 questions for which all diagnostic arms are available, after removing five questions from the corrupted scene `d755b3d9d8`. The questions span ScanNet, ScanNet++, and ARKitScenes and use 32 video frames.

RESTI-TINY. RESTI-TINY is our diagnostic subset of RESTI (Sun et al., 2026), not an official split. It keeps the six ScanNet indoor tasks whose source 3D annotations and camera trajectories support GT-context construction, excludes scenes `scene0002_00` and `scene0069_00` after the target checks below, and contains 775 questions that use 30 video frames.

Within each subset, all arms share the questions, frame budget, and answer parser, and averages weight questions equally. The context and code-writing pipelines differ in reconstruction settings and evaluation harness, so we compare base and CROSS runs only within each interface.

A.2 IMPLEMENTATION DETAILS

Table 7 lists the inference settings. Context-augmented runs use sampled decoding, and each arm is evaluated once. SpatialClaw runs keep SpatialClaw's agent loop and add the CROSS tools described in Appendix C.4.

Table 7: Inference settings for the two interfaces.

	Context augmentation	Code writing (SpatialClaw)
Model	Qwen3-VL-32B-Thinking	Qwen3-VL-32B-Thinking (REVSI-TINY, RESTI-TINY); Gemma-4-31B-FP8 (ReVSI, DSI-Bench)
Serving	vLLM, bf16	vLLM
Decoding	Sampling with temperature 1.0, top-p 0.95, top-k 20; up to 40,960 new tokens	Temperature 0.6; up to 32,768 (Qwen) or 131,072 (Gemma) tokens per call
Video	Uniform frames: 32 (REVSI-TINY), 30 (RESTI-TINY, STI-Bench), 64 (ReVSI, VSI-Bench)	32 key frames shown to the model; up to 64 frames for reconstruction
Perception	SAM3 and DA3 in streaming mode (predicted context; Appendix B)	SAM3 and DA3 in batch mode, called as agent tools
Agent loop	–	Up to 30 code steps with a 600 s limit per step; planning enabled
Scoring	Answers extracted with the VLMEvalKit spatial-benchmark matcher; MRA over thresholds 0.50–0.95 for numerical answers	SpatialClaw's evaluation harness with the same benchmark metrics

A.3 PROMPTS

Figure 5 gives the prompts of the three context arms. The arms differ only in the blocks inserted before the question. When the gate rejects a CROSS derivation, the question uses the Video CoT prompt.

Video CoT

```
You are answering a spatial reasoning question about an indoor scene
shown in the video frames above.
```

[Time note]

```
Question: {question}
Options: {options}
Think step by step, then provide your final answer in the format
<answer>X</answer>.
```

Time note

```
This question refers to the time interval from t = {start} s to t =
{end} s, measured from the start of the video.
```

Context

After the Video CoT introduction, the prompt adds `Below is a structured representation of the scene, provided as additional context. Use it to inform your answer.` and the structured spatial context as JSON; the time note, question, options, and answer instruction follow.

Context + CROSS

After the filtered context, the prompt adds `The block below was computed deterministically from the structured scene representation above, using the coordinate contract and units it declares. It is not a guess and it is not derived from the answer options.` and the derived block as JSON.

Figure 5: Prompt templates for the context arms. Typewriter text is verbatim and braces mark inserted fields. The time note appears only for time-scoped questions, with a single-instant variant, and "an indoor scene" becomes "a scene" for outdoor and tabletop sources.

A.4 EVIDENCE AND TARGET CHECKS

Before attributing a GT-context error to reasoning, we recompute the requested answer deterministically from the supplied context under the benchmark's conventions. If the answer is recoverable, the error counts toward the reasoning residual. If the context omits a required variable, as with sparse trajectories or missing room footprints, the example is context-insufficient. If recomputation contradicts the released target, the example is invalid for diagnosis.

A.5 GT-CONTEXT FAILURE AUDIT

Figure 2 audits the GT-context Base runs of Table 2 on the eight subtasks whose GT-context score is below 90, with path length separated from endpoint displacement. A failure is an incorrect multiple-choice answer or a numerical prediction with MRA below 1. For each failure, we evaluate the rules in Table 8 on the supplied context, the prediction, and the reference answer, and assign the first match. A match shows consistency with a mechanism rather than a unique cause.

Wrong measurement and frame. The context's closest-surface gap reproduces all 239 absolute-distance targets, yet 225 of the 227 errors equal the center distance within 5%. Deterministic geometry reproduces all 53 relative-direction labels. In route planning, 21 of the 40 errors differ from the target only by left–right turn swaps, which account for 43 of the 64 wrong turn tokens. Correct frame composition recovers 138 of 142 pose answers; the three pose hypotheses overlap (22, 15, and 13 matches), and the table reports first-match counts.

Missing information. In room size, 25 of the 27 errors equal the product of the scene length and width within 5%, and all 25 overshoot. The path-length context keeps a median of 37 of 1,157 valid poses; the chord sum over these poses recovers a median 77.8% of the true length and selects the correct option in only 12 of 66 questions. Grounding accuracy is 12/44 when the queried box is

Table 8: Rules and denominators for Figure 2. Counts are matching errors over all errors in the same GT-context Base run as Table 2; percentages are conditional on failure. Multiple rules within a row use first-match assignment.

Subtask	Matching rule	Matched/errors
Wrong measurement		
Absolute distance	Prediction within 5% of center-to-center distance	225/227
Wrong frame or state		
Relative direction	Left–right mirror (14), reversed facing (5), or fixed scene axes (4)	23/25
Route planning	Incorrect answer differs from the target only by left–right turn swaps	21/40
Pose estimation	Nearest option to ego-as-world (22), unrotated displacement (6), or initial pose (4)	32/48
Missing information		
Room size	Prediction within 5% of scene length times width	25/27
Path length	Selected option is nearest the chord sum of stored trajectory samples	31/53
3D video grounding	Queried box absent (32), or present category has multiple instances (11)	43/45
Dimensional measurement	Value not recoverable (10), or target category absent (6)	16/23

absent from the context, 72/83 when it shares its category with other instances, and 48/50 when it is unique. Dimensional-measurement accuracy is 20/27 when the context can recover the value and 9/25 otherwise.

B Structured Spatial Context

Coordinate contract. All geometric fields use one ego-anchored, gravity-aligned frame in meters. The first valid camera center is the origin, $+Z$ points up, $+Y$ is the horizontal projection of the first camera's viewing direction, and $+X = Y \times Z$. Camera poses are stored as $T_{\text{ego}\leftarrow\text{cam}}$, which maps points from the OpenCV camera frame ($+X$ right, $+Y$ down, $+Z$ forward) into the ego frame.

Construction. GT context is compiled from source 3D annotations and camera trajectories. Predicted context uses SAM3 masks and tracks, prompted with object categories from the question, together with DA3 metric depth and camera poses; predicted gravity is estimated from the camera up-vectors. The predictor receives the video, the question, and the requested timestamps, but no GT geometry. Both sources share the schema, coordinate contract, and timestamp schedule, and fields that reveal the source are removed before serialization.

Example. Figure 6 shows an abridged GT context for a REVSI-TINY absolute-distance question, together with the derived block that CROSS adds when the gate applies its derivation. The context stores object centers and oriented extents, so the closest-surface gap and the center distance are both recoverable; the derived block reports the requested quantity.

Predicted-context quality. Predicted camera positions deviate from GT by a median of 0.39 m (90th percentile 1.55 m). The median gravity tilt is 9.2°, but 59 of 404 scene records exceed 30°, mostly because of rotational drift. These errors bound what operators can recover from predicted context.

C CROSS Implementation

C.1 Atomic operator inventory

Table 9 lists the operators in each family of Table 1. The context safeguard g (Equation 1) gates and filters their outputs but is not itself an operator.

```
{"scene_id": "41069021", "dataset": "arkitscenes",
 "coordinate_contract": {
   "frame_definition": "origin = first valid video-frame camera center; ...",
   "pose_matrix_rule": "T_A_from_B maps a homogeneous point expressed in B into A",
   "camera_pose": "T_ego_from_camera_opencv maps OpenCV-camera points into ego", ...},
 "scene_dimensions_lwh_m": [7.28, 3.99, 1.99],
 "entities": [
   {"id": "picture_27", "category": "picture", "aliases": ["picture", "wall picture"],
    "center_ego_m": [-0.92, 0.15, 0.73], "size_local_axes_m": [0.77, 0.05, 0.79],
    "axes_ego_from_object": [[0.94, -0.34, 0.0], [0.34, 0.94, 0.0], [0.0, 0.0, 1.0]]},
   {"id": "microwave_22", "category": "microwave", "aliases": ["microwave"],
    "center_ego_m": [2.96, 0.42, 0.37], "size_local_axes_m": [0.44, 0.28, 0.27], ...},
   ...],
 "camera_trajectory": {"full_valid_pose_count": 1878, "samples": [
   {"frame_index": 59, "time_s": 5.88, "position_ego_m": [0.07, -1.24, 0.86],
    "pose_ego_from_camera_opencv":
      [[0.86, 0.12, -0.49, 0.07], [0.50, -0.13, 0.85, -1.24],
       [0.04, -0.99, -0.17, 0.86], [0.0, 0.0, 0.0, 1.0]]},
   ...]}}
```

```
{"computed_by": "deterministic geometry over the representation above",
 "skill": "abs_distance",
 "quantities": {"closest_surface_distance_m":
     {"value": 3.3268, "unit": "m", "frame": "ego", "uncertainty": 0.1165}},
 "roles": {"a": {"phrase": "wall picture", "matched_entities": ["picture_27"]},
           "b": {"phrase": "microwave", "matched_entities": ["microwave_22"]}},
 "center_distance_m_for_contrast": 3.9152, ...}
```


Figure 6: Abridged GT context (top) and derived block (bottom) for "Measuring from the closest point of each object, what is the direct distance between the wall picture and the microwave (in meters)?" The reference answer is 3.3 m. Context values are rounded, and "..." marks omitted fields and entries; the full context contains 19 entities and 32 trajectory samples.

Table 9: Full atomic operator inventory and family responsibilities. The shared safeguard g controls whether derived evidence enters the augmented context and is separate from these spatial computations.

Family	Operators	Responsibility
Frame	`relative_vector`, `build_ego_basis`, `project_local`, `relative_se3`, `compose_se3`, `pose_distance`	Construct directed vectors and a handedness-aware forward/right/up basis, transport poses between declared frames, and score pose candidates.
Geometry/shape	`clean_geometry`, `aggregate_geometry`, `surface_distance`, `oriented_extent`, `floor_polygon`, `polygon_area`	Robustly aggregate predicted evidence and compute the benchmark-defined surface, extent, or footprint quantity.
Set/decision	`deduplicate`, `cardinality`, `reduce_set`, `rank_with_margin`, `match_candidate`	Operate over instance sets and match a computed result to candidates only when its margin exceeds estimated error.
Temporal/state	`endpoint_displacement`, `path_integral`, `interval_rate`, `classify_sector`, `advance_agent`	Bind motion to an interval, distinguish endpoint from path quantities, classify local turns, and persistently update route position and heading.

C.2 TASK-SPECIFIC SKILL GRAPHS

Table 10 gives the operator composition and main fallback condition of each skill. In context augmentation, each skill runs on the full structured context before g; code-writing agents call the same operator sequence directly.

Table 10: Subtask skills and principal validity failures. Each operator sequence runs over the full structured spatial context and is followed by gating and filtering in the context-augmented realization.

Skill	Fixed operator graph	Main fallback condition
Counting	`deduplicate` → `cardinality`	Missing/merged objects or severe track fragmentation.
Absolute distance	`select/clean/aggregate` → `surface_distance` → `reduce(min)`	A role is absent or geometric uncertainty is high.
Relative distance	Absolute-distance graph per candidate, then `reduce(min)` → `rank_with_margin`	Candidate coverage is incomplete or the rank margin is too small.
Relative direction	`relative_vector` → `basis/project/classify`	Missing role, degenerate facing, or boundary ambiguity.
Object size	`select/clean/aggregate` → `oriented_extent` → `reduce(max)`	Truncation or unstable multi-view extent.
Room area	`select_region` → `clean` → `floor_polygon` → `polygon_area`	Only a global envelope or incomplete floor is available.
Route turns	`relative_vector` → `basis/project/classify` → `advance_agent` per waypoint, then `match_candidate`	Ambiguous landmark or unspecified terminal heading.
Displacement	`endpoint_displacement`	Interval endpoints or timestamps are missing.
Path length	`path_integral`	The context is sparse or trajectory jitter exceeds the error budget.
Interval speed	`endpoint_displacement` → `interval_rate`	Interval binding fails or the quantity definition is inconsistent.
Pose estimation	`relative_se3` → `compose_se3` → `pose_distance` → `rank`	Frames are incompatible or a queried pose is missing.

C.3 Routing, gating, and filtering

Routing. The router is rule-based. It selects a skill from the benchmark's task label and uses fixed patterns to extract the question's semantic roles, such as the anchor, facing, and target objects, the route landmarks, or the queried time interval, together with the answer convention. Each role phrase is bound to context entities by exact category, synonym, head-noun, or token-overlap matching; multiple matches are flagged as ambiguous rather than resolved silently.

Gating. Every operator returns its value with a unit, a frame, a declared error, and validity codes, and the gate maps the resulting report to one of three decisions. It abstains when the context cannot represent the requested quantity, such as a floor area without a room footprint or a path length from sparse poses. It falls back when a role is unresolved, the geometry is degenerate or the frames are incompatible, a direction lies within 5° of a sector boundary, the relative uncertainty exceeds a skill-specific limit, or the margin between the best and second-best options is below a skill-specific multiple of the declared error. Otherwise, it applies the derivation. Both abstention and fallback lead to video-only answering. The skill-specific limits, and whether ambiguous roles or overlapping boxes are rejected, are selected by grid search on a scene-disjoint development split (30% of scenes) of the predicted-context diagnostic runs, maximizing expected accuracy when rejected questions are answered from video alone. Table 2 reports all diagnostic questions, including these development scenes.

Filtering and the derived block. For an applied derivation, the filter keeps the coordinate contract, the full records of the entities bound to the question's roles, and the trajectory samples at the start and end of the queried interval, or only the first sample for static questions. Counting keeps all entities, because restricting them to the queried category would reveal the answer. In predicted context, the retained entity records include reconstruction-support fields, such as depth confidence and supporting frames, which the base context omits. The derived block reports the value, unit, frame, uncertainty, operator sequence, and role bindings (Figure 6). Answer options enter only the final matching step, after the estimate is fixed.

C.4 Integration with SpatialClaw

For code writing, CROSS is added to SpatialClaw's tools as a Python module next to its reconstruction and segmentation tools. Each skill is a single function that takes the session's reconstruction, segmentation masks, and role arguments, and returns the value with its operator sequence, supporting evidence, and validity codes; the individual operators remain callable for quantities that no skill returns. The tool description asks the agent to verify masks and evidence before measuring and to

heed validity codes, and the agent decides whether to call a skill. For DSI-Bench, four additional skills—camera motion, object motion, distance trend, and observer bearing change—compose the same frozen operators and return the matching option text.

D Additional Results and Ablations

D.1 Per-subtask effects of CROSS

Tables 2 and 3 report per-subtask scores. This section identifies the subtasks behind each average change, in percentage points.

GT context. After relative direction (+47.2) and absolute distance (+41.6), the largest ReVSI-Tiny gain is route planning (+19.3, from 29.8% to 49.1%), consistent with the left–right turn swaps among the GT-context failures (Appendix A.5).

Predicted context. Pose estimation (+42.9, from 27.5% to 70.4%) and relative direction (+22.7, from 39.6% to 62.3%) gain most. Counting, object size, and room size match their video-only scores exactly (58.2%, 67.9%, and 55.0%), which indicates that the gate routed these questions to video-only answering (Appendix C.3). Their changes relative to predicted context (+12.0, +13.3, and −7.0) therefore reflect fallback rather than operator outputs. Relative distance (56.8% to 52.3%) and dimensional measurement (28.9% to 25.0%) also decrease.

Code writing. Relative direction gains most (+34.0, from 28.3% to 62.3%). Speed and acceleration decreases slightly (56.5% to 55.7%), and egocentric orientation is unchanged (37.3%).

ReVSI. In the context runs, relative direction drives the average gain (+22.7, from 41.6% to 64.3%), whereas object size (68.7% to 67.8%) and route planning (51.6% to 49.1%) fall below direct inference. With SpatialClaw, CROSS improves all seven subtasks, most on route planning (+7.1) and absolute distance (+4.8).

D.2 Ablations on VSI-Bench

Table 11 compares five configurations on VSI-Bench with Qwen3-VL-32B-Thinking and 64 video frames. The 32- and 64-frame contexts differ only in the temporal density of the trajectory in the predicted context; the answering model always sees 64 video frames. Screened context removes question-irrelevant entities without applying operators.

Table 11: Ablations on VSI-Bench with 64 video frames. Average gives equal weight to eight subtasks. Numerical tasks use MRA and categorical tasks use accuracy (higher is better). Bold marks the best result in each column, including ties.

Setting	Numerical questions (MRA)				Multiple-choice questions (Acc.)				Average ↑
	Obj. cnt.	Abs. dist.	Obj. size	Room size	Rel. dist.	Rel. dir.	Route plan	App. order	
Direct	**60.8**	50.2	**73.9**	61.3	**57.2**	57.5	48.7	68.7	59.8
32-frame context	40.6	39.4	45.4	**65.5**	56.5	53.1	42.4	68.9	51.5
64-frame context	41.9	39.0	46.0	64.7	56.8	53.4	38.7	69.2	51.2
Screened context	42.3	41.2	46.9	57.0	55.9	**62.6**	46.6	**69.7**	52.8
CROSS	**60.8**	**53.4**	**73.9**	**65.5**	**57.2**	**62.6**	**49.2**	68.7	**61.4**

A denser trajectory does not help (51.2% with 64 samples versus 51.5% with 32), and both context variants fall below direct inference (59.8%). Screening improves relative direction and route planning but hurts room size, leaving the average at 52.8%. CROSS reaches 61.4%, exceeding screened context by 8.6 points and direct inference by 1.6, with gains on absolute distance, room size, relative direction, and route planning.

E Limitations

The GT-context intervention shows whether accurate evidence is useful, but it does not make perception and reasoning independent causal mechanisms. Predicted context remains limited by open-vocabulary recall, monocular metric scale, and rotational drift. The gate is heuristic: its checks

cannot catch every incorrect operator output, and its room-scale assumptions may not transfer to outdoor scenes. The operator basis does not cover acceleration or appearance order; the indoor RESTI subset contains no acceleration questions, and our diagnosis does not study appearance order. Spatial evaluation also depends on explicit target definitions: total 3D rotation and gravity-relative yaw are both valid but measure different capabilities. Deterministic operators precompute part of the answer; their value lies in enforcing conventions reliably, not in showing that the base model has acquired an internal spatial algorithm. Finally, published methods differ in backbone and frame budget, so their scores provide descriptive context rather than matched comparisons, and our tables report point estimates without significance tests.

F QUALITATIVE EXAMPLES

Figures 7 and 8 show one GT-context failure for each subtask in Figure 2, with REVSI-TINY cases in the upper rows and RESTI-TINY cases in the lower rows. Each panel gives the question, video frames, supplied context, an excerpt of the model's reasoning, the diagnosed failure, and the predicted and reference answers.

Figure 9 follows one RESTI-TINY pose question across four arms. Both baselines choose a wrong option; the SpatialClaw baseline compares reconstructed poses with the question's world-frame options without aligning them to the given initial pose. Both CROSS runs compose the recovered camera motion with the initial pose in a common frame and select the correct option. The context and agent arms use separate reconstructions, and this context run disables the gate's uncertainty test.

Figures 10, 11, and 12 extend the paired layout to three other REVSI-TINY subtasks, comparing the predicted-context baseline with CROSS on the same predicted context. The distance case replaces center distance with a surface gap; the route case supplies turns computed with heading-state updates. The counting case instead recovers through video fallback: forcing the derived count preserves the baseline's error. These selected examples illustrate observed recoveries, not correction rates or isolated operator effects.

Absolute distance

Q: Closest-point distance between the sofa chair and couch (m)?



GROUND-TRUTH CONTEXT

Exact oriented boxes for both objects: center_ego_m, size_local_axes_m, axes_ego_from_object Centre-to-centre distance 2.404 m Closest-surface gap 1.113 m Both are recoverable from the context.

MODEL REASONING

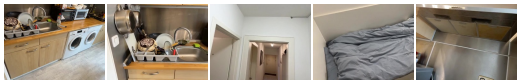
"Differences: Δx = 0.49898 Δy = 2.35131 Δz = -0.01154 Now, distance = sqrt((0.49898)^2 + (2.35131)^2 + (-0.01154)^2) sqrt(5.777813) ≈ 2.404 meters. ... So the distance between them is about 2.4 metres. 2.4 is plausible."

Uses centres; ignores object extents.

MODEL 2.4 m | CORRECT 1.1 m

Room size

Q: Size of the scene, combining rooms when multiple rooms are shown (m²)?



GROUND-TRUTH CONTEXT

scene_dimensions_lwh_m = [13.37, 6.40, H] That is the axis-aligned bounding box of the WHOLE captured scene. No floor polygon. No per-room extent.

MODEL REASONING

*"So the correct answer here would be 85.57 ... the scene dimensions are given as 13.37 x 6.4. So 13.37 * 6.4 = 85.568, which is 85.57 when rounded to two decimal places. So length is 13.37, width is 6.4, so area is L*W."*

Scene envelope substituted for floor area.

MODEL 85.57 m² | CORRECT 33.6 m²

Path length

Q: Total distance travelled by the camera between 0 and 38.5 s (m)?



GROUND-TRUTH CONTEXT

camera_trajectory.samples 39 poses camera_trajectory. full_valid_pose_count 1155 poses → the context stores 1 pose in every 30. Chord sum over the 39 stored samples = 9.34 m True arc length = 11.86 m Recovered fraction = 78.7%

MODEL REASONING

"Adding up these 9 segments ... 0.212 + 0.1115 + 0.2897 + ... ≈ 9.34 ... perhaps the correct answer is B. 8.96 m. Given that my manual sum is about 9.34, which is closest to 8.96 (B), so the answer is B." (It reads full_valid_pose_count = 1155, then dismisses it.)

Stored chords under-measure the trajectory.

MODEL B — 8.96 m | CORRECT C — 11.86 m

Pose estimation

Q: Given the initial pose, select the camera pose at 48 s in the same frame.



GROUND-TRUTH CONTEXT

Context is EGO-ANCHORED (origin = the t=0 camera centre, +Z = gravity). ego position at t = 48 s = (-4.784, -0.611, 0.173) Correct composition t0 + (R0 · R_e0^T)(e - e0) = (6.13, 3.27, 1.68) → option A ✓ Recovers the true option in 138/142 (97.2%).

MODEL REASONING

"The horizontal position (x) would be 1.8 + (-4.78385721) = -2.98385721 (which is the expected x) The vertical position (y) would be 5.35 + (-0.61105062) = 4.73894938 ... the x value is -2.98, which is to the left of 1.8, which is correct ... Therefore ... option C."

Adds ego displacement without rotating it.

MODEL C — (−1.76, 5.85, 1.87) | CORRECT A — (6.21, 3.38, 1.79)

Figure 7: GT-context failures on absolute distance and room size (top) and path length and pose estimation (bottom): center distance used for the surface gap, the scene envelope used for floor area, a sparse trajectory that underestimates path length, and an ego displacement not rotated into the world frame.

Relative direction

Q: At the wooden chest facing the TV stand, is the hutch left, right, or back?



GROUND-TRUTH CONTEXT

anchor box_19 (-0.61, 7.56, -1.02) facing tv_5 (2.00, 7.01, -0.94) target cabinet_16 (-1.30, 2.36, -0.17) Deterministic bearing from these three points: -85.65° (convention: positive = LEFT) → unambiguously RIGHT, and 85.65° away from the left/right boundary.

MODEL REASONING

"So, the TV stand is to the user's right (ΔX > 0) ... The vector from the user to the hutch: ΔX = -0.686 (left), ΔY = -5.204 (forward). The previous cross product calculation was considering the wrong axes. Since X in the ego coordinates corresponds to the camera's right, a negative ΔX would be to the user's left. Therefore, the correct answer is C. left."

Ego-axis sign replaces observer-frame bearing.

MODEL C — left | CORRECT B — right

Route planning

Q: Start at TV facing piano; via couch and bar stool to tree: which two turns?



GROUND-TRUTH CONTEXT

Exact positions for every landmark: couch, piano, bar stools, christmas tree Everything needed to simulate an agent with a position AND a heading is present.

MODEL REASONING

"The bar stools are in the kitchen area, which is in the direction of the entrance. So if the robot is facing the piano, to get to the kitchen, it would turn right. Because the kitchen is to the RIGHT OF THE PIANO ROOM. Now, from the bar stool, the christmas tree is to the left (the tree is next to the bar stools)."

Fixed room relations replace heading updates.

MODEL B — right, left | CORRECT C — right, right

3D video grounding

Q: At 57.4 s, which camera-frame box is the blue chair with metal legs under a wooden table?



GROUND-TRUTH CONTEXT

Entity fields available: id · category · aliases center_ego_m · size_local_axes_m axes_ego_from_object aliases is literally [category]. 'chair' matches 22 entities in this scene. No colour. No material. No spatial context. When a phrase DOES bind uniquely, the context box matches the labelled option to 0.001 m.

MODEL REASONING

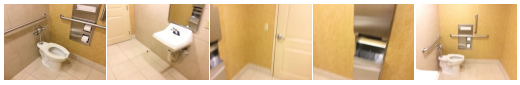
"So for option B, the C_heading is calculated correctly ... Now, the question is, which of the options correctly describes the 3D box. The correct answer is likely B, but to confirm, we need to check ... BUT WITHOUT MORE DATA, IT'S HARD TO SAY. However ... Alternatively, the answer could be B."

Referring expression fails to identify an instance.

MODEL B — one candidate chair | CORRECT D — intended chair

Dimensional measurement

Q: Approximate gravity-aligned height of the toilet paper dispenser on a beige wall (m)?



GROUND-TRUTH CONTEXT

Oriented boxes for the bound entities, with extents in metres. Options: 0.43 · 0.51 · 0.57 · 0.63 · 0.71 m ≈ 13% steps — about 6 cm apart. The box extent the context supports is not precise to 6 cm.

MODEL REASONING

"... approximately 0.46, and the options have B as 0.43. That's a 0.03 difference. Maybe. If the height of the toilet (in Z) is 0.459 m, but the question is about the dispenser ... I'M STUCK. Maybe the answer is B. 0.43m. Or A. 0.51m. ... I think I'll go with A. 0.51 m."

Reasoning remains uncertain between options.

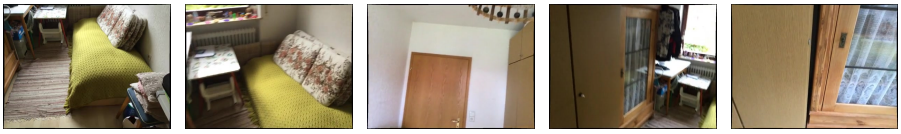
MODEL A — 0.51 m | CORRECT E — 0.57 m

Figure 8: GT-context failures on relative direction and route planning (top) and 3D grounding and dimensional measurement (bottom): a handedness reversal, a heading that is never updated, a referring expression that matches many entities, and answer options closer than the context's precision.

Pose estimation

Q: Given the initial pose below, select the camera pose at t = 9 s in the same coordinate frame, using only the RGB video.

Initial pose: t_o = (1.01, 1.18, 1.46) m; R_o = [−0.18, −0.61, 0.78; −0.98, 0.10, −0.15; 0.01, −0.79, −0.61]



0 s · initial pose | 9 s · queried | 18 s | 27 s | 35 s

BASELINE · Direct video

Model reasoning (excerpt)

"Option A has tx=0.78, ty=0.43, tz=1.6. Let's see. The other options have higher tx values (2.0, 3.42, 3.22, 2.7), which might be too far. The initial tx was 1.01, so at t=9s, it's probably not that far. So A is more plausible."

Chooses a nearby x-coordinate without recovering camera motion.

A: (0.78, 0.43, 1.60) m | incorrect

OURS · Predicted context + CROSS

1 Bind time and compose the pose

t = 9 s binds to frame 270 (zero snap error).
$T_{q←c_9} = T_{q←c_o} \cdot \text{inv}(T_{e←c_o}) \cdot T_{e←c_9}$
q: question frame; e: context frame; c_o/c_9: camera at 0/9 s.
Position = (2.0018, 1.3786, 1.4706) m.

2 Gate and screen the predicted context

Pose frozen before option matching; decision APPLY (uncertainty test off). Keeps 0/11 entities and 2/37 poses (7.0% of the context).

3 Model reads the derived position

"But the key here is that the estimated_translation from the computed block is [2.0018, 1.3786, 1.4706], which matches option B's (2.0, 1.43, 1.47) closely. So the answer is B."

Composes the camera motion into the question frame.

B: (2.00, 1.43, 1.47) m | correct

BASELINE · SpatialClaw

Generated code (excerpts)

```
T_270 = recon.extrinsics.camera_poses[recon_idx]
[...]
ssd[key] = np.sum((params_270 - opt) ** 2)
```

Output (step 3)

```
A: 13.933608
B: 16.599027
[...]
Best match: A
```

"The computed SSD values show that option A has the smallest sum of squared differences (13.93) compared to other options (B: 16.59, C: 26.95, D: 25.78, E: 21.06)."

Scores raw reconstruction coordinates against world-frame options.

A: (0.78, 0.43, 1.60) m | incorrect

OURS · SpatialClaw + CROSS

1 Call the CROSS pose operator (step 1)

```
p = tools.Ops.camera_pose_at(recon, T_initial,
    t=9.0, fps=30.0)
```

Support: translation = [2.2535, 1.412, 1.5189] in "the frame of the given initial pose"; t_asked_s = t_used_s = 9.0.

2 Compare aligned poses (step 3)

```
error = d_m.value + 0.01 * d_deg.value
```

A: 1.7726 m, 26.40°, error 2.0366
B: 0.2588 m, 17.44°, error 0.4332
Best candidate: B, error 0.4332

Aligns the pose first, then compares translation and rotation.

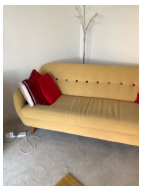
B: (2.00, 1.43, 1.47) m | correct

Reference: B: (2.00, 1.43, 1.47) m

Figure 9: Pose-estimation correction through context augmentation (top) and SpatialClaw (bottom). Both baselines choose A; both CROSS runs compose the camera motion with the initial pose and choose B.

Absolute distance

Q: Measuring from the closest point of each object, what is the direct distance between the couch and the radiator (in meters)?



Couch · 15 s



Radiator · 39 s

BASELINE · Predicted context

Supplied predicted geometry

Couch center: (1.7266, 2.0085, -0.8698) m.
Radiator center: (-1.4430, 3.2944, -0.8519) m.
Center distance = 3.4206 m.

Model reasoning (excerpts)

"To find the closest points, we need to consider the distance between the closest points on the two objects. But since we don't have the full 3D bounding boxes, maybe the question expects the distance between the centers."

"3.42 is approximately 3.4 m."

Substitutes center distance for the requested closest-surface gap.

3.4 m | score 0.0

OURS · Predicted context + CROSS

1 Bind roles and compute surface gap

couch: sofa_000, sofa_001;
radiator: radiator_000.
SURFACE_GAP computes candidate gaps;
REDUCE_INSTANCES(min) selects the minimum.

2 Gate and append derived context

Decision: APPLY; ROLE_AMBIGUOUS.
closest_surface_distance_m = 2.1655
Reported uncertainty = 0.0933 m.

3 Model uses the derived quantity

"Yes, the computed value is 2.1655, so the answer is 2.17."

Uses the surface-gap operator on the same predicted geometry.

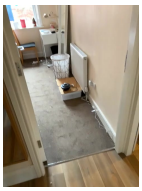
2.17 m | score 1.0

Reference: 2.1 m

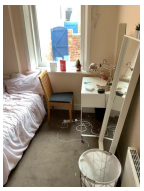
Figure 10: Absolute-distance correction. The baseline uses center distance and returns 3.4 m; CROSS derives the closest-surface gap and returns 2.17 m against a 2.1 m reference. The gate retains the ambiguous couch-role warning while reducing candidate pairs by minimum gap.

Route planning

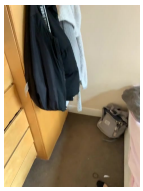
Q (abridged): Start at the backpack, facing the pillow on the bed. [Turn], go forward to the laundry basket, [turn], then go forward to the chair. Choose the two turns.



Laundry basket · 0 s



Bed / chair · 13 s



Backpack · 31 s

BASELINE · Predicted context

Model reasoning (excerpt)

"Assuming that the first turn (from the backpack, facing the pillow) is a turn left, then moving forward reaches the laundry basket. Then, turn right to face the chair. Hence, option A."

Missing explicit route state

The model assumes a left-first route, then a right turn. The excerpt does not compute either turn from the current position and heading.

Assumes turns without explicitly tracking the arrival heading.

A: left, right | incorrect

OURS · Predicted context + CROSS

1 Bind start, facing and waypoints

Start: backpack; facing: pillow on bed.
Waypoints: laundry basket, then chair.

2 Compute turns and advance state

EGO_BASIS / LOCAL_BEARING / CLASSIFY_SECTOR; ADVANCE_STATE updates the arrival heading before the next turn.
turn_tokens = [turn right, turn left]
n_wildcards = 0
Decision: APPLY; ROLE_AMBIGUOUS.

3 Model returns the matching option

"So step 1: turn right. Step 3: turn left. So the answer is C: Turn Right, Turn Left."

Supplies a turn sequence computed with heading-state updates.

C: right, left | correct

Reference: C: turn right, turn left

Figure 11: Route-planning correction. The baseline selects A (left, right); CROSS supplies the sequence computed with heading-state updates and the model selects C (right, left). Multiple bed and chair tracks remain in the bound roles.

Object counting

Q: How many pink pillow(s) are in the scene?



Pink pillows · 37 s

BASELINE · Predicted context

Supplied predicted instances

pink_pillow_000
pink_pillow_001
pink_pillow_002
pink_pillow_003

Model reasoning (excerpt)

"Yes, that's four entries. So the number of pink pillows is 4."

Force-derived diagnostic

REDUCE_INSTANCES(count) = 4.
Exposing this derived count still produces 4 (score 0.0). Counting the predicted tracks preserves the error.

Treats four predicted tracks as four distinct physical pillows.

4 pillows | score 0.0



Pink pillows · 43 s

OURS · Predicted context + CROSS

1 Gate the predicted counting input

Decision: FALLBACK_DIRECT.
The deployed counting policy routes this question to direct video reasoning.

2 Answer from video

"So that's two."

3 Compare with the diagnostic

Video fallback returns 2 (score 1.0), whereas the force-derived run returns 4. The gain comes from routing away from the predicted instance count.

Recovers the count through video fallback; no track deduplication.

2 pillows | score 1.0

Reference: 2 pillows

Figure 12: Object-counting recovery through video fallback. The baseline counts four predicted pink-pillow tracks; the deployed gate routes to video and the model returns the reference count of two. Forcing the derived count still returns four, distinguishing the routing benefit from instance reduction.