REMEMBERING ISN'T UPDATING: MOTION-GUIDED SPATIAL REASONING IN VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spatial reasoning requires understanding spatial relationships and keeping them consistent across changing viewpoints. Yet retaining past observations does not ensure that a vision-language model can interpret them in the reference frame required by a question. We propose FrameShift, a geometry-guided method for spatial reasoning that represents camera motion in both the initial and final reference frames. This dual-reference representation preserves complementary spatial information about the same trajectory. A learned readout uses the question to combine evidence from the two reference frames. FrameShift then combines question relevance with geometric confidence to control the contribution of spatial evidence. The resulting spatial tokens are integrated with visual and language representations for answer generation. We further introduce geometric supervision and consistency objectives that connect continuous spatial information with language predictions. FrameShift improves overall accuracy on SAW-Bench by 7.19 percentage points and relative-direction accuracy by 22.30 points over the base model. These results highlight the value of representing the same trajectory in complementary reference frames. They support using motion and question context together to interpret spatial relationships in the appropriate coordinate system as the viewpoint changes. Code is available at `https://anonymous.4open.science/r/FrameShift-0368/`.

1 INTRODUCTION

Spatial reasoning requires understanding where objects are, how they relate to one another, and how these relationships change across observations. For a vision-language model (VLM), this means combining visual evidence with a question to reason about locations, directions, and distances. Image-language pretraining and visual instruction tuning provide a foundation for this task (Radford et al., 2021; Liu et al., 2023b). SpatialVLM and SpatialRGPT improve spatial grounding through targeted supervision and depth information (Chen et al., 2024; Cheng et al., 2024). More recent methods incorporate geometric structure into VLMs: Spatial-MLLM uses VGGT-derived features, while VLM-3R encodes scene context and camera motion as geometric tokens (Wu et al., 2025; Fan et al., 2026).

Despite this progress, dedicated evaluations reveal persistent errors in spatial relations, object orientation, and reasoning across views (Liu et al., 2023a; Kamath et al., 2023; Ma et al., 2025). Video-based evaluation further exposes the difficulty of combining sequential observations into a coherent understanding of space (Yang et al., 2025). A central challenge is to interpret earlier observations from a changed viewpoint. Figure 1(a) illustrates this requirement: a stationary table is initially in front of an observer, but lies to their left after a right turn. The object has not moved, yet its relative direction has changed. Access to the earlier observation alone does not resolve this change; the model must also account for the intervening motion. This motivates the distinction between remembering an observation and updating its spatial interpretation.

Visual memory methods improve access to past observations (He et al., 2024; Song et al., 2024), and spatial memory models also incorporate geometric representations (Liu et al., 2026; Hu et al., 2025). We examine whether a previously observed object's current direction can be recovered from intermediate VLM features when the object is no longer visible. Figure 1(b) measures the angular error of this readout, while Figure 1(c) measures its direction classification accuracy. At the prese-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

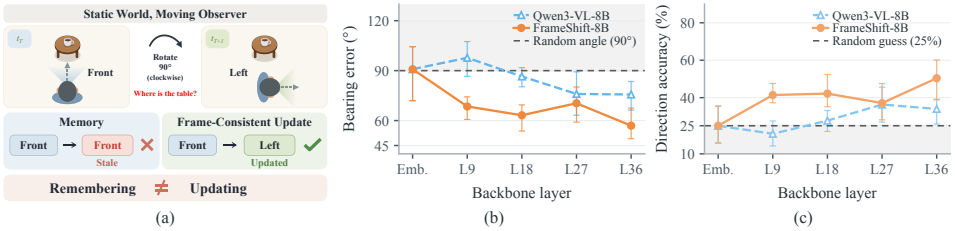


Figure 1: **Remembering is not updating.** (a) Viewpoint changes require updating object directions; (b,c) FrameShift improves the readout of current spatial information from backbone features.

lected middle layer, the base model's error is close to the random-angle reference and its accuracy is near chance. These probe results indicate limited accessibility of current-direction information, rather than identifying a specific memory failure (Hewitt & Liang, 2019; Belinkov, 2022). They motivate our question: how can motion evidence help a VLM interpret spatial relationships in the reference frame requested by the question?

To address this question, we propose FrameShift. It expresses camera motion in the initial and final reference frames and uses a question-conditioned readout to combine them. A question-conditioned readout combines evidence from the two frames, while a soft gate uses question relevance and geometric confidence to control its contribution. The resulting spatial tokens enter the VLM alongside its visual and language representations. Geometric supervision and training consistency objectives connect the continuous motion evidence with answer generation. The comparisons in Figure 1(b,c) show that FrameShift lowers bearing error and improves direction accuracy at the preselected layer, making current spatial information more accessible in the backbone features.

Our main contributions are as follows:

- We propose FrameShift, a geometry-guided method for spatial reasoning that uses camera motion to help VLMs interpret spatial relationships across changing viewpoints.
- We introduce a dual-reference trajectory representation that expresses the same motion in the initial and final reference frames. This representation makes complementary spatial relationships available to the model without committing to a single reference frame in advance.
- We design a question-conditioned readout with confidence gating that combines evidence from both reference frames and controls its contribution using question relevance and geometric confidence. Geometric supervision and consistency objectives connect this spatial evidence with language predictions.
- We demonstrate improved spatial reasoning on SAW-Bench through evaluation across all task categories with a matched local baseline. FrameShift improves overall accuracy by 7.19 percentage points and relative-direction accuracy by 22.30 points, showing particular gains in reasoning about relative directions.

2 Related Work

2.1 Visual Spatial Reasoning

Visual spatial reasoning studies locations, directions, distances, and relationships from visual observations. Models such as Flamingo, BLIP-2, and Qwen3-VL connect visual representations with language generation, providing a foundation for visual question answering (Alayrac et al., 2022; Li et al., 2023; Bai et al., 2025). CLEVR and GQA provide structured questions for compositional reasoning (Johnson et al., 2017; Hudson & Manning, 2019). VSR examines spatial relations and reference-frame ambiguity, while What'sUp isolates relational changes with controlled object configurations (Liu et al., 2023a; Kamath et al., 2023). BLINK and 3DSRBench extend evaluation to multiview perception and 3D relationships (Fu et al., 2024; Ma et al., 2025). Video benchmarks further require reasoning across observations: VSI-Bench evaluates spatial understanding from videos (Yang et al., 2025), SAW-Bench covers six spatial task categories (Li et al., 2026).

Methods for improving spatial reasoning develop both targeted supervision and geometric grounding. SpatialVLM generates qualitative and quantitative spatial supervision (Chen et al., 2024), SpatialRGPT adds region-level grounding and depth information (Cheng et al., 2024), and SpatialReasoner-R1 optimizes spatial reasoning through fine-grained preferences (Shen et al., 2025). 3D-LLM incorporates point-cloud features and localization information (Hong et al., 2023). Spatial-MLLM combines semantic features with a VGGT-initialized spatial encoder (Wu et al., 2025), while VLM-3R represents scene structure and camera motion using spatial and view tokens (Fan et al., 2026). Building on these advances, FrameShift focuses on expressing trajectory geometry in complementary reference frames and reading it according to the question.

2.2 Spatial Representations and Memory

Spatial representations organize observations in a form that supports reasoning across views. DROID-SLAM jointly refines camera poses and depth (Teed & Deng, 2021). DUSt3R aligns predicted pointmaps in a common reference frame, and MASt3R improves visual correspondence estimation with learned local features (Wang et al., 2024; Leroy et al., 2024). VGGT predicts camera and scene attributes, while VGGT-SLAM 2.0 aligns its submaps incrementally (Wang et al., 2025; Maggio & Carlone, 2026). These methods establish geometric correspondences between observations and recover camera motion, providing a basis for interpreting spatial relationships as the viewpoint changes.

Memory extends spatial reasoning to observations that are no longer visible. Scene Memory Transformer attends to stored observation embeddings for long-horizon decisions (Fang et al., 2019). MA-LMM maintains memory banks for online video understanding, and MovieChat compresses video tokens into sparse memory (He et al., 2024; Song et al., 2024). For spatial tasks, 3DLLM-Mem retrieves spatial and temporal features from episodic memory using working-memory queries (Hu et al., 2025). VLM² combines working and episodic memory with view-consistent geometric representations (Liu et al., 2026). These approaches couple access to past observations with the organization of spatial evidence. FrameShift complements this direction by using question relevance and geometric confidence to select trajectory evidence expressed in the reference frames needed for an answer.

3 Preliminaries

3.1 Problem Definition

Spatial reasoning involves understanding object locations, directions, distances, and spatial relationships. We evaluate this ability through question answering (QA). Given a video V with N frames, we extract an ordered sequence of $N_k \leq N$ keyframes, $\mathcal{K} = (I_0, \ldots, I_{N_k-1})$, where I_i denotes the i-th selected frame in temporal order, indexed from zero. The model receives $\mathcal{K}$ and a spatial question q and predicts an answer $a = f_\theta(\mathcal{K}, q)$. We focus on the performance of VLMs on spatial reasoning tasks, with f_θ denoting a VLM and θ its parameters.

3.2 Representation Diagnostics

Spatial reasoning across viewpoints requires recovering the current direction of previously observed objects, even when they are no longer visible. We examine this ability using matched linear probes on intermediate backbone features. At the preselected layer 18, Qwen3-VL-8B has a bearing error of $86.50°$, close to the $90°$ random-angle reference (Figure 1(b)), and a direction accuracy of 27.73%, near the 25% chance level (Figure 1(c)). These results indicate that current-direction information is difficult to recover from the base model's features. To address this gap, we introduce FrameShift, which conditions VLM representations on motion-derived geometry. The comparisons in Figure 1(b,c) show improved readout with FrameShift: bearing error decreases to $63.18°$ and direction accuracy increases to 42.19%.

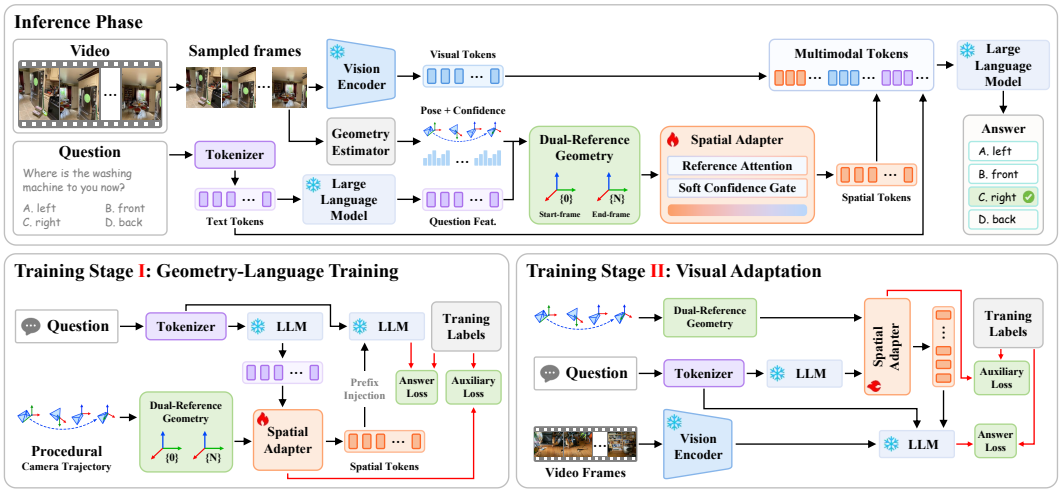


Figure 2: **Overview of FrameShift.** At inference phase, shared video frames provide visual tokens and estimated camera poses; a question-conditioned adapter encodes geometry in the initial and final reference frames into spatial tokens, which are prepended to the multimodal input for answer generation. Training phase combines geometry-language learning on procedural trajectories without images with visual adaptation using video and ground-truth camera geometry, updating only the spatial adapter and auxiliary heads.

4 METHOD

4.1 OVERVIEW

FrameShift provides a VLM with continuous geometric evidence for spatial reasoning (Figure 2). It has two branches that share the same video frames. The visual branch extracts visual tokens. The geometry branch estimates camera motion and expresses the trajectory in both the initial and final camera frames. A lightweight adapter uses the question to combine evidence from these two reference frames. It then scales the spatial tokens by question relevance and geometric confidence. The tokens enter the frozen VLM alongside the visual and text inputs. The VLM alone produces the answer; geometry provides evidence rather than an answer rule. Only the adapter and its auxiliary training heads are updated during training.

4.2 CAMERA POSE ESTIMATION

We sample each video once to obtain the keyframe sequence $\mathcal{K}$ and retain its frame indices and timestamps. This single sequence is fed to both branches: the visual branch applies the VLM's visual preprocessing and encodes the frames into visual tokens, while the geometry branch applies VGGT's image preprocessing to the same frames. The branches therefore share frame identities and temporal order, while using their own input representations. Sampling settings are given in Section 5.1. Optional reference images are included in the VLM's visual input.

The geometry branch processes the keyframes in temporal order using a frozen VGGT-SLAM 2 estimator. The first group contains up to nine frames. Each subsequent group retains the previous group's last frame and adds up to 8 new frames; the final group may be shorter. For example, a sequence of 17 keyframes is processed as $(I_0, \ldots, I_8)$ followed by $(I_8, \ldots, I_{16})$. VGGT predicts camera poses, depth, and geometric confidence within each group. The shared frame links adjacent local reconstructions, allowing VGGT-SLAM to align their scales and coordinate systems and optimize the pose graph. We then extract one camera-to-world pose per keyframe, remove duplicate overlap entries, and express the trajectory relative to the first camera. Confidence scores are derived from the support of reliable geometric predictions. This process runs independently for each video clip, without assuming a common coordinate system across clips.

4.3 CONTINUOUS DUAL-REFERENCE REPRESENTATION

A trajectory can be described from its starting viewpoint or its final viewpoint. The same displacement has different coordinates in these two frames. We retain both descriptions so that the model can use the reference frame relevant to the question.

For a clip with keyframes $t = 0, \ldots, T$, let $P_t = [R_t, c_t]$ be the camera-to-world pose. Here, R_t is the camera rotation and c_t is its position. We compute $P_0^{-1}P_t$ and $P_T^{-1}P_t$, which express each pose in the initial and final camera frames, respectively. We divide translations by the total path length $L = \sum_{t=1}^{T} \|c_t - c_{t-1}\|$, using $\max(L, 10^{-6})$ as the denominator. This reduces dependence on the arbitrary translation scale of the estimated trajectory. For numerical stability, we subtract camera centers before applying the reference rotation.

We summarize each clip at 8 evenly spaced temporal anchors, including its endpoints. Each anchor has a 53-dimensional feature vector f_k. It contains four transforms: the final pose relative to the start, the start relative to the final pose, and the anchor relative to each endpoint. Each transform contributes a 6D rotation representation (Zhou et al., 2019), a three-dimensional normalized translation v, and the three-dimensional vector $v/\sqrt{\|v\|^2 + 10^{-4}}$. The remaining five features encode the net-displacement-to-path-length ratio, path progress, normalized time, anchor confidence, and clip confidence. These features describe camera motion, not an explicit map of individual objects. They remain continuous and are not converted into direction labels at inference.

4.4 QUESTION-CONDITIONED READOUT AND SOFT GATING

Different questions may require different reference frames. We therefore use the question to read the geometric representation. The frozen language backbone encodes the question into a contextual vector $\mathbf{q}$. Answer options are excluded from this readout but remain available to the final answering model.

A multilayer perceptron E maps each feature vector f_k to a 256-dimensional representation. We add a learned slot embedding e_k to identify the anchor. Question-conditioned feature modulation (FiLM) (Perez et al., 2018) then scales and shifts this representation. We also encode the two endpoint transforms as reference vectors m_0 and m_T. The question predicts soft attention weights α_0 and α_T, which sum to one. Their weighted readout is added to each anchor representation. The resulting spatial token is

$$z_k = gW_o \left(\mathrm{FiLM}(E(f_k) + e_k, \mathbf{q}) + W_r \sum_{r \in \{0,T\}} \alpha_r(\mathbf{q}) m_r \right),$$

where W_r projects the reference readout. The output map W_o applies layer normalization and projects the result to the language embedding space. The soft gate is

$$g = \sigma(a(\mathbf{q}))\, c_{\mathrm{global}}.$$

Here, $\sigma(a(\mathbf{q}))$ is the learned question-relevance score and $c_{\mathrm{global}} \in [0, 1]$ is the clip's geometric confidence. The gate is shared by the 8 tokens within a clip. Low relevance or low confidence reduces their contribution without selecting a different answer path. Confidence is multiplied once in this gate and is also included in the input features.

4.5 SPATIAL-TOKEN FUSION AND ANSWER GENERATION

The adapter produces eight spatial tokens per clip. We concatenate tokens from different clips in input order and insert them as a prefix to the VLM input. Visual tokens, question text, answer options, and optional reference images are retained. The frozen VLM generates every answer through the same decoder. A common answer parser handles all task categories within the evaluation protocol.

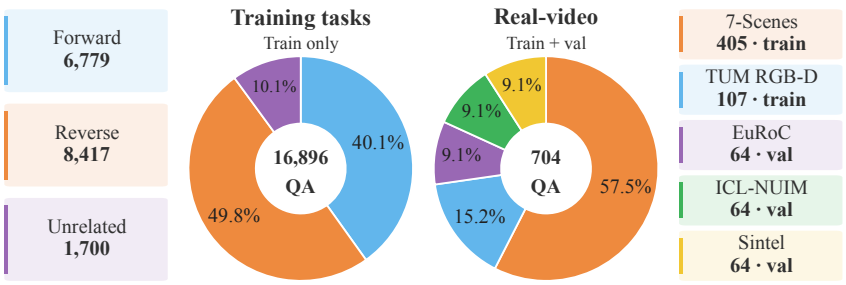


Figure 3: Composition of the training data by source and task.

4.6 Training Objectives

We train the lightweight adapter while keeping the visual encoder, language backbone, and motion estimator frozen. Training uses procedural numerical QA and video QA derived from public ground-truth camera trajectories. Numerical QA connects geometry to language; video QA trains the adapter to use geometry together with visual evidence.

The final visual adaptation objective is

$$\mathcal{L}_{visual} = \mathcal{L}_{CE} + 0.1\mathcal{L}_{rel} + 0.05\mathcal{L}_{recon} + 0.2\mathcal{L}_{ref} + 0.5\mathcal{L}_{angle} + 0.5\mathcal{L}_{pairedKL}.$$

The answer loss $\mathcal{L}_{CE}$ is cross-entropy on the target answer. Three auxiliary terms supervise question relevance ($\mathcal{L}_{rel}$), endpoint reconstruction ($\mathcal{L}_{recon}$), and reference attention ($\mathcal{L}_{ref}$). These terms help the adapter preserve useful geometry and read it according to the question.

The angular loss $\mathcal{L}_{angle}$ penalizes probability assigned to directions that disagree with the continuous target bearing. The consistency term $\mathcal{L}_{pairedKL}$ aligns the answer distribution with a frozen numerical teacher that receives geometry without video. We apply this term only when the teacher's top prediction matches the independent training label. Thus, an incorrect teacher prediction does not become a distillation target. Direction labels and option-to-direction mappings are used only to compute training losses, not to select answers at inference.

We also replay four numerical examples per update, with loss weight 0.5. Every fourth update includes an unrelated visual question. For this question, a KL loss encourages the adapter's answer distribution to match the original backbone, and a relevance loss encourages a low gate.

5 Experiments

5.1 Experimental Setup

5.1.1 Training Data

We train on 16,384 numerical and 512 video QA examples (Figure 3). Numerical QA uses in-house 3D camera trajectories, including closed and near-closed paths, with answers derived from endpoint geometry and no visual input. Forward-reference, reverse-reference, and unrelated questions account for 40.1%, 49.8%, and 10.1% of training data, respectively.

Video QA uses eight-frame clips with trajectory-derived labels: 405 training questions from 7-Scenes (Shotton et al., 2013) and 107 from TUM RGB-D (Sturm et al., 2012). Validation includes 512 numerical questions and 192 video questions, with 64 each from EuRoC (Burri et al., 2016), ICL-NUIM (Handa et al., 2014), and Sintel (Butler et al., 2012).

5.1.2 Training Setting

We use Qwen3-VL-8B-Instruct (Bai et al., 2025) as the backbone and frozen VGGT-SLAM 2 (Maggio & Carlone, 2026) for camera pose estimation. The visual encoder, language backbone, and motion estimator remain frozen; only the spatial adapter and its auxiliary heads are trained, totaling

Table 1: Spatial reasoning accuracy (%) on SAW-Bench. Unless otherwise specified, all models process videos at 2 fps (frames per second). Bold and underlined scores indicate the best and second-best results within each model comparison block, respectively, including ties. Our local 4B and 8B baselines and FrameShift models use the corresponding Qwen3-VL Instruct backbones; the published Qwen3-VL results use Thinking variants.

Model Name	Overall Accuracy	Self-Localization	Relative Direction	Route Shape	Reverse Route Plan	Spatial Memory	Spatial Affordance
Reference Baselines							
Human Level	91.55	94.00	89.39	97.62	93.01	88.50	79.01
Chance Level (Random)	27.49	34.00	25.90	21.43	27.51	28.00	56.17
Proprietary Multimodal Foundation Models							
Gemini 3 Flash	**53.89**	48.50	**41.13**	64.84	**61.57**	**66.00**	**70.99**
Gemini 2.5 Pro	50.80	45.50	37.05	**66.12**	51.53	**66.00**	66.05
Gemini 3 Pro	45.97	**50.00**	38.61	52.01	36.24	63.00	61.73
GPT-5.2	41.04	45.50	25.78	50.55	44.98	63.00	62.96
Gemini 2.5 Flash	39.79	44.00	25.30	57.33	37.99	49.00	46.91
GPT-5 Mini	33.80	43.50	27.46	36.08	22.27	56.00	49.38
Open-Source Multimodal Foundation Models							
Qwen3-VL 235B-A22B	**41.40**	43.50	**33.41**	**53.11**	**30.13**	46.00	54.32
Qwen3-VL 32B	38.58	44.00	29.14	48.35	29.26	52.00	52.47
Qwen3-VL 30B-A3B	36.55	39.00	29.62	43.04	27.07	**54.00**	50.00
Qwen2.5-VL 32B	36.46	**53.00**	28.06	41.03	24.89	45.00	54.94
Qwen2.5-VL 72B	36.17	51.50	26.74	41.76	25.33	45.00	**56.79**
Qwen3-VL 8B	36.12	40.00	27.82	46.70	23.58	48.00	48.77
LLaVA OneVision 72B (32 frames)	33.70	39.00	22.30	46.15	24.45	41.00	52.47
InternVL3 8B (32 frames)	33.70	43.50	26.86	36.45	27.95	46.00	48.15
LLaVA-Video 72B (32 frames)	32.98	32.50	23.86	43.04	24.45	41.00	53.70
InternVL3 14B (32 frames)	32.69	49.00	17.27	45.05	24.02	**54.00**	49.38
Qwen2.5-VL 7B	31.48	38.50	19.06	43.59	26.20	38.00	49.38
LLaVA-NeXT-Video 32B (32 frames)	31.24	41.00	24.46	35.35	22.27	34.00	51.23
LLaVA-Video 7B (32 frames)	30.81	41.00	25.06	32.78	24.45	32.00	49.38
InternVL2 40B (8 frames)	30.13	45.00	17.75	38.28	24.89	32.00	54.32
InternVL2 8B (32 frames)	29.84	43.00	14.99	41.94	24.89	40.00	50.00
LLaVA OneVision 7B (32 frames)	29.45	34.50	20.26	34.80	25.33	44.00	49.38
InternVL3 38B (8 frames)	27.71	35.50	23.50	37.55	24.45	46.00	51.23
Qwen3-VL-4B-Instruct	33.85	**43.00**	23.74	46.70	**23.58**	32.00	46.91
FrameShift-4B	**36.46**	42.00	**30.94**	**48.35**	16.16	**33.00**	**48.77**
Qwen3-VL-8B-Instruct	31.14	**43.50**	18.11	**39.74**	**24.02**	**49.00**	**53.09**
FrameShift-8B	**38.34**	39.50	**40.41**	37.18	18.78	46.00	**53.09**

3.53M parameters. The adapter has a hidden width of 256 and produces eight 4,096-dimensional spatial tokens per clip. Following the preceding numerical and visual training stages, the final adaptation stage uses AdamW with learning rate 3×10^{-5}, weight decay 0.01, gradient clipping at 1.0. Each update uses one video example and four numerical replay examples; an unrelated visual question is added every fourth update. We select the checkpoint at step 256 using the mean of video and numerical validation accuracy.

5.1.3 Evaluation Setting

We evaluate on all 2,071 questions in SAW-Bench (Li et al., 2026). Videos use the official 2-fps preparation. The visual and geometry branches receive the same selected frames. The local baseline uses the same backbone, visual inputs, questions, and decoding settings. We report answer accuracy, count invalid responses as incorrect, and estimate paired differences with 10,000 source-connected cluster bootstrap resamples.

5.2 Evaluation on Spatial Reasoning

Table 1 compares FrameShift with open- and closed-source models across six SAW-Bench tasks. FrameShift-8B achieves 38.34% overall accuracy, outperforming reported models of comparable size and surpassing Qwen2.5-VL-72B (36.17%). On Relative Direction, it reaches 40.41%, exceeding all open-source models reported in the benchmark paper, including Qwen3-VL-235B-A22B (33.41%). Published models differ in checkpoint variants and inference settings. Against the matched local 8B backbone, FrameShift improves overall accuracy by 7.19 percentage points and

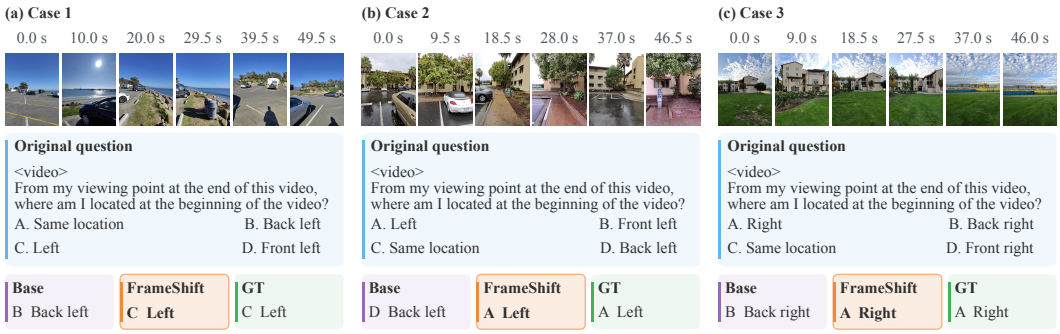


Figure 4: Qualitative examples from SAW-Bench. Qwen3-VL-8B gives incorrect spatial answers, while FrameShift-8B answers correctly.

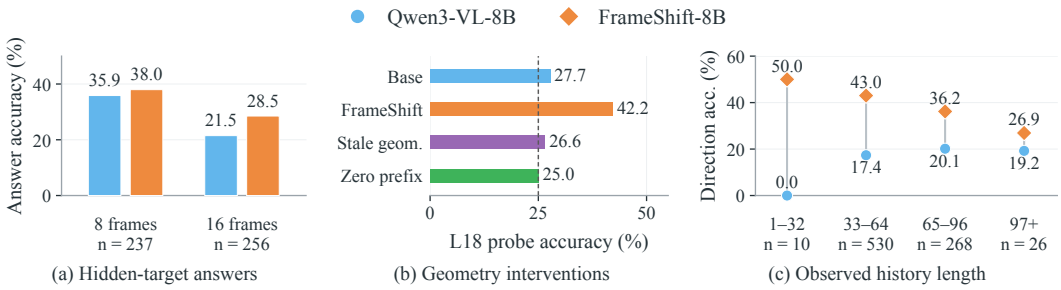


Figure 5: Geometric reasoning analysis: (a) hidden-target answer accuracy on a 256-episode ARKit diagnostic; (b) current-direction probe accuracy under geometry interventions; and (c) SAW direction accuracy grouped by observed frame count. The dashed line in (b) marks 25% chance; connections in (c) compare models within each group.

Relative Direction by 22.30 points. Improvements also hold for the 4B version: overall accuracy rises from 33.85% to 36.46% (+2.61 points), and Relative Direction from 23.74% to 30.94% (+7.19 points). These results show gains at both model sizes, although their training histories differ. The gains are concentrated in direction reasoning, while the other task scores decrease or remains unchanged. These drops may occur because our training focuses on direction reasoning, while other tasks require different spatial cues.

Qualitative evidence. Figure 4 presents unmodified SAW video frames, questions, options, and predictions for reverse-reference questions answered incorrectly by the base model and correctly by FrameShift. These questions concern the initial camera location in the final viewpoint.

5.3 Geometric Reasoning Analysis

We study whether FrameShift uses geometric input, how this affects hidden-target reasoning, and how performance changes with observation length.

Does the Model Use Geometry? On 512 numerical questions, FrameShift achieves 93.16% accuracy with correct geometry, compared with 37.70% under shuffled geometry and 41.80% without geometry (Table 2). These inputs contain motion features and text, without images or videos. Unrelated-question accuracy remains at 100% in all three conditions. This supports dependence on the supplied geometric content. With real video frames and ground-truth camera poses, FrameShift answers 178 of 192 development questions correctly (92.71%): 98.67% for the forward relation and 86.46% for the reverse relation. These questions concern camera endpoints. Both development sets contributed to checkpoint selection, so these scores diagnose geometry use rather than establish held-out performance with estimated poses or hidden-object tracking.

Hidden-Target Reasoning. We evaluate generated answers at the 8-frame and 16-frame endpoints of the same 256 ARKit (Baruch et al., 2021) episodes. At the final endpoint, all targets are hidden.

As shown in Figure 5(a), FrameShift improves answer accuracy from 21.48% to 28.52%, a gain of 7.03 percentage points. The paired 95% confidence interval is [-0.53, 14.87]. At the earlier endpoint, hidden-target accuracy rises from 35.86% to 37.97%.

Does Current Geometry Matter? We compare FrameShift with stale geometry and a zero-valued spatial prefix on the same final hidden-target queries. Stale geometry keeps the trained adapter but freezes estimated camera poses at the initial pose. Zero prefix keeps the token positions but removes their values. As shown in Figure 5(b), at backbone layer 18, current-direction probe accuracy falls from 42.19% to 26.56% and 25.00%, respectively. Bearing error also increases from 63.18° to about 89°. Thus, the improved direction readout depends on the geometric content, rather than the presence of extra tokens alone.

Performance across Observation Lengths. Figure 5(c) shows that FrameShift outperforms the local backbone in every SAW Relative Direction frame-count group. The gain is 25.66 percentage points for 33–64 frames and 16.04 points for 65–96 frames. Each group contains different questions. This comparison describes performance across observed input lengths rather than isolating the effect of a longer history.

Table 2: Answer accuracy (%) under different geometric inputs.

Geometry	Forward	Reverse	Unrelated	Overall
Numerical QA ($n = 512$)				
None	31.22	38.67	100.00	41.80
Shuffled	30.24	31.25	100.00	37.70
Correct	**94.15**	**91.02**	100.00	**93.16**
Real video ($n = 192$)				
GT poses	98.67	86.46	100.00	92.71

Table 3: Inference-time module interventions on all 834 SAW-Bench Relative Direction questions. All rows use the released FrameShift-4B weights without retraining. Changes are relative to Full.

Variant	Accuracy (%) ↑	Δ (pp)
FrameShift-4B (Full)	30.94	–
w/o initial reference	28.30	-2.64
w/o reference attention	25.30	-5.64
w/o reference selection	27.70	-3.24
w/o confidence multiplier	30.46	-0.48
w/o soft gating	29.62	-1.32

5.4 Ablation Study

Table 3 compares inference-time interventions on all 834 SAW-Bench Relative Direction questions using the same FrameShift-4B weights, visual inputs, and decoding settings. Full achieves 30.94% accuracy. Removing the initial reference lowers accuracy to 28.30%, supporting the use of complementary reference frames. Removing reference attention causes a larger drop to 25.30%, showing the contribution of the reference-readout path. Replacing question-conditioned reference weights with an equal mixture yields 27.70%, supporting question-dependent reference selection. Setting the confidence multiplier to one gives 30.46%, while disabling both gate multipliers gives 29.62%.

6 Conclusion

We introduced FrameShift, a geometry-guided method for spatial reasoning in vision-language models. FrameShift represents camera motion in both the initial and final reference frames. A question-conditioned readout combines these complementary representations. We train a lightweight spatial adapter while keeping the VLM and geometry estimator frozen. On SAW-Bench, the 8B model improves overall and Relative Direction accuracy by 7.19 and 22.30 percentage points over the matched baseline. Representation diagnostics further show that current-direction information becomes easier to recover from intermediate features, and geometry interventions confirm that this readout depends on the supplied motion evidence.

Limitations. (1) FrameShift represents camera motion but does not maintain explicit object states or align geometry across independent clips. Future work will explore object tracking and cross-clip alignment to support persistent spatial representations. (2) Training focuses on short clips and direction reasoning, and the gate does not prevent performance drops on other tasks. Future work will expand training to longer videos and more diverse spatial tasks.

AI USE STATEMENT

In the process of developing this work, we utilized LLMs for grammar correction and occasionally as a rewriting tool. In addition, we involved LLMs in the process of data visualization.

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, Mikoł aj 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. Curran Associates, Inc., 2022. doi: 10.52202/068431-1723. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/960a172bc7fbf0177ccccbb411a7d800-Paper-Conference.pdf.

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

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Yuri Feigin, Peter Fu, Thomas Gebauer, Daniel Kurz, Tal Dimry, Brandon Joffe, Arik Schwartz, and Elad Shulman. ARKitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile RGB-d data. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. URL https://openreview.net/forum?id=tjZjv_qh_CE.

Yonatan Belinkov. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*, 48(1):207–219, 2022. doi: 10.1162/coli_a_00422. URL https://aclanthology.org/2022.cl-1.7/.

Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W. Achtelik, and Roland Siegwart. The EuRoC Micro Aerial Vehicle Datasets. *The International Journal of Robotics Research*, 2016. URL https://journals.sagepub.com/doi/abs/10.1177/0278364915620033.

Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A Naturalistic Open Source Movie for Optical Flow Evaluation. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*, pp. 611–625. Springer-Verlag, 2012. doi: 10.1007/978-3-642-33783-3_44. URL https://doi.org/10.1007/978-3-642-33783-3_44.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing Vision-Language Models with Spatial Reasoning Capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14455–14465, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Chen_SpatialVLM_Endowing_Vision-Language_Models_with_Spatial_Reasoning_Capabilities_CVPR_2024_paper.html.

An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. SpatialRGPT: Grounded Spatial Reasoning in Vision-Language Models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 135062–135093. Curran Associates, Inc., 2024. doi: 10.52202/

079017-4293. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f38cb4cf9a5eaa92b3cfa481832719c6-Paper-Conference.pdf.

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Peihao Wang, Huaizhi Qu, Shijie Zhou, Dilin Wang, Zhicheng Yan, Hongyu Xu, Justin Theiss, Tianlong Chen, Jiachen Li, Zhengzhong Tu, Zhangyang Wang, and Rakesh Ranjan. VLM-3R: Vision-Language Models Augmented with Instruction-Aligned 3D Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31054–31065, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Fan_VLM-3R_Vision-Language_Models_Augmented_with_Instruction-Aligned_3D_Reconstruction_CVPR_2026_paper.html.

Kuan Fang, Alexander Toshev, Li Fei-Fei, and Silvio Savarese. Scene Memory Transformer for Embodied Agents in Long-Horizon Tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 538–547, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Fang_Scene_Memory_Transformer_for_Embodied_Agents_in_Long-Horizon_Tasks_CVPR_2019_paper.html.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal Large Language Models Can See but Not Perceive. In *Computer Vision – ECCV 2024*, pp. 148–166, 2024. doi: 10.1007/978-3-031-73337-6_9. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/3356_ECCV_2024_paper.php.

Ankur Handa, Thomas Whelan, John McDonald, and Andrew J. Davison. A Benchmark for RGB-D Visual Odometry, 3D Reconstruction and SLAM. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1524–1531, 2014. doi: 10.1109/ICRA.2014.6907054.

Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. MA-LMM: Memory-Augmented Large Multimodal Model for Long-Term Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13504–13514, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/He_MA-LMM_Memory-Augmented_Large_Multimodal_Model_for_Long-Term_Video_Understanding_CVPR_2024_paper.html.

John Hewitt and Percy Liang. Designing and Interpreting Probes with Control Tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1275. URL https://aclanthology.org/D19-1275/.

Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3D-LLM: Injecting the 3D World into Large Language Models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 20482–20494. Curran Associates, Inc., 2023. doi: 10.52202/075280-0900. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/413885e70482b95dcbeeddc1daf39177-Paper-Conference.pdf.

Wenbo Hu, Yining Hong, Yanjun Wang, Leison Gao, Zibu Wei, Xingcheng Yao, Nanyun Peng, Yonatan Bitton, Idan Szpektor, and Kai-Wei Chang. 3DLLM-Mem: Long-Term Spatial-Temporal Memory for Embodied 3D Large Language Model. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, 2025. doi: 10.52202/085713-2278. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/61f527a737e4ba61f3e10d6c3f0c4b55-Paper-Conference.pdf.

Drew A. Hudson and Christopher D. Manning. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6693–6702, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Hudson_GQA_A_New_Dataset_for_Real-World_Visual_Reasoning_and_Compositional_CVPR_2019_paper.html.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross Girshick. CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Johnson_CLEVR_A_Diagnostic_CVPR_2017_paper.html.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. What's "up" with vision-language models? Investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 9161–9175. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.568. URL https://aclanthology.org/2023.emnlp-main.568/.

Vincent Leroy, Yohann Cabon, and Jerome Revaud. Grounding Image Matching in 3D with MASt3R. In *Computer Vision – ECCV 2024*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/9080_ECCV_2024_paper.php.

Chuhan Li, Rilyn Han, Joy Hsu, Yongyuan Liang, Rajiv Dhawan, Jiajun Wu, Ming-Hsuan Yang, and Xin Eric Wang. SAW-Bench: Learning Situated Awareness in the Real World, 2026. URL https://arxiv.org/abs/2602.16682.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 19730–19742. PMLR, 2023. URL https://proceedings.mlr.press/v202/li23q.html.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual Spatial Reasoning. *Transactions of the Association for Computational Linguistics*, 11:635–651, 2023a. doi: 10.1162/tacl_a_00566. URL https://aclanthology.org/2023.tacl-1.37/.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual Instruction Tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023b. doi: 10.52202/075280-1516. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

Zuntao Liu, Yi Du, Taimeng Fu, Shaoshu Su, Cherie Ho, and Chen Wang. Vision-Language Memory for Spatial Reasoning. In *Computer Vision – ECCV 2026*, pp. 297–316, Cham, 2026. Springer Nature Switzerland. ISBN 978-3-032-37356-4. doi: 10.1007/978-3-032-37356-4_17. URL https://link.springer.com/chapter/10.1007/978-3-032-37356-4_17.

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3DSRBench: A Comprehensive 3D Spatial Reasoning Benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6924–6934, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Ma_3DSRBench_A_Comprehensive_3D_Spatial_Reasoning_Benchmark_ICCV_2025_paper.html.

Dominic Maggio and Luca Carlone. VGGT-SLAM 2.0: Real-time Dense Feed-forward Scene Reconstruction, 2026. URL https://arxiv.org/abs/2601.19887.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Yifan Shen, Yuanzhe Liu, Jingyuan Zhu, Xu Cao, Xiaofeng Zhang, Yixiao He, Wenming Ye, James Rehg, and Ismini Lourentzou. Fine-Grained Preference Optimization Improves Spatial Reasoning in VLMs. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 17929–17960. Curran Associates, Inc., 2025. doi: 10.52202/085713-0606. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/1a17a06de88cf77f25cda0da91615a54-Paper-Conference.pdf.

Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. URL https://www.cv-foundation.org/openaccess/content_cvpr_2013/html/Shotton_Scene_Coordinate_Regression_2013_CVPR_paper.html.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. MovieChat: From Dense Token to Sparse Memory for Long Video Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18221–18232, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Song_MovieChat_From_Dense_Token_to_Sparse_Memory_for_Long_Video_CVPR_2024_paper.html.

Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A Benchmark for the Evaluation of RGB-D SLAM Systems. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 573–580, 2012. doi: 10.1109/IROS.2012.6385773.

Zachary Teed and Jia Deng. DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 16558–16569. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/89fcd07f20b6785b92134bd6c1d0fa42-Paper.pdf.

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 (CVPR)*, pp. 5294–5306, 2025. 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 Jerome Revaud. DUSt3R: Geometric 3D Vision Made Easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20697–20709, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wang_DUSt3R_Geometric_3D_Vision_Made_Easy_CVPR_2024_paper.html.

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, Main Conference, pp. 13569–13597. Curran Associates, Inc., 2025. doi: 10.52202/085713-0457. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/142b166c8b18b43774c4f0e0c1e9948e-Paper-Conference.pdf.

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in Space: How Multimodal Large Language Models See, Remember, and Recall Spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yang_Thinking_in_Space_How_Multimodal_Large_Language_Models_See_Remember_CVPR_2025_paper.html.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

A Appendix

A.1 Implementation and Reproducibility

Shared input and token construction. The visual encoder and geometry estimator receive the same selected frames. For each clip, camera poses are expressed relative to both the initial and final frames. Eight endpoint-preserving anchors form an 8×53 feature array: 24 endpoint-transform features, 24 anchor-transform features, three trajectory/time ratios, and two quality values. The adapter combines these features with the question representation and produces eight spatial tokens. For multiple clips, their spatial tokens are concatenated in clip order. The spatial prefix precedes the original text and visual sequence, whose media order is retained.

Question encoding and multimodal answer generation share one frozen Qwen language backbone. Text passes through its tokenizer and embedding layer; spatial tokens are injected as continuous embeddings. Native multimodal position handling is retained. Only the adapter and auxiliary training heads are optimized. The 8B configuration uses a 256-dimensional adapter width, 4,096-dimensional output tokens, and 3.53M trainable parameters.

Data construction. Procedural trajectories contain 8, 16, 32, 64, or 128 poses. Positions combine endpoint displacement with lateral and vertical variation; orientations include yaw, pitch, and roll. Random world-frame transformations provide coordinate variation. Closed and near-closed paths are included. Forward and reverse labels are computed from the two endpoint reference frames. Direction options use eight angular sectors; a net-displacement-to-path-length ratio below 0.075 receives the same-location label. Language expansion varies question wording and option order while retaining geometric labels. Numerical questions have no visual inputs.

The final inventory contains 16,384 numerical questions (6,554 forward, 8,192 reverse, and 1,638 unrelated) and 512 video questions (225 forward, 225 reverse, and 62 unrelated). Video training uses eight-frame clips from 7-Scenes and TUM RGB-D; validation contains 64 questions each from EuRoC, ICL-NUIM, and Sintel. Numerical validation contains 512 questions.

Training objective. In the final adaptation stage, one visual example contributes

$$\mathcal{L}_v = \mathcal{L}_{\mathrm{QA}} + 0.1\mathcal{L}_{\mathrm{rel}} + 0.05\mathcal{L}_{\mathrm{rec}} + 0.2\mathcal{L}_{\mathrm{ref}} + 0.5\mathcal{L}_{\mathrm{angle}} + 0.5\mathcal{L}_{\mathrm{pair}}.$$

Answer cross-entropy supervises the correct option token over the full vocabulary. Relevance uses binary cross-entropy. Reconstruction uses mean squared error on the 24 endpoint features. Reference selection uses two-way cross-entropy on forward/reverse questions. Angular risk is the option-probability-weighted angular cost, with the correct option assigned zero cost. Paired distillation matches the frozen numerical teacher's full-vocabulary distribution only when its top-1 answer agrees with the training label.

Each update also includes four numerical replay examples, weighted by 0.5, using answer, relevance, reconstruction, and reference-selection losses. Every fourth update adds an unrelated visual question with baseline-preservation KL and relevance supervision toward zero. AdamW uses learning rate 3×10^{-5}, weight decay 0.01, and gradient clipping at 1.0. The selected final-stage checkpoint is step 256, chosen by mean numerical and video validation accuracy. Training-stage labels in the pipeline summarize functional phases of multiple continuation runs.

Statistical protocol. SAW comparisons use paired predictions and 10,000 source-connected cluster bootstrap resamples. Probe comparisons resample scenes; simulation comparisons keep noise seeds nested within scenes. Intervals are percentile 95% intervals. Invalid answers remain in the denominator.

A.2 Supplementary SAW-Bench Results

Gains at two model sizes. The matched comparisons in Section 5.2 use all 2,071 questions, including 834 Relative Direction questions. FrameShift-4B improves overall accuracy from 33.85% to 36.46% and direction accuracy from 23.74% to 30.94%. FrameShift-8B improves the corresponding scores from 31.14% to 38.34% and from 18.11% to 40.41%. The overall paired gains have source-cluster intervals of [0.92, 4.30] and [5.12, 9.21] percentage points, respectively. Each

Table 4: Complete local 8B SAW-Bench task accuracy (%). Both models use the same inputs and decoding settings.

Task	Questions	Base	FrameShift
Self-Localization	200	43.50	39.50
Relative Direction	834	18.11	40.41
Route Shape	546	39.74	37.18
Reverse Route Plan	229	24.02	18.78
Spatial Memory	100	49.00	46.00
Spatial Affordance	162	53.09	53.09

comparison holds the local backbone and evaluation inputs fixed; training histories differ between scales.

Complete task breakdown. Table 4 complements the published-model comparison with all local task scores and denominators. The strongest gain is in direction reasoning, the task directly aligned with the geometric training objective. The 8B method corrects 296 baseline errors across SAW-Bench. Together with the complete paired outcomes, this supports a net gain of 149 correct answers. Published Thinking models and local Instruct models remain separate comparison groups.

A.3 Representation Diagnostics and Controls

Geometric motivation. For camera-to-world pose $P_t = [R_t, c_t]$, a stationary point X has camera-relative coordinates $x_t = R_t^\top (X - c_t)$. Its coordinates after motion satisfy

$$x_{t+1} = R_{t+1}^\top R_t x_t + R_{t+1}^\top (c_t - c_{t+1}).$$

The relation motivates probing the object's direction in the current frame. Our probes measure how readily this information can be recovered from the frozen backbone representation.

Protocol. We use 8- and 16-frame prefixes from 768 ARKit trajectories, giving 1,536 queries. Probe fitting, development, and testing contain 920/104/512 queries from 49/10/10 mutually exclusive scenes. These are project-defined splits. VGGT estimates each prefix independently; object coordinates and GT camera poses are used only for diagnostic labels. Features are extracted at the final prefill token from layers 0, 9, 18, 27, and 36. Layer 18 was fixed as the primary layer before the diagnostic inference. Ridge probes predict the sine and cosine of bearing; regularization is selected on development scenes. The two models use separately fitted probes under the same protocol.

Current-direction readout. The primary endpoint contains 256 hidden-target queries balanced across the four directions. FrameShift raises probe accuracy from 27.73% to 42.19%, while shuffled-label controls yield 23.13% and 23.28% for the base and FrameShift representations. Mean angular error decreases from 86.50° to 63.18°, with a paired reduction of 23.32° [14.75, 32.92]. The categorical and angular metrics describe the same probe predictions. Generated answers are evaluated separately, using a 16-token diagnostic budget rather than the 96-token benchmark budget.

Geometry interventions. Stale geometry fixes estimated poses to the initial pose. Zero prefix retains the spatial-token positions but replaces their values with zero. Both reuse the FrameShift-trained probe without refitting. Table 5 reports paired accuracy gains over the base and both controls. These results support the contribution of geometric content to current-direction readout. The interventions hold model weights fixed and are distinct from training a Static Memory model.

A.4 Prediction and Correction: Simulation Details

All 6,000 ARKit-derived episodes retain their recorded trajectories and visibility masks. A visible target supplies its GT position plus Gaussian measurement noise with standard deviation 0.1 m. Translation/yaw perturbations per step use standard deviations of (0 m, 0 degrees), (0.01 m, 1 degree), or (0.05 m, 3 degrees). Three fixed seeds produce paired perturbations across estimators. The 18,000 seed–episode pairs per setting represent 6,000 unique episodes.

Table 5: FrameShift current-direction probe gains at layer 18 on 256 hidden-target queries. Intervals use paired scene-cluster bootstrap.

Compared with	Gain (pp)	95% interval
Base	14.45	[9.60, 22.39]
Stale geometry	15.62	[8.14, 21.19]
Zero prefix	17.19	[9.43, 28.66]

Table 6: Combined-minus-motion-only hidden-accuracy differences in the analytic simulation. Noise seeds remain nested within scenes during paired bootstrap.

Motion noise	Gain (pp)	95% interval
Exact	0.86	[0.33, 1.52]
Low	0.95	[0.45, 1.54]
High	1.71	[1.03, 2.44]

Observation-only replaces its state when a measurement is available and otherwise holds the last coordinates. Motion-only initializes from the first measurement and propagates with camera motion. The combined estimator uses the same prediction and performs a Kalman correction on visible frames. Noise assumptions and trajectory inputs are identical within each comparison.

Table 6 reports the combined estimator's hidden-accuracy gain over motion-only at every tested noise level. Under high motion noise, the combined estimator reaches 88.32% hidden accuracy, compared with 86.61% for motion-only and 45.50% for observation-only. At target reappearance, its mean angular error decreases from 8.96° before correction to 3.12° afterward. These paired results illustrate the complementary roles of prediction during occlusion and correction after new observations. They concern analytic estimators with specified sensor noise, not the learned FrameShift model or an implemented correction module.

A.5 Complete Module Interventions

Table 7 supplements Section 5.4 with every measured condition on the same 834 SAW-Bench Relative Direction questions. All comparisons use the trained weights without retraining and identical visual inputs and decoding.

Reference pathways. Removing the initial reference zeros its endpoint and anchor feature blocks and reads only the final-reference memory. Removing reference attention bypasses the reference-readout residual. Removing reference selection retains the residual with equal weights of 0.5. On 4B, these conditions yield 28.30%, 25.30%, and 27.70%, compared with 30.94% for Full. The results support complementary reference information and question-conditioned reading. The corresponding measured 8B initial-reference and equal-mixture scores are 37.89% and 39.45%, versus 40.41% for Full; their paired intervals are reported in the source data.

Modulation and gate definitions. FiLM removal bypasses question-conditioned scaling and shifting. Query and confidence interventions fix the corresponding output multiplier to one; the geometry confidence remains available in the encoder inputs. Disabling soft gating fixes both multipliers to one. The complete table separates these operations so that reference reading, modulation, and token scaling can be evaluated independently. Results quantify sensitivity of the released computations rather than the accuracy of separately retrained architectures. Direction-only gate measurements are not a test of preservation on other task categories.

A.6 Additional Qualitative Evidence

The additional SAW example asks for the starting camera location from the final viewpoint. For source question direction:177, the correct answer is Left. The local baseline answers Back left, whereas FrameShift answers Left. The input contains 94 prepared frames over 46.5 seconds. The

Table 7: Complete measured inference-time module interventions on all 834 SAW-Bench Relative Direction questions. Each scale uses its released weights without retraining; changes are relative to Full at the same scale. Dashes indicate unmeasured conditions. Bold marks the highest measured accuracy within each scale.

	FrameShift-4B		FrameShift-8B	
Variant	Acc. (%)	Δ (pp)	Acc. (%)	Δ (pp)
Full FrameShift	30.94	0.00	40.41	0.00
w/o initial reference	28.30	-2.64	37.89	-2.52
w/o reference attention	25.30	-5.64	–	–
w/o reference selection	27.70	-3.24	39.45	-0.96
w/o FiLM	**31.06**	+0.12	–	–
w/o query multiplier	30.94	+0.00	–	–
w/o confidence multiplier	30.46	-0.48	–	–
w/o soft gating	29.62	-1.32	**51.20**	+10.79

case illustrates the reverse-reference relation addressed by the method using authentic benchmark frames and unchanged question/answer records.

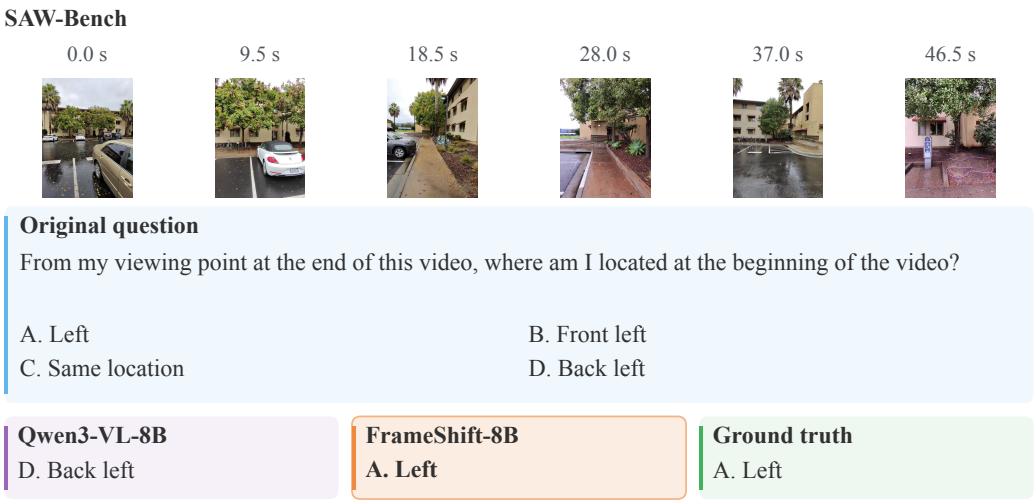


Figure 6: An additional SAW-Bench reverse-reference example. The baseline predicts Back left and FrameShift correctly predicts Left. Six chronological frames are shown with the original question, options, and model answers.

The qualitative inventory selects the three longest reverse-reference questions corrected by FrameShift, breaking ties by sample ID. These are illustrative selections, not estimates of success frequency. The complete 834-question length breakdown in Section 5.3 includes every direction question: the 33–64 and 65–96 frame groups show gains of 25.66 and 16.04 percentage points. These groups contain different questions, so their curves describe observed input lengths rather than within-question changes.

A.7 Failure Case and Evaluation Boundaries

Hidden-target example. The illustrated ARKit development case shows a television monitor initially in front and later behind the camera. Both models answer Left. The target is absent from the final view, so success requires linking its earlier visual identity and location to the current reference frame. This example relates to the geometric-representation limitation in **Limitations** in Section 6: the current model provides camera-motion features but has no explicit persistent object state.

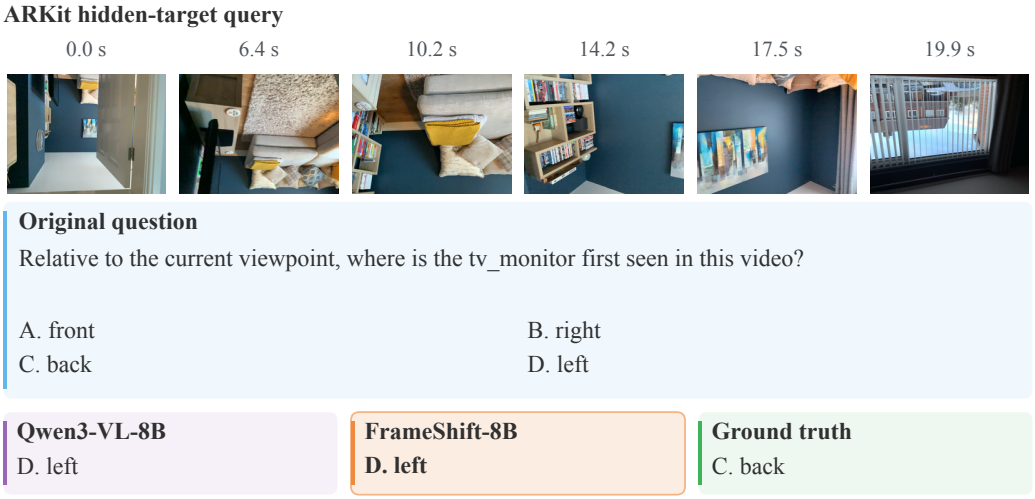


Figure 7: ARKit hidden-target failure case. The correct final direction is Back; both the base model and FrameShift answer Left. Six chronological frames retain the original image orientation; the question, options, and answer letters follow the recorded evaluation.

Scope of the object-level evidence. The 256-query diagnostic in Section 5.3 and the full formal ARKit test use different protocols. The completed 6,000-episode test contains 1,671 visible and 4,329 hidden queries. Base/FrameShift Visible, Hidden, and Overall accuracies are 5.45/6.10%, 31.44/25.76%, and 24.20/20.28%, respectively, with no invalid answers. Thus the diagnostic gain is not a full-test improvement. The full-test paired overall difference is -3.92 percentage points with scene-cluster interval $[-10.55, 1.43]$. This result belongs with the representation and cross-task generalization boundaries already stated in **Limitations** in Section 6; it does not isolate a causal failure mechanism.

Current-bearing probes also should not be read as direct proof of retaining and updating object state. Initial-bearing error is 14.01° for both models, compared with 13.55° for the training-mean control. Among 192 changed-direction queries, generated old-direction answers occur in 1.04% of base responses and 6.25% of FrameShift responses. These controls delimit the spatial-updating interpretation described in **Limitations** in Section 6 while leaving the measured current-direction readout gains intact.

Cross-task gate scope. The complete SAW task table and module interventions provide the measured context for the cross-task limitation. Removing both gate multipliers raises 8B direction accuracy to 51.20%, so the current gate cannot be claimed to maximize direction accuracy. Whether its use improves preservation elsewhere requires matched gate-on/off evaluation on those tasks. The simulation in Appendix A.4 similarly supports a prediction–correction principle, not a learned correction mechanism in the current model.