LEARNING TO THINK AND EXPLORE IN 3D WORLD

Anonymous authors
Paper under double-blind review

ABSTRACT

We study how vision-language models (VLMs) can learn to think and actively explore in 3D space. Instead of assuming informative input and providing multi-view images, videos, or complete 3D representations directly to the model, we preserve the native visual encoding of VLMs, i.e., a single image, one at a time, just like humans, and treat the 3D representation only as an environment with which the model can interact through actions (e.g., move left) and observations (an rendered image). Each rollout begins from a fixed initial first-person view together with a visual question. The model can then iteratively reason, navigate, and reorient its viewpoint, with each action inducing a new observation, before committing to a final answer. Following an initial stage of trajectory-supervised fine-tuning, we optimize the interactive policy with reinforcement learning (RL) using rewards driven primarily by final-answer correctness. Crucially, RL provides no direct supervision over which viewpoints the model should visit or which exploration actions it should take. Instead, the model must learn an effective exploration policy solely from the extent to which its interactions improve the performance.

Across six 3D benchmarks, our policy achieves the highest average among active models (54.1%), improving its base Qwen model by 22.9% and surpassing proprietary Gemini 3.6 Flash by 2.6% under the same interaction protocol. The result includes four out-of-domain benchmarks that shows generalized improvement of 7.2% on average. It also outperforms every evaluated open-source 3D model despite using 50–70% fewer input frames and lower-resolution renders. Further analysis shows that RL improves accuracy across all interaction levels ($+3.2$–6.2%) and induces question-conditioned exploration: the policy searches for occluded targets, briefly confirms visible ones, and gathers multiple viewpoints for relational comparisons. These results Our RL design enables VLMs not only to reason over observed evidence, but also to decide when and how to acquire the evidence needed for generalizable 3D question answering.

1 INTRODUCTION

Despite substantial progress in 3D vision-language models, much of the existing work studies spatial reasoning under observations that are specified *a priori*, providing models with complete RGB-D scans(Azuma et al., 2022; Zhu et al., 2025), multi-view images(Yang et al., 2025), videos(Zheng et al., 2025; Yang et al., 2026), or explicit 3D scene representations such as point clouds and meshes (Hong et al., 2023; Huang et al., 2024b). Yet under partial observability, a fundamental challenge remains: the model must not only reason over the evidence currently available, but also determine how to act in order to acquire the evidence that is missing. This motivates the central question of our work: *can a vision-language model learn to actively explore and acquire task-relevant visual evidence as an integral part of the 3D reasoning process?*

We address this question by first formulating reasoning in 3D as an interactive visual decision process under partial observability. Rather than providing the VLM with a complete 3D representation, the 3D scene is kept external to the VLM and serves as an interactive environment that the model accesses through sequential action-observation interactions. Starting from a fixed initial egocentric view and a visual question, the model alternates between reasoning and interaction, iteratively acquiring observations through navigation and viewpoint control before committing to a final answer. This entire episode of interaction constitutes a policy rollout, in which the model jointly determines *what information to acquire, how to interact with the environment, and how to reason over the*

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.

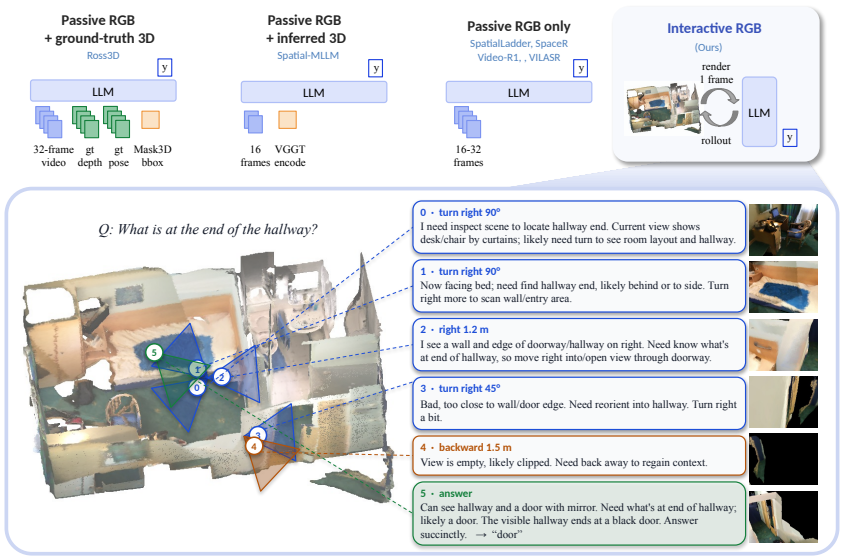


Figure 1: **Interactive 3D reasoning from a single view.** Unlike passive systems that receive a fixed collection of observations, our policy alternates reasoning, camera actions, and newly rendered views before producing an answer.

accumulated evidence. To learn a policy that integrates visual information acquisition, spatial interaction, and reasoning, we adopt a two-stage training procedure: trajectory-supervised fine-tuning first establishes the interaction protocol, after which reinforcement learning optimizes the full interactive policy over the rollout described above using rewards driven primarily by final-answer correctness. During reinforcement learning, no direct supervision is provided for intermediate viewpoints, actions, or exploration trajectories. Effective exploration must instead be learned through credit assignment from task performance. This formulation enables us to examine whether task-level supervision alone can induce task-directed visual information acquisition during 3D reasoning.

Experiments across six 3D benchmarks show that outcome-based RL makes active reasoning both more accurate and more adaptive. Our mixed-data policy improves its matched SFT initialization by 5.5% on average, achieves the highest average among active models (54.1%), and exceeds proprietary Gemini 3.6 Flash by 2.6% under the same interaction protocol; notably, Gemini is the strongest model by a substantial margin under passive evaluation. Our policy also surpasses every evaluated open-source 3D method despite using 50–70% fewer input frames and lower-resolution renders. The gains extend beyond the training domains: performance on four out-of-domain benchmarks rises by 7.2% on average. Together, these results show that RL enables VLMs not only to reason over what they observe, but also to decide when and how to acquire the visual evidence needed for strong, generalizable 3D question answering.

2 Related Work

3D VQA, scene understanding and spatial reasoning. Prior work on 3D vision-language models operates over observations assumed to be complete and directly presented to the model, including 3D representations such as point clouds or meshes (Hong et al., 2023; Huang et al., 2024a;b), fixed multi-view images (Zhu et al., 2025; Zheng et al., 2025), with reconstruction supervision (Wang et al., 2025a). More closely related are methods that actively select viewpoints. Chain-of-View performs training-free coarse-to-fine view selection through a discrete camera-action interface (Zhao et al., 2026). Think3D equips models with 3D reconstruction tools and optimizes their use with RL (Zhang et al., 2026b), ViewSuite distills self-exploration into a view graph (Wang et al., 2026). In contrast, we keep the 3D scene external to the VLM as an interactive environment: the model observes only egocentric 2D views and learns, jointly with reasoning, to navigate and acquire the visual evidence needed to answer the question.

Multimodal agentic reasoning. Reinforcement learning (e.g., GRPO (Shao et al., 2024)) has emerged as an effective paradigm for improving reasoning through verifiable rewards (Guo et al., 2025). It has further been applied to agentic systems with multi-turn interaction and tool use (Jiang et al., 2026; Wang et al., 2025b), as well as to multimodal reasoning (Feng et al., 2025). These methods primarily manipulate or revisit visual evidence already contained in the provided input. In contrast, our model learns to explore the 3D environment, actively acquiring new visual evidence while jointly optimizing reasoning and interaction in a single rollout.

Exploration in embodied and 3D environments. EmbodiedQA introduced navigation-based question answering (Das et al., 2017), later extended to open-vocabulary evaluation by OpenEQA (Majumdar et al., 2024). Recent benchmarks further show that active observation can outperform passive viewing and expose failures in spatial exploration and long-horizon planning (Hong et al., 2026; Zhang et al., 2026c;a; Wei et al., 2026; Wang et al., 2026). We abstract embodiment to direct camera control over a scanned 3D scene and jointly learn exploration and reasoning for question answering.

3 METHOD

We study how a VLM can learn to reason in 3D under partial observability. Learning such a policy poses two complementary challenges. The model must first acquire a basic interaction competence in 3D environments before it can learn from the interaction with environments despite the absence of ground-truth exploration trajectory. We therefore adopt a two-stage training procedure. We first perform *trajectory-supervised fine-tuning* using curated interaction demonstrations to initialize a coherent reason–act–observe policy. We then apply *reinforcement learning with verifiable reward* to optimize the policy with rewards driven only by final performance, without supervision of any intermediate viewpoints or actions. This allows the model to learn task-directed exploration end-to-end from the task performance of the evidence it acquires.

3.1 REASONING IN 3D UNDER PARTIAL OBSERVABILITY

Problem formulation. We formalize reasoning in 3D as a partially observable Markov decision process (POMDP) (Kaelbling et al., 1998). A task instance $w = (\mathcal{M}, q, y^\star) \sim \mathcal{D}$ consists of a reconstructed 3D scene $\mathcal{M}$, a question q, and the ground-truth answer $y^\star = g(\mathcal{M}, q) \in \mathcal{Y}$. At step t, the environment state is $s_t = (\mathcal{M}, c_t)$, where $c_t = (e_t, f_t)$ specifies the camera position $e_t \in \mathbb{R}^3$ and viewing direction $f_t \in \mathbb{S}^2$, and c_0 is a fixed initial viewpoint of the scene. The scene $\mathcal{M}$ remains fixed and hidden throughout the episode, while the camera state evolves under the model's actions. The policy observes the rendered egocentric view $o_t = \mathrm{Render}(\mathcal{M}, c_t)$ and, after each action, the resulting camera configuration c_t.

Action space. A navigation action $a_t = (\xi_t, n_t) \in \mathcal{A}_{\mathrm{nav}} = \mathcal{U} \times \{1, \dots, 8\}$ pairs a motion primitive ξ_t with an integer multiplier n_t. The set $\mathcal{U}$ contains six translations (forward/backward, left/right, up/down) and four rotations (yaw and pitch, in either direction), and each primitive ξ has a unit step δ_ξ. Let $r_t = (f_t \times \hat{z})/\|f_t \times \hat{z}\|$ denote the horizontal lateral direction, defined whenever f_t is not vertical. Camera dynamics are deterministic:

$$(e_{t+1}, f_{t+1}) = \begin{cases} (\Pi_{\mathcal{B}}(e_t + n_t \delta_{\xi_t} u_t),\ f_t), & a_t \text{ is a translation,} \\ (e_t,\ \mathrm{Rot}_{k_t}(n_t \delta_{\xi_t}) f_t), & a_t \text{ is a rotation,} \end{cases} \tag{1}$$

where $u_t \in \{\pm f_t, \pm r_t, \pm \hat{z}\}$ and $k_t \in \{\pm \hat{z}, \pm r_t\}$ are the translation direction and rotation axis selected by ξ_t, and $\Pi_{\mathcal{B}}$ projects the camera position onto a box $\mathcal{B}$ enclosing the scene. We augment navigation with the terminal answer actions $\mathcal{A}_{\mathrm{stop}} = \{\textsc{stop}(y) : y \in \mathcal{Y}\}$,

$$\bar{\mathcal{A}} = \mathcal{A}_{\mathrm{nav}} \cup \mathcal{A}_{\mathrm{stop}}. \tag{2}$$

Executing $\textsc{stop}(y)$ terminates the episode and returns y as the prediction. Given the task instance, both the transition and observation processes are deterministic; stochasticity arises only from the policy itself. We consider two parameterizations. Under the *absolute* parameterization, the policy directly predicts a camera pose in the scene coordinate frame, represented by a position and a look-at point. Under the *relative* parameterization defined above, the policy instead predicts an egocentric transformation of the current pose, corresponding to translation along or rotation about the current camera axes. We adopt the latter throughout training and evaluation. Absolute actions require the policy to predict coordinates in a global reference frame that is not directly observed and

whose scale varies across scenes. In contrast, relative actions are scale-invariant, camera-centric, and map naturally to interpretable motions such as "move left" or "turn right". We compare the two parameterizations empirically in Appendix B.2.

History and implicit belief. At step t, the policy conditions on the complete interaction history

$$h_t = (q, o_0, z_0, a_0, c_1, o_1, z_1, \ldots, a_{t-1}, c_t, o_t, z_t), \tag{3}$$

where z_t denotes the model's intermediate reasoning before taking the next action. The policy is therefore $a_t \sim \pi_\theta(\cdot \mid h_t),\ a_t \in \bar{\mathcal{A}}$. Because the scene is hidden but static, the posterior $b_t(\mathcal{M}) = \Pr(\mathcal{M} \mid h_t)$, together with the observed camera configuration c_t and the elapsed step t, forms a sufficient statistic for optimal decision making. In practice, we do not explicitly maintain b_t; instead, the VLM conditions on the full interaction context, which implicitly encodes its belief over the accumulated observations. The reasoning z_t is generated by the policy and belongs to the history, not to the environment state.

Objective. An episode terminates when the policy emits an answer or reaches the interaction budget, with $T = \min\{t : a_t \in \mathcal{A}_{\text{stop}}\} \wedge T_{\max}$; if no answer is emitted within the budget, the prediction y is empty and counted as incorrect. The resulting rollout is $\tau = (h_T, y)$, and we optimize the policy for expected finite-horizon, undiscounted return:

$$\max_\theta J(\theta) = \mathbb{E}_{w\sim\mathcal{D}}\mathbb{E}_{\tau\sim\pi_\theta\otimes\mathcal{E}}\left[R(\tau)\right], \tag{4}$$

where $\mathcal{E}$ denotes the deterministic interactive environment defined above and R is the reward introduced in Sec. 3.3, which depends on the task instance through $y^\star$ and the annotated answer object. We define the answer accuracy of a policy π as

$$\mathrm{Acc}(\pi) = \Pr_{w\sim\mathcal{D},\,\tau\sim\pi\otimes\mathcal{E}}(y = y^\star). \tag{5}$$

This objective jointly evaluates the policy's ability to acquire informative observations and reason over them to produce the correct answer.

We illustrate the necessity of interaction with environment for reasoning under partial observability in Appendix A.

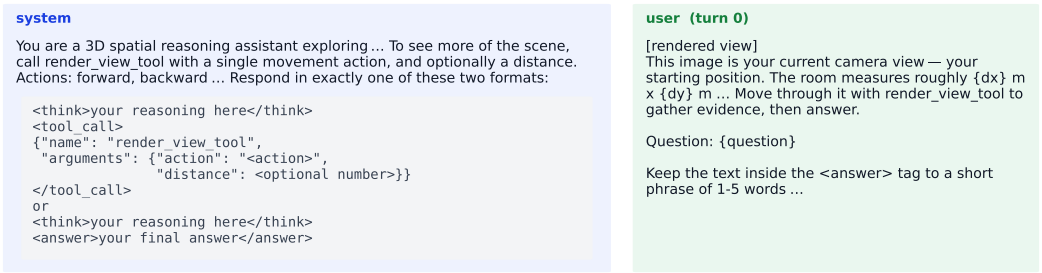


Figure 2: Prompt template overview. The full text is in Appendix D.1.

3.2 Trajectory-Supervised Policy Initialization

We initialize the policy with trajectory-supervised fine-tuning to establish basic multi-step interaction competence in 3D environments before policy optimization with reinforcement learning.

Trajectory curation. We use GPT-5.4 to generate reason–act–observe trajectories in the environment from Sec. 3.1. We discard invalid starting views and retain trajectories with correct answers, valid interactions, well-formed reasoning and responses, and sufficient visible scene geometry; MP3D examples must also observe the evidence required by the question. After answer canonicalization, the corpus contains 5,712 trajectories, 20,603 rendered observations, and 830 scenes. These demonstrations initialize the interaction protocol rather than prescribe a unique exploration strategy.

Supervised objective. Let $\tau^\star \in \mathcal{D}_{\text{traj}}$ denote a curated demonstration, serialized as a token sequence, and let $\mathcal{I}(\tau^\star)$ denote the positions of its model-generated tokens, i.e., reasoning, actions,

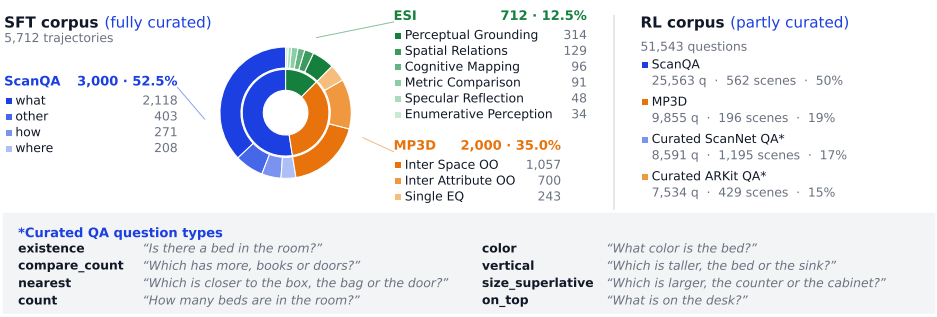


Figure 3: Training data breakdown.

and the final answer. Environment observations and prompt tokens are masked from the loss. We minimize

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{\tau^\star \sim \mathcal{D}_{\mathrm{traj}}} \sum_{\ell \in \mathcal{I}(\tau^\star)} \log \pi_\theta \left(\tau^\star_\ell \mid \tau^\star_{<\ell}\right). \tag{6}$$

The resulting policy initializes the reinforcement learning stage described next. We emphasize that these trajectories provide *policy initialization* rather than canonical exploration targets. Trajectory supervision therefore establishes the interaction protocol and a useful behavioral prior, while the acquisition strategy is subsequently optimized with RL.

3.3 Outcome-Driven Reinforcement Learning

Starting from the trajectory-supervised initialization, we optimize the full interactive policy using outcome-driven reinforcement learning. Each rollout follows the decision process in Sec. 3.1

Training data. RL requires only a question–answer pair and a valid initial viewpoint, allowing substantially broader training data than trajectory supervision. We collect 51,543 questions from ScanQA, MP3D, and generated QAs over ScanNet and ARKitScenes.

For each initial condition $x = (q, o_0)$, we sample a group of G rollouts $\{\tau_i\}_{i=1}^{G}$ from $\pi_{\theta_{\mathrm{old}}} \otimes \mathcal{E}$, all starting from x. Because actions determine subsequent observations, rollouts from the same initial condition may acquire substantially different evidence, enabling direct comparison of alternative information-acquisition strategies under the same task objective.

Verifiable Reward. For a rollout τ with final answer y, the reward is a weighted sum of answer-correctness, evidence-acquisition, and output-format terms:

$$R(\tau) = \underbrace{\lambda_{\mathrm{judge}} R_{\mathrm{judge}} + \lambda_{\mathrm{EM}} R_{\mathrm{EM}}}_{\text{answer correctness}} + \underbrace{\lambda_{\mathrm{tool}} R_{\mathrm{judge}} \min\{n_{\mathrm{tool}}, N_{\max}\} + \lambda_{\mathrm{gnd}} R_{\mathrm{gnd}}}_{\text{evidence acquisition}} - \underbrace{\lambda_{\mathrm{fmt}} R_{\mathrm{fmt}}}_{\text{output format}}, \tag{7}$$

where $\lambda_{(\cdot)} \geq 0$ are specified, together with $N_{\max}$ in Sec. 4.1. All terms depend on τ; we omit this for brevity. $R_{\mathrm{judge}} \in \{0, 1\}$ is the verdict of an LLM judge that, given the question, determines whether y is semantically equivalent to the ground-truth answer $y^\star$. $R_{\mathrm{EM}} \in \{0, 1\}$ indicates whether y exactly matches any of the annotated GT answers; it complements the judge by favoring the concise canonical answers on which the benchmark metric is computed. In the tool-use term, n_{tool} denotes the number of tool calls in τ, of which at most $N_{\max}$ are credited; because the term is multiplied by R_{judge}, interaction is rewarded only in rollouts that end in a correct answer. The grounding term $R_{\mathrm{gnd}} \in \{-1, 0, 1\}$ uses the annotated 3D location of the object of interest: 1 if the object is visible in any rendered view of τ, -1 otherwise, and 0 for questions without an annotated answer object. Finally, $R_{\mathrm{fmt}} \in \{0, 1\}$ indicates a malformed output, namely a missing or empty answer. None of these terms prescribes a particular intermediate viewpoint, action, or trajectory; the grounding term only verifies whether the answer object was observed.

Policy optimization. We optimize the policy with Group Relative Policy Optimization (GRPO) (Shao et al., 2024). For each rollout group, we standardize the outcome reward to obtain

$$\hat{A}_i = \frac{R(\tau_i) - \mathrm{mean}_j\, R(\tau_j)}{\mathrm{std}_j\, R(\tau_j)}. \tag{8}$$

The same trajectory-level advantage $\hat{A}_i$ is assigned to all model-generated tokens in τ_i, while environment observations are excluded from optimization. GRPO then optimizes

$$\mathcal{J}_{\mathrm{GRPO}}(\theta) = \mathbb{E}_{w\sim\mathcal{D},\,\{\tau_i\}\sim\pi_{\theta_{\mathrm{old}}}\otimes\mathcal{E}}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|\mathcal{I}(\tau_i)|}\sum_{\ell\in\mathcal{I}(\tau_i)}\min\left(\frac{\pi_\theta(\tau_{i,\ell}\mid\tau_{i,<\ell})}{\pi_{\theta_{\mathrm{old}}}(\tau_{i,\ell}\mid\tau_{i,<\ell})}\hat{A}_i,\right.\right.$$
$$\left.\left.\mathrm{clip}\left(\frac{\pi_\theta(\tau_{i,\ell}\mid\tau_{i,<\ell})}{\pi_{\theta_{\mathrm{old}}}(\tau_{i,\ell}\mid\tau_{i,<\ell})},1-\epsilon,1+\epsilon\right)\hat{A}_i\right)\right] - \beta D_{\mathrm{KL}}\left(\pi_\theta\,\|\,\pi_{\mathrm{ref}}\right), \tag{9}$$

where $\mathcal{I}(\tau_i)$ denotes the model-generated token positions of τ_i and $\tau_{i,<\ell}$ its preceding tokens, including the rendered observations, ϵ is the clipping range, and β weights a KL penalty toward a reference policy π_{ref}. Because reasoning and interaction are generated by the same autoregressive policy, the episode-level reward assigns credit across the entire rollout. The policy can therefore learn not only how to reason over acquired evidence, but also which evidence to acquire in the first place.

4 Experiment

4.1 Setup

Models and training. All experiments use Qwen3.5-9B. We compare a ScanQA-only lineage with our final mixed-data lineage, trained on ScanQA, MP3D, and ESI trajectories for supervised fine-tuning (SFT), followed by GRPO on ScanQA, MP3D, and generated ARKit questions. We report the instruction-tuned base, SFT checkpoint, and RL policy to isolate the contribution of each stage. SFT uses 5,712 teacher trajectories, and GRPO uses 51,543 questions with learning rate 10^{-6}, rollout temperature 0.7, group size $G = 8$, clipping range $\epsilon = 0.2$, and KL coefficient $\beta = 0.005$. Reward weights are $(\lambda_{\mathrm{judge}}, \lambda_{\mathrm{EM}}, \lambda_{\mathrm{fmt}}, \lambda_{\mathrm{tool}}, \lambda_{\mathrm{gnd}}) = (0.3, 0.2, 0.2, 0.05, 0.1)$, and $N_{\max} = 12$; since this equals the interaction budget $T_{\max}$, every tool call on a correct rollout is rewarded. Grounding supervision applies only to the ScanNet portion with annotated answer boxes; no intermediate view or action labels are used. Prompts are provided in Appendices D.1 and D.2.

Evaluation. All systems receive one 640×480 initial render, at most twelve tool calls; baselines use the same harness. Qwen3-30B-A3B-Instruct used as answer judge. We evaluate on SPBench-SI (1,009 questions, 170 scenes), SPBench-MV (319, 127), ARKit (1,000, 80), ScanQA validation (4,675, 71), SQA3D (3,519, 67), Real-3DQA (1,485, 67), and MP3D (1,000, 60). These datasets range from single-room question answering to multi-scene spatial reasoning. The single-draw standard error is approximately 0.7% on ScanQA and 1.4% on the 1,000-question sets; differences within these margins are not interpreted as meaningful.

5 Main Results

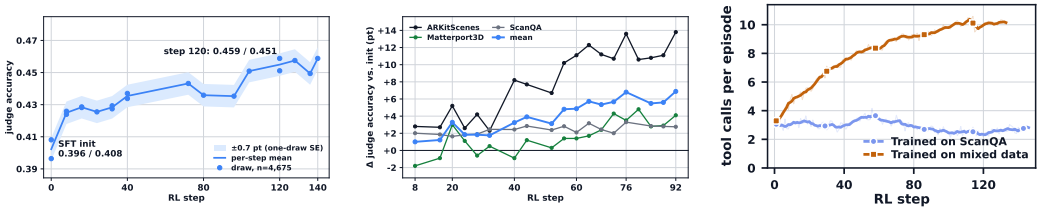
Main Result 1: RL delivers strong active performance and cross-domain transfer. Table 1 shows that our mixed-data policy achieves the highest average among active models (54.1%). Under the same render-and-move interface, it surpasses proprietary Gemini 3.6 Flash by 2.6% and leads it on four of six benchmarks. It also outperforms every evaluated open-source 3D model, including InternVL3.5-38B by 10.6%, despite using 50–70% fewer input frames and lower-resolution renders. Our result demonstrates the effectiveness of learned interaction rather than making a protocol-independent state-of-the-art claim. Against the matched mixed-data SFT initialization, RL raises the six-benchmark average from 48.6% to 54.1% (+5.5%) and improves every benchmark. In the ScanQA-only training, it raises the average from 47.8% to 50.6% (+2.8%) and ScanQA from 39.4% to 45.5% (+6.1%), the best active ScanQA result.

Mixed-data RL also improve four out-of-domain benchmarks by 7.2% on average, led by ARKit (+13.7%) and SPBench-MV (+11.3%), versus 2.2% across the two in-domain benchmarks. Broader training coverage turns outcome optimization into a generalizable 3D reasoning signal rather than a narrow source-domain specialization.

Main Result 2: mixed-domain RL steadily improves performance and incentivizes exploration. Figure 4(a) shows that ScanQA-only training raises validation accuracy from 0.402 to 0.455 at step 120 (+5.3%). The largest jump occurs early, but performance continues to improve after step 40

model	ScanQA	SQA3D	SPB MV	SPB SI	MP3D	ARKit	Exp.
Active: renders its own views through the tool							
Ours, trained on mixed data	42.5	47.7	50.5	41.2	62.9	79.6	*54.1*
Gemini 3.6 Flash	39.1	53.8	45.5	44.4	53.4	73.0	*51.5*
Ours, trained on ScanQA	45.5	49.8	47.6	42.3	50.7	67.6	*50.6*
Ours, mixed-data SFT init	39.3	47.4	39.2	37.7	61.8	65.9	*48.6*
Ours, ScanQA SFT init	39.4	46.0	47.3	41.2	44.7	68.3	*47.8*
InternVL3.5-38B	30.9	41.5	41.1	41.6	50.1	55.8	*43.5*
Qwen3.5-9B	36.7	33.5	21.3	24.7	30.9	40.1	*31.2*
Qwen2.5-VL-72B	16.8	26.8	20.8	19.1	26.7	46.8	*26.2*
Passive: fixed sampled frames							
Gemini 3.6 Flash	47.1	61.1	49.9	42.2	71.4	77.2	*58.1*
SpaceR-7B	31.6	44.7	52.6	41.0	52.0	70.1	*48.7*
Ross3D (+ depth, pose, BEV)	50.7	62.6	32.0	31.9	n/a	n/a	*47.0*†
SpatialLadder-3B	29.3	40.4	61.2	45.2	45.2	58.7	*46.7*
ViLaSR-7B	33.3	47.6	52.6	43.4	34.6	66.0	*46.2*
Video-R1-7B	30.8	41.1	42.0	39.6	42.6	63.4	*43.2*
Qwen3.5-9B	19.4	17.4	3.2	3.1	17.3	29.0	*14.9*

Table 1: **Performance of active and passive models on six 3D benchmarks.** Green marks a cell whose domain appears in that model's training data. Ross3D cannot run on MP3D or ARKit; its marked average (†) imputes those with the mean of the other passive models on each benchmark.



(a) ScanQA-only run, ScanQA validation

(b) Multi-domain run

(c) Tool calls per episode

Figure 4: **Validation results along training curve.** *(a)* ScanQA-only validation. *(b)* Mixed-data validation gains relative to SFT initialization. *(c)* Mean tool calls per episode for ScanQA-only and mixed-domain training.

despite temporary reversals. The mixed-data run improves on all three validation domains, with the largest gain on ARKit and smaller gains on ScanQA and MP3D (Figure 4(b)). Tool use remains roughly constant during ScanQA-only training but grows from about three to ten calls under mixed-domain training (Figure 4(c)). Thus, RL does not impose a fixed amount of exploration: interaction adapts to the training distribution and grows as the model learns to explore more complex, mixed environments.

Main Result 3: RL improves accuracy across interaction levels. Figure 5 groups episodes by the trained policy's tool-call count and measures accuracy gain over each policy's SFT initialization. Every stratum improves by $+3.2\%$ to $+6.2\%$, showing that RL benefits both low- and high-interaction episodes rather than a particular training regime or call range.

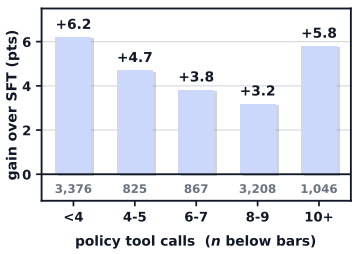


Figure 5: Gain over SFT tool-call count.

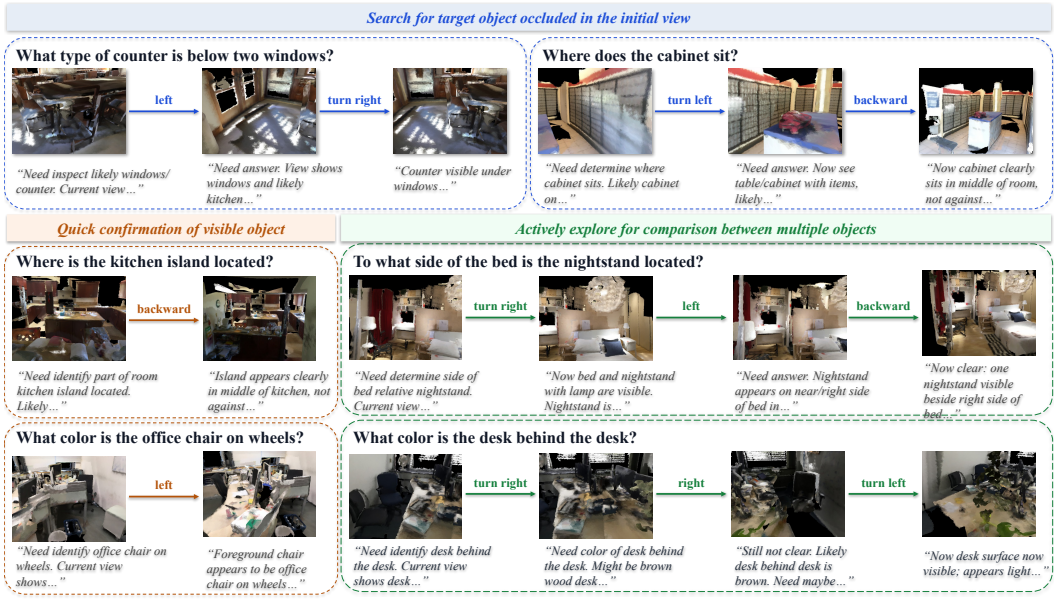


Figure 6: **Qualitative exploration behaviors.** Arrows show the executed camera actions; excerpts below each frame show the policy's evolving reasoning.

Main Result 4: the policy adapts how it explores to the question. In Figure 6 when the target is initially occluded, the policy rotates or translates until the target enters view; when the relevant object is already visible, it makes a short confirmation move and answers; and when the question compares multiple objects, it acquires several viewpoints before committing. Across these cases, the reasoning trace updates with the returned observation and stops once the evidence is sufficient. These examples connect the aggregate tool-use increase trends to concrete question-conditioned actions that gives informational and confirmation benefits.

6 Ablation Results

Ablations of the turn-0 input and action parameterization appear in Appendices B.1 and B.2, respectively.

6.1 Image Rendering

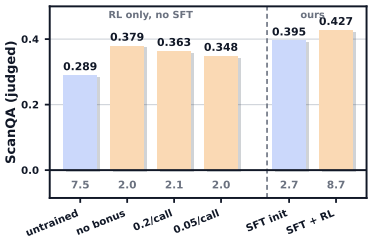
We compared sourcing image input from real ScanNet image and from 3D mesh rendering. Figure 7 exemplifies that the nearest captured photograph can be substantially displaced from the requested view, whereas rendering supplies an observation at every pose the policy can choose–a tradeoff between image quality and position faith. Under matched training — same base, data fraction, schedule, tool and reward — rendering is also the stronger source: across the thirteen steps where both runs report validation, the rendered-view run leads the nearest-real-photo run at every one, by 10.6% on average.



Figure 7: Pose-faithful renders and the nearest captured photographs.

6.2 SUPERVISED INITIALIZATION

Direct RL initially collapses interaction: all no-SFT variants fall from 7.5 to about 2.0 calls per episode, and increasing the tool bonus to 0.05 or 0.2 does not prevent it. In our SFT-initialized lineage, RL instead raises tool use from 2.7 to 8.7 calls and accuracy from 0.395 to 0.427; because the lineages differ in backbone and training step, only within-lineage trends are comparable. By step 470, pure RL recovers 0.414 accuracy and 4.0 calls, showing that SFT improves efficiency and stability rather than enabling an otherwise impossible behavior.



condition	val acc.	turns
instruction-tuned, untrained	0.289	7.5
no tool reward RL, step 80	0.379	2.0
tool reward RL 0.2/call, step 80	0.363	2.1
tool reward RL, step 470	**0.414**	4.0

Figure 8: **Direct RL without SFT.** *Left:* judged ScanQA val accuracy (bars) and tool calls per episode (numbers). *Right:* early and late no-SFT checkpoints.

6.3 PER-CALL TOOL REWARD

Table 2 shows that the tool bonus improves OOD performance with mild sacrifices in in-domain validation performance. We retain it as a behavioral control that encourages active exploration and adaptation to new domains and environments, rather than as a tuned source of accuracy.

		accuracy			tool calls / episode	
benchmark		bonus	none	Δ	bonus	none
SPBench-MV	OOD	**49.2**	42.8	+6.4	9.86	6.26
SPBench-SI	OOD	**38.4**	37.4	+1.0	9.83	6.30
SQA3D	OOD	**47.7**	47.2	+0.5	9.41	5.12
ARKit	OOD	78.4	**79.3**	−0.9	10.11	5.92
MP3D	in-domain	62.7	**65.0**	−2.3	9.99	5.70
ScanQA	in-domain	41.0	**41.3**	−0.4	9.54	5.17

Table 2: Effect of a 0.05-per-call tool bonus versus no tool bonus, with all other settings matched.

7 CONCLUSION

We formulate 3D question answering as an interactive decision process in which a VLM must acquire the evidence needed to answer, rather than receive complete observations in advance. Starting from a single rendered view, trajectory-supervised fine-tuning establishes the interaction protocol and outcome-based RL learns the full reason–act–observe policy without direct supervision of viewpoints or actions. Across six benchmarks, our mixed-data policy achieves the highest average among active models (54.1%), improves its matched supervised initialization by 5.5%, and surpasses proprietary Gemini 3.6 Flash by 2.6% under the same interaction protocol. It also outperforms every evaluated open-source 3D model despite using 50–70% fewer input frames and lower-resolution renders. The gains transfer beyond the training domains (+7.2% on average) and span all interaction levels (+3.2–6.2%). Training and qualitative analyses further show that the learned policy adapts its exploration to the question: it searches for occluded targets, briefly confirms visible ones, and gathers multiple views for relational comparisons. Together, these results establish outcome-based RL as an effective route to generalizable 3D reasoning under partial observability.

AI USE STATEMENT

Generative AI models are part of both our research pipeline and manuscript preparation. We used GPT-5.4 to generate candidate reason–act–observe trajectories for trajectory-supervised fine-tuning;

these candidates were filtered for answer correctness, valid interaction, well-formed reasoning and responses, and sufficient scene visibility. We also used generative AI to create synthetic question–answer pairs over ScanNet and ARKitScenes for reinforcement-learning training. A frozen Qwen3-30B-A3B-Instruct model served as a semantic-correctness judge during training and evaluation, and Gemini models were evaluated as baselines. We additionally used generative AI tools to improve manuscript grammar, clarity, and readability and to refine the presentation of figures and tables. We did not use generative AI to select the reported measurements or determine the scientific conclusions. The authors reviewed the model-generated training data, model-assisted evaluations, and AI-assisted manuscript changes, and take full responsibility for the final manuscript, including all text, claims, data processing, figures, and other artifacts.

Ethics Statement

This work studies active visual reasoning in offline 3D environments built from publicly released indoor-scene and question-answering datasets, including ScanNet/ScanQA, Matterport3D, ARKitScenes, and ESI. The learned policy can only move a virtual camera and return an answer; it cannot act in physical spaces, contact people, or modify external systems. We collect no new personal data and recruit no human subjects. Nevertheless, source scans may depict real homes or workplaces and can contain incidental sensitive details. Dataset assets and derived traces should therefore be handled under their original licenses and privacy protections, without redistributing restricted raw data or unnecessary personal information. Generated questions and teacher trajectories may also inherit biases from the source data and foundation models. Human validation concerns model outputs and visual evidence, not decisions about individuals. Performance should not be interpreted as approval for autonomous robotics, surveillance, or other consequential deployment; such use would require additional privacy, safety, robustness, and human-oversight safeguards.

References

Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19107–19117, 2022. doi: 10.1109/CVPR52688.2022.01854.

Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering, 2017. URL https://arxiv.org/abs/1711.11543.

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

Daya Guo et al. DeepSeek-R1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645:633–638, 2025. doi: 10.1038/s41586-025-09422-z.

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 *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Yining Hong, Jiageng Liu, Han Yin, Manling Li, Leonidas Guibas, Li Fei-Fei, Jiajun Wu, and Yejin Choi. Esi-bench: Towards embodied spatial intelligence that closes the perception-action loop, 2026. URL https://arxiv.org/abs/2605.18746.

Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, and Zhou Zhao. Chat-Scene: Bridging 3d scene and large language models with object identifiers. In *Advances in Neural Information Processing Systems*, volume 37, 2024a.

Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024b.

Dongfu Jiang, Yi Lu, Zhuofeng Li, Zhiheng Lyu, Ping Nie, Haozhe Wang, Alex Su, Hui Chen, Kai Zou, Chao Du, Tianyu Pang, and Wenhu Chen. VerlTool: Towards holistic agentic reinforcement learning with tool use. *Transactions on Machine Learning Research*, 2026.

Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1-2):99–134, 1998.

Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Alexander Sax, and Aravind Rajeswaran. OpenEQA: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16488–16498, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024. URL https://doi.org/10.48550/arXiv.2402.03300.

Haochen Wang, Yucheng Zhao, Tiancai Wang, Haoqiang Fan, Xiangyu Zhang, and Zhaoxiang Zhang. Ross3D: Reconstructive visual instruction tuning with 3D-awareness. In *ICCV*, 2025a. arXiv:2504.01901.

Kangrui Wang, Linjie Li, Zhengyuan Yang, Shiqi Chen, Zihan Wang, Li Fei-Fei, Jiajun Wu, Leonidas Guibas, Lijuan Wang, and Manling Li. Planning with the views. *arXiv preprint arXiv:2605.29563*, 2026.

Zihan Wang, Kangrui Wang, Qineng Wang, Pingyue Zhang, Linjie Li, Zhengyuan Yang, Xing Jin, Kefan Yu, Minh Nhat Nguyen, Licheng Liu, Eli Gottlieb, Yiping Lu, Kyunghyun Cho, Jiajun Wu, Li Fei-Fei, Lijuan Wang, Yejin Choi, and Manling Li. RAGEN: Understanding self-evolution in llm agents via multi-turn reinforcement learning. *arXiv preprint arXiv:2504.20073*, 2025b.

Yuxi Wei, Wei Huang, Qirui Chen, Lu Hou, and Xiaojuan Qi. See, remember, explore: A benchmark and baselines for streaming spatial reasoning, 2026. URL https://arxiv.org/abs/2603.23864.

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, 2025. URL https://arxiv.org/abs/2412.14171.

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

Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, Ranjay Krishna, Jiajun Wu, Li Fei-Fei, and Manling Li. Theory of space: Can foundation models construct spatial beliefs through active exploration? In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=8iPwqr6Adk.

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

Zhongbo Zhang, Jiayi Jin, Yifan Wang, Zaibin Zhang, Haiwen Diao, Lijun Wang, and Huchuan Lu. Vabench: Measuring embodied spatial intelligence through visual demonstrations, active perception, and metric control, 2026c. URL https://arxiv.org/abs/2609.19554.

Haoyu Zhao, Akide Liu, Zeyu Zhang, Weijie Wang, Feng Chen, Ruihan Zhu, Gholamreza Haffari, and Bohan Zhuang. Cov: Chain-of-view prompting for spatial reasoning. In *ACL (Findings)*, pp. 32427–32440, 2026. URL https://doi.org/10.18653/v1/2026.findings-acl.1623.

Duo Zheng, Shijia Huang, and Liwei Wang. Video-3D LLM: Learning position-aware video representation for 3d scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8995–9006, 2025.

Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. LLaVA-3D: A simple yet effective pathway to empowering lmms with 3d capabilities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4295–4305, 2025.

A WHY IS INTERACTION NECESSARY?

We use the notation of Sec. 3.1. To characterize whether the observations acquired so far are sufficient to determine the answer, we define the *consistency set*

$$\mathcal{K}(h_t) = \left\{ \mathcal{M}' : \mathrm{Render}(\mathcal{M}', c_s) = o_s,\ \forall s \le t \right\}, \tag{10}$$

where $c_0, \ldots, c_t$ is the camera path induced by the actions in h_t. $\mathcal{K}(h_t)$ contains all scenes indistinguishable from the true scene under the interaction history h_t. We say that q is *answerable* from h_t when all scenes consistent with the acquired observations induce the same answer:

$$g(\mathcal{M}', q) = g(\mathcal{M}'', q), \qquad \forall\, \mathcal{M}', \mathcal{M}'' \in \mathcal{K}(h_t). \tag{11}$$

Because the environment is deterministic, each additional observation can only further constrain the set of compatible scenes,

$$\mathcal{K}(h_{t+1}) \subseteq \mathcal{K}(h_t), \tag{12}$$

and can therefore reduce uncertainty about the answer. Correspondingly, treating the interaction history as a random variable, conditioning on additional observations cannot increase the expected uncertainty about the answer:

$$\mathbb{H}(y^\star \mid h_{t+1}) \le \mathbb{H}(y^\star \mid h_t). \tag{13}$$

Unlike the set inclusion, which holds along every trajectory, this inequality holds only in expectation over histories: a particular observation may leave the uncertainty unchanged or even increase it. Successful interaction therefore requires selecting viewpoints that are informative for the current question.

This formulation exposes a fundamental limitation of passive perception. A model restricted to the initial observation o_0 cannot recover task-relevant information absent from that view. Its Bayes-optimal passive accuracy is

$$\mathrm{Acc}^{\mathrm{pass}} = \mathbb{E}_{q,o_0}\left[\max_{y \in \mathcal{Y}} \Pr(y^\star = y \mid q, o_0) \right]. \tag{14}$$

Whenever (q, o_0) is consistent with multiple scenes that induce different answers, this ambiguity cannot be resolved from the initial observation alone, regardless of model capacity. By contrast, interactive policies can acquire additional observations that progressively refine the set of compatible scenes. Let Π_{int} denote the class of history-dependent interactive policies over the augmented action space. Since passive prediction is recovered as the special case of terminating immediately at $t = 0$,

$$\sup_{\pi \in \Pi_{\mathrm{int}}} \mathrm{Acc}(\pi) \ge \mathrm{Acc}^{\mathrm{pass}}. \tag{15}$$

Interaction can therefore overcome the passive observation ceiling by acquiring task-dependent evidence unavailable in the initial view. The resulting learning problem is not merely to reason over observed evidence, but to determine which observations should be acquired to make the question answerable.

B ADDITIONAL ABLATIONS

B.1 INPUT DATA FORMAT

Across turn-0 visual inputs, the single-view condition reaches 0.410 validation accuracy, within 0.4% of the best condition, a bird's-eye view (BEV) plus four candidate renders (0.414), while cutting the initial context from five images to one and reducing the confound of the target being visible before the policy acts. Adding a BEV to that single view is worse than either (0.367). We therefore adopt the single starting view, which is also the condition that best resembles a human viewpoint.

turn-0 visual input	Val. Acc.
BEV + 4 renders	**0.414**
1 render, no BEV	<u>0.410</u>
BEV + 1 render	0.367

Table 3: ScanQA validation accuracy for runs differing only in the turn-0 visual input.

B.2 Action Space

Relative camera moves outperform absolute pose prediction on a model trained with 15% of the data (0.414 vs. 0.367; Figure 9). We therefore use relative actions throughout.

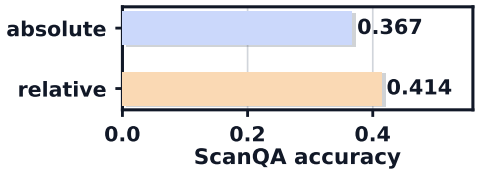


Figure 9: ScanQA validation accuracy by action parameterization.

C Reproducing Ross3D

We evaluate the released Ross3D checkpoint (LLaVA-Video-Qwen2-7B) with its official dataloader, frame sampler, scorer, and bird's-eye-view input. The only implementation change is calling the evaluation function directly because the original eight-GPU Ray launcher is incompatible with our scheduler.

metric	published	reproduced
EM	30.8	30.7
CIDEr	107.0	107.1
BLEU-4	17.9	18.2
METEOR	20.9	20.8
ROUGE-L	50.7	50.7

Table 4: Agreement with the published Ross3D results on all $4{,}675$ ScanQA validation questions.

The reproduced scores closely match the published results across every metric, validating our Ross3D evaluation pipeline and confirming that its scores are directly comparable with the other ScanQA results reported in this paper.

D PROMPT TEMPLATES

This appendix records the prompts used by the reported policy, supervised-data teacher, accuracy judge, and Gemini baselines. The bracketed image labels in the cards denote multimodal message parts and are not literal text sent to the model. For print consistency, the renderer uses ASCII double hyphens where the source string contains an em dash; all wording, placeholders, tags, and JSON structure are otherwise preserved. The cards are generated by `figs/prompt_cards/make_prompt_cards.py`.

D.1 INTERACTIVE POLICY AND EVALUATION

System prompt. The v10md policy uses the following system message during RL and evaluation. It defines the action space, optional movement magnitude, and the only two valid response formats.

system

You are a 3D spatial reasoning assistant exploring an indoor scene. You are given ONE image: your current camera view -- this is your starting position. To see more of the scene, call render_view_tool with a single movement action, and optionally a distance. Actions: forward, backward, left, right, up, down, turn_left, turn_right, turn_up, turn_down. If you do not give a distance, the camera takes its default step -- about 1 metre forward or backward, about 0.6 metres for the other moves, and about 30 degrees for turns. Give a distance in metres (or degrees for turns) when a larger or smaller step is more efficient -- one 4 m move is better than four small ones. You will see the new view after each action and can then decide to move again or to answer. Only answer when confident; if a view is empty or low-coverage, move to a better viewpoint first. Respond in exactly one of these two formats:

<think>your reasoning here</think>
<tool_call>
{"name": "render_view_tool", "arguments": {"action": "<action>", "distance": <optional number>}}
</tool_call>

or

<think>your reasoning here</think>
<answer>your final answer</answer>

Initial user turn. The first turn contains exactly one starting-pose render. The card shows the form used when scene bounds are available; the dimensions sentence is omitted when they are unavailable.

user (turn 0)

[rendered view]
This image is your current camera view -- your starting position. The room measures roughly {dx} m x {dy} m, with about {dz} m of height. Move through it with render_view_tool to gather evidence, then answer.

Question: {question}

Keep the text inside the <answer> tag to a SHORT PHRASE of 1-5 words -- no full sentence, no explanation. Still reason inside <think> first.

Tool observation. Every valid action returns a rendered image together with the compact observation below; {ep} and {ap} are the camera and look-at 3-vectors rounded to 0.01 m.

tool (after each action)

[rendered view]
Action '{action}'. Camera now at {ep} looking toward {ap}.

Evaluation-only recovery messages. Training does not add a final-turn hint. During evaluation, a tool call after the final allowed turn receives the first message below. A malformed response receives the second.

user (turn budget exhausted)

You have used all render calls. Do NOT call the tool. Give your final answer as <answer>...</answer>.

user (invalid output)

Respond ONLY as a <tool_call> or a final <answer>.

D.2 TRAJECTORY-SUPERVISED TEACHER

The GPT-5.4 trajectory teacher reuses the policy system prompt and initial user template above. Its per-action observation is more verbose so that the teacher can audit coverage and scene bounds while producing the curated trajectory.

user (teacher observation)

[rendered view]
Action '{act}'. Camera now near {ep} looking toward {ap}. Rendered view from camera_position={ep} looking at look_at={ap}. Scene coverage: {cov}% of pixels contain geometry. Scene bounds: min={mn}, max={mx}.

The exact-match training variant uses no additional LLM prompt. It deterministically replaces the final teacher answer with the closest ScanQA ground-truth string after normalization; exact normalized matches take precedence over token-F1, with first-listed order breaking ties.

D.3 ACCURACY-REWARD JUDGE

The training reward sends the following system and user messages to the local judge with temperature 0.1 and a 64-token output limit. The binary reward is one only when the response contains `CORRECT` and does not contain `INCORRECT`.

system (accuracy judge)

You are an expert evaluator for 3D spatial reasoning questions. Determine whether the model's answer is semantically equivalent to the ground truth, given the question. COHERENCE REQUIREMENT: first judge whether the model's answer is coherent, fluent natural language that actually answers the question. If the answer is gibberish, word-salad, self-contradictory, contains broken/garbled or non-English junk tokens, or is a rambling string of unrelated phrases -- respond INCORRECT even if it happens to contain some keywords from the ground truth. Only a coherent answer that is also semantically equivalent to the ground truth is CORRECT. Respond with exactly one word: CORRECT or INCORRECT.

user (accuracy judge)

Evaluate the model answer against the ground truth.

[Question]: {question_text}
[Ground Truth]: {ground_truth}
[Model Answer]: {answer_text}
[Judgement]:

Evaluation-harness judge. The Gemini baseline and saved prediction dumps use a shorter judge prompt with temperature zero and a four-token output limit. It explicitly permits terse answer fragments.

system (evaluation judge)

You are an expert evaluator for 3D spatial reasoning questions. Determine whether the model's answer is semantically equivalent to the ground truth, given the question. COHERENCE: if the answer is gibberish/word-salad/non-English junk, respond INCORRECT. Terse fragments are acceptable answers. Respond with exactly one word: CORRECT or INCORRECT.

user (evaluation judge)

Question: {question}
Ground truth answer(s): {gts}
Model answer: {pred}

Is the model answer correct?