RENDER TO REASON: NOVEL-VIEW SEMANTIC PREDICTION IMPROVES SPATIAL UNDERSTANDING IN VLMS

Anonymous authors
Paper under double-blind review

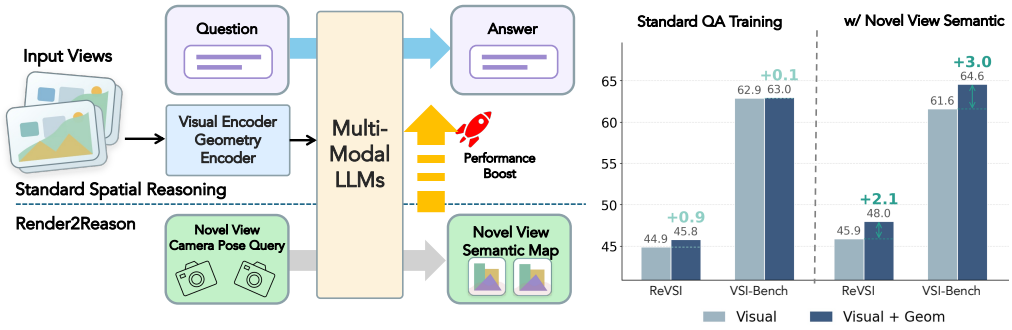


Figure 1: **Render to Reason.** Left: We introduce novel-view semantic rendering as an auxiliary training task, which takes input images and a target camera token, and renders the semantic layout of the unseen view. Right: On ReVSI (Zhang et al., 2026c) and VSI-Bench (Yang et al., 2025a), adding geometry features under standard QA training yields only marginal gains (+0.9, +0.1). When trained with our novel-view semantic rendering, adding geometry features yields larger gains (+2.1, +3.0), showing that the auxiliary task encourages the model to integrate geometry features more effectively.

ABSTRACT

Recent works augment Vision-Language Models with geometry features from pretrained 3D models, expecting that the geometric signal will boost spatial reasoning. However, we find that simply fusing geometry features and training on standard spatial QA yields only marginal improvements on high-level multi-hop tasks. We attribute this gap to a training-signal problem: standard spatial QA can be largely answered from visual features and language priors, so the geometry pathway receives weak gradients and fails to integrate with the visual features. To provide a training signal that requires geometry, we propose **novel-view semantic rendering** as an auxiliary training task that requires the model to predict the semantic layout of an unobserved viewpoint, inspired by humans' ability to mentally simulate novel viewpoints during spatial reasoning. This task encourages joint use of both pathways: geometry provides pose-dependent visibility, while vision provides semantic content. Our auxiliary task yields consistent improvements over the geometry-augmented baseline across all three benchmarks (up to +1.6 on VSI-Bench, +2.2 on ReVSI, +2.9 on our 3D-Point-QA dataset) and our full model surpasses prior open-source methods on VSI-Bench and on ReVSI. The code and data will be openly released.

1 INTRODUCTION

Spatial reasoning from visual observations is important for embodied AI, robotic navigation, and augmented reality applications, and Vision-Language Models (VLMs) (OpenAI, 2024; Gemini Team, 2025; Wang et al., 2025c; Bai et al., 2025) have emerged as a promising architecture for such tasks due to their strong generalizability. To improve VLMs' spatial reasoning abilities, recent works (Wu et al., 2025; Fan et al., 2025; Zheng et al., 2025; Zhao et al., 2025; Zhang et al., 2026b) have augmented standard visual features with geometry features encoded from pretrained geometry models (Wang

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.

et al., 2025a;b) that support dense geometry predictions such as point clouds and camera poses, expecting that the geometric signal would boost spatial reasoning. However, this expectation does not hold in practice: simply fusing geometry features and training on standard spatial QA yields only marginal improvements over a vision-only baseline on high-level multi-hop tasks (Fig. 1).

We attribute this gap to a training-signal problem: standard spatial QA can be largely answered from visual features and language priors, so the geometry pathway is underused and fails to integrate with the visual features. To encourage stronger use of geometry, we propose **novel-view semantic rendering** as an auxiliary training task that requires both pathways. Our design draws inspiration from classic findings in cognitive science: humans solve spatial reasoning tasks by *mentally simulating* novel viewpoints (Shepard & Metzler, 1971; Hegarty, 2004). We adapt this perspective to VLMs: if the model is trained to predict the semantic layout of an unseen viewpoint, it must combine the geometric pathway (*which surfaces are visible from the target pose*) with the visual pathway (*what those surfaces are*), and neither alone is sufficient (Fig. 3). The task can be supervised at scale from any video, with semantic labels generated by a pretrained semantic segmentation model (Wang et al., 2023), and requires no additional human annotation.

We evaluate on the multi-hop spatial reasoning benchmarks VSI-Bench (Yang et al., 2025a) and ReVSI (Zhang et al., 2026c), and on **3D-Point-QA**, a low-level benchmark we introduce that isolates geometric reasoning from object grounding via point prompts. Adding novel-view semantic rendering as auxiliary supervision improves a geometry-augmented baseline by +1.6 on VSI-Bench, +2.2 on ReVSI, and +2.9 on 3D-Point-QA. Our full model achieves 74.4 average on VSI-Bench and 54.5 on ReVSI, surpassing prior open-source methods on ReVSI and on VSI-Bench.

In summary, our contributions are:

- We propose **novel-view semantic rendering**, an auxiliary task that requires both visual and geometric reasoning during training and consistently improves spatial QA performance over a geometry-augmented baseline.
- Our full model outperforms prior open-source methods on VSI-Bench and ReVSI.
- We introduce **3D-Point-QA**, a low-level spatial reasoning dataset that isolates geometric reasoning from object grounding through point-prompt queries.

2 Related Work

2.1 Vision-Language Models for Spatial Understanding

Building on general-purpose Vision-Language Models (VLMs) (Li et al., 2024; Wang et al., 2025c; Bai et al., 2025), recent works have specialized these architectures for spatial reasoning by integrating explicit 3D representations into VLMs. 3D-LLM (Hong et al., 2023) feeds 3D point cloud features into the language model, enabling 3D scene captioning, grounding, and question answering. Follow-up work explores point cloud encoders (Xu et al., 2024; Chen et al., 2024) and scene graph inputs (Zhu et al., 2023). These methods assume access to dense 3D inputs at inference, motivating a line of work that targets the same spatial competencies from images alone.

To remove this dependency, recent methods (Wu et al., 2025; Fan et al., 2025; Zheng et al., 2025; Zhang et al., 2026b; Zhao et al., 2025; Li et al., 2026) inject geometry features extracted by pretrained 3D models (Wang et al., 2025a; 2024; 2025b) directly into VLMs from images alone. These methods report consistent gains on spatial benchmarks, demonstrating that geometric features carry useful signal. Chen et al. (2026) and Huang et al. (2026) improve spatial reasoning by training the VLM to predict geometry features at input views, distilling knowledge without taking geometry as direct input. In this work we further examine how this signal is integrated during training, and observe that under standard QA supervision the visual pathway dominates while the geometry pathway remains underused on high-level multi-hop questions. Zhang et al. (2026a) are motivated by a similar concern and encourage greater use of geometry features through random appearance feature dropping and geometry-guided fusion. We instead target the training signal itself with an auxiliary objective that requires both pathways. Closer to our work, several recent methods add predictive auxiliary objectives to VLM training for spatial reasoning. On the generation side, Omni-View (Hu et al., 2025) jointly trains 3D scene generation and understanding. Cambrian-S (Yang et al., 2025b) trains a next-latent-frame predictor to drive long-video memory, and Cambrian-P (Yang et al., 2026)

additionally supervises camera pose as an auxiliary output. Our approach, given the camera pose of an arbitrary unseen view, instead predicts the discrete semantic layout via patch-level classification rather than generating appearance or predicting pose, supervised jointly with the QA objective.

2.2 Spatial Reasoning Datasets and Benchmarks

Early 3D vision-language datasets focus on scene captioning and question answering with ground-truth 3D inputs, including ScanQA (Azuma et al., 2022), SQA3D (Ma et al., 2022), and 3D-LLM's instruction data (Hong et al., 2023). More recent benchmarks evaluate spatial reasoning directly from videos or image sequences without requiring 3D scans: VSI-Bench (Yang et al., 2025a) introduces eight categories of multi-hop spatial questions, ViewSpatial-Bench (Li et al., 2025) and MindCube (Wang et al., 2026) target perspective-taking and mental rotation, and OmniSpatial (Jia et al., 2025) broadens coverage with cognitive-psychology categories such as spatial logic and dynamic reasoning. On the training side, several large-scale datasets have been released to improve spatial VLMs: SPAR (Zhang et al., 2025) provides 7M spatial QA pairs, VLM-3R (Fan et al., 2025) extracts 208K reconstruction-aware QAs, and VSI-590K (Yang et al., 2025b) contributes 590K samples spanning multiple datasets (Dai et al., 2017; Yeshwanth et al., 2023; Dehghan et al., 2021). ReVSI (Zhang et al., 2026c) refines existing 3D annotations and provides ground truth that depends on the input frame count, enabling more controlled evaluation. These datasets cover a wide range of multi-hop questions to evaluate high-level spatial reasoning. We complement these benchmarks with 3D-Point-QA, a low-level benchmark that extends DepthLM's (Cai et al., 2025) single-image point-prompt design to multi-image queries (including cross-image point matching and 3D coordinate mapping), isolating geometric reasoning from text-based grounding.

3 Method

Our method (Fig. 2) targets spatial reasoning given multiview images. We first describe the base architecture with geometry features(§3.1), then present our novel-view semantic rendering task (§3.2), and the overall training procedure (§3.3).

3.1 Preliminaries

VLMs with geometry features. Standard VLMs take N input views $\{I_i\}_{i=1}^{N}$ and extract per-view visual tokens $\mathbf{v}_i \in \mathbb{R}^{L_v \times d_v}$ through a vision encoder (e.g., SigLIP (Zhai et al., 2023; Tschannen et al., 2025)). These tokens are passed through a learned projector f_{proj} that maps them into the language model's hidden dimension, and then concatenated with text tokens $\mathbf{t}_{\text{text}}$ representing the question or instruction. The decoder autoregressively generates the answer given the concatenated tokens.

While standard VLMs exhibit strong general visual understanding, the visual encoder is primarily trained for high-level semantic understanding, and is not well suited for geometry reasoning tasks. Recent works (Sec. 2.1) augment the VLMs with geometry features. A pretrained geometry encoder VGGT (Wang et al., 2025a) jointly processes all N views to produce per-view geometry patch tokens $\mathbf{g}_i \in \mathbb{R}^{L_g \times d_g}$ and camera tokens $\mathbf{c}_i \in \mathbb{R}^{1 \times d_g}$, encoding dense geometry and camera pose information respectively. To fuse geometry information into the visual representations, we concatenate the geometry and camera tokens as keys and values in a cross-attention module, with the visual tokens as queries, followed by a gated MLP with a residual connection:

$$\mathbf{v}_i' = \mathbf{v}_i + \alpha \cdot \texttt{MLP}(\texttt{CrossAttn}(Q = \mathbf{v}_i,\ K = [\mathbf{g}_i; \mathbf{c}_i],\ V = [\mathbf{g}_i; \mathbf{c}_i]))\,, \tag{1}$$

where $[\cdot\,;\cdot]$ denotes concatenation along the token dimension and α is a learnable gate. The fused visual tokens are then projected and fed to the decoder, allowing its predictions to incorporate dense 3D structure and camera-pose information encoded in the geometry features. We compared this fusion strategy against direct addition and decoder-level concatenation, and found that cross-attention performs best while keeping the LLM input length unchanged. We use a decoder-concatenation variant only for the modality ablation in Fig. 3, since it allows each pathway to be removed at inference by dropping the corresponding tokens.

Limitations of standard QA training. Standard QA training, however, provides limited training signal for the geometry pathway. Adding VGGT features to a VLM trained on the same QA data

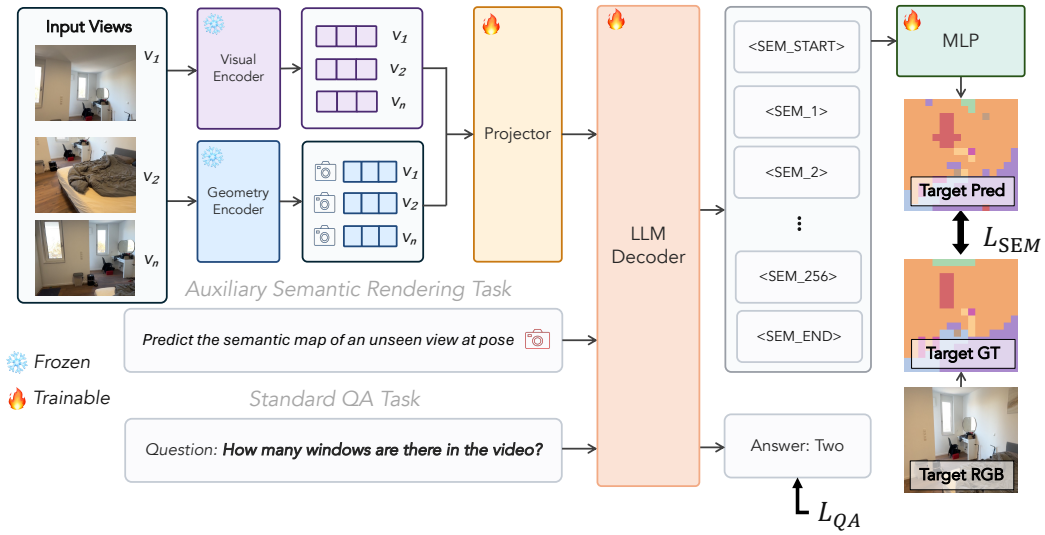


Figure 2: **Model Architecture.** Our model jointly supports semantic novel-view rendering and standard QA, sharing a backbone and differing only in prompt and output head. Input views are encoded into visual and geometry features, fused via cross-attention, and passed to the LLM decoder. For novel view semantic rendering, we extract a camera token from the target view via the geometry encoder and prepend it to the prompt; the LLM produces 256 output tokens, one per target patch in raster order, each mapped to a semantic class by a 2-layer MLP. For standard QA, the prompt only contains the question, and the LLM generates the answer without using the rendering MLP.

improves VSI-Bench scores by only +0.1 and +1.0 and ReVSI by +0.9 and +0.8 on InternVL3.5-4B and Qwen3-VL-4B respectively (Tab. 3). Visual features already contain implicit spatial cues sufficient to answer many spatial questions, leaving the standard QA objective with weak signal through the geometry pathway.

3.2 Novel-View Semantic Rendering

Motivation. To encourage the model to use both pathways, we design an objective that cannot be solved with either alone. Predicting the semantic layout from an unseen viewpoint satisfies this requirement: determining what is visible from the novel pose requires dense 3D geometry and camera pose reasoning, captured by the geometry features, and determining the semantic class of each visible surface requires the appearance and category information provided by the visual features. To verify this, we train the decoder-concatenation variant with semantic rendering and remove each pathway while keeping the target camera token at inference. As shown in Fig. 3, visual-only predictions fail to localize the target view, while geometry-only predictions fail to produce reasonable class labels. Simpler alternatives, such as better geometry–visual alignment or predicting VGGT features at novel views, do not improve spatial QA, since they do not require the model to combine both pathways (Appendix A.3). We choose semantic layouts over RGB because semantic prediction couples scene structure with object identity, providing a signal aligned with spatial reasoning, while remaining a discrete prediction target that integrates naturally with the LLM's next-token prediction machinery.

Task formulation. We formulate the task as predicting the semantic map from the target viewpoint given the N input views and a target camera token $\tilde{\mathbf{c}}_t$ corresponding to an unseen view. We obtain $\tilde{\mathbf{c}}_t$ by including the target frame as an additional input to the VGGT encoder alongside the N input views, and extracting the target frame's camera token. We keep VGGT frozen and discard the target view's output patch tokens, retaining only its camera token and the context-view features for the auxiliary prediction task, and we verify that target-frame information leaking into the context features is negligible (Appendix A.5). Since our goal is to encourage geometric reasoning rather than fine-grained rendering, we keep the output simple to classify once the viewpoint is correctly resolved. By default, we use a 16×16 grid (256 query tokens), which balances object merging at coarser resolutions against sequence length at finer ones (Tab. 3f). We pair this resolution with a compact taxonomy of $C = 9$ room-level classes: `floor`, `wall`, `ceiling`, `opening` (doors, windows),

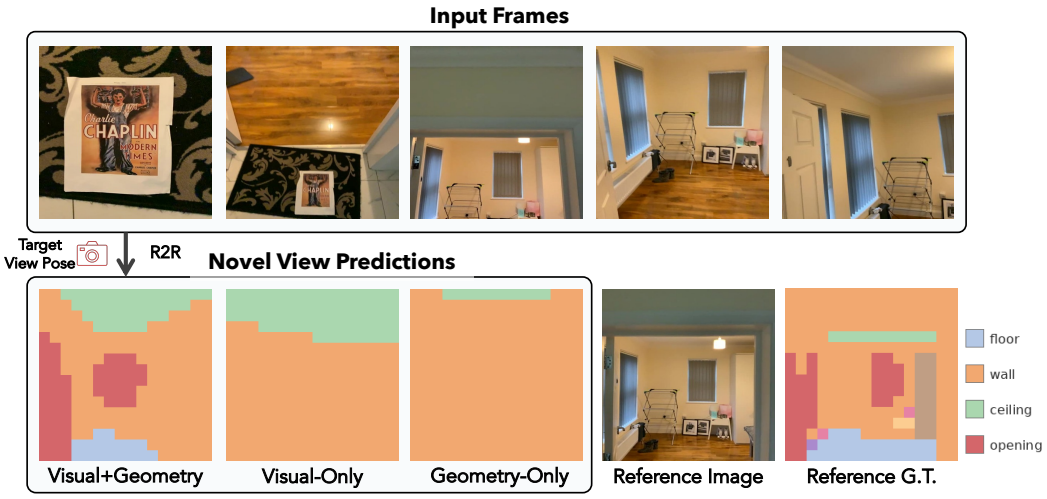


Figure 3: **Modality Ablation for Novel View Semantic Rendering.** The model predicts a semantic map of an unseen viewpoint, given input frames and the target camera pose. Top row: input frames. Bottom row: target view. With both modalities, the prediction matches the target geometry. With vision only, the prediction fails to reach the target pose and **resembles the last input frame**, indicating that the geometry features are required to render at the correct viewpoint.

`table`, `seating`, `cabinet`, `object`, and `void`. Each cell of the grid corresponds to the most common semantic class in that spatial region of the target view.

Model output and supervision. To produce the semantic grid, a natural choice would be to extend the language model's vocabulary with C class tokens and have the decoder autoregressively generate the $16 \times 16 = 256$ semantic labels. However, under teacher forcing, ground-truth labels of previously predicted patches are fed as input, making the task too easy: the model can infer each patch from neighboring labels, bypassing cross-modal reasoning. To prevent this shortcut and enable parallel decoding without overloading the LLM head with patch-level classification, we instead (i) add 256 learnable query tokens (`<SEM_1>` to `<SEM_256>`) to the vocabulary, one per spatial location in the 16×16 grid, following common practice in referring segmentation (Lai et al., 2024; Ren et al., 2024); and (ii) attach a separate two-layer MLP head $f_{\text{sem}} : \mathbb{R}^d \to \mathbb{R}^C$ for class prediction. The query tokens are appended to the input sequence after the input views and the target camera token. The LLM decoder produces a hidden representation $\mathbf{h}_j \in \mathbb{R}^d$ for each query token j in a single forward pass, and f_{sem} maps each hidden representation to a class distribution in parallel: $p_j = \text{softmax}(f_{\text{sem}}(\mathbf{h}_j)), j = 1, \ldots, 256$. The task is supervised with standard cross-entropy loss:

$$\mathcal{L}_{\text{sem}} = -\frac{1}{|\mathcal{V}|} \sum_{j \in \mathcal{V}} \log p_j[y_j], \tag{2}$$

where y_j is the ground-truth class at patch j, and $\mathcal{V}$ is the set of valid non-void patches.

Inference behavior. The model's behavior at inference depends on which task is invoked. For spatial QA, the model takes the question as input and generates the answer autoregressively, leaving the semantic query tokens and MLP head unused. For novel-view rendering, the prompt contains the target camera token followed by the 256 query tokens. The LLM then produces hidden representations in a single forward pass, and f_{sem} maps them to class labels for all patches in parallel.

Data generation. For each sample, we extract six views from consecutive video frames with random temporal intervals of 0.5 to 1.5 seconds. One view is randomly selected as the target, and the remaining five serve as inputs. We empirically find that this simple sampling strategy is more effective than explicitly maximizing viewpoint diversity for improving geometric reasoning performance, likely because it preserves realistic temporal coherence. We generate semantic labels at scale using a pretrained segmentation model (Mask2Former (Cheng et al., 2022) with the InternImage-H backbone (Wang et al., 2023)). The predicted segmentation maps are downsampled to a 16×16 grid via max-pooling and mapped to the 9-class taxonomy by assigning the dominant class for each cell.



Across 2 images, colored arrows indicate points on three objects:
- **Red arrow (Image 1): a block**
- **Green arrow (Image 2): a sofa**
- **Blue arrow (Image 2): a sofa**

Which point is closer to the one indicated by the red arrow, the green or the blue?
Answer with just the color: Green or Blue.
Answer: Green

Figure 4: **Visualization of P2P relative distance in 3D-Point-QA.** We render arrows as prompts to make recognition easier for VLMs (Xu et al., 2025). See more visualizations in Appendix A.6.

3.3 Training

Joint objective. We jointly train the model on spatial QA and novel-view semantic rendering. During training, each batch contains a mixture of QA samples and rendering samples, so the model receives supervision from both tasks at every update. For each sample, we compute only the loss associated with its task: QA samples contribute to the QA loss, while rendering samples contribute to the semantic rendering loss. The overall objective is:

$$\mathcal{L} = \mathcal{L}_{\text{QA}} + \lambda_{\text{sem}} \mathcal{L}_{\text{sem}}, \tag{3}$$

where $\mathcal{L}_{\text{QA}}$ is the standard autoregressive token-level cross-entropy loss on the answer tokens for QA samples, $\mathcal{L}_{\text{sem}}$ is the loss on novel-view semantic rendering samples, and λ_{sem} controls the relative weight of the auxiliary rendering objective. We set $\lambda_{\text{sem}} = 1$ in all experiments, since the per-sample values of $\mathcal{L}_{\text{sem}}$ and $\mathcal{L}_{\text{QA}}$ are comparable in scale.

Model training. We adopt InternVL3.5 (Wang et al., 2025c) and Qwen3-VL (Bai et al., 2025) as base models. The visual encoder and geometry encoder remain frozen. We train the geometry adapters, camera adapter, shared projector, and the semantic prediction head, and follow prior work (Zhu et al., 2025) to update only the MLP gating and up-projection layers in the LLM decoder. This design enables efficient training while preserving the model's pretrained capabilities.

4 Experiments

Our main experimental analysis centers on controlled experiments across two 4B base models, isolating the contributions of geometry features and auxiliary semantic supervision, as dense ablations at larger scale are computationally prohibitive. We additionally evaluate the complete training recipe with 8B backbones for system-level comparisons with existing methods, which differ in backbone, training data, and schedule.

4.1 Setup

Benchmarks. We evaluate on VSI-Bench (Yang et al., 2025a), ReVSI (Zhang et al., 2026c), and a set of low-level spatial QAs that we introduce. Existing spatial reasoning benchmarks (Sec. 2.2) focus on high-level, multi-hop tasks where objects are referred to only by name. This makes it difficult to disentangle whether errors stem from geometric reasoning or from object grounding given text alone. SPAR (Zhang et al., 2025) extracts low-level spatial QAs but still relies on object centroids for answering, coupling geometric reasoning with object grounding. We instead construct 3D-Point-QA, a dataset with low-level reasoning tasks (with visual and textual point prompts) that should be straightforward to answer given camera poses and point clouds without object-level reasoning. The dataset includes point-to-camera distance, point-to-point distance, relative distance comparison, point matching, and 3D coordinate mapping (Fig. 4 and Fig. 7). We extract 100K training and 5K validation QAs from ground-truth meshes in ScanNet++ (Yeshwanth et al., 2023) following the official splits.

Training data composition. Our training data consists of spatial QA pairs and semantic rendering samples constructed from indoor 3D datasets, including ScanNet, ScanNet++, and ARKitScenes. We use 494K total samples: 290K QA pairs from VSI-590K (Yang et al., 2025b), 4K route-planning QA pairs from VLM-3R (Fan et al., 2025), 100K 3D-Point-QA pairs, and 100K semantic rendering pairs.

Metrics. For VSI-Bench and ReVSI, we follow prior work (Yang et al., 2025a) and report accuracy for multiple-choice questions and mean relative accuracy for numerical questions. For 3D-Point-QA,

Table 1: **VSI-Bench sub-task breakdown.** Best results within each model group are **bolded**. *#QA* denotes the number of spatial training QAs. 9B parameters include the frozen geometry encoder.

Methods	Backbone	#Params	#QA	Avg.	*Obj. Count*	*Abs. Dist.*	*Obj. Size*	*Room Size*	*Rel. Dist.*	*Rel. Dir.*	*Route Plan*	*Appr. Order*
					Numerical Answer				Multiple-Choice Answer			
Baseline												
Chance (Frequency)	–	–	–	34.0	62.1	32.0	29.9	33.1	25.1	47.9	28.4	25.2
Proprietary Models (API)												
GPT-4o	–	–	–	34.0	46.2	5.3	43.8	38.2	37.0	41.3	31.5	28.5
Gemini-2.5 Pro	–	–	–	51.5	43.8	34.9	64.3	42.8	61.1	47.8	45.9	71.3
Open-source Fine-tuned Models												
VLM-3R	LLaVA-NeXT	7B	208K	60.9	70.2	49.4	69.2	67.1	65.4	80.5	45.4	40.1
VG-LLM	Qwen2.5-VL	9B	442K	62.2	71.4	56.8	69.0	69.1	67.9	83.2	47.4	32.5
SenseNova-SI	InternVL3	8B	8M	68.8	72.0	53.5	76.8	72.8	69.6	80.8	48.5	76.4
GeoThinker	Qwen3-VL	9B	1.1M	72.6	–	–	–	–	–	–	–	–
Cambrian-S	Qwen2.5	7B	590K	67.5	73.2	50.5	74.9	72.2	71.1	76.2	41.8	80.1
Cambrian-P	Cambrian-S	7B	590K	73.7	74.9	60.1	76.0	**76.9**	74.8	**89.5**	**52.6**	85.0
Render2Reason	InternVL3.5	9B	394K	67.3	72.0	51.9	71.8	68.2	68.2	83.5	45.9	76.4
Render2Reason	Qwen3-VL	9B	394K	**74.4**	**75.0**	**64.1**	**78.3**	75.0	**80.0**	88.4	48.5	**85.9**

we use $\delta_{1.25}$ (Eigen et al., 2014) for point-to-camera and point-to-point distance estimation, accuracy for relative distance inference, PCK@0.05 (Yang & Ramanan, 2013) for point matching, and the percentage of predicted points within 0.5m of the ground truth for 3D mapping.

4.2 Comparisons with Existing Methods

We compare with existing methods on ReVSI (Zhang et al., 2026c) and VSI-Bench (Yang et al., 2025a). Our method outperforms prior open-source methods while using less annotated training data.

Results on VSI-Bench. We evaluate our method with 128 input views. In Tab. 1, R2R with the Qwen3-VL backbone achieves 74.4 average accuracy, the best among all compared methods, surpassing Cambrian-P-7B (73.7) and GeoThinker-9B (72.6) with less training data. With the InternVL3.5 backbone, R2R reaches 67.3, surpassing VLM-3R-7B (60.9) and VG-LLM-9B (62.2) and matching Cambrian-S-7B (67.5). Across categories, our Qwen3-VL variant achieves the best results on five of eight question types. The largest margins over Cambrian-P are on relative distance (+5.2) and absolute distance (+4.0), and the absolute-distance gain is consistent with our results on ReVSI. Overall, these results show that our auxiliary novel-view semantic rendering objective is an effective and data-efficient approach to improving spatial reasoning in VLMs.

Results on ReVSI. ReVSI evaluates spatial reasoning across seven question types with manually refined annotations conditioned on the input frame count. We evaluate our method with 64 input frames and compare all methods under the same frame setting in Tab. 2. R2R with the Qwen3-VL backbone reaches 54.5 average accuracy, the best among open-source models. It exceeds Cambrian-P by 2.5 points, Cambrian-S by 5.4 points, and GPT-5.2 by 3.6 points, while the much larger Gemini 3 Pro remains ahead at 60.9. Our InternVL3.5 variant reaches 51.4, exceeding Cambrian-S by 2.3 points and GPT-5.2 by 0.5 points. Across categories, R2R shows the largest gains on absolute distance (+5.7 over Cambrian-P) and room size (+7.4 over the strongest baseline, VLM-3R). However, it performs worse than Cambrian-S on object counting and worse than VST and VLM-3R on relative distance. This pattern suggests that our auxiliary supervision benefits scene-level geometry and layout reasoning more than object-centric estimation such as counting.

4.3 Ablations and Auxiliary Task Analysis

We provide ablation studies in Tab. 3 to isolate the contributions of geometry features and the auxiliary novel-view semantic rendering task, compare semantic and RGB supervision as the rendering target, vary the semantic grid resolution, and compare with GeoSR (Zhang et al., 2026a), which we retrain in our setup for a fair comparison. Finally, we evaluate the rendering quality of the auxiliary head.

Effect of Geometry Features. On InternVL (Tab. 3a), adding VGGT features improves 3D-Point-QA by +1.7 and ReVSI by +0.9, with only a marginal change on VSI-Bench (+0.1). This pattern is

Table 2: **ReVSI sub-task breakdown.** Best results within each section are **bolded**. *#QA* denotes the number of geometry QAs generated with ground truth. 9B parameters include the geometry encoder.

Methods	Backbone	#Params	#QA	Avg.	*Obj. Count*	*Abs. Dist.*	*Obj. Size*	*Room Size*	*Rel. Dist.*	*Rel. Dir.*	*Route Plan*
					Numerical Answer				Multiple-Choice Answer		
Baseline											
Chance (Frequency)	–	–	–	31.4	52.2	40.1	17.4	20.9	25.8	31.9	30.2
Proprietary Models (64+ Frames)											
GPT-5.2	–	–	–	50.9	56.2	41.5	73.9	63.0	48.4	34.9	38.2
Gemini 3 Pro	–	–	–	60.9	60.1	54.7	79.3	51.9	68.1	56.0	56.4
Open-source Fine-tuned Models (64+ Frames)											
VST	Qwen2.5-VL	7B	4.2M	46.4	35.4	52.6	**67.9**	47.2	**49.2**	36.9	35.4
VLM-3R	LLaVA-NeXT	7B	208K	50.2	42.0	61.4	64.6	51.1	46.2	49.2	36.9
Cambrian-S	Qwen2.5	7B	590K	49.1	**48.4**	60.5	65.5	46.7	37.1	48.5	37.0
Cambrian-P	Cambrian-S	7B	590K	52.0	41.4	68.3	66.7	45.9	40.2	48.7	53.1
Render2Reason	InternVL-3.5	9B	394K	51.4	38.4	60.0	60.5	52.3	42.9	50.3	**55.2**
Render2Reason	Qwen3-VL	9B	394K	**54.5**	42.2	**74.0**	67.3	**58.5**	41.3	**51.6**	46.6

consistent with the design of 3D-Point-QA. Geometry features contribute most to the benchmark that isolates geometric reasoning, and less to high-level multi-hop benchmarks where visual features and language priors already provide useful signal. On Qwen (Tab. 3b), the gain is again largest on 3D-Point-QA (+4.9), with smaller gains on VSI-Bench (+1.0) and ReVSI (+0.8).

Effect of Auxiliary Task. Adding novel-view semantic rendering on top of VGGT features improves all three benchmarks on both backbones (+1.6, +2.2, and +2.9 on VSI-Bench, ReVSI, and 3D-Point-QA for InternVL; +1.2, +1.0, and +2.9 for Qwen). As shown in the per-category results (Appendix A.4), both backbones gain on point-to-camera distance, point-to-point distance, and 3D mapping in 3D-Point-QA, and on relative distance in both high-level benchmarks, while room size in VSI-Bench drops and other categories vary. Without VGGT (Tab. 3d), the auxiliary task alone improves 3D-Point-QA and ReVSI but hurts VSI-Bench (-1.3), while VGGT alone yields only marginal gains, indicating the two are complementary.

Semantic vs. RGB Supervision. We compare semantic-class prediction against patch-level RGB prediction as the rendering target with the same architecture and VGGT inputs. RGB supervision improves low-level metrics, particularly point matching, where patch-level reconstruction directly aids dense correspondence. However, on high-level reasoning (VSI-Bench, ReVSI), RGB supervision underperforms semantic supervision, likely because appearance-level prediction does not tie geometry to object-level semantics, which is important for multi-hop spatial questions. Combining both supervisions yields the best 3D-Point-QA score but degrades multi-hop performance relative to semantic supervision alone. We therefore use semantic rendering as our default supervision target.

Performance without Geometry at Inference. To test whether the auxiliary task increases reliance on geometry, we disable the geometry pathway at inference (α=0) and compare the performance

Table 3: **Ablation studies.** We report results on VSI-Bench and ReVSI (16 frames) and on 3D-Point-QA. *Sem.* refers to semantic rendering. Best results are **bolded**. In (e), all models include VGGT features; we disable the geometry pathway at inference. See Appendix A.4 for full results.

(a) Components (InternVL).

Methods	VSI	ReVSI	3DPQA
Finetuned	62.9	44.9	55.3
+ VGGT	63.0	45.8	57.0
+ VGGT + Sem.	**64.6**	**48.0**	**59.9**

(b) Components (Qwen).

Methods	VSI	ReVSI	3DPQA
Finetuned	66.0	50.7	72.2
+ VGGT	67.0	51.5	77.1
+ VGGT + Sem.	**68.2**	**52.5**	**80.0**

(c) GeoSR comparison (Qwen).

Methods	VSI	ReVSI	3DPQA
Baseline	67.0	51.5	77.1
GeoSR	67.2	50.7	**84.8**
Ours	**68.2**	**52.5**	80.0

(d) Rendering target (InternVL).

Target	VGGT	VSI	ReVSI	3DPQA
Sem.	✗	61.6	45.9	60.6
RGB	✓	62.4	45.2	60.2
RGB + Sem.	✓	62.6	46.6	**61.1**
Sem.	✓	**64.6**	**48.0**	59.9

(e) Geometry removal in inference.

Methods	VSI	ReVSI	3DPQA
InternVL	−6.3	−5.5	−6.4
+ Sem.	−6.7	−7.3	−7.0
Qwen	−2.3	−5.2	−5.2
+ Sem.	−4.2	−8.9	−8.7

(f) Grid resolution (InternVL).

Grid	VSI	ReVSI	3DPQA
8×8	**64.9**	46.2	53.0
16×16	64.6	48.0	**59.9**
32×32	64.3	**48.4**	56.6

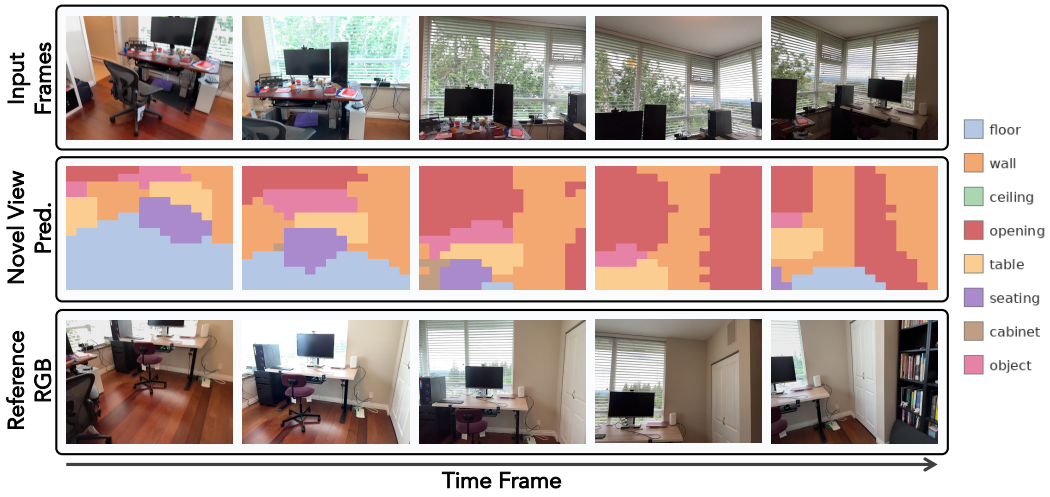


Figure 5: **Semantic Rendering Visualization.** We visualize novel-view semantic renderings along a forward-moving camera trajectory. The first row shows the five initial RGB input frames. Each prediction uses the five RGB frames right before its target frame: the first prediction uses all five frames in row 1; the second uses frames 2–5 from row 1 and the first RGB frame in row 3; and so on.

drops (Tab. 3e). This is off-distribution for both models (both fall below the finetuned baseline), so only the difference between their drops is meaningful. On both backbones and all three benchmarks, removing geometry hurts the auxiliary-task models more, especially on Qwen, suggesting that our auxiliary task encourages the model to rely more on geometry features.

Comparison with GeoSR. Under the same setup (Tab. 3c), GeoSR achieves a higher 3D-Point-QA score (84.8 vs. 80.0), as its appearance masking directly strengthens the geometry readout that 3D-Point-QA measures. However, our method outperforms GeoSR on VSI-Bench (+1.0) and ReVSI (+1.8), since multi-hop spatial reasoning requires both geometric and object-level semantic understanding, which our semantic rendering task provides.

Semantic Grid Resolution. As shown in Tab. 3f, the 16×16 grid gives the best overall balance. Both coarser and finer grids perform comparably on the high-level benchmarks but drop on 3D-Point-QA.

Novel View Rendering Results. We evaluate auxiliary semantic rendering on 5K held-out samples where the target view occurs 1 second after the last input frame. Our model (InternVL3.5-4B) achieves 44.7% mIoU and 67.0% patch accuracy. An oracle baseline that copies the best-matching input frame's semantic map for each target frame achieves 35.6% mIoU and 57.6% patch accuracy, indicating that the model accounts for viewpoint changes rather than copying an observed layout. Fig. 5 shows qualitative examples: the predicted layouts are largely consistent with the target view, capturing room structure and object placement. Predicted videos are provided in the supplement.

5 Conclusion

We present R2R, a vision-language model for spatial reasoning. To strengthen the supervision signal beyond standard spatial QA, we propose novel-view semantic rendering as an auxiliary training task that requires combining geometric and visual reasoning to predict the semantic layout of an unobserved viewpoint. We show that this auxiliary supervision consistently improves QA performance over a geometry-augmented baseline. We additionally introduce 3D-Point-QA, a low-level dataset that isolates geometric reasoning from object grounding. R2R achieves competitive performance on VSI-Bench, ReVSI, and 3D-Point-QA.

Limitations. Our auxiliary task can be supervised from any video data without human annotation, but we validate it only at a limited scale (Dai et al., 2017; Yeshwanth et al., 2023; Dehghan et al., 2021). Scaling to larger and more diverse scenes is a promising direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools to draft sections and polish paper writing. We have not used generative AI tools to design or provide feedback on research methodology or experiments, assist with translation, clean or reformat datasets, support qualitative and thematic data analysis. Generating synthetic data sets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses are not applicable to this work. Additionally, we used generative AI tools to suggest experimental parameters, summarize or analyze existing literature, and identify relevant literature. We have reviewed all AI-assisted text for accuracy and take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The model architecture, and the training objectives are described in Section 3.2 and Section 3.3. Full implementation details, including base model, trainable components, optimizer and learning-rate schedule, batch size, and training cost are given in Appendix A.1. The composition of the training mixture and the evaluation protocol and metrics for all three benchmarks are specified in Section 4.1. The code and the 3D-Point-QA data will be openly released. Reference code for the method and data samples are attached in the supplementary material.

REFERENCES

Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19129–19139, 2022.

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

Zhipeng Cai, Ching-Feng Yeh, Hu Xu, Zhuang Liu, Gregory Meyer, Xinjie Lei, Changsheng Zhao, Shang-Wen Li, Vikas Chandra, and Yangyang Shi. Depthlm: Metric depth from vision language models. *arXiv preprint arXiv:2509.25413*, 2025.

Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding, reasoning, and planning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Zhangquan Chen, Manyuan Zhang, Xinlei Yu, Xufang Luo, Mingze Sun, Zihao Pan, Xiang An, Yan Feng, Peng Pei, Xunliang Cai, and Ruqi Huang. Think with 3d: Geometric imagination grounded spatial reasoning from limited views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2432–2443, 2017.

Afshin Dehghan, Gilad Baruch, Zhuoyuan Chen, 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 *NeurIPS Datasets and Benchmarks*, 2021.

David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Neural Information Processing Systems*, 2014.

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Huaizhi Qu, 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. *ArXiv*, abs/2505.20279, 2025.

Google Gemini Team. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *ArXiv*, abs/2507.06261, 2025.

Mary Hegarty. Mechanical reasoning by mental simulation. *Trends in cognitive sciences*, 2004.

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. *Advances in Neural Information Processing Systems*, 36: 20482–20494, 2023.

JiaKui Hu, Shanshan Zhao, Qing-Guo Chen, Xuerui Qiu, Jialun Liu, Zhao Xu, Weihua Luo, Kaifu Zhang, and Yanye Lu. Omni-view: Unlocking how generation facilitates understanding in unified 3d model based on multiview images. *arXiv preprint arXiv:2511.07222*, 2025.

Xiaohu Huang, Jingjing Wu, Qunyi Xie, and Kai Han. 3drs: Mllms need 3d-aware representation supervision for scene understanding. *Advances in Neural Information Processing Systems (NeurIPS)*, 2026.

Mengdi Jia, Zekun Qi, Shaochen Zhang, Wenyao Zhang, Xinqiang Yu, Jiawei He, He Wang, and Li Yi. Omnispatial: Towards comprehensive spatial reasoning benchmark for vision language models. *arXiv preprint arXiv:2506.03135*, 2025.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9579–9589, 2024.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *ArXiv*, abs/2408.03326, 2024.

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

Haoyuan Li, Qihang Cao, Tao Tang, Kun Xiang, Zihan Guo, Jianhua Han, JiaWang Bian, Hang Xu, and Xiaodan Liang. Thinking with geometry: Active geometry integration for spatial reasoning. In *International Conference on Machine Learning (ICML)*, 2026.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. *arXiv preprint arXiv:2210.07474*, 2022.

OpenAI. Gpt-4o technical report. Technical report, OpenAI, 2024. URL https://openai.com/index/hello-gpt-4o/.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. In *SC20: international conference for high performance computing, networking, storage and analysis*, pp. 1–16. IEEE, 2020.

Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. Pixellm: Pixel reasoning with large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26374–26383, 2024.

Roger N. Shepard and Jacqueline Metzler. Mental rotation of three-dimensional objects. *Science*, 171:701 – 703, 1971.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim M. Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier H'enaff, Jeremiah Harmsen, Andreas Steiner, and Xiao-Qi Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *ArXiv*, abs/2502.14786, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotný. Vggt: Visual geometry grounded transformer. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306, 2025a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10510–10522, 2025b.

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

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*, pp. 20697–20709, 2024.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025c.

Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14408–14419, 2023.

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence. *arXiv preprint arXiv:2505.23747*, 2025.

Mingjie Xu, Jinpeng Chen, Yuzhi Zhao, Jason Chun Lok Li, Yue Qiu, Zekang Du, Mengyang Wu, Pingping Zhang, Kun Li, Hongzheng Yang, Wenao Ma, Jiaheng Wei, Qinbin Li, Kangcheng Liu, and Wenqiang Lei. Vp-bench: A comprehensive benchmark for visual prompting in multimodal large language models. *ArXiv*, abs/2511.11438, 2025.

Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pp. 131–147. Springer, 2024.

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

Jihan Yang, Zifan Zhao, Xichen Pan, Shusheng Yang, Junyi Zhang, Bingyi Kang, Hu Xu, Shang-Wen Li, and Saining Xie. Cambrian-p: Pose-grounded video understanding. In *European Conference on Computer Vision (ECCV)*, 2026.

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Daohan Lu, Rob Fergus, Yann LeCun, Fei-Fei Li, and Saining Xie. Cambrian-s: Towards spatial supersensing in video. *ArXiv*, abs/2511.04670, 2025b.

Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:2878–2890, 2013.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12–22, 2023.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952, 2023.

Jiahui Zhang, Yurui Chen, Yanpeng Zhou, Yueming Xu, Ze Huang, Jilin Mei, Junhui Chen, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. From flatland to space: Teaching vision-language models to perceive and reason in 3d. *ArXiv*, abs/2503.22976, 2025.

Shihua Zhang, Qiuhong Shen, Shizun Wang, Tianbo Pan, and Xinchao Wang. Make geometry matter for spatial reasoning. In *European Conference on Computer Vision (ECCV)*, 2026a.

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

Yiming Zhang, Jiacheng Chen, Jiaqi Tan, Yongsen Mao, Wenhu Chen, and Angel X. Chang. Revsi: Rebuilding visual spatial intelligence evaluation for accurate assessment of vlm 3d reasoning. *arXiv preprint arXiv:2604.24300*, 2026c.

Ruosen Zhao, Zhikang Zhang, Jialei Xu, Jiahao Chang, Dong Chen, Lingyun Li, Weijian Sun, and Zizhuang Wei. Spacemind: Camera-guided modality fusion for spatial reasoning in vision-language models. *ArXiv*, abs/2511.23075, 2025.

Duo Zheng, Shijia Huang, Yanyang Li, and Liwei Wang. Learning from videos for 3d world: Enhancing mllms with 3d vision geometry priors. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Zhen Zhu, Yiming Gong, Yao Xiao, Yaoyao Liu, and Derek Hoiem. How to teach large multimodal models new skills? *https://arxiv.org/abs/2510.08564*, 2025.

Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2911–2921, 2023.

A Appendix

A.1 Implementation Details

We adopt InternVL3.5 (Wang et al., 2025c) and Qwen3-VL (Bai et al., 2025) as base models. The visual encoder and geometry encoder remain frozen. We train the geometry adapters, camera adapter, shared projector, and semantic prediction head, and follow prior work (Zhu et al., 2025) to update only the MLP gating and up-projection layers in the LLM decoder. This enables efficient training while preserving the pretrained capabilities of the model. We use 16 input frames for the 4B variants and 32 for the 8B variants, but the 8B variants still perform well with more input frames at inference.

For both semantic and RGB rendering, the prediction head is a two-layer MLP that outputs a 16×16 grid. Semantic rendering is supervised with cross-entropy over class indices, and RGB rendering with an MSE loss weighted by 20. We add new anchor tokens (e.g., <SEM>-style tokens) to the tokenizer vocabulary rather than repurposing existing tokens, and initialize their embeddings via per-dimension std sampling with row-norm renormalization. During development, we found that excluding these anchor tokens from the supervision signal yields better performance, so the auxiliary loss is computed only on the patch-level prediction tokens.

We train with AdamW (Loshchilov & Hutter, 2019) using a learning rate of 1×10^{-5}, batch size 384, and zero weight decay, with a cosine schedule and 3% linear warmup. We train for 1 epoch in all experiments, except Qwen3-VL-8B, which we train for 3 epochs as it converges more slowly. All ablations use identical training schedules. Models are trained in bfloat16 with gradient checkpointing, using DeepSpeed ZeRO Stage 2 (Rajbhandari et al., 2020) on 8 NVIDIA GH200 GPUs. Training takes approximately 12 hours for the 4B variants, 30 hours for InternVL3.5-8B, and 96 hours for Qwen3-VL-8B.

A.2 Additional Rendering Results

We visualize additional novel-view semantic rendering predictions in Fig. 6.

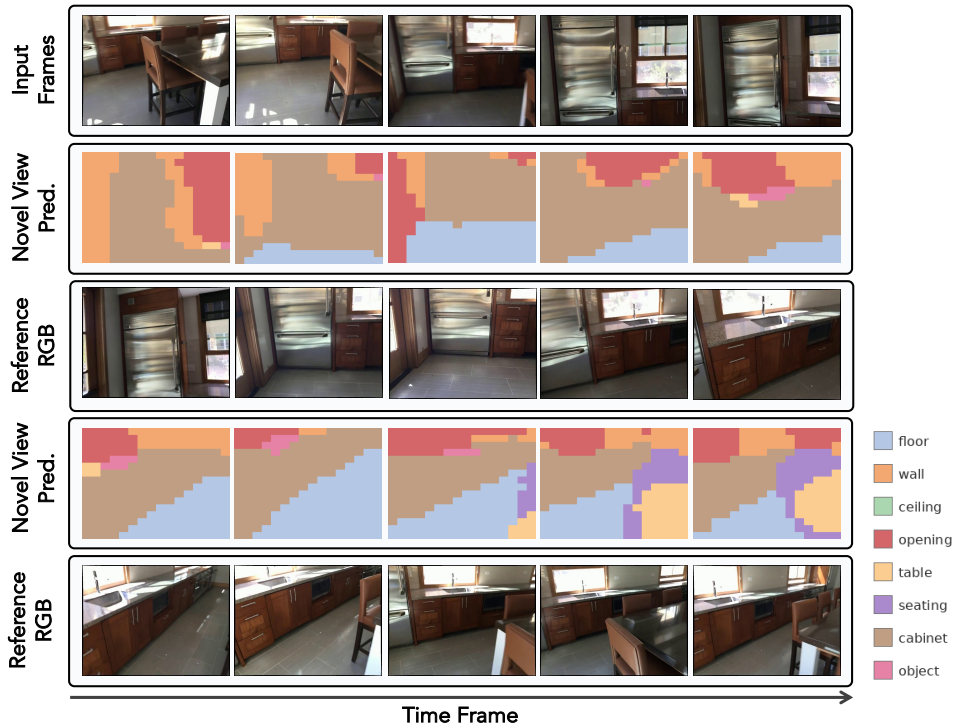


Figure 6: **Semantic Rendering Visualization.** We visualize sequential novel-view semantic renderings in a forward-moving sequence, with one-second intervals between adjacent views. The first row shows the 5 input frames. Each subsequent prediction uses the previous 5 frames as input (sliding window): the first prediction in row 2 uses all 5 frames from row 1; the second prediction uses frames 2-5 of row 1 plus the first RGB frame in row 3; and so on.

A.3 Alternative Auxiliary Objectives

As discussed in Sec. 3.1, standard QA training provides weak gradient signal through the geometry pathway. Before arriving at novel-view semantic rendering, we explored two alternatives to address this.

Geometry–visual alignment pretraining. We first tested whether the weak geometry signal stems from misalignment between geometry and visual features. We pretrained the geometry projector with geometry-only captioning and instruction following before joint spatial QA training. This improved geometry-only inference but not joint inference, indicating that better alignment alone does not lead the model to combine the two pathways.

Novel-view geometry feature prediction. We then trained the model to predict VGGT latent features at novel views, supervised through depth and camera poses decoded by the frozen VGGT heads. This reduced VSI-Bench accuracy by 1.5 points relative to the geometry-augmented baseline. We attribute this to the prediction target lying entirely in the geometry feature space, so it can be solved by the geometry pathway alone without engaging the visual pathway.

Both results suggest that an effective auxiliary objective should require both pathways. This motivates novel-view semantic rendering (Sec. 3.2), where localizing visible surfaces from the target pose requires geometry, and labeling them requires appearance.

A.4 Full Ablation Results

We report the per-category results for the ablations in Tab. 3. Tabs. 4 and 5 cover the component and rendering-target ablations on InternVL3.5-4B (Tabs. 3a and 3d). Tabs. 6 and 7 cover the component ablation and GeoSR comparison on Qwen3-VL-4B (Tabs. 3b and 3c). Tabs. 8 and 9 cover the grid resolution ablation (Tab. 3f).

Table 4: **Full results of the component and rendering-target ablations on VSI-Bench.** All models use InternVL3.5-4B with 16 input frames. Best results are **bolded**.

Methods	Avg.	*Obj. Count*	*Abs. Dist.*	*Obj. Size*	*Room Size*	*Rel. Dist.*	*Rel. Dir.*	*Route Plan*	*Appr. Order*
		Numerical Answer				Multiple-Choice Answer			
Finetuned	62.9	**70.6**	45.5	**70.2**	62.6	59.3	70.8	40.7	**83.3**
+ VGGT	63.0	69.7	**47.6**	67.8	**66.3**	58.9	75.5	42.8	75.2
+ Sem.	61.6	69.3	41.3	69.5	63.4	61.0	66.0	40.7	81.6
+ VGGT + RGB	62.4	69.7	42.7	69.2	64.2	61.0	72.2	45.4	74.9
+ VGGT + RGB + Sem.	62.6	69.0	42.8	68.1	65.5	60.1	74.0	44.3	77.0
+ VGGT + Sem.	**64.6**	70.4	44.7	69.3	64.6	**61.8**	**78.3**	**47.9**	79.6

Table 5: **Full results of the component and rendering-target ablations on ReVSI-16-Frame and 3D-Point-QA.** All models use InternVL3.5-4B. Best results are **bolded**.

Methods	ReVSI-16-Frame						3D-Point-QA					
	Avg.	*Obj. Cnt.*	*Abs. Dist.*	*Obj. Size*	*Rel. Dist.*	*Rel. Dir.*	Avg.	*P2C Dist.*	*P2P Dist.*	*P2P R-Dist.*	*Point Match*	*3D Mapping*
Finetuned	44.9	33.6	53.4	57.7	35.0	45.0	55.3	66.0	41.2	90.1	34.8	44.7
+ VGGT	45.8	32.3	55.0	57.0	36.4	48.1	57.0	59.6	44.0	92.5	44.1	45.0
+ Sem.	45.9	34.6	51.7	58.4	36.1	48.5	60.6	69.5	**54.6**	91.3	38.2	49.3
+ VGGT + RGB	45.2	33.1	50.5	58.1	35.3	49.1	60.2	65.3	44.4	92.1	**48.6**	**50.7**
+ VGGT + RGB + Sem.	46.6	35.4	55.0	**58.7**	36.1	47.9	**61.1**	**70.3**	52.9	**93.3**	38.5	50.6
+ VGGT + Sem.	**48.0**	**37.4**	**55.6**	57.1	**38.4**	**51.6**	59.9	67.9	52.9	91.3	38.3	49.2

Table 6: **Full results of the component ablation and GeoSR comparison on VSI-Bench.** All models use Qwen3-VL-4B with 16 input frames. Best results are **bolded**.

Methods	Avg.	*Obj. Count*	*Abs. Dist.*	*Obj. Size*	*Room Size*	*Rel. Dist.*	*Rel. Dir.*	*Route Plan*	*Appr. Order*
		Numerical Answer				Multiple-Choice Answer			
Finetuned	66.0	70.7	47.0	**75.8**	**71.8**	64.9	71.1	43.8	83.2
+ VGGT	67.0	68.8	51.6	73.6	71.6	63.0	79.3	43.3	85.0
+ VGGT + GeoSR	67.2	**71.4**	53.7	73.0	63.2	67.5	**80.8**	**44.3**	84.0
+ VGGT + Sem.	**68.2**	70.6	**56.0**	75.7	65.8	**72.8**	77.5	41.2	**85.6**

Table 7: **Full results of the component ablation and GeoSR comparison on ReVSI-16-Frame and 3D-Point-QA.** All models use Qwen3-VL-4B. Best results are **bolded**.

Methods	ReVSI-16-Frame						3D-Point-QA					
	Avg.	*Obj. Cnt.*	*Abs. Dist.*	*Obj. Size*	*Rel. Dist.*	*Rel. Dir.*	Avg.	*P2C Dist.*	*P2P Dist.*	*P2P R-Dist.*	*Point Match*	*3D Mapping*
Finetuned	50.7	38.6	59.8	**68.4**	37.8	49.0	72.2	81.7	63.1	92.7	64.5	58.9
+ VGGT	51.5	**39.2**	65.1	64.3	38.2	**50.4**	77.1	82.5	68.6	95.6	69.9	68.9
+ VGGT + GeoSR	50.7	39.0	65.4	63.1	**39.6**	46.6	**84.8**	**90.2**	**78.8**	**97.0**	**75.1**	**83.0**
+ VGGT + Sem.	**52.5**	39.0	**72.6**	62.8	39.1	48.8	80.0	85.3	69.9	96.6	72.6	75.7

Table 8: **Full results of the grid resolution ablation on VSI-Bench.** All models use InternVL3.5-4B with VGGT features, semantic rendering, and 16 input frames. Best results are **bolded**.

Grid	Avg.	*Obj. Count*	*Abs. Dist.*	*Obj. Size*	*Room Size*	*Rel. Dist.*	*Rel. Dir.*	*Route Plan*	*Appr. Order*
		Numerical Answer				Multiple-Choice Answer			
8×8	**64.9**	69.5	**47.0**	68.0	**64.9**	**64.8**	75.7	**48.5**	**80.7**
16×16	64.6	70.4	44.7	**69.3**	64.6	61.8	**78.3**	47.9	79.6
32×32	64.3	**70.6**	46.3	69.2	61.7	63.4	77.0	45.9	80.6

Table 9: **Full results of the grid resolution ablation on ReVSI-16-Frame and 3D-Point-QA.** All models use InternVL3.5-4B with VGGT features and semantic rendering. Best results are **bolded**.

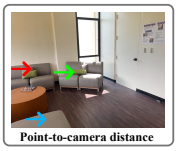
Grid	ReVSI-16-Frame						3D-Point-QA					
	Avg.	*Obj. Cnt.*	*Abs. Dist.*	*Obj. Size*	*Rel. Dist.*	*Rel. Dir.*	Avg.	*P2C Dist.*	*P2P Dist.*	*P2P R-Dist.*	*Point Match*	*3D Mapping*
8×8	46.2	34.2	55.4	55.3	**38.7**	47.7	53.0	65.2	44.5	85.0	26.4	43.9
16×16	48.0	**37.4**	55.6	57.1	38.4	**51.6**	**59.9**	67.9	**52.9**	**91.3**	**38.3**	**49.2**
32×32	**48.4**	36.4	**60.5**	**59.4**	38.4	47.4	56.6	**71.5**	51.2	86.8	29.1	44.1

A.5 Target Frame Information Leakage Analysis

For novel-view semantic rendering, we check whether target-frame information leaks into the context-view features through VGGT's global attention, by recomputing the context-view features without the target frame at inference, while keeping the target camera token unchanged. On the 5K held-out rendering samples, only 2.3% of predicted patches change, and performance is nearly identical (44.3% vs. 44.7% mIoU; 66.7% vs. 67.0% patch accuracy). This indicates that the model does not rely on target-frame information leaked through the context features.

A.6 3D-Point-QA Visualization

We visualize the full question prompts alongside the corresponding inputs in Fig. 7. Note that the arrows are enlarged here for clarity. The actual inputs use smaller arrows, following the size convention from DepthLM (Cai et al., 2025).

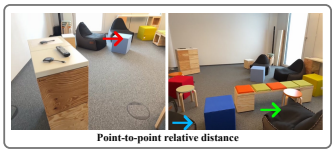


Question: In the image, colored arrows indicate points on three objects:
- **Red arrow: a sofa**
- **Green arrow: a sofa**
- **Blue arrow: a sofa**

What is the distance (in meters) from each indicated point to the camera that captured it?
Answer in the format: Red: X.X Green: X.X Blue: X.X
Answer: Red: 3.4 Green: 3.1 Blue: 0.7



A red arrow indicates a point on a speaker and a green arrow indicates a point on a hood.
What is the 3-D distance in meters between these two indicated points?
Answer in the format: X.X
Answer: 1.8



Across 2 images, colored arrows indicate points on three objects:
- **Red arrow (Image 1): a block**
- **Green arrow (Image 2): a sofa**
- **Blue arrow (Image 2): a sofa**

In 3-D space, which point is closer to the one indicated by the red arrow, the green or the blue?
Answer with just the color: Green or Blue.
Answer: Green



A red arrow in Image 1 indicates a point on a sofa chair.
Where does the same point appear in Image 2?
Answer with its position as normalized coordinates (x, y), where x is horizontal from left (0=left, 1000=right) and y is vertical from top (0=top, 1000=bottom).

Answer: (548, 621)



Colored arrows indicate points on objects across the images:
- **Red arrow (Image 1): cabinet**
- **Green arrow (Image 1): a table**
- **Blue arrow (Image 3): a bike**
- **Purple arrow (Image 2): a door**

Based on these views, produce a 3-D spatial map of the marked points (units = meters, rounded to 0.1 m), where x is right, y is forward, z is up, with the camera of Image 1 as the origin.
Answer in the format: Red: (x.x, y.y, z.z)… (one line per arrow)
Answer: Red: (-0.6, 2.0, -0.2) Green: (-0.5, 1.6, 0.6) Blue: (1.3, 4.0, 1.6) Purple: (-2.4, 6.5, 2.5)

Figure 7: **Visualization of 3D-Point-QA examples.** We visualize the question and answer for each sample.

A.7 Zero-Shot Evaluation on MindCube

To test whether the benefit of novel-view semantic rendering transfers beyond the benchmarks used during development, we evaluate zero-shot on MindCube-Tiny (Wang et al., 2026), a benchmark for spatial mental modeling from limited views. We compare the original pretrained InternVL3.5-4B with the three variants from Tab. 3a (Finetuned, + VGGT, and + VGGT + Sem.), using the same checkpoints without any additional training. MindCube-Tiny groups questions into three settings by camera configuration: *Rotation*, *Among*, and *Around*. Results are shown in Tab. 10.

Table 10: **Zero-shot results on MindCube-Tiny.** All models are based on InternVL3.5-4B and use the checkpoints from Tab. 3a without additional training. Best results are **bolded**.

Methods	Rotation	Among	Around	Overall
Pretrained	**37.5**	35.8	42.0	37.6
Finetuned	30.5	39.3	48.4	39.8
+ VGGT	37.0	38.2	34.4	37.1
+ VGGT + Sem.	32.0	**41.5**	**50.8**	**41.9**

Adding geometry features alone hurts overall performance (39.8 → 37.1), most obviously on *Around* (48.4 → 34.4). Adding novel-view semantic rendering makes the geometry features beneficial, achieving the best overall accuracy (41.9), a gain of +2.1 over the finetuned baseline and +4.3 over the pretrained model. The gains concentrate on *Among* and *Around*, where camera poses are object-centric and resemble the target-pose distribution produced by our view sampling (Sec. 3.2). In contrast, *Rotation* consists of orthogonal views from an in-place rotating camera, which rarely occur among our sampled training views, and all finetuned variants fall below the pretrained model.