Beyond 3D VQAs: Injecting 3D Spatial Priors into Vision-Language Models for Enhanced Geometric Reasoning

Supplementary Material

Overview

In this supplementary material, we provide details on our geometric training data collection in Section A. Next, we provide full implementation details, including the correspondence head architecture ($\mathcal{H}_c$) and all training hyperparameters, in Section B. Following this, we detail the evaluation protocol used to measure correspondence in both our model and the baselines in Section C. We then provide a quantitative analysis of the VSI-Bench dataset, exploring its inherent biases and the performance of SFT-trained models in Section D. We subsequently provide a brief theoretical overview of the gradient backpropagation from our geometric losses in Section E. Finally, we discuss the fundamental distinction between our learned geometric correspondence and standard rotary positional embeddings (RoPEs) in Section F.

A. Training Dataset Collection

We leverage the multi-view video sequences and depth maps from DL3DV [7] and follow the VGGT's annotation recipe [11] to generate dense point correspondence annotations for training.

For each scene, we use the provided camera intrinsics $K \in \mathbb{R}^{3\times3}$ and extrinsics $[R|t] \in \mathbb{R}^{3\times4}$ from COLMAP's Structure-from-Motion reconstruction in DL3DV [7] and VGGT [11]. Given a query frame (frame 0) with depth map $D_0 \in \mathbb{R}^{H\times W}$, we back-project valid pixels to 3D world coordinates using $\mathbf{p}_w = K^{-1}D_0(u,v)[u,v,1]^T$, where (u,v) denotes the pixel coordinate. These world points are then projected to subsequent frames using $\mathbf{p}_i = K[R_i|t_i]\mathbf{p}_w$ to establish correspondences. We validate each correspondence through depth consistency: a projected point is considered valid only if the depth difference satisfies $|D_{proj} - D_{map}| < 0.05 \times \min(D_{proj}, D_{map})$, where D_{proj} is the projected depth and D_{map} is the depth map value at the projected location. Also, we enforce a boundary margin of 4 pixels from image edges to avoid projection artifacts.

To construct a balanced training signal, we sample both positive and negative correspondences. Positive tracks are sampled from validated 3D projections, prioritizing points that remain visible across multiple frames (at least 2 frames). We target 8×8 and 24×24 points per video frame and retain the top 50% of tracks ranked by visibility duration. Negative samples are generated by applying random spatial perturbations (within 50%).

B. Additional Implementation Details

Here, we provide the specific architectural and training details required for reproducibility.

Correspondence Head ($\mathcal{H}_c$) Architecture. The correspondence head $\mathcal{H}_c$ is implemented as a 2-layer MLP consisting of a Linear layer that projects from hidden dimension d_h to $d_h/2$, followed by GELU activation, and a second Linear layer projecting back to d_h. For our experiments, $d_h = 3584$ for Qwen2.5-VL-7B [2] and $d_h = 4096$ for LLaVA-NeXT-Video-7B [15]. The head is initialized using SVD decomposition of the query projection matrix ($\mathbf{W}_Q$) from the corresponding attention layer.

Training Hyperparameters. We employ LoRA fine-tuning with rank $r = 512$ for LLaVA-NeXT-Video-7B and $r = 128$ for Qwen2.5-VL-7B, applied only to attention projection matrices (W_Q, W_K, W_V, W_O). The correspondence head is trained with full precision. We use cosine learning rate scheduling with 10% warmup over 3 epochs. For the loss function (Equation 9 in the main paper), we set the contrastive loss weight $\lambda_c = 0.3$ and distance loss weight $\lambda_d = 1.0$.

Joint Training Data Composition. Our joint training combines the DL3DV-derived 3D scene dataset (1.75M point correspondence annotations) with LLaVA-Video-178K (100K general video QA samples). This composition ensures the model maintains strong general video understanding capabilities while acquiring fine-grained spatial reasoning abilities.

C. Correspondence Evaluation Protocol

This section details the exact methodology used to compute correspondence accuracy (PCK) for both baseline models (LLaVA-NeXT-Video-7B, Qwen2.5-VL-7B) and our GASP models. For baseline models lacking explicit correspondence heads, we extract query states Q and key states K from each transformer layer during forward pass. Visual tokens are isolated by slicing the sequence from position T_s to T_e where T_s denotes the first visual token position and $T_e = T_s + N_f \times N_p$ with N_f being the number of frames and N_p the patches per frame. The extracted features are reshaped to $[N_f, N_p, d_h]$ where d_h is the hidden dimension. For models employing Grouped-Query Attention (GQA), we average over attention heads to obtain $[N_p, \bar{d}]$ where $\bar{d} = d_h/n_h$. Given source frame features $Q_0 \in \mathbb{R}^{N_p\times\bar{d}}$ and target frame features $K_j \in \mathbb{R}^{N_p\times\bar{d}}$ for frame j, we compute the correspondence matrix using cosine similarity:

Table 1. **Analysis of VSI-Bench dataset bias.** We compare the baseline models against themselves when provided with the dataset's average object and room sizes as a textual "prior" in the prompt. Deltas for VLM-3R are shown relative to the LLaVA-NeXT-Video Baseline (7B&72B).

Task	Metric	Baseline (7B)	Baseline (7B) + Avg. Prior	Baseline (72B)	Baseline (72B) + Avg. Prior	VLM-3R
Object Size Estimation	MRA@.5:.95:.05	0.47	0.64 (Δ **+0.17**)	0.57	0.65 (Δ **+0.08**)	0.69 (Δ **+0.22**)
Room Size Estimation	MRA@.5:.95:.05	0.24	0.38 (Δ **+0.14**)	0.36	0.46 (Δ **+0.10**)	0.67 (Δ **+0.43**)
Object Abs Distance	MRA@.5:.95:.05	0.14	0.61 (Δ **+0.47**)	0.23	0.57 (Δ **+0.34**)	0.49 (Δ **+0.36**)

$S = \text{CosineSim}(Q_0, K_j^T)$, and the predicted target patch for source patch i is $\hat{p}_i = \arg\max_j S_{ij}$.

We convert both ground-truth and predicted patch indices to 2D grid coordinates and compute the Euclidean distance $d = \|(r_{gt}, c_{gt}) - (r_{pred}, c_{pred})\|_2$ in patch space. We separately compute confidence on correct predictions ($d < 2$) versus incorrect predictions to obtain the calibration gap, which measures whether the model exhibits awareness of its prediction quality.

D. Analysis of VSI-Bench Dataset Bias

A potential criticism of high performance on benchmarks like VSI-Bench is that models may "hack" the benchmark by learning superficial dataset-specific biases (e.g., "all microwaves are 0.5m wide") rather than performing genuine 3D reasoning.

Bias Hacking Experiment. To investigate the extent to which VSI-Bench scores can be "hacked" by exploiting dataset-level biases, we conducted an experiment using a text-based prior. We first quantified these biases by extracting the object and room sizes from the VSI-bench QAs and averaging them. This yielded a dictionary of average object sizes (e.g., `'sofa': 181.30`, `'bed': 216.06`) and an average room size of 20.5 square meters.

Instead of a "bias-only" model, we provided these averaged values directly to the baseline VLMs as part of the input prompt, e.g., *"The average room size is 20.5 square meters. Use this information to guide your estimate."* As shown in Table 1, this simple textual prior dramatically boosts performance. For example, the LLaVA-NeXT-Video-7B baseline's "Object Abs Distance" score skyrockets from 0.14 to 0.61 (+0.47), and the LLaVA-NeXT-Video-72B model's score jumps from 0.23 to 0.57 (+0.34). Notably, on this task, the baseline models with this simple prior (0.61 and 0.57) both significantly outperform the SFT-trained VLM-3R (0.49). This finding indicates that *a significant portion of the benchmark's challenge can be solved by exploiting these easily-averaged dataset statistics, rather than relying solely on complex, visual-based spatial reasoning.*

Our observation mirrors the recent findings [4] where they demonstrated that VSI-Bench contains pervasive non-visual shortcuts that allow models to bypass genuine visual reasoning. Their diagnostic "Test-set Stress-Test" revealed

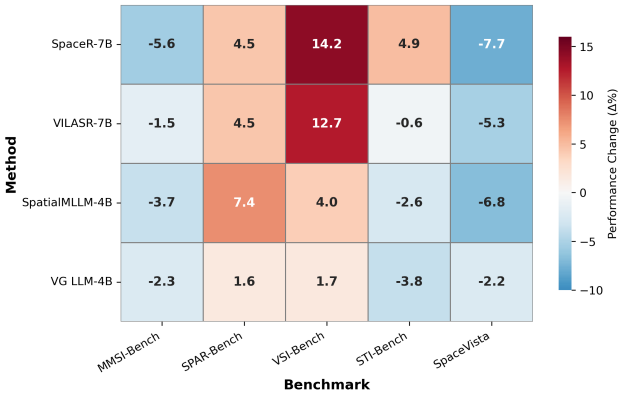

Figure 1. **Generalization Gap in 3D-VQA Fine-Tuning.** We illustrate the performance change (Δ%) of specialized spatial VLMs relative to their underlying pre-trained backbones across five distinct spatial benchmarks. While fine-tuning yields significant improvements on specific datasets (e.g., VSI-Bench, highlighted in red), it consistently leads to performance degradation (blue cells) on out-of-distribution benchmarks like MMSI-Bench and SpaceVista. This performance profile suggests that standard SFT strategies suffer from severe overfitting to dataset-specific biases, whereas genuine spatial understanding should generalize across domains.

that statistical regularities in the answer distribution enable high performance even without visual input, a vulnerability our experiment empirically validates by explicitly exploiting these statistical priors.

Generalization Analysis of 3D-VQA Models. To empirically validate the generalization limitations of standard 3D-VQA fine-tuning, we conduct a cross-dataset performance analysis in Figure 1. We report the relative performance change (Δ%) of state-of-the-art spatial VLMs compared to their respective pre-trained base models (e.g., Qwen2.5-VL). A clear pattern of *task-specific overfitting* emerges: models like SpaceR-7B [8] and VILASR-7B [12] achieve substantial gains on VSI-Bench [13] (+14.2% and +12.7%), likely due to high similarity between their training mixtures and this specific benchmark.

However, this comes at the cost of negative transfer on other spatial benchmarks. Notably, performance degrades significantly on MMSI-Bench [14], STI-Bench [6], and SpaceVista [10] (dropping by as much as -7.7%), indicat-

ing that these models are memorizing dataset-specific distributions rather than acquiring robust, generalized spatial reasoning. This stark contrast underscores the necessity of our GASP approach, which injects fundamental geometric priors to avoid such brittle memorization.

E. Analysis of Gradient Backpropagation

The total loss for our framework is $\mathcal{L}_{total} = \mathcal{L}_{LM} + \lambda_c \mathcal{L}_{corr} + \lambda_d \mathcal{L}_{depth}$. The key to our method is how the geometric-aware gradients from $\mathcal{L}_{corr}$ and $\mathcal{L}_{depth}$ backpropagate through the correspondence head to update the backbone's parameters, specifically the Query (Q) and Key (K) projectors within the transformer layers.

Formally, let $\theta^{(l)} = \{W_Q^{(l)}, W_K^{(l)}, W_V^{(l)}\}$ denote the learnable weights of the Self-Attention mechanism at transformer layer l. The visual tokens $V^{(l)}$ output by this layer serve as the input to our lightweight correspondence head $\mathcal{H}_c$. The gradient of the total loss with respect to the backbone weights $\theta^{(l)}$ can be decomposed as:

$$\frac{\partial \mathcal{L}_{total}}{\partial \theta^{(l)}} = \underbrace{\frac{\partial \mathcal{L}_{LM}}{\partial \theta^{(l)}}}_{\text{Language Modeling}} + \underbrace{\lambda_c \frac{\partial \mathcal{L}_{corr}}{\partial \theta^{(l)}} + \lambda_d \frac{\partial \mathcal{L}_{depth}}{\partial \theta^{(l)}}}_{\text{Geometric Supervision}} \quad (1)$$

We focus on the geometric term. Since the correspondence embeddings are defined as $E = \mathcal{H}_c(V^{(l)})$ (Eq. 4 in the main paper), the gradients flow via the chain rule:

$$\frac{\partial \mathcal{L}_{corr}}{\partial \theta^{(l)}} = \frac{\partial \mathcal{L}_{corr}}{\partial E} \cdot \frac{\partial \mathcal{H}_c(V^{(l)})}{\partial V^{(l)}} \cdot \frac{\partial V^{(l)}}{\partial \theta^{(l)}} \quad (2)$$

The term $\frac{\partial V^{(l)}}{\partial \theta^{(l)}}$ acts as a *Gradient Bridge*. Recall that self-attention is defined as $Z = \text{Softmax}(\frac{QK^T}{\sqrt{d_k}})V = A \cdot V$, where $Q = X^{(l-1)}W_Q^{(l)}$, $K = X^{(l-1)}W_K^{(l)}$, $V = X^{(l-1)}W_V^{(l)}$, and $A = \text{Softmax}(\frac{QK^T}{\sqrt{d_k}})$. The output visual tokens are $V^{(l)} = Z + X^{(l-1)}$. Applying the chain rule through the attention mechanism:

$$\frac{\partial V^{(l)}}{\partial W_Q^{(l)}} = \frac{\partial (A \cdot V)}{\partial A} \cdot \frac{\partial A}{\partial (QK^T)} \cdot \frac{\partial (QK^T)}{\partial Q} \cdot \frac{\partial Q}{\partial W_Q^{(l)}} \quad (3)$$

The key components are: $\frac{\partial Q}{\partial W_Q^{(l)}} = (X^{(l-1)})^T$, $\frac{\partial (QK^T)}{\partial Q} = K$, the softmax Jacobian $\frac{\partial A_{ij}}{\partial S_{kl}} = A_{ij}(\delta_{ik}\delta_{jl} - A_{il})$ where $S = \frac{QK^T}{\sqrt{d_k}}$, and $\frac{\partial (A \cdot V)}{\partial A} = V^T$. Combining these, the gradient with respect to $W_Q^{(l)}$ becomes:

$$\frac{\partial \mathcal{L}_{corr}}{\partial W_Q^{(l)}} = (X^{(l-1)})^T \cdot \left[\frac{1}{\sqrt{d_k}} K \cdot \nabla_A^{\text{softmax}} \cdot V \cdot \frac{\partial \mathcal{L}_{corr}}{\partial V^{(l)}} \right] \quad (4)$$

where $\nabla_A^{\text{softmax}} = \text{diag}(A)(I - \mathbf{1}A)$ is the softmax gradient term. Similarly, for $W_K^{(l)}$:

$$\frac{\partial \mathcal{L}_{corr}}{\partial W_K^{(l)}} = (X^{(l-1)})^T \cdot \left[\frac{1}{\sqrt{d_k}} Q^T \cdot \nabla_A^{\text{softmax}} \cdot V \cdot \frac{\partial \mathcal{L}_{corr}}{\partial V^{(l)}} \right] \quad (5)$$

Geometric Gradient Structure. The correspondence loss $\mathcal{L}_{corr}$ is a contrastive objective over correspondence embeddings. For frames $(I_t, I_{t'})$ with matched points (p_i, p_j) and embeddings (e_i, e_j):

$$\mathcal{L}_{corr} = -\log \frac{\exp(\text{sim}(e_i, e_j)/\tau)}{\sum_{k \in \mathcal{N}} \exp(\text{sim}(e_i, e_k)/\tau)} \quad (6)$$

where $\mathcal{N}$ includes positive and negative samples. The derivative is:

$$\frac{\partial \mathcal{L}_{corr}}{\partial e_i} = \frac{1}{\tau} \left[\sum_{k \in \mathcal{N}} p_k \cdot \frac{\partial \text{sim}(e_i, e_k)}{\partial e_i} - \frac{\partial \text{sim}(e_i, e_j)}{\partial e_i} \right] \quad (7)$$

where $p_k = \frac{\exp(\text{sim}(e_i, e_k)/\tau)}{\sum_l \exp(\text{sim}(e_i, e_l)/\tau)}$. This gradient pushes e_i towards its correspondence e_j while pulling away from negatives, creating view-invariance. Crucially, backpropagating through $\mathcal{H}_c$:

$$\frac{\partial \mathcal{L}_{corr}}{\partial V^{(l)}} = \left(\frac{\partial \mathcal{H}_c}{\partial V^{(l)}} \right)^T \cdot \frac{\partial \mathcal{L}_{corr}}{\partial E} \quad (8)$$

produces a spatially localized gradient that differs fundamentally from the dense semantic gradient $\frac{\partial \mathcal{L}_{LM}}{\partial V^{(l)}}$. This teaches the attention mechanism to distinguish tokens by 3D spatial positions, not just semantic categories.

Impact on Query-Key Similarity. The similarity between tokens i and j is:

$$S_{ij} = \frac{q_i^T k_j}{\sqrt{d_k}} = \frac{x_i^T W_Q^T W_K x_j}{\sqrt{d_k}} \quad (9)$$

The gradient update due to $\mathcal{L}_{corr}$ is:

$$\Delta S_{ij} = -\eta \lambda_c \frac{\partial \mathcal{L}_{corr}}{\partial S_{ij}} = -\eta \lambda_c \left[\frac{\partial \mathcal{L}_{corr}}{\partial V^{(l)}} \cdot \frac{\partial V^{(l)}}{\partial A} \cdot \frac{\partial A}{\partial S_{ij}} \right] \quad (10)$$

where η is the learning rate. This update increases S_{ij} for spatially corresponding tokens and decreases it for geometrically distinct tokens, even if semantically similar. Over training, the projector product $W_Q^T W_K$ learns to encode geometric correspondence:

$$W_Q^{T,(l)} W_K^{(l)} \approx M_{\text{geo}} + M_{\text{sem}} \quad (11)$$

where M_{geo} encodes geometric alignment (high values for corresponding 3D locations) and M_{sem} encodes semantic

similarity (from $\mathcal{L}_{LM}$). The geometric term emerges from the accumulated gradients:

$$M_{\text{geo}} = \sum_{t=1}^{T} \eta \lambda_c \left[\frac{\partial \mathcal{L}_{corr}}{\partial W_Q^{(l)}} \right]^T \left[\frac{\partial \mathcal{L}_{corr}}{\partial W_K^{(l)}} \right] \tag{12}$$

Depth Consistency Regularization. The depth loss $\mathcal{L}_{depth} = \sum_{i,j} A_{ij} \cdot \mathcal{D}(d_i, d_j)$ penalizes depth-inconsistent matches, where $\mathcal{D}(\cdot, \cdot)$ measures depth discrepancy. The gradient is:

$$\frac{\partial \mathcal{L}_{depth}}{\partial A_{ij}} = \mathcal{D}(d_i, d_j) \tag{13}$$

Backpropagating through softmax:

$$\frac{\partial \mathcal{L}_{depth}}{\partial S_{ij}} = \mathcal{D}(d_i, d_j) \cdot A_{ij}(1 - A_{ij}) \tag{14}$$

The term $A_{ij}(1 - A_{ij})$ amplifies gradients for mid-confidence predictions ($A_{ij} \approx 0.5$), teaching the model to suppress geometrically invalid matches. This creates depth-aware projectors:

$$\frac{\partial \mathcal{L}_{depth}}{\partial W_Q^{(l)}} = (X^{(l-1)})^T \cdot \left[\frac{1}{\sqrt{d_k}} K \cdot \text{diag}(\mathcal{D}) \cdot \nabla_A^{\text{softmax}} \cdot V \right] \tag{15}$$

where $\text{diag}(\mathcal{D})$ is a diagonal matrix of depth discrepancies. This modulates the attention mechanism to respect 3D boundaries, effectively learning:

$$S_{ij}^{\text{effective}} = \frac{x_i^T W_Q^T W_K x_j}{\sqrt{d_k}} - \lambda_d \cdot \mathcal{D}(d_i, d_j) + \text{noise} \tag{16}$$

where the depth penalty is implicitly encoded in $W_Q^T W_K$.

QK Enhancement Mechanism. The correspondence head creates two synergistic effects. First, *Geometric Subspace Alignment*: the gradient update

$$W_Q^{(l)} \leftarrow W_Q^{(l)} - \eta \lambda_c \frac{\partial \mathcal{L}_{corr}}{\partial W_Q^{(l)}} \tag{17}$$

incorporates $K \cdot \nabla_A^{\text{softmax}} \cdot V \cdot \frac{\partial \mathcal{L}_{corr}}{\partial V^{(l)}}$ (from Eq. 4), which couples the current Key representations with geometric error signals. Over iterations, W_Q and W_K co-evolve:

$$\langle W_Q^{(l)} x_i, W_K^{(l)} x_j \rangle \to \max \quad \text{if } (x_i, x_j) \text{ corresponds} \tag{18}$$

Second, *Depth-Aware Pruning*: the depth gradient (Eq. 16) forces attention weights to respect 3D structure. The combined effect yields learned attention weights:

$$A_{ij}^{\text{learned}} = \text{Softmax}\left(\frac{x_i^T W_Q^T W_K x_j}{\sqrt{d_k}} \right) \tag{19}$$

that are high for geometrically corresponding and depth-consistent token pairs, and low otherwise. Consequently, although $\mathcal{H}_c$ is discarded at inference, these geometric priors are permanently baked into $\theta^{(l)}$. The learned projectors $W_Q^{(l)}$ and $W_K^{(l)}$ encode: (1) spatial correspondence—tokens at corresponding 3D locations produce high S_{ij}; (2) view invariance—the QK space is invariant to perspective/lighting changes; (3) depth awareness—attention respects 3D scene structure. This enables the standard VLM to perform robust spatial reasoning without auxiliary inputs, as the attention mechanism itself has been geometrically restructured. The correspondence head guides the backbone to internalize 3D-aware attention patterns.

F. Relation to Positional Embeddings

Rotary Positional Embeddings. Standard Vision Transformers (ViTs) and VLMs utilize Positional Embeddings (PEs), such as absolute learnable embeddings [5] or Rotary Positional Embeddings (RoPE) [9], to inject grid location information into the sequence. Similarly, Video Transformers often extend this to 3D-RoPEs [1, 3] by adding a temporal or depth dimension. However, these RoPEs provide only *static coordinate information* (e.g., "this token is at location (x, y)"). They do not encode *visual correspondence* or *object permanence*. As evidenced in our main paper (Figure 3), the baseline models (Qwen2.5-VL and LLaVA-NeXT)—which are already equipped with advanced RoPE—achieve near-zero correspondence accuracy. This empirically demonstrates that providing coordinate information via RoPE is insufficient for the model to learn that an object at location (x_1, y_1) in Frame t is the same entity as the one at (x_2, y_2) in Frame $t + 1$.

Our GASP: From Coordinates to Correspondence. In contrast to RoPE, which is an *input-level* signal, GASP operates on the *interaction mechanism* (QK^T) of the model.

- **Content-Aware vs. Location-Aware:** RoPE is content-agnostic; it is identical for a blank wall or a complex face. GASP, supervised by our contrastive loss $\mathcal{L}_{corr}$, forces the visual features to be *content-aware*. It ensures that the query representation of an object matches its key representation in another view, regardless of their disparate positional encodings.
- **Implicit 3D Consistency vs. Explicit 3D Input:** While approaches like 3D-RoPE require explicit 3D inputs (e.g., depth maps or point clouds) to encode geometry, GASP internalizes 3D consistency into the 2D weights of the LLM. By training with $\mathcal{L}_{depth}$, our model learns to implicitly respect 3D boundaries (e.g., occlusion) using only 2D RGB inputs during inference.

Therefore, GASP does not replace RoPE but complements it: *RoPE provides the "where" within the image grid, while GASP teaches the "what" and "which" across the spatio-temporal manifold.*

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021. 4

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 1

[3] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, page 4, 2021. 4

[4] Ellis Brown, Jihan Yang, Shusheng Yang, Rob Fergus, and Saining Xie. Benchmark designers should" train on the test set" to expose exploitable non-visual shortcuts. *arXiv preprint arXiv:2511.04655*, 2025. 2

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 4

[6] Yun Li, Yiming Zhang, Tao Lin, XiangRui Liu, Wenxiao Cai, Zheng Liu, and Bo Zhao. Sti-bench: Are mllms ready for precise spatial-temporal world understanding? *arXiv preprint arXiv:2503.23765*, 2025. 2

[7] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22160–22169, 2024. 1

[8] Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. Spacer: Reinforcing mllms in video spatial reasoning. *arXiv preprint arXiv:2504.01805*, 2025. 2

[9] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 4

[10] Peiwen Sun, Shiqiang Lang, Dongming Wu, Yi Ding, Kaituo Feng, Huadai Liu, Zhen Ye, Rui Liu, Yun-Hui Liu, Jianan Wang, et al. Spacevista: All-scale visual spatial reasoning from mm to km. *arXiv preprint arXiv:2510.09606*, 2025. 2

[11] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025. 1

[12] Junfei Wu, Jian Guan, Kaituo Feng, Qiang Liu, Shu Wu, Liang Wang, Wei Wu, and Tieniu Tan. Reinforcing spatial reasoning in vision-language models with interwoven thinking and visual drawing. *arXiv preprint arXiv:2506.09965*, 2025. 2

[13] Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10632–10643, 2025. 2

[14] Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, et al. Mmsi-bench: A benchmark for multi-image spatial intelligence. *arXiv preprint arXiv:2505.23764*, 2025. 2

[15] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 1