GEOMETRIZE WHAT REALLY MATTERS: TOWARD LIGHTER AND FASTER SPATIAL MLLMS

Anonymous authors
Paper under double-blind review

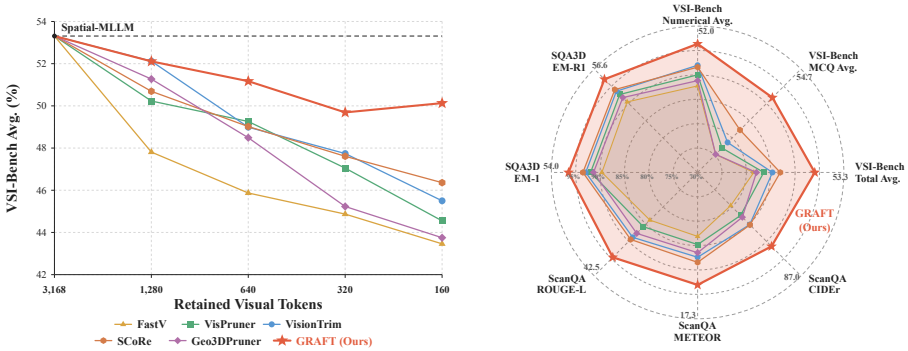

Figure 1: Performance of the proposed GRAFT. **Left**: As the token budget decreases, GRAFT consistently outperforms existing pruning methods on VSI-Bench. **Right**: GRAFT achieves the strongest performance retention under the most severe token budget compression (95%) across all benchmarks. The outer boundary represents the performance of unpruned Spatial-MLLM.

ABSTRACT

Spatial multimodal large language models augment vision-language reasoning with explicit geometric representations, enabling understanding of object locations, distances, directions, and cross-view relations. This capability, however, introduces redundancy at two levels: dense multi-view observations must first pass through expensive geometric encoders, and the resulting geometry-enhanced visual tokens then increase language-model prefill computation and KV-cache storage. Existing pruning methods typically optimize only the latter stage after geometry has already been computed, while geometric diversity alone may fail to preserve evidence that is critical to the frozen model. We present **GRAFT** (**G**eometry **R**outing **A**cross **F**rames and **T**okens), a training-free hierarchical routing framework that allocates geometric computation and evidence along the inference pipeline. Before full geometric encoding, GRAFT uses shallow features from a frozen geometry backbone to select views with reliable cross-view relations and temporal coverage, reducing the observations processed by the full geometry encoder. After visual-geometric encoding, it estimates model sensitivity to construct a compact semantic evidence set and exploits aligned 3D coordinates to correct spatially invalid token substitutions under an exact budget. Extensive experiments on three benchmarks demonstrate that GRAFT consistently maintains competitive spatial reasoning accuracy while reducing both geometric encoding and language inference costs. On VSI-Bench, retaining only 5% of visual tokens preserves 94% of the accuracy, while reducing inference latency by 82% and peak memory usage by 55%. Anonymously released code: `https://anonymous.4open.science/r/graft-88DA`.

1 INTRODUCTION

Spatial intelligence requires models to understand not only object identity but also locations, distances, orientations, and cross-view relationships Wu et al. (2025); Yang et al. (2025). Spatial multimodal large language models (Spatial MLLMs) Wu et al. (2025); Fan et al. (2026); Zhang et al. (2026); Zheng et al. (2025b) integrate vision-language models Wang et al. (2024); Bai et al. (2025a) with geometry foundation models Wang et al. (2025a;b) to recover depth, camera poses, or world coordinates from multi-frame observations, bringing explicit 3D evidence into language reasoning. However, this capability introduces two major costs: dense multi-view geometry encoding, followed by language-model prefill computation and KV-cache overhead for thousands of geometry-enhanced visual tokens. Reducing both costs while preserving spatial reasoning is therefore essential for efficient deployment.

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.

Since Spatial MLLMs also process videos or multi-frame images, visual token compression for general-purpose and video MLLMs Chen et al. (2024); Alvar et al. (2025); Xu et al. (2026); Zhang et al. (2025) offers a natural starting point. Existing methods use attention, local sensitivity, or feature diversity to remove visual redundancy and shorten language-model inputs. Without an explicit geometry branch, a Spatial MLLM Wu et al. (2025) reduces to this conventional setting. Nevertheless, these methods still optimize two-dimensional spatiotemporal representations and the corresponding language-model inputs, without addressing the additional computation and representational structure introduced by an explicit geometry branch.

In contrast, research on efficiency methods specifically designed for spatial or 3D MLLMs remains relatively limited. Recent approaches exploit voxelization, 3D distances, or geometry-aware attention for spatial token compression, exploring how to preserve multi-view consistency and scene completeness Baek & Kong (2026); Li et al. (2026). Yet geometry typically guides grouping, ranking, or merging only after spatial representations have been computed, leaving upstream geometry-encoding costs unchanged. Moreover, voxel coverage or spatial diversity does not explicitly ensure preservation of the semantic evidence required by the frozen model: spatially dispersed tokens may have limited model utility, and visually similar tokens at different physical locations may not be interchangeable. These approaches do not jointly account for sufficient coverage of model-critical evidence and the spatial validity of representative relationships. Consequently, upstream geometry computation and downstream evidence compression remain optimized in isolation.

Overcoming these challenges requires two decisions along the computational pipeline of a spatial MLLM. Before geometry modeling, reliable world coordinates are not yet available, and the model must infer from inexpensive 2D cross-view relationships which observation views warrant full geometry computation. After geometry-enhanced representations have been formed, the model must preserve critical semantic evidence under a fixed token budget and use 3D structure to determine whether the retained tokens validly represent the full input in physical space. These two decisions share the objective of allocating geometric resources, yet each must adapt to the information available at its respective stage. This work therefore studies the following unified question: *How can geometric computation and geometric evidence be progressively allocated along the computational pipeline of a Spatial MLLM, such that views truly worth geometrizing are identified before full geometry modeling, while a compact token set that both covers model-critical semantic evidence and remains spatially valid is constructed afterward?*

We propose **GRAFT** (**G**eometry **R**outing **A**cross **F**rames and **T**okens), a training-free hierarchical geometry routing framework. GRAFT operates at two consecutive stages of Spatial-MLLM Wu et al. (2025). **Stage I: Frame Sampling** uses shallow cross-view representations from the frozen geometry backbone to select 16 frames from 128 candidate frames that jointly provide reliable correspondences, informative viewpoint changes, and temporal coverage, thereby reducing the number of observations subjected to full geometry modeling. **Stage II: Spatial-MLLM Inference** performs visual-geometric encoding and initializes a compact token set using model sensitivity and semantic coverage. It then uses 3D coordinates to reduce the spatial mismatch of semantic representative assignments, while constraining semantic coverage errors to remain no higher than their initialization values. The final token set exactly satisfies the target budget. GRAFT directly retains the fused tokens generated by the original model and requires no updates to any model parameters. We conduct extensive experiments on VSI-Bench Yang et al. (2025), ScanQA Azuma et al. (2022), and SQA3D Ma et al. (2023). The results demonstrate that GRAFT effectively reduces both upstream geometry-modeling costs and downstream language-inference costs while preserving spatial reasoning capabilities.

Our main contributions are summarized as follows:

- Starting from the selective geometrization of important visual evidence, we formulate the efficiency problem of Spatial MLLMs as hierarchical geometric-resource allocation across both view and token granularities, and accordingly propose the training-free GRAFT framework.
- We develop shallow cross-view selection to reduce geometric encoding costs and geometric token refinement under explicit semantic coverage constraints. The latter improves spatial representativeness while keeping both the sensitivity-weighted mean semantic coverage error and the worst-case coverage radius within their initialization bounds.
- We evaluate GRAFT on three public benchmarks. The results show that it maintains competitive accuracy across four token budgets while substantially improving computational efficiency.

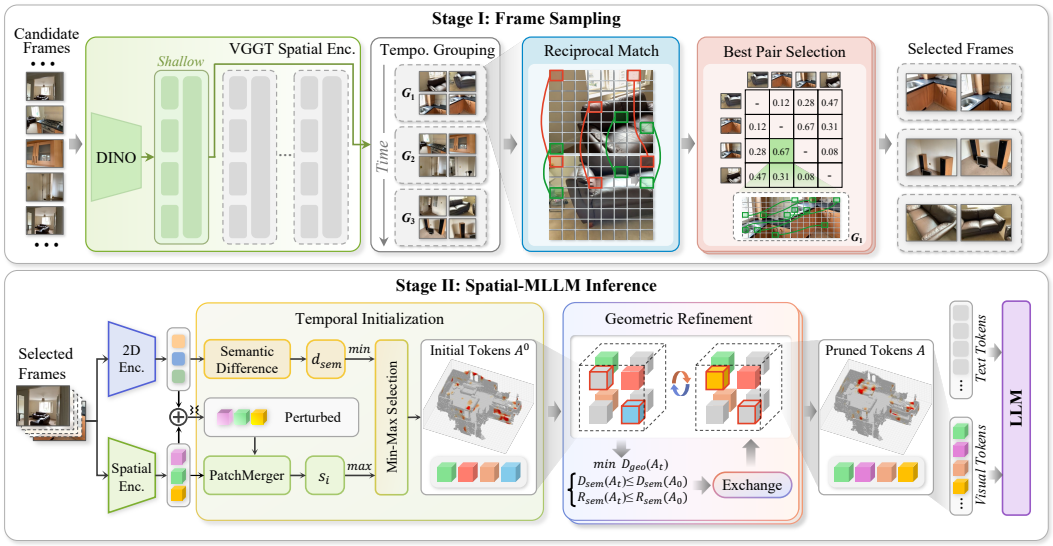

Figure 2: Overall framework of the proposed GRAFT.

2 Related Work

Coupling semantic visual encoders with geometry branches has become a major paradigm for RGB-based spatial MLLMs, exemplified by Spatial-MLLM Wu et al. (2025), VG-LLM Zheng et al. (2025b), *etc*. This architecture introduces costly multi-view geometry encoding alongside long visual-token sequences. Its video-based backbone allows existing attention, diversity, and text-guided pruning methods to be adapted Zhang et al. (2025); Alvar et al. (2025); Yu et al. (2026), but these approaches do not explicitly account for 3D scene structure. Recent spatial compression methods incorporate voxel structure and geometry-aware selection or merging Li et al. (2026); Baek & Kong (2026); Ling et al. (2026). Nevertheless, preserving important semantic content while reducing spatial redundancy remains challenging. These methods also operate after geometry has been computed, leaving the cost of geometric encoding largely unchanged. GRAFT addresses both issues by selecting frames before full geometry encoding and subsequently using geometry to refine token selection while preserving semantic evidence, significantly reducing computation at both stages. A more detailed version of Related Work can be found in Appendix A.

3 Proposed Method

3.1 Overall Framework

Spatial-MLLM Wu et al. (2025) inference incurs two major costs: full geometric encoding of a large set of candidate views during frame sampling, and the computational and memory overhead of processing long visual sequences in the language model during inference. To address both, GRAFT performs hierarchical geometry routing at the frame and token levels, as shown in Fig. 2. Frame-level routing selects observations for full geometric modeling, while token-level routing compresses the resulting geometry-enhanced representations for language model inference.

Specifically, given a candidate frame set $\mathcal{C} = \{I_i\}_{i=1}^{F}$, a question q, a frame budget L, and a visual token budget K, the frame router $\mathcal{R}_\mathrm{f}$ selects L views before full geometric encoding, and the token router $\mathcal{R}_\mathrm{t}$ retains exactly K of the resulting fused tokens $\mathbf{E}$. The overall pipeline is:

$$\underbrace{\mathcal{C} \xrightarrow{\mathcal{R}_\mathrm{f}} \mathcal{C}_\mathcal{S}}_{\text{Stage I: Frame Sampling } (F \to L)} \quad \Longrightarrow \quad \underbrace{\mathcal{C}_\mathcal{S} \xrightarrow{\mathcal{E}_\mathrm{Spatial}} (\mathbf{E}, \mathbf{P}) \xrightarrow{\mathcal{R}_\mathrm{t}(K)} \widetilde{\mathbf{E}}, \quad (\widetilde{\mathbf{E}}, q) \xrightarrow{\mathrm{LLM}} y}_{\text{Stage II: Spatial-MLLM Inference } (L \to y)}, \tag{1}$$

where $\mathbf{E} = [\mathbf{e}_i]_{i=1}^{N}$ and $\mathbf{P} = [\mathbf{p}_i]_{i=1}^{N} \in \mathbb{R}^{N\times 3}$ denote the full sequence of fused tokens and their aligned world coordinates, respectively, and $\widetilde{\mathbf{E}}$ is a sparse visual sequence of length K. The question q is supplied to the language model only after routing.

3.2 Stage I: Frame Sampling before Full Geometry Encoding

Stage I performs low-cost frame routing before full geometric encoding. From shallow cross-view features, GRAFT evaluates frame pairs as the basic multi-view evidence unit: their shared content

supports correspondence, while their viewpoint difference provides the geometric baseline. It then selects one high-utility pair per temporal group and passes both frames to the full geometry encoder.

Shallow Cross-View Representations. A full VGGT forward pass involves multiple layers of alternating frame-wise and global aggregation, followed by the prediction of camera parameters, depth maps, and world points Wang et al. (2025a). To avoid executing this entire pipeline before frame selection, GRAFT runs only a shallow prefix $\mathcal{E}_{\text{shallow}}$ of the geometry backbone. After removing the special tokens, the patch features of each frame are pooled into a $P \times P$ grid and normalized:

$$\mathbf{H} = \mathcal{E}_{\text{shallow}}(\mathcal{C}), \qquad \mathbf{f}_i^u = \text{Norm}(\text{Pool}_{P\times P}(\mathbf{H}_i)_u), \quad u = 1, \dots, M, \quad M = P^2. \tag{2}$$

The shallow prefix performs limited cross-view interaction across candidate frames and preserves local appearance and grid layout before explicit 3D prediction. Its output therefore captures both per-frame appearance and preliminary inter-frame relations, which can be used directly for reciprocal matching and view scoring. Because computation stops before the remaining aggregation layers and the camera, depth, and world-point prediction heads, GRAFT avoids running full geometric inference on every candidate frame.

View Utility Estimation and Temporal Selection. To preserve temporal coverage, we first partition the candidate sequence into G contiguous temporal groups and set $L = 2G$. Within each group, we score candidate frame pairs by combining reciprocal correspondence confidence with normalized image-plane displacement. Similarity-only scores favor near-duplicate views, whereas displacement-only scores may select pairs with little overlap. We therefore establish forward nearest-neighbor matches using cosine similarity and retain only reciprocal correspondences that are consistent in both directions:

$$\begin{aligned} \pi_{i\to j}(u) &= \arg\max_{v\in\{1,\dots,M\}} (\mathbf{f}_i^u)^\top \mathbf{f}_j^v, \\ r_{ij}(u) &= \mathbb{I}[\pi_{j\to i}(\pi_{i\to j}(u)) = u]. \end{aligned} \tag{3}$$

Let $\mathbf{c}_u \in [0,1]^2$ denote the normalized coordinates of grid location u. We define the utility of a view pair as:

$$Q_{ij} = \frac{1}{M}\sum_{u=1}^{M} r_{ij}(u) \left[(\mathbf{f}_i^u)^\top \mathbf{f}_j^{\pi_{i\to j}(u)}\right]_+ \frac{\left\|\mathbf{c}_u - \mathbf{c}_{\pi_{i\to j}(u)}\right\|_2}{\sqrt{2}}. \tag{4}$$

The score Q_{ij} aggregates correspondence confidence weighted by normalized image-plane displacement, where $[\cdot]_+$ clips negative similarities to zero and $\sqrt{2}$ normalizes displacement by the maximum distance in the normalized image plane. Low-overlap pairs have few reciprocal correspondences, while near-duplicate views exhibit limited displacement; neither therefore receives a high score.

Selecting the maximum-scoring pair from each group yields the candidate index set:

$$(i_g^*, j_g^*) = \arg\max_{i<j,\, i,j\in\mathcal{G}_g} Q_{ij}, \quad \mathcal{S}_{\text{obs}} = \bigcup_{g=1}^{G} \{i_g^*, j_g^*\}, \quad |\mathcal{S}_{\text{obs}}| = L. \tag{5}$$

The endpoints of the selected pairs are sorted in temporal order to form the final frame sequence. This sequence contains exactly L distinct views and is passed to the full geometry encoder for Stage II.

3.3 Stage II: Sparse Spatial-MLLM Inference

Stage II performs geometry-aware token routing after the selected frames are encoded by Spatial-MLLM. GRAFT first combines model sensitivity and semantic coverage to construct a temporally balanced evidence set, and then uses world coordinates to refine spatially invalid representative relations under an exact token budget. The selected original fused tokens are finally passed to the language model.

Fused Semantic-Geometric Representations. Given the selected L frames, the frozen semantic vision encoder produces semantic tokens $\mathbf{Z}$, while the frozen VGGT spatial branch runs the full frame and global aggregator to obtain spatial features $\mathbf{G}$ and feeds the aggregated features to a DPT head, which predicts the dense world point map $\mathbf{P}^{\text{raw}}$. The spatial features are projected through the connector Φ Wu et al. (2025) and added to the semantic tokens. The world point map is first resized to the visual grid and then pooled using the temporal and spatial merging scales τ and σ to align one coordinate with each fused visual token:

$$\begin{aligned} \mathbf{Z} &= [\mathbf{z}_i]_{i=1}^N \in \mathbb{R}^{N\times d}, \quad \mathbf{E} = \mathbf{Z} + \Phi(\mathbf{G}) = [\mathbf{e}_i]_{i=1}^N, \\ \mathbf{P} &= \text{Pool}_{\tau,\sigma}(\text{Resize}(\mathbf{P}^{\text{raw}})) = [\mathbf{p}_i]_{i=1}^N \in \mathbb{R}^{N\times 3}. \end{aligned} \tag{6}$$

Each token position is thus associated with a semantic representation $\mathbf{z}_i$, a fused representation $\mathbf{e}_i$, and world coordinates $\mathbf{p}_i$. GRAFT determines the retained indices using semantic representations,

model sensitivity, and world coordinates, but passes the corresponding fused tokens $\mathbf{e}_i$ to the language model without modification. Routing therefore changes only the composition and length of the visual sequence, without altering the token semantics learned by the base model.

Model-Sensitivity-Guided Temporal Initialization. The initial evidence set should prioritize informative tokens while covering distinct semantic content across time. Semantic distance captures content variation, while sensitivity measures the local response to feature perturbations; selecting solely by sensitivity may concentrate the budget on a few views. GRAFT therefore estimates the zeroth-order sensitivity of the frozen PatchMerger in Spatial-MLLM Wu et al. (2025) as a proxy for token importance and combines it with semantic coverage under balanced temporal allocation. Let $\mathbf{x}_i$ denote the input that produces $\mathbf{z}_i$, h the perturbation step size, $\{\mathbf{u}_r\}_{r=1}^{R}$ random directions sampled from the unit sphere, and $\mathcal{P}$ the PatchMerger:

$$s_i = \frac{1}{R}\sum_{r=1}^{R} \frac{\|\mathcal{P}(\mathbf{x}_i + h\mathbf{u}_r) - \mathcal{P}(\mathbf{x}_i - h\mathbf{u}_r)\|_2}{2h}. \tag{7}$$

This estimator requires repeated evaluations only of the PatchMerger, without backpropagation or rerunning the full visual encoder. To prevent global selection from concentrating most retained tokens in a few views, we partition the visual sequence into T contiguous temporal groups $\{\mathcal{I}_t\}_{t=1}^{T}$ and distribute the total budget as evenly as possible, preserving cross-view evidence in the sparse sequence. Within each group, semantic differences are measured by cosine distance:

$$K_t = \left\lfloor \frac{K}{T} \right\rfloor + \mathbb{I}[t \le K \bmod T], \quad \sum_{t=1}^{T} K_t = K, \quad d_{\text{sem}}(i,j) = 1 - \frac{\mathbf{z}_i^\top \mathbf{z}_j}{\|\mathbf{z}_i\|_2 \|\mathbf{z}_j\|_2}. \tag{8}$$

We normalize the sensitivity scores within each group to obtain $\bar{s}_i \in [0,1]$. Each group first retains the token with the highest sensitivity, then iteratively adds tokens that combine strong model responses with semantic coverage gains:

$$\begin{aligned} i_1^* &= \arg\max_{i \in \mathcal{I}_t} \bar{s}_i, \qquad \mathcal{A}_{t,1}^0 = \{i_1^*\}, \\ i_l^* &= \arg\max_{i \in \mathcal{I}_t \setminus \mathcal{A}_{t,l-1}^0} \bar{s}_i \min_{a \in \mathcal{A}_{t,l-1}^0} d_{\text{sem}}(i,a), \quad l = 2, \dots, K_t, \\ \mathcal{A}_{t,l}^0 &= \mathcal{A}_{t,l-1}^0 \cup \{i_l^*\}. \end{aligned} \tag{9}$$

Here, $\mathcal{A}_t^0 = \mathcal{A}_{t,K_t}^0$. The multiplicative criterion requires both model sensitivity and marginal coverage for a high selection score: tokens with low sensitivity or already well covered by the current set receive limited gain. It therefore avoids an additional mixing weight between importance and diversity. When $K \ge T$, each temporal group retains at least one token, and the groups jointly form an initial set $\mathcal{A}^0 = \bigcup_t \mathcal{A}_t^0$ that satisfies the exact total budget.

Geometric Refinement under Semantic Fidelity Constraints. The initial set determines which evidence should remain represented, but semantic proximity alone does not establish spatial substitutability: semantically similar tokens may correspond to different physical locations. GRAFT therefore applies geometric reasoning after semantic initialization, using the initial semantic coverage as a feasible reference and improving the spatial validity of the induced representative relations within that region. Given a retained set $\mathcal{A}_t$ within a temporal group, we first assign each original token to its semantically nearest representative:

$$\pi_{\mathcal{A}_t}(i) = \arg\min_{a \in \mathcal{A}_t} d_{\text{sem}}(i,a). \tag{10}$$

The representative mapping is used to evaluate coverage and guide token refinement. The final sequence retains the selected original fused tokens without aggregating features from discarded tokens. We normalize the nonnegative sensitivity scores within each group to obtain weights w_i, using uniform weights if their sum is zero. Based on the same coverage mapping, we define

$$\begin{aligned} D_{\text{sem}}(\mathcal{A}_t) &= \sum_{i \in \mathcal{I}_t} w_i \, d_{\text{sem}}(i, \pi_{\mathcal{A}_t}(i)), \quad R_{\text{sem}}(\mathcal{A}_t) = \max_{i \in \mathcal{I}_t} d_{\text{sem}}(i, \pi_{\mathcal{A}_t}(i)), \\ D_{\text{geo}}(\mathcal{A}_t) &= \sum_{i \in \mathcal{I}_t} w_i \left\| \mathbf{p}_i - \mathbf{p}_{\pi_{\mathcal{A}_t}(i)} \right\|_2. \end{aligned} \tag{11}$$

Here, D_{sem} and R_{sem} measure the sensitivity-weighted mean coverage error and the worst-case coverage radius, respectively, while D_{geo} measures the spatial mismatch of the same representative assignments in world coordinates. The first two measures constrain semantic fidelity, whereas the third assesses the spatial validity of these assignments.

Table 1: Performance comparison on VSI-Bench. "GRAFT^{S2}" refers to the version that uses only the Stage II token pruning and does not use the Stage I view selection mechanism.

Method	Venue	Numerical Questions				Multiple-Choice Questions				Avg.	Rank
		Obj. Cnt.	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order		
16 Frames, All 3,168 Tokens											
Spatial-MLLM	NeurIPS'25	64.8	35.6	58.0	54.5	40.6	64.5	36.1	61.5	53.3	–
16 Frames, Retain 1,280 Tokens (↓ 60%)											
FastV	ECCV'24	62.7	34.2	58.2	48.0	38.2	50.6	35.1	47.3	47.8	8
VisPruner	ICCV'25	63.8	33.9	58.0	52.3	40.3	56.7	34.0	53.2	50.2	6
DivPrune	CVPR'25	63.9	35.0	58.7	50.1	39.7	55.3	33.5	54.9	50.2	6
VisionTrim	ICLR'26	64.3	36.6	58.4	52.1	39.2	61.0	37.6	57.9	52.1	2
SCoRe	CVPR'26	65.2	34.5	58.5	51.3	40.1	56.9	34.5	54.5	50.7	5
Geo3DPruner	CVPR'26	65.1	33.1	58.3	50.1	37.2	60.1	35.6	60.2	51.3	4
GRAFT^{S2}	–	64.1	35.6	57.8	52.8	40.3	63.0	36.6	58.3	52.4	1
GRAFT	–	64.4	36.4	58.5	51.6	39.2	60.0	35.6	60.2	52.1	2
16 Frames, Retain 640 Tokens (↓ 80%)											
FastV	ECCV'24	62.0	32.3	56.9	43.1	39.0	43.8	36.1	48.1	45.9	8
VisPruner	ICCV'25	63.1	33.2	58.1	46.8	41.1	53.0	35.1	53.7	49.3	3
DivPrune	CVPR'25	62.9	32.2	58.2	47.1	36.9	54.4	33.5	54.2	48.8	6
VisionTrim	ICLR'26	63.1	33.0	57.8	47.9	37.8	53.3	34.5	55.3	49.0	4
SCoRe	CVPR'26	63.1	32.7	57.4	46.0	39.7	55.1	36.1	51.9	49.0	4
Geo3DPruner	CVPR'26	64.6	32.5	58.3	41.7	36.3	54.8	34.5	51.9	48.5	7
GRAFT^{S2}	–	63.9	35.6	58.0	50.9	39.2	57.8	36.1	58.3	51.2	1
GRAFT	–	64.4	36.0	58.4	50.2	39.6	57.5	34.5	57.3	51.2	1
16 Frames, Retain 320 Tokens (↓ 90%)											
FastV	ECCV'24	61.0	32.2	55.8	35.8	37.8	44.5	35.6	46.3	44.9	8
VisPruner	ICCV'25	63.2	31.4	56.7	40.9	39.3	50.9	34.5	48.2	47.1	6
DivPrune	CVPR'25	61.3	31.8	57.2	39.0	39.2	50.4	34.5	55.3	47.6	4
VisionTrim	ICLR'26	61.9	31.6	57.4	43.9	40.7	47.1	35.1	56.5	47.7	3
SCoRe	CVPR'26	63.2	31.4	57.4	41.6	39.0	46.4	35.1	58.6	47.6	4
Geo3DPruner	CVPR'26	62.8	31.1	57.6	34.3	36.8	46.6	35.1	45.0	45.2	7
GRAFT^{S2}	–	64.8	35.5	57.9	45.7	38.9	57.0	34.5	57.1	50.5	1
GRAFT	–	63.6	34.8	58.4	47.3	39.7	52.7	35.1	56.2	49.7	2
16 Frames, Retain 160 Tokens (↓ 95%)											
FastV	ECCV'24	61.2	29.3	55.7	29.4	37.9	41.3	35.6	46.4	43.5	8
VisPruner	ICCV'25	61.7	28.9	57.2	35.0	36.3	46.1	35.6	44.8	44.6	6
DivPrune	CVPR'25	60.7	30.3	56.7	34.0	40.0	50.9	34.5	53.2	46.9	3
VisionTrim	ICLR'26	62.0	30.8	57.3	38.1	39.2	43.9	35.6	48.5	45.5	5
SCoRe	CVPR'26	62.5	30.8	56.8	36.9	39.2	46.2	36.6	52.6	46.4	4
Geo3DPruner	CVPR'26	60.7	29.8	57.7	27.1	35.1	41.8	36.6	48.5	43.8	7
GRAFT^{S2}	–	64.4	34.4	57.2	43.0	40.0	54.9	34.5	53.1	49.3	2
GRAFT	–	63.6	34.9	57.8	42.3	43.8	54.9	34.0	55.2	50.1	1

Using $\mathcal{A}_t^0$ as the semantic reference, GRAFT minimizes geometric mismatch while maintaining the exact budget and preventing deterioration in either semantic measure:

$$\min_{\mathcal{A}_t \subseteq \mathcal{I}_t} D_{\text{geo}}(\mathcal{A}_t) \quad \text{s.t.} \quad |\mathcal{A}_t| = K_t, \quad D_{\text{sem}}(\mathcal{A}_t) \le D_{\text{sem}}(\mathcal{A}_t^0), \quad R_{\text{sem}}(\mathcal{A}_t) \le R_{\text{sem}}(\mathcal{A}_t^0). \tag{12}$$

The mean-error constraint controls aggregate semantic coverage, while the worst-case constraint guards against severe omissions that an average can conceal. Together, they guide geometric refinement without relaxing either coverage bound established by initialization. To solve this problem under an exact budget, we use deterministic, budget-preserving swaps within each temporal group. Specifically, we select the unretained token c with the largest contribution to the current geometric mismatch and consider replacing each retained token a in turn:

$$q_i = w_i \left\| \mathbf{p}_i - \mathbf{p}_{\pi_{\mathcal{A}_t}(i)} \right\|_2, \quad i \in \mathcal{I}_t \setminus \mathcal{A}_t, \quad c = \arg\max_{i \in \mathcal{I}_t \setminus \mathcal{A}_t} q_i, \quad \mathcal{A}_t^{a \to c} = (\mathcal{A}_t \setminus \{a\}) \cup \{c\}. \tag{13}$$

We recompute the coverage mapping for each candidate set and accept only swaps satisfying

$$\begin{aligned} D_{\text{sem}}(\mathcal{A}_t^{a \to c}) &\le D_{\text{sem}}(\mathcal{A}_t^0), \\ R_{\text{sem}}(\mathcal{A}_t^{a \to c}) &\le R_{\text{sem}}(\mathcal{A}_t^0), \\ D_{\text{geo}}(\mathcal{A}_t^{a \to c}) &< D_{\text{geo}}(\mathcal{A}_t). \end{aligned} \tag{14}$$

The algorithm chooses the feasible swap with the lowest D_{geo}, breaking ties in favor of lower D_{sem}. The coverage mapping is recomputed after each update. Each group runs for at most $|\mathcal{I}_t| - K_t$ iterations and terminates early if no feasible swap exists for the current candidate. Every accepted update preserves the group budget and semantic fidelity while strictly reducing geometric mismatch.

Sparse Visual Sequence Construction. After geometric refinement in all temporal groups, the final index set and sparse visual sequence are: $\mathcal{A} = \bigcup_{t=1}^{T} \mathcal{A}_t$, $\widetilde{\mathbf{E}} = [\mathbf{e}_i \mid i \in \mathcal{A}]$, $|\mathcal{A}| = K$. The retained indices are sorted in their original order, and GRAFT directly selects the corresponding fused tokens without merging, reconstruction, or interpolation. To preserve M-RoPE alignment, the model first computes the three-axis position indices for the full input, then applies the same selection mask to the visual tokens, attention mask, and position indices. All non-visual tokens are retained,

Table 2: Performance comparison on ScanQA and SQA3D benchmarks.

Method	Venue	ScanQA (val)							SQA3D (test)			
		BLEU-1	BLEU-4	METEOR	ROUGE-L	CIDEr	Avg.	Rank	EM-1	EM-R1	Avg.	Rank
16 Frames, All 3,168 Tokens												
Spatial-MLLM	NeurIPS'25	41.6	13.0	17.3	42.5	87.0	40.3	–	54.0	56.6	55.3	–
16 Frames, Retain 1,280 Tokens (↓ 60%)												
FastV	ECCV'24	38.5	11.8	15.6	38.3	75.4	35.9	8	51.6	54.2	52.9	8
VisPruner	ICCV'25	40.5	13.0	16.6	40.7	81.6	38.5	7	52.0	54.4	53.2	7
DivPrune	CVPR'25	41.0	13.2	16.9	41.5	83.8	39.3	3	53.1	55.6	54.4	3
VisionTrim	ICLR'26	41.1	13.4	16.8	41.3	83.4	39.2	4	52.9	55.4	54.2	5
SCoRe	CVPR'26	40.3	13.3	16.5	40.7	82.1	38.6	6	53.1	55.5	54.3	4
Geo3DPruner	CVPR'26	40.7	12.4	16.8	41.2	82.9	38.8	5	52.9	55.5	54.2	5
GRAFT^{S2}	–	41.5	11.9	17.2	42.4	86.2	39.8	1	53.7	56.2	54.9	2
GRAFT	–	41.1	12.7	16.8	41.7	84.8	39.4	2	53.7	56.4	55.0	1
16 Frames, Retain 640 Tokens (↓ 80%)												
FastV	ECCV'24	37.4	11.4	15.0	37.3	72.7	34.8	8	50.4	52.9	51.7	8
VisPruner	ICCV'25	39.1	12.0	15.9	39.4	78.1	36.9	7	51.2	54.1	52.7	7
DivPrune	CVPR'25	39.5	12.1	16.0	39.5	79.0	37.2	6	52.2	54.8	53.5	4
VisionTrim	ICLR'26	39.8	12.0	16.2	40.0	79.5	37.5	4	51.7	54.3	53.0	6
SCoRe	CVPR'26	39.8	12.0	16.1	39.7	79.4	37.4	5	51.9	54.4	53.2	5
Geo3DPruner	CVPR'26	40.8	12.6	16.4	40.5	81.0	38.3	3	52.8	55.3	54.1	2
GRAFT^{S2}	–	41.5	13.1	17.1	42.3	85.0	39.8	1	53.5	56.2	54.9	1
GRAFT	–	40.3	11.2	16.6	41.2	82.6	38.4	2	52.7	55.5	54.1	2
16 Frames, Retain 320 Tokens (↓ 90%)												
FastV	ECCV'24	36.4	11.2	14.5	35.9	70.2	33.6	8	49.7	52.3	51.0	8
VisPruner	ICCV'25	38.3	11.7	15.4	38.2	75.6	35.8	6	50.6	53.5	52.0	6
DivPrune	CVPR'25	37.5	10.9	14.9	37.1	72.8	34.6	7	50.6	53.4	52.0	6
VisionTrim	ICLR'26	38.9	11.6	15.8	39.2	77.5	36.6	3	50.9	53.7	52.3	4
SCoRe	CVPR'26	38.4	10.8	15.5	38.7	75.9	35.9	5	51.0	53.8	52.4	3
Geo3DPruner	CVPR'26	39.1	11.3	15.7	38.6	76.2	36.2	4	51.0	53.5	52.2	5
GRAFT^{S2}	–	40.4	13.2	16.5	40.8	82.5	38.7	1	52.3	55.0	53.7	2
GRAFT	–	40.4	11.5	16.4	40.8	81.5	38.1	2	52.5	55.1	53.8	1
16 Frames, Retain 160 Tokens (↓ 95%)												
FastV	ECCV'24	36.2	11.1	14.4	35.6	69.3	33.3	8	48.4	51.2	49.8	8
VisPruner	ICCV'25	37.2	11.1	14.7	36.4	71.8	34.2	6	49.6	52.4	51.0	6
DivPrune	CVPR'25	36.9	10.8	14.5	36.2	71.5	34.0	7	50.1	52.8	51.4	5
VisionTrim	ICLR'26	37.9	11.3	15.1	37.7	74.1	35.2	4	50.2	52.9	51.5	4
SCoRe	CVPR'26	38.0	10.9	15.3	38.0	74.2	35.3	3	50.5	53.2	51.8	3
Geo3DPruner	CVPR'26	37.9	10.8	15.0	37.3	72.2	34.6	5	49.3	51.9	50.6	7
GRAFT^{S2}	–	40.2	12.3	16.2	40.4	81.4	38.1	1	52.2	55.0	53.6	1
GRAFT	–	40.0	11.3	16.1	40.2	79.6	37.4	2	52.1	54.9	53.5	2

and the positional offset for generation is adjusted to account for the compressed sequence length. Consequently, $\widetilde{\mathbf{E}}$ remains compatible with the original model's KV-cache decoding pipeline.

Stage I restricts full geometric encoding to the selected L frames, while Stage II reduces the language model's visual prefill sequence to K tokens. Both routing stages are training-free, introduce no additional model parameters, and satisfy the prescribed budgets exactly.

4 Experiments

4.1 Experimental Setup

We evaluate GRAFT on three spatial reasoning benchmarks: VSI-Bench Yang et al. (2025), ScanQA Azuma et al. (2022), and SQA3D Ma et al. (2023). For VSI-Bench, numerical tasks are evaluated using Mean Relative Accuracy (MRA), and multiple-choice tasks are evaluated using accuracy. ScanQA is evaluated using BLEU-1, BLEU-4, METEOR, ROUGE-L, and CIDEr, while SQA3D is evaluated using EM-1 and EM-R1.

We use the official Spatial-MLLM-v1.1-Instruct-135K Wu et al. (2025) as our base model. Stage I selects 16 frames from 128 candidates using a single shallow VGGT attention cycle, an 8×8 pooling grid, and 8 temporal groups. Stage II retains exactly K visual tokens for $K \in \{160, 320, 640, 1280\}$, with zeroth-order sensitivity estimated using 64 random directions and a finite-difference step size of $h = 0.01$. All configurations are shared across tasks and token budgets without dataset-specific tuning. Except for efficiency measurements, all experiments are conducted on four NVIDIA A800 80GB GPUs. For baseline methods, we prioritize official implementations and reproduce methods without publicly available code following the settings reported in their papers.

4.2 Comparison with Existing Methods

Accuracy. Tables 1 and 2 compare GRAFT with existing visual token pruning methods across compression budgets. Both GRAFT variants rank among the top two configurations across all three benchmarks and four budgets. On VSI-Bench, full GRAFT's gains over the best compared baseline increase from 1.9 points at 640 tokens to 2.0 at 320 and 3.2 at 160. At the smallest budget, it achieves an average score of 50.1, retaining approximately 94% of the dense baseline's score with

Table 3: Two-stage efficiency comparison between Spatial-MLLM and GRAFT. GRAFT retains $K = 160$ visual tokens in Stage II, while Spatial-MLLM retains all 3,168 visual tokens.

Method	Stage I			Stage II				Total	
	Latency (s)	TFLOPs	VRAM (GB)	TTFT (s)	TFLOPs	VRAM (GB)	KV-Cache (MB)	Latency (s)	TFLOPs
Spatial-MLLM	14.80	2,072.3	35.7	1.23	185.1	14.5	119.4	16.04	2,257.4
GRAFT (Ours)	2.41	299.3	16.0	0.46	73.8	11.6	8.5	2.86	373.1

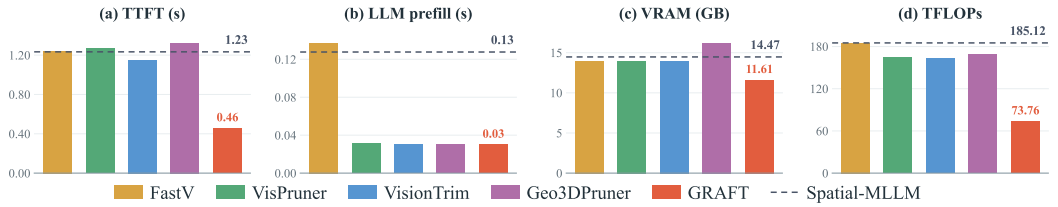

Figure 3: Stage II efficiency comparison at $K = 160$. From left to right: TTFT, LLM prefill, peak VRAM, and TFLOPs. Dashed lines indicate the Spatial-MLLM Dense baseline ($K = 3168$).

only 5% of its visual tokens. At budgets of 320 and 160 tokens, full GRAFT also improves over the best compared baselines by 1.5 and 2.1 points on ScanQA, and by 1.4 and 1.7 points on SQA3D, respectively. Relative to GRAFT^{S2}, full GRAFT substantially reduces upstream geometric encoding costs, with score changes of at most 0.8 points in either direction on VSI-Bench and SQA3D, and decreases of 0.4-1.4 points on ScanQA. These modest accuracy changes enable the full pipeline to extend efficiency gains beyond token compression, while the two configurations offer flexibility in balancing task accuracy and computational cost.

Computational Efficiency. All efficiency experiments are conducted on a single NVIDIA A800 80GB GPU. Table 3 compares the two-stage computational costs of Spatial-MLLM and the full GRAFT. In Stage I, GRAFT reduces latency from 14.80 s to 2.41 s (83.7%) and computation by 85.6%. In Stage II, at $K = 160$, time to first token (TTFT) and computation decrease by 62.9% and 60.2%, respectively, while the key-value cache (KV-cache) size decreases from 119.4 MB to 8.5 MB (92.9%). Figure 3 provides a further fine-grained comparison of Stage II efficiency. Under the same budget of $K = 160$, GRAFT achieves a TTFT of 0.46 s, 60.3% lower than that of the next-best method, VisionTrim. Its computational cost and peak GPU memory usage are 54.9% and 16.9% lower than those of the respective next-best methods. These results indicate that GRAFT incurs lower computational and GPU memory costs under the same compression budget.

4.3 Ablation Study

Analysis of the Two-Stage Components. To analyze the contributions of the two routing stages, we conduct a component ablation, as shown in Table 4. The configuration with neither Uniform nor FR enabled uses the default frame sampling strategy of Spatial-MLLM. With all visual tokens retained, Stage I frame routing increases the average score from 51.5 with uniform sampling to 52.4, indicating that shallow cross-view relations help select more informative views. With Stage II token routing fixed at $K = 160$, frame routing improves the score from 49.6 with uniform sampling to 50.1, while the default frame sampling strategy yields 49.3. These results show that frame routing provides consistent gains both with and without token compression.

Table 4: Ablation of the two-stage components. Uniform denotes uniform frame sampling; FR and TR denote frame routing and token routing, respectively. Choice and Numer. denote multiple-choice questions and numerical questions, respectively.

Stage I		Stage II	K	Choice	Numer.	Avg. Score
Uniform	FR	TR				
✓	✗	✗	3168	50.8	52.3	51.5
✗	✓	✗	3168	52.1	52.6	52.4
✗	✗	✓	160	48.6	50.0	49.3
✓	✗	✓	160	49.8	49.4	49.6
✗	✓	✓	160	50.2	50.1	50.1

Stage I View Scoring Strategy. We further analyze the contribution of each view scoring term in Stage I, with results reported in Table 5. The full model achieves the highest score of 50.1. Removing reciprocal correspondence, similarity weighting, and image-plane displacement reduces the score to 48.7, 48.3, and 49.1, respectively, indicating that all three factors contribute to effective view selection. Removing similarity weighting causes the largest performance drop, suggesting that reliable feature matching confidence is an important basis for assessing view utility.

Table 5: Ablation of the Stage I scoring terms. MC, SW, and ID denote Mutual Correspondence, Similarity Weight, and Image Displacement, respectively.

Configuration	Choice	Numer.	Avg. Score
w/o MC	48.1	49.2	48.7
w/o SW	47.1	49.4	48.3
w/o ID	49.4	48.8	49.1
Full Model	50.2	50.1	50.1

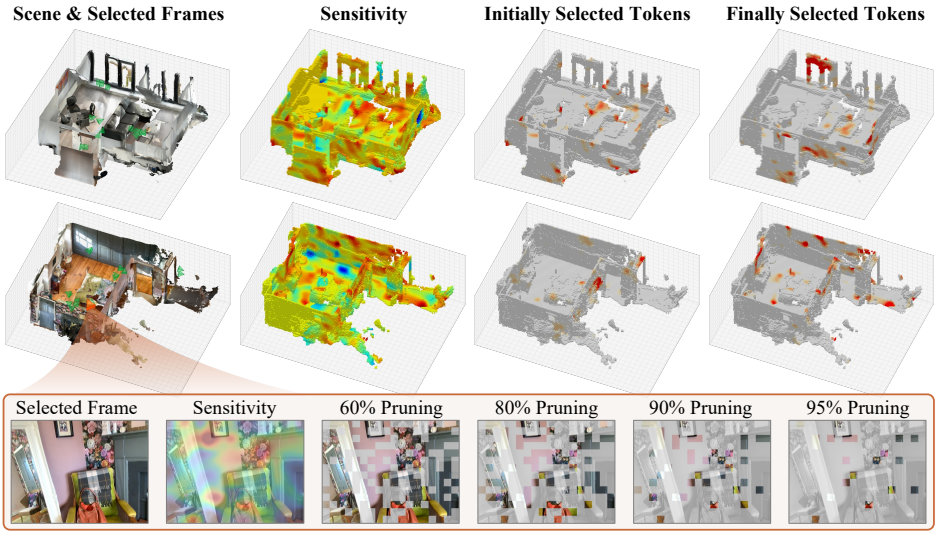

Figure 4: Visualizations of GRAFT's frame and token routing process. Top: original scene, sensitivity response, initialized tokens, and tokens after geometric refinement. Green and red square pyramids indicate Stage I-selected views and the view corresponding to the bottom panel, respectively. Bottom: original frame, sensitivity response, and visual tokens retained at different pruning ratios.

Stage II Geometric Swaps and Semantic Constraints. As shown in Table 6, we ablate the geometric refinement module and its constraints to analyze their roles after semantic initialization. Removing geometric refinement reduces the average score from 50.1 to 48.0. The highest score is achieved when both the mean coverage error constraint and the worst-case coverage radius constraint are retained, supporting their complementary roles in geometric refinement.

Table 6: Ablation of Stage II geometric refinement and constraints. GS, Mean, and Worst denote geometric swaps, the mean coverage error constraint, and the worst-case coverage radius constraint, respectively.

GS	Mean	Worst	Choice	Numer.	Avg. Score
✗	✗	✗	46.4	49.5	48.0
✓	✗	✗	47.1	49.4	48.3
✓	✓	✗	46.5	49.2	47.9
✓	✗	✓	46.4	49.5	48.0
✓	✓	✓	50.2	50.1	50.1

Qualitative Analysis. As shown in Fig 4, GRAFT effectively preserves critical visual evidence in both the 3D scene and single-frame visualizations. The views selected by Stage I cover the main scene regions, while sensitivity responses are concentrated on key objects and their boundaries. Semantic initialization followed by geometric refinement further improves the token distribution. As the single-frame pruning ratio increases from 60% to 95%, redundant tokens are progressively removed, while core spatial evidence remains well preserved. These results indicate that GRAFT can construct representative sparse visual evidence even under high compression ratios.

Generalization to Other Spatial MLLMs. To evaluate the applicability of GRAFT to different base models, we apply it to VG-LLM and compare it with other representative pruning methods on VSI-Bench. As shown in Table 7, when all methods retain 160 visual tokens, GRAFT achieves an average score of 47.2, outperforming SCoRe by 2.2 points. Compared with the unpruned VG-LLM, GRAFT retains approximately 94.0% of the performance, demonstrating that it effectively preserves key evidence for spatial reasoning under substantial visual token compression.

Table 7: Performance comparison on VSI-Bench using VG-LLM. All pruning methods retain 160 visual tokens.

Method	Choice	Numer.	Avg. Score
VG-LLM	46.1	54.1	50.2
FastV	35.6	46.2	41.1
SCoRe	41.4	48.4	45.0
Geo3DPruner	38.6	40.7	39.7
GRAFT	43.9	50.2	**47.2**

5 Conclusion

We propose GRAFT, a training-free two-stage routing framework for spatial multimodal large language models. Stage I selects observations based on shallow cross-view relations to reduce geometric encoding costs, while Stage II combines model sensitivity, semantic coverage, and 3D geometric information for token selection and refinement. Experimental results show that GRAFT maintains strong spatial reasoning performance while substantially compressing visual inputs and effectively reducing computation, latency, and memory overhead.

AI USE STATEMENT

We used generative AI tools to improve the clarity and readability of the manuscript and to check grammar. The authors reviewed all AI-assisted revisions to ensure that they accurately reflected the intended meaning and take full responsibility for the final content of this paper.

REPRODUCIBILITY STATEMENT

To support reproducibility, we describe the GRAFT framework and its two routing stages in Sec. 3. Sec. 4.1 specifies the evaluation benchmarks, metrics, base model, and experimental settings. Appendix B provides further implementation details, including hyperparameters, the random seed, hardware and software configurations, and the protocols for implementing comparison methods. We also provide an anonymous link to our source code to facilitate reproduction of the reported results.

REFERENCES

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. DivPrune: Diversity-based visual token pruning for large multimodal models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9392–9401, 2025.

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

Changwoo Baek and Kyeongbo Kong. 3DZip: Spatial-aware feature diversity-guided token compression for 3D question answering. *arXiv preprint arXiv:2608.01185*, 2026.

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision*, pp. 19–35, 2024.

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Peihao Wang, Huaizhi Qu, Shijie Zhou, et al. VLM-3R: Vision-language models augmented with instruction-aligned 3D reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 31054–31065, 2026.

Tianyu Fu, Tengxuan Liu, Qinghao Han, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. FrameFusion: Combining similarity and importance for video token reduction on large vision language models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22654–22663, 2025.

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.

Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, et al. Chat-Scene: Bridging 3D scene and large language models with object identifiers. *Advances in Neural Information Processing Systems*, 37: 113991–114017, 2024.

Hsiang-Wei Huang, Fu-Chen Chen, Wenhao Chai, Che-Chun Su, Lu Xia, Sanghun Jung, Cheng-Yen Yang, Jenq-Neng Hwang, Min Sun, and Cheng-Hao Kuo. Zero-shot 3D question answering via voxel-based dynamic token compression. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19424–19434, 2025.

Youngeun Kim, Youjia Zhang, Huiling Liu, Aecheon Jung, Sunwoo Lee, and Sungeun Hong. ZOO-Prune: training-free token pruning via zeroth-order gradient estimation in vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39572–39582, 2026.

Han Li, Zehao Huang, Jiahui Fu, Naiyan Wang, and Si Liu. Geometry-guided 3d visual token pruning for video-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9615–9625, 2026.

Peng Ling, Yingda Yin, Lingting Zhu, Weikai Chen, Shengju Qian, Zeyu Hu, Xin Wang, and Wenming Yang. CoverPrune: Coverage-driven token pruning for 3D VLMs via optimal transport. In *European Conference on Computer Vision*, pp. 499–516, 2026.

Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. SQA3D: Situated question answering in 3D scenes. In *International Conference on Learning Representations*, 2023.

Hui Sun, Shiyin Lu, Huanyu Wang, Qing-Guo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Ming Li. MDP3: a training-free approach for list-wise frame selection in Video-LLMs. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24090–24101, 2025.

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

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3D perception model with persistent state. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10510–10522. IEEE, 2025b.

Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-MLLM: Boosting MLLM capabilities in visual-based spatial intelligence. *Advances in neural information processing systems*, 38:13569–13597, 2025.

Tong Xu, Hailong Shi, and Xingyu Gao. SCoRe: Salience-coverage reduction for vision token pruning in vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24686–24695, 2026.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643, 2025.

Hanxun Yu, Wentong Li, Xuan Qu, Song Wang, Junbo Chen, and Jianke Zhu. VisionTrim: Unified vision token compression for training-free mllm acceleration. *The Fourteenth International Conference on Learning Representations*, 2026.

Jian Zhang, Shijie Zhou, Bangya Liu, Achuta Kadambi, and Zhiwen Fan. SpatialStack: Layered geometry-language fusion for 3D VLM spatial reasoning. *arXiv preprint arXiv:2603.27437*, 2026.

Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond text-visual attention: Exploiting visual cues for effective token pruning in vlms. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20857–20867, 2025.

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

Duo Zheng, Yanyang Li, Liwei Wang, et al. Learning from videos for 3D world: Enhancing MLLMs with 3d vision geometry priors. *Advances in neural information processing systems*, 38: 20560–20586, 2025b.

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 *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4295–4305, 2025.

Appendix

CONTENTS

A Related Work **13**
A.1 Spatial MLLM 13
A.2 MLLM Token Pruning 13

B Implementation Details **14**

C Supplementary Experiments **14**
C.1 Quantitative Results 14
C.2 Qualitative Results 15

A RELATED WORK

A.1 SPATIAL MLLM

Spatial MLLMs extend language reasoning to three-dimensional scenes. Early methods introduce scene-level or object-centric 3D representations into language models Hong et al. (2023); Huang et al. (2024), while LLaVA-3D and Video-3D LLM augment image features with positions derived from depth and camera parameters Zhu et al. (2025); Zheng et al. (2025a). With geometry foundation models such as VGGT and CUT3R Wang et al. (2025a;b), combining a pretrained semantic visual encoder with a dedicated geometry branch has emerged as a major architectural paradigm for RGB-based spatial MLLMs. Spatial-MLLM and VG-LLM exemplify this design by fusing semantic and geometric features Wu et al. (2025); Zheng et al. (2025b); VLM-3R and SpatialStack further explore instruction-aligned reconstruction features and multi-level geometry–language fusion Fan et al. (2026); Zhang et al. (2026).

This paradigm introduces substantial computation beyond conventional video understanding: multi-view geometry encoding processes dense observations, while long sequences of geometry-enhanced tokens increase language-model inference costs. Even frame sampling can require expensive geometric inference. Spatial-MLLM, for example, predicts camera parameters and depth for candidate views before selecting frames by voxel coverage Wu et al. (2025). Improving inference speed and memory efficiency therefore requires attention to both geometric encoding and the subsequent processing of visual evidence.

A.2 MLLM TOKEN PRUNING

Structurally, removing the geometry branch from this paradigm recovers a video MLLM, making existing visual-token compression a natural starting point. Such methods exploit attention and feature coverage Chen et al. (2024); Alvar et al. (2025); Zhang et al. (2025); Xu et al. (2026), projector sensitivity Kim et al. (2026), or textual and temporal redundancy Yu et al. (2026); Fu et al. (2025); frame selection also uses query relevance and diversity Sun et al. (2025). These criteria reduce token costs but do not explicitly determine whether visually similar observations are interchangeable in 3D, leaving spatially distinct evidence vulnerable to compression.

Geometry-aware compression remains a comparatively small line of research. DTC and Geo3DPruner exploit voxel structure and cross-view redundancy Huang et al. (2025); Li et al. (2026), while 3DZip combines feature diversity with spatially constrained merging Baek & Kong (2026). CoverPrune further integrates feature, spatial, and temporal discrepancies through optimal-transport coverage Ling et al. (2026). These methods couple semantic and geometric criteria through selection or aggregation, leaving semantic coverage free to change during geometric compression.

Moreover, these spatial compression methods assume geometry is already available, leaving the upstream cost of producing it largely unaddressed. GRAFT targets both stages: it routes frames before full geometry encoding, then refines a sensitivity-aware token subset using 3D coordinates under explicit semantic coverage constraints. This reduces both geometric computation and language-model input while retaining semantically representative, spatially consistent evidence.

B Implementation Details

We adopt the official Spatial-MLLM-v1.1-Instruct-135K Wu et al. (2025) as the base model, following its Qwen2.5-VL Bai et al. (2025b) vision-language backbone and frozen VGGT-1B spatial encoder Wang et al. (2025a). GRAFT operates directly on the pretrained model at inference time without updating any model parameters. Except for the computational efficiency experiments, all experiments are conducted on 4× NVIDIA A800 80GB GPUs. Different samples are distributed across GPUs using data parallelism, with a per-GPU batch size of 1. The model runs in BF16 precision with FlashAttention.

For Stage I, we uniformly sample $F = 128$ candidate frames from each video and execute only one frame-wise and global attention cycle of VGGT. The patch representations of each frame are then adaptively pooled into an 8×8 grid. The candidate sequence is evenly partitioned into $G = 8$ consecutive temporal groups, from each of which one view pair is selected, resulting in a fixed target frame budget of $L = 16$. This configuration is kept fixed across all tasks and downstream token budgets.

For Stage II, the 16 input frames are encoded by the base model into $N = 3{,}168$ geometry-enhanced visual tokens. We evaluate four token budgets, $K \in \{160, 320, 640, 1280\}$, and verify during each inference run that the actual number of retained tokens is exactly equal to K. Zeroth-order sensitivity estimation uses $R = 64$ random directions with a finite-difference step size of $h = 0.01$, and the random seed is fixed to 20260815. The directions are processed in blocks of four; this blocking strategy only controls peak computational cost and does not alter the sampled directions or the final selection results. Temporal grouping and world-coordinate alignment follow the temporal merging scale and 2×2 spatial merging scale of the base model. With 16 input frames, this yields $T = 8$ temporal groups, across which the token budget is balanced according to Eq. (9). All configurations are shared across tasks and token budgets, without dataset-specific tuning.

For comparison methods, we prioritize official implementations whenever available. If the code is not publicly released, we reproduce the method according to the algorithm and default hyperparameters reported in the corresponding paper. All methods use the same base checkpoint, inference precision, per-GPU batch size, and exact token budget, without additional tuning for any specific evaluation dataset.

C Supplementary Experiments

C.1 Quantitative Results

VGGT Encoding Depth. We further analyze the effect of the VGGT encoding depth in Stage I, as shown in Table 8. Using only one frame-global attention cycle achieves the highest score of 50.1. Increasing the encoding depth to two and three cycles reduces the score to 48.4 and 48.5, respectively, while even the full VGGT geometry encoder achieves only 49.4. These results indicate that shallow representations already provide sufficient cross-view relations for frame selection.

Table 8: Ablation of VGGT encoding depth.

Depth	Choice	Numerical	Avg. Score
1	50.2	50.1	50.1
2	46.9	49.9	48.4
3	47.0	49.9	48.5
24 (full)	49.6	49.1	49.4

Pooling Grid for Shallow Features. We analyze the effect of the pooling grid resolution for shallow features on model performance, with results reported in Table 9. An 8×8 grid achieves the highest score of 50.1. Reducing the resolution to 4×4 decreases the score to 49.0, while increasing it to 16×16 and 32×32 reduces the score to 47.6 and 48.4, respectively. Coarser grids weaken the discriminative capacity of local correspondences, whereas excessively fine grids make routing more sensitive to local noise. Considering both accuracy and computational cost, we use 8×8 as the default pooling grid.

Efficiency under Different Token Budgets. As shown in Fig. 5, GRAFT maintains consistent Stage II efficiency advantages across different visual token budgets. When $K = 320, 640$, and 1280, GRAFT achieves TTFTs of 0.48, 0.52, and 0.60 s, respectively, reducing latency by 58.5%, 55.5%,

Table 9: Ablation of the pooling grid.

Pooling Grid	Choice	Numerical	Avg. Score
4×4	48.5	49.5	49.0
8×8	50.2	50.1	50.1
16×16	46.1	49.0	47.6
32×32	46.9	49.8	48.4

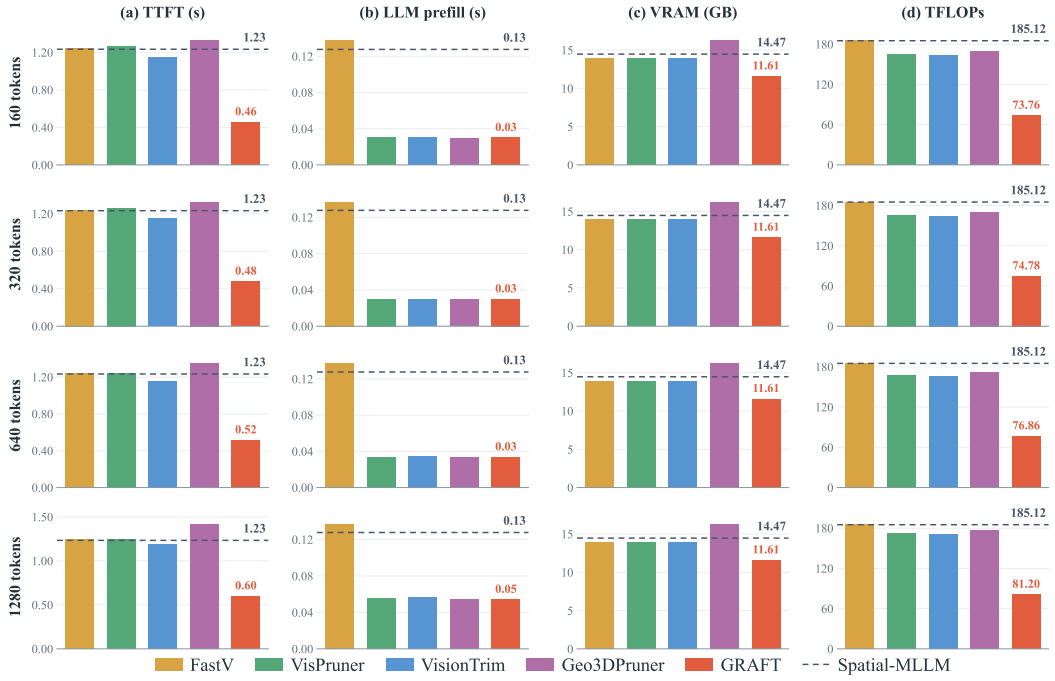

Figure 5: Stage II efficiency comparison under different visual token budgets ($K = 160, 320, 640, 1280$). From left to right, each group reports time to first token (TTFT), LLM prefill time, peak GPU memory (VRAM), and computation. Dashed lines indicate the reference values of the original Spatial-MLLM ($K = 3168$).

and 49.5% compared with the best-performing competing pruning method under the same budget. Its computational cost is also reduced by 54.5%, 53.9%, and 52.5%, respectively, indicating that GRAFT effectively controls the additional computation introduced by longer visual sequences. Meanwhile, GRAFT maintains a peak GPU memory usage of 11.61 GB across all three budgets, 16.9% lower than the best competing method. Overall, GRAFT preserves stable advantages in latency, computation, and memory usage as the token budget increases, further demonstrating its efficiency and scalability across different compression settings.

Generalization to Other Spatial MLLMs. To evaluate the applicability of GRAFT to a different base model, we apply it to VG-LLM Zheng et al. (2025b) and compare it with other pruning methods on VSI-Bench. As shown in Table 10, with all pruning methods retaining 160 visual tokens, GRAFT achieves an average score of 47.2, exceeding the strongest competing pruning method, SCoRe (45.0), by 2.2 percentage points. It also achieves the highest score among pruning methods in seven of the eight question categories. Compared with unpruned VG-LLM (50.2), GRAFT retains approximately 94.0% of the average score. These results indicate that GRAFT preserves much of the evidence required for VG-LLM's spatial reasoning under substantial visual token compression.

C.2 Qualitative Results

To further validate GRAFT's ability to preserve critical visual evidence across different scenes, we provide additional visualizations of frame routing and token routing in 3D scenes in the appendix. As shown in Fig. 6, the views selected by Stage I provide good coverage of the major spatial regions, while the model sensitivity responses are concentrated on key objects and their structural boundaries. Based on these responses, semantic initialization first retains visual tokens with high

Table 10: Performance comparison on VSI-Bench using VG-LLM. All pruning methods retain 160 (5%) visual tokens.

Method	Numerical Questions				Multiple-Choice Questions				Avg.
	Obj. Cnt.	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order	
VG-LLM	68.1	37.2	58.2	61.8	46.5	40.5	34.0	60.2	50.2
FastV	61.9	31.6	53.4	34.1	31.0	40.2	29.4	37.9	41.1
SCoRe	62.5	33.2	54.2	45.8	37.0	42.6	29.4	49.5	45.0
Geo3DPruner	52.1	26.8	49.0	31.3	31.3	41.0	31.4	47.9	39.7
GRAFT (Ours)	64.0	36.0	52.2	57.7	42.3	42.9	32.0	52.6	**47.2**

model relevance and semantic representativeness, after which geometric refinement further adjusts their spatial distribution, improving 3D spatial representativeness while preserving critical semantic evidence.

To further analyze how GRAFT reduces redundancy while preserving visual evidence required for spatial reasoning across its two compression stages, we visualize the candidate frames, selected views, and visual token routing results from the same scene. As shown in Fig. 7, the 128 candidate frames contain many visually similar observations, whereas the 16 views selected by Stage I are distributed across different scene locations and still cover the major objects and spatial regions. Fig. 8 further shows the sensitivity responses of the selected frames and the retained tokens under different pruning ratios. As the pruning ratio increases from 60% to 95%, the retained tokens become progressively sparser while remaining concentrated on major objects and structural regions of the scene. These results show that GRAFT effectively removes redundant views and visual tokens while preserving an informative multi-view representation of the scene.

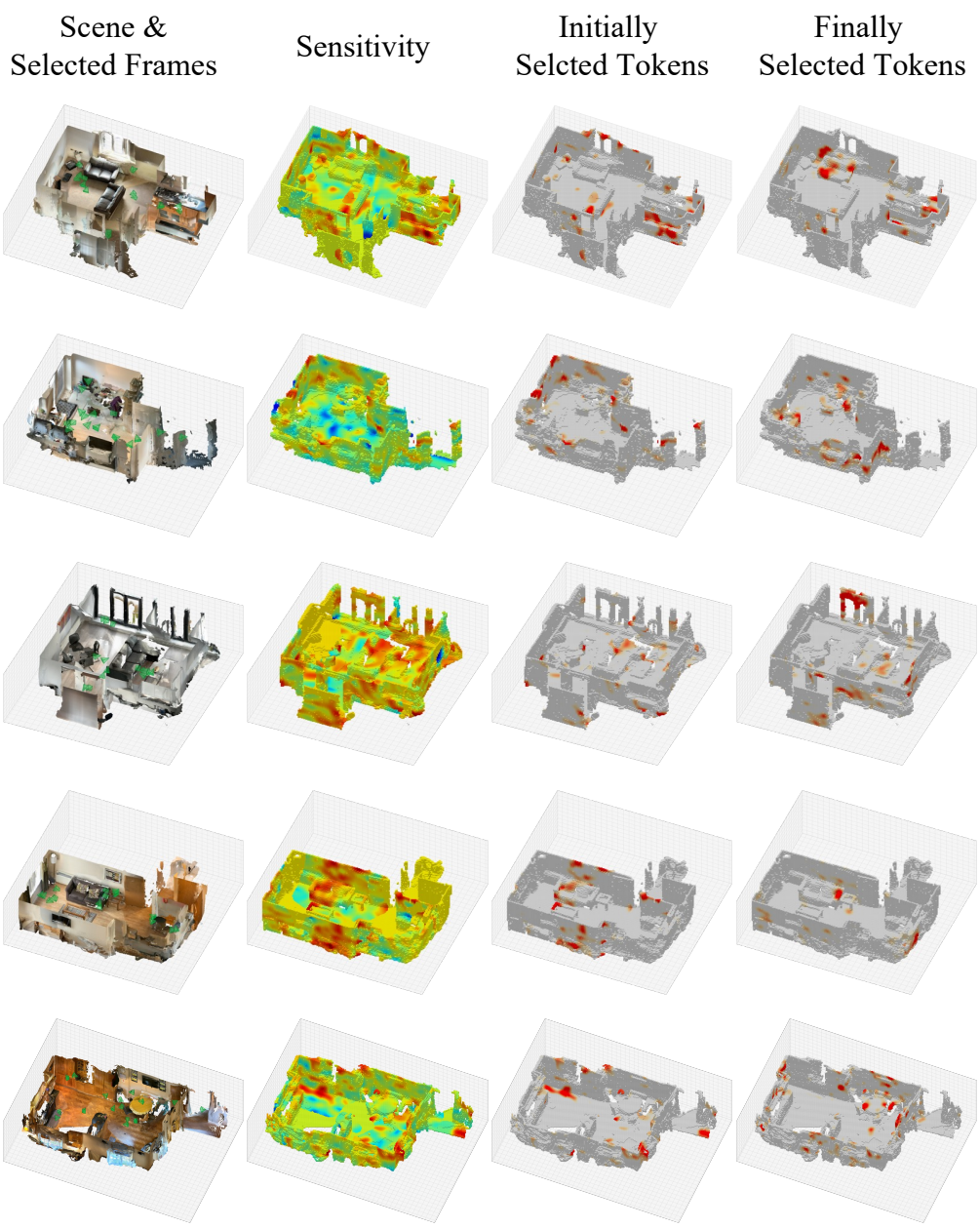

Figure 6: Additional visualizations of GRAFT's frame and token routing across 3D scenes. From left to right in each group: the original scene, sensitivity response, semantically initialized tokens, and final tokens after geometric refinement. Green square pyramids indicate the observation views selected by Stage I.

(a) Candidate Frames



(b) Scene & Selected Views

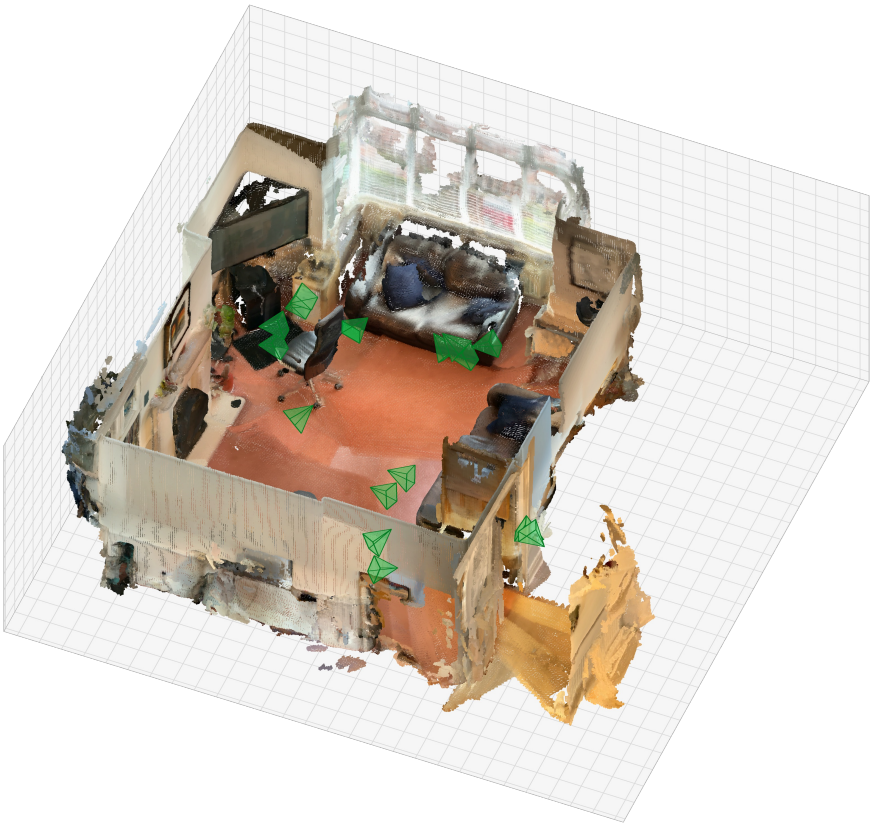

Figure 7: Frame routing visualization. (a) All 128 candidate frames from the input sequence. (b) The corresponding 3D scene and the 16 observation views selected from the candidates; green square pyramids indicate the selected views.



Figure 8: Token routing visualization for the selected views. Each row corresponds to a frame selected by Stage I. From left to right: the original frame, sensitivity response, and visual tokens retained at different pruning ratios.