3DZip: Spatial-Aware Feature Diversity-Guided Token Compression for 3D Question Answering

Changwoo Baek and Kyeongbo Kong✉

Department of Electrical and Electronics Engineering, Pusan National University, Busan, Republic of Korea
{higok18, kbkong}@pusan.ac.kr
https://cvsp-lab.github.io/3DZip

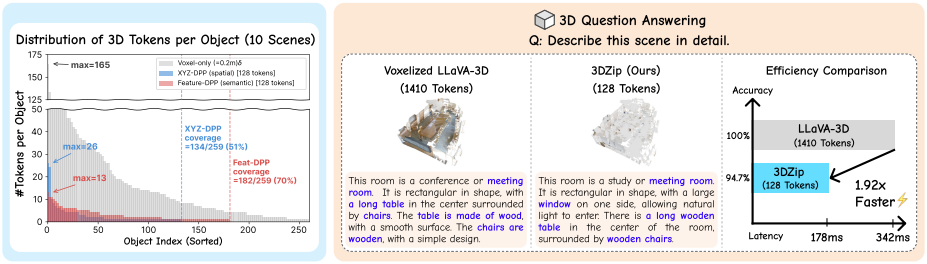
Abstract. Recent 3D vision-language models (3D VLMs) construct geometry aware tokens by projecting 2D visual features into world coordinates, enabling spatial reasoning for tasks such as 3D question answering. However, this design generates thousands of tokens per scene, resulting in substantial computational and memory overhead. While token compression has been extensively studied in 2D VLMs, existing approaches rely on semantic relevance or attention-based selection that overlook the structured spatial nature of 3D tokens. Moreover, redundancy in 3D representations cannot be resolved by spatial proximity alone, as object-level token imbalance persists even after spatial aggregation. To address this, we propose 3DZip, a three-stage token compression framework that first applies coarse voxelization to remove point-level redundancy, then selects anchor tokens based on feature-space diversity via a Determinantal Point Process, and finally merges remaining tokens under spatial constraints to preserve geometric coherence. Experiments on three 3D question answering benchmarks demonstrate that 3DZip consistently outperforms existing compression methods, retaining 94.7% of the original performance with only 128 tokens, achieving a 1.92× faster inference speed.

Keywords: 3D Vision-Language Models · 3D Question Answering · Token Compression

1 Introduction

Recent advances in large language models (LLMs) [1, 2] and vision-language models (VLMs) [3–5] have extended multimodal understanding beyond 2D image perception to 3D Vision-Language Models (3D VLMs) [6–9], which are capable of reasoning over complex three-dimensional environments. In particular, 3D Question Answering [10–12], which requires understanding object attributes, spatial relations, and scene-level context from natural language queries, has emerged as a key task for embodied AI and 3D robotics applications.

✉ Corresponding author.



(a) (b)

Fig. 1: (a) Per-object token allocation under different selection strategies. Object coverage is defined as the fraction of GT object instances that contain at least one selected token, computed using GT object masks and aggregated over 10 scenes from SQA3D. Spatial sampling (XYZ-DPP) concentrates tokens on large objects (51% coverage), whereas Feature-DPP distributes tokens more evenly across objects, increasing coverage to 70%. **(b) 3DZip overview.** 3DZip compresses dense 3D tokens to only 128 while retaining 94.7% accuracy and achieving 1.92× faster inference.

Early attempts to address 3D Question Answering relied on point cloud representations [7]. However, the lack of large-scale 3D language datasets and powerful pretrained 3D encoders limited their performance compared to approaches leveraging strong 2D visual backbones. As a result, recent works adopt a projection-based paradigm that lifts 2D visual features into a shared 3D coordinate system using depth and camera pose information [6]. This approach allows models to reuse powerful 2D pretrained representations while incorporating spatial structure for 3D reasoning.

Despite these advantages, projection-based multi-view aggregation introduces two distinct forms of redundancy in 3D VLM representations. First, *point-level redundancy* arises when identical physical surfaces are repeatedly observed across multiple viewpoints. When lifted into a shared 3D coordinate system, these repeated observations generate overlapping tokens that correspond to nearly identical spatial locations. This type of redundancy can be partially mitigated through voxel-based aggregation, which collapses tokens occupying similar 3D positions. Second, *object-level redundancy* occurs when a single object is represented by tokens corresponding to multiple surface regions. For example, an object such as a sofa may produce tokens from its front, side, and back surfaces when observed from different viewpoints. Although these tokens describe the same physical instance, their spatial coordinates remain separated due to the object's geometric extent, making it difficult to collapse them into a single representation using spatial aggregation alone.

As a consequence, object-level redundancy often leads to a highly imbalanced token distribution across objects. As illustrated in Fig. 1a, even after voxel aggregation, token allocation across objects exhibits a pronounced long-tailed distribution in which a small subset of objects dominates the token budget. Spatially driven diversity strategies such as XYZ-DPP alleviate this issue only partially, as tokens remain concentrated on a limited subset of objects, covering only 51% of object instances in our analysis. In contrast, selecting tokens

based on feature-level diversity produces a markedly more balanced distribution, increasing object coverage to 70%. These observations suggest that spatial aggregation alone is insufficient to mitigate object-level redundancy in multi-view 3D VLM representations.

Motivated by these observations, we design a structured token compression framework that explicitly addresses different sources of redundancy in multi-view 3D representations. Specifically, we decompose redundancy into three complementary components: point-level density, object-level duplication, and geometric consistency. First, we apply voxel-based spatial aggregation to reduce point-level redundancy caused by multi-view projection. By merging tokens within fixed-size voxels, this step removes tokens corresponding to nearly identical 3D locations while preserving the coarse spatial structure of the scene. Second, we perform feature-level diversity selection to suppress object-level duplication. As discussed above, spatial grouping alone cannot reliably determine whether tokens belong to the same object instance. Instead, selecting tokens based on feature diversity enables the model to retain semantically distinct objects while avoiding repeated representations of the same object surfaces. Finally, the remaining tokens are merged to their nearest anchors according to spatial proximity. This spatially constrained merging step preserves geometric consistency between tokens and maintains the structural relationships required for spatial reasoning.

Based on this design, we propose **3DZip** (Fig. 1b), a geometry-aware token compression framework for multi-view 3D VLMs. By aligning token selection with the redundancy structure induced by multi-view aggregation, our method significantly improves efficiency while maintaining strong reasoning performance. Extensive experiments on three 3D question answering benchmarks demonstrate that 3DZip consistently outperforms existing token compression methods. Our approach retains 94.7% of the original performance using only 128 tokens, achieving a 1.92× faster inference speed, highlighting its effectiveness for scalable multi-view 3D VLM deployment.

In summary, our contributions are three-fold:

- We empirically reveal a severe object-level token imbalance induced by projection based multi-view aggregation in 3D VLMs, which persists even after spatial aggregation.
- We show that spatial grouping alone cannot resolve this imbalance and demonstrate that feature-level diversity is crucial for balancing object coverage in multi-view 3D scenes.
- We propose **3DZip**, a structured three-stage token compression framework combining voxel aggregation, feature-diversity–based anchor selection, and spatially constrained merging.

2 Related Works

2.1 3D Vision-Language Models

Following the success of 2D VLMs [3, 5], research has shifted toward extending LLMs to 3D scene understanding. Existing approaches vary by their represen-

tation strategies. Prior to LLM-based methods, transformer-based 3D vision-language models such as 3D-VisTA [13] align 3D scenes and text through pre-training and are competitive on 3D question answering. Building on LLMs, early works such as LL3DA [14], LEO [9], and Chat-Scene [8] directly integrate 3D point clouds into LLMs [7, 15–19], and Robin3D [20] further improves such 3D LLMs through robust instruction tuning. While effective for geometric modeling, these methods rely on specialized 3D encoders and large-scale 3D vision-language datasets and benchmarks [10–12,21], limiting their scalability and generalization.

To address this, recent studies have adopted projection-based paradigms that leverage pretrained 2D models [22,23]. Early efforts like 3D-LLM [22] and Scene-LLM [23] aggregate 2D features into 3D space but depend on complex pipelines and external segmentation, which are computationally intensive and cause information loss.

In contrast, LLaVA-3D [6] offers a more unified solution by integrating 3D positional embeddings into 2D patches. This design enables explicit 3D modeling without dedicated encoders, preserving the original LMM's reasoning capabilities. However, LLaVA-3D generates thousands of tokens per scene, leading to substantial computational and memory overhead. Thus, efficient token compression is essential to make these powerful 3D VLMs practical for large-scale deployment.

2.2 Token Compression

Token compression has been widely studied in VLMs to reduce attention cost and memory overhead caused by a large number of visual tokens. Most existing methods are developed for 2D image–text settings, where visual tokens are assumed to be highly redundant [24–33]. Early approaches leveraged cross-attention distributions within the LLM, retaining visual tokens that receive high attention from text tokens [24, 25, 34]. However, such methods depend on the decoding process, making them inefficient and difficult to combine with acceleration techniques such as FlashAttention [35]. Moreover, attention tends to be biased toward later visual tokens, leading to unstable performance and limited representativeness. To address these issues, recent methods rely on CLS-token attention from the vision encoder [26,27,36,37]. While CLS-based selection avoids additional decoding cost and captures globally salient tokens, it often overlooks locally important objects and fine-grained geometric structures. Moreover, these methods are primarily designed for 2D image–text settings and do not explicitly account for 3D spatial structure. To the best of our knowledge, their effectiveness in projection-based 3D VLMs has not been systematically studied.

In projection-based 3D VLMs, multi-view features are lifted into world coordinates to form 3D tokens, substantially increasing token count. While DTC [38] incorporates depth information for token pruning, its focus remains on multi-view 2D VLM settings that do not explicitly learn 3D spatial representations. In contrast, our method jointly considers feature-space diversity and geometric constraints, preserving both semantic representativeness and spatial coherence in projection-based 3D VLMs.

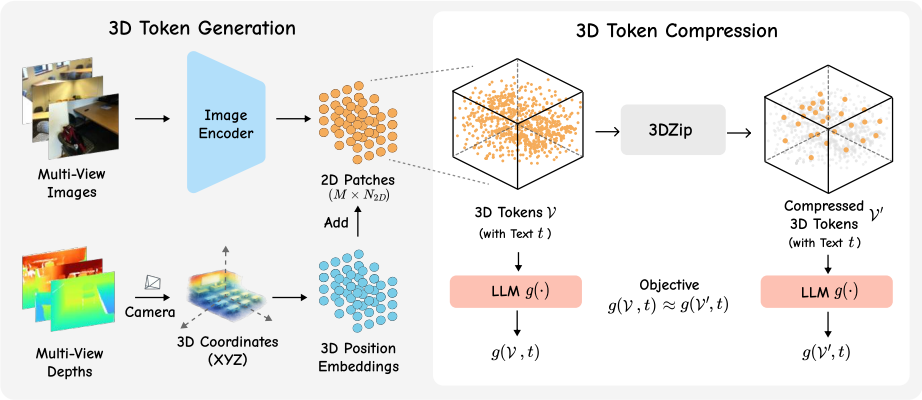


Fig. 2: Overview of geometry-aware 3D token construction and the compression objective. Multi-view RGB-D inputs are projected into world coordinates to form geometry-aware 3D tokens $\mathcal{V}$ via 3D positional embedding. Due to the large token cardinality $N = M \times N_{2D}$, an efficient token compression strategy is crucial.

3 Preliminary: Projection-based 3D VLMs

Projection-based 3D VLMs construct geometry-aware tokens by lifting multi-view 2D visual features into a shared 3D coordinate system. As illustrated in Fig. 2, multi-view RGB-D observations are first processed by a pretrained image encoder to extract 2D visual tokens. Using the corresponding depth maps and camera poses, these tokens are then back-projected into the world coordinate system and augmented with 3D positional embeddings to form geometry-aware 3D tokens. Let M denote the number of views, and each view produces N_{2D} visual tokens extracted by the image encoder. For view m, the 2D token features are

$$\{\mathbf{f}_{m,i}\}_{i=1}^{N_{2D}}, \quad \mathbf{f}_{m,i} \in \mathbb{R}^d. \tag{1}$$

Given the depth map D_m and camera pose $T_m \in SE(3)$, each token is back-projected into the world coordinate system:

$$\mathbf{p}_{m,i} = \mathit{\Pi}^{-1}(\mathbf{u}_{m,i}, D_m, T_m), \tag{2}$$

where $\mathbf{u}_{m,i}$ denotes the 2D pixel location corresponding to token i. To incorporate geometric information, the 3D coordinate $\mathbf{p}_{m,i}$ is encoded using a learnable positional embedding $\phi(\mathbf{p}_{m,i}) \in \mathbb{R}^d$ and combined with the visual feature:

$$\tilde{\mathbf{f}}_{m,i} = \mathbf{f}_{m,i} + \phi(\mathbf{p}_{m,i}). \tag{3}$$

Each geometry-aware 3D token is therefore represented as

$$v_{m,i} = (\tilde{\mathbf{f}}_{m,i}, \mathbf{p}_{m,i}). \tag{4}$$

Aggregating tokens across all views yields a token set

$$N = M \times N_{2D}. \tag{5}$$

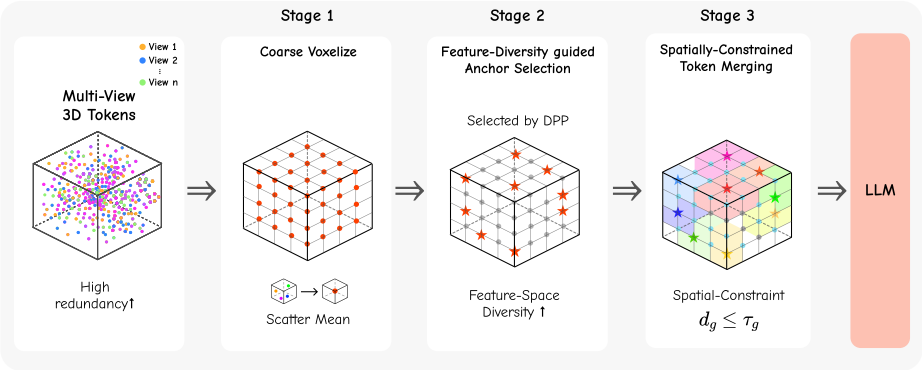


Fig. 3: Overview of the proposed three-stage token compression pipeline. Given geometry-aware 3D tokens $\mathcal{V}$, we first apply coarse voxelization to obtain a spatially reduced set $\mathcal{V}_{\text{vox}}$. We then perform diversity-aware anchor selection via DPP on $\mathcal{V}_{\text{vox}}$ to identify semantically representative anchors $\mathcal{A}$. Finally, spatially-constrained token merging aggregates nearby non-anchor tokens into anchors to produce the compressed set $\mathcal{V}'$.

For notational simplicity, we flatten the token indices and denote the resulting 3D tokens as

$$v_i = (\tilde{\mathbf{f}}_i, \mathbf{p}_i), \quad i = 1, \ldots, N. \tag{6}$$

Let the original token set be

$$\mathcal{V} = \{v_i\}_{i=1}^{N}. \tag{7}$$

Our goal is to construct a compressed token set

$$|\mathcal{V}'| = N', \quad N' \ll N, \tag{8}$$

that preserves the information necessary for downstream reasoning while reducing computational cost. Formally, given a language model $g(\cdot)$ operating on the token set with a text query t, we aim to ensure that

$$g(\mathcal{V}, t) \approx g(\mathcal{V}', t). \tag{9}$$

meaning that the compressed representation maintains the semantic and spatial cues required for reasoning tasks while significantly reducing token cardinality.

4 Method

Our goal is to compress the large set of geometry-aware 3D tokens $\mathcal{V}$ while preserving the semantic and spatial information required for downstream reasoning. As illustrated in Fig. 3, we design a three-stage token compression pipeline that progressively removes redundancy while maintaining geometric consistency. Starting from the dense token set $\mathcal{V}$ generated by multi-view projection, we first apply **Stage 1: coarse voxelization** to reduce point-level redundancy caused

Algorithm 1 Overview of the proposed 3DZip

Require: Original token set $\mathcal{V} = \{v_i\}_{i=1}^{N}$, voxel size δ, anchor number K
Ensure: Compressed token set $\mathcal{V}'$
Stage 1: Coarse Voxelization
Partition tokens into voxel groups $\mathcal{G} = \{G_k\}$
Aggregate tokens within each voxel using mean pooling
Obtain voxel tokens $\mathcal{V}_{\text{vox}} = \{v_k\}$
Stage 2: Feature-Diversity Guided Anchor Selection
Normalize voxel features $\hat{\mathbf{f}}_k$
Construct similarity kernel $L_{kl} = \hat{\mathbf{f}}_k^\top \hat{\mathbf{f}}_l$
Select anchor set $\mathcal{A}$ via greedy DPP maximization with $|\mathcal{A}| = K$
Stage 3: Spatially-Constrained Token Merging
for each $j \in \mathcal{V}_{\text{vox}} \setminus \mathcal{A}$ **do**
    Find nearest anchor $a^*(j)$ in feature space
    if $d_g(j, a^*(j)) \leq \tau_g$ **then**
        Assign token j to anchor $a^*(j)$
    else
        Discard token j
    end if
end for
Aggregate assigned tokens into anchors
return Compressed token set $\mathcal{V}'$

by overlapping observations. Tokens within each voxel are aggregated to obtain a spatially reduced token set $\mathcal{V}_{\text{vox}}$. Next, we perform **Stage 2: feature-diversity guided anchor selection**. From the voxelized token set $\mathcal{V}_{\text{vox}}$, we select a subset of representative anchors $\mathcal{A}$ using a Determinantal Point Process (DPP) [39], which encourages diversity in feature space and preserves semantically distinct tokens. Finally, we apply **Stage 3: spatially-constrained token merging**. Non-anchor tokens are assigned to nearby anchors under a spatial distance constraint, allowing anchors to aggregate complementary contextual information while maintaining geometric consistency. This produces the compressed token set $\mathcal{V}'$, which is subsequently provided to the language model for downstream reasoning. The overall procedure of the proposed compression pipeline is summarized in Algorithm 1.

4.1 Stage 1: Coarse Voxelization

Multi-view projection often produces densely overlapping tokens corresponding to nearly identical 3D locations. To mitigate this *point-level redundancy*, we partition the 3D space into axis-aligned voxels with size δ.

Let $\mathcal{G} = \{G_k\}_{k=1}^{N_v}$ denote the set of occupied voxels, where each cell contains tokens whose positions fall inside the k-th voxel. Tokens within each voxel are aggregated via mean pooling

$$\mathbf{f}_k = \frac{1}{|G_k|} \sum_{v_i \in G_k} \mathbf{f}_i, \qquad \mathbf{p}_k = \frac{1}{|G_k|} \sum_{v_i \in G_k} \mathbf{p}_i. \tag{10}$$

The resulting voxel tokens

$$v_k = (\mathbf{f}_k, \mathbf{p}_k) \tag{11}$$

form the reduced set

$$\mathcal{V}_{\text{vox}} = \{v_k\}_{k=1}^{N_v}, \quad N_v \ll N. \tag{12}$$

4.2 Stage 2: Feature-Diversity Guided Anchor Selection

Although voxelization reduces point-level redundancy, many tokens still correspond to repeated observations of the same object surfaces, resulting in *object-level redundancy*. To retain semantically distinct representations, we select a subset of representative anchors.

Given $\mathcal{V}_{\text{vox}}$, we aim to select an anchor set $\mathcal{A} \subset \mathcal{V}_{\text{vox}}$ with $|\mathcal{A}| = K$. We first normalize voxel features

$$\hat{\mathbf{f}}_k = \frac{\mathbf{f}_k}{\|\mathbf{f}_k\|_2}.$$

A cosine similarity kernel is then constructed

$$L_{kl} = \hat{\mathbf{f}}_k^\top \hat{\mathbf{f}}_l. \tag{13}$$

Using a DPP, anchors are selected by maximizing $\det(L_{\mathcal{A}})$. We adopt the standard greedy approximation based on Cholesky decomposition [39].

4.3 Stage 3: Spatially-Constrained Token Merging

While anchors capture diverse semantic information, the remaining tokens may still contain complementary context. To incorporate such information while preserving spatial consistency, we merge nearby tokens into their assigned anchors.

For each non-anchor token index $j \in \mathcal{V}_{\text{vox}} \setminus \mathcal{A}$, we assign it to the most similar anchor in feature space,

$$a^*(j) = \arg\min_{a \in \mathcal{A}} \left(1 - \hat{\mathbf{f}}_j^\top \hat{\mathbf{f}}_a\right), \tag{14}$$

and allow merging only when the spatial distance between tokens is sufficiently small. Specifically, let

$$\mathbf{c}_k = \left\lfloor \frac{\mathbf{p}_k}{\delta} \right\rfloor \tag{15}$$

denote the voxel grid index of token v_k, and define the grid-space distance

$$d_g(j, a) = \|\mathbf{c}_j - \mathbf{c}_a\|_2. \tag{16}$$

We define the set of tokens assigned to anchor a as

$$\mathcal{S}_a = \{\, j \mid a^*(j) = a,\ d_g(j,a) \le \tau_g \,\}. \tag{17}$$

Tokens for which no anchor satisfies the spatial constraint (i.e., $d_g(j, a^*(j)) > \tau_g$) are discarded, as they likely correspond to spatially isolated observations that do not contribute complementary context to any anchor. The anchor feature is then updated as the mean of itself and all assigned tokens:

$$\mathbf{f}'_a = \frac{\mathbf{f}_a + \sum_{j \in \mathcal{S}_a} \mathbf{f}_j}{1 + |\mathcal{S}_a|}, \qquad \mathbf{p}'_a = \mathbf{p}_a. \tag{18}$$

The final compressed token set is

$$\mathcal{V}' = \{v'_k\}_{k=1}^{N'}, \quad v'_k = (\mathbf{f}'_k, \mathbf{p}'_k), \quad N' = |\mathcal{A}| = K \ll N. \tag{19}$$

The resulting token set $\mathcal{V}'$ is then provided to the language model for downstream reasoning.

5 Experiments

5.1 Experimental Settings

Baselines and Models We compare our method with representative token compression approaches, including voxelization-only aggregation and the voxel-based DTC. In addition, we evaluate methods originally developed for 2D VLM settings, including FastV [24], SparseVLM [25], VisionZip [27], and VisPruner [26], to examine their effectiveness when extended to 3D spatial representations. As the backbone model, we adopt LLaVA-3D [6] as our projection-based 3D VLM. To ensure a fair comparison, all competing methods are implemented on top of the same backbone architecture.

Implementation Details We set the voxel size $\delta = 0.2\,\mathrm{m}$ for coarse voxelization and the grid-space threshold $\tau_g = 5$ for spatially-constrained merging. All experiments are conducted with temperature set to 0 to ensure reproducibility.

5.2 Dataset

SQA3D This benchmark [10] extends 3D Question Answering to a situated and embodied setting, introducing 6.8K agent-centric situations over 650 ScanNet [40] scenes, with 33K reasoning questions incorporating situation understanding and situated reasoning. We report exact match (EM) accuracy on the test set.

OpenEQA Proposed by [11], this is the first open-vocabulary benchmark for embodied reasoning in 3D environments, containing over 1.6K human-authored questions across seven reasoning categories from ScanNet and HM3D [41] environments. It employs an automated LLM-based evaluation pipeline, and we report the GPT-4o LLM-Match score following the standard protocol [42].

Table 1: Comparison of token compression methods. The 3D-Aware column indicates whether spatial geometry is explicitly considered during token selection.

Method	3D-Aware	ScanQA	SQA3D	OpenEQA	Rel.
All 1410 Tokens					
LLaVA-3D (ICCV'25)	✓	26.5	55.7	60.3	100.0%
Retain 128 Tokens ↓(9.1%)					
FastV (ECCV'24)	✗	21.9	50.9	56.0	88.9%
SparseVLM (ICML'25)	✗	21.8	50.3	53.9	87.3%
VisionZip (CVPR'25)	✗	22.2	51.5	56.2	89.8%
VisPruner (ICCV'25)	✗	22.3	52.0	55.8	90.0%
Voxelization	✓	23.6	51.5	54.6	90.7%
DTC (CVPR'25)	✓	22.1	51.3	54.8	88.8%
3DZip (Ours)	✓	**24.2**	**53.2**	**58.6**	**94.7%**
Retain 64 Tokens ↓(4.5%)					
FastV (ECCV'24)	✗	21.1	49.5	53.8	85.9%
SparseVLM (ICML'25)	✗	20.9	49.8	53.6	85.7%
VisionZip (CVPR'25)	✗	20.0	49.1	53.4	84.1%
VisPruner (ICCV'25)	✗	21.9	50.1	53.9	87.3%
Voxelization	✓	21.9	49.8	54.6	87.5%
DTC (CVPR'25)	✓	21.3	50.2	54.3	86.8%
3DZip (Ours)	✓	**23.3**	**52.8**	**56.7**	**92.3%**
Retain 32 Tokens ↓(2.3%)					
FastV (ECCV'24)	✗	19.9	47.7	52.5	82.6%
SparseVLM (ICML'25)	✗	20.4	48.7	52.2	83.7%
VisionZip (CVPR'25)	✗	19.6	47.0	52.0	81.5%
VisPruner (ICCV'25)	✗	20.9	49.1	53.5	85.2%
Voxelization	✓	20.7	47.9	53.4	84.2%
DTC (CVPR'25)	✓	20.6	48.8	52.6	84.2%
3DZip (Ours)	✓	**21.9**	**51.1**	**55.7**	**88.9%**

ScanQA This benchmark [12] is built on RGB-D reconstructions from ScanNet, containing 41K question–answer pairs over 800 indoor scenes. Questions require reasoning over object attributes, spatial relations, counting, and scene layout. We report EM and standard captioning metrics on the validation split.

5.3 Results

Performance Comparison Table 1 compares our method with representative token compression approaches under different token budgets on ScanQA, SQA3D, and OpenEQA. Methods originally designed for 2D VLMs, such as FastV, SparseVLM, VisionZip, and VisPruner, consistently underperform in 3D Question Answering settings. In particular, VisionZip and VisPruner select tokens based on CLS-attention from the vision encoder, emphasizing global semantic relevance. While effective for 2D tasks, such global attention-based selection does not explicitly preserve local geometric relationships required for spatial reasoning in 3D Question Answering. Similarly, LLM-attention-based pruning selects tokens with high relevance to the input text, which can bias the selection toward a small set of semantically salient regions while discarding surrounding tokens that encode spatial context, potentially disrupting geometric coherence in 3D scenes.

However, existing 3D-aware methods still show limited performance under constrained token budgets. Purely spatial aggregation, such as voxelization, may overlook semantic representativeness. Although DTC incorporates visual semantics, it still shows performance degradation as the token budget becomes more constrained. Across all token budgets (128, 64, and 32 tokens), our method consistently achieves the best performance. Notably, 3DZip maintains the highest

Table 2: Category-wise OpenEQA results. 3DZip achieves the highest average at all token budgets, with the largest gains in attribute recognition and object recognition.

Method	OpenEQA							Avg.
	attribute recognition	functional reasoning	object localization	object recognition	object state recognition	spatial understanding	world knowledge	
All 1410 Tokens								
LLaVA-3D (ICCV'25)	64.0	62.7	51.5	55.9	73.7	54.2	59.1	60.3
Retain 128 Tokens ↓(9.1%)								
FastV(ECCV'24)	57.8	58.9	46.8	47.7	**75.6**	45.3	59.4	56.0
SparseVLM(ICML'25)	54.0	54.1	48.3	45.9	74.1	44.5	55.5	53.9
VisionZip(CVPR'25)	58.4	57.8	48.4	51.0	69.4	47.2	**60.9**	56.2
Voxelize	56.3	58.5	47.2	47.2	68.6	49.1	54.9	54.6
DTC(CVPR'25)	56.4	57.7	48.5	46.2	71.5	47.2	55.3	54.8
3DZip (Ours)	**64.2**	**60.0**	**50.0**	**53.2**	72.5	**49.5**	60.1	**58.6**
Retain 64 Tokens ↓(4.5%)								
FastV(ECCV'24)	51.2	58.2	47.1	43.2	**76.3**	43.2	56.7	53.8
SparseVLM(ICML'25)	50.6	57.8	46.5	45.3	74.3	45.2	55.5	53.6
VisionZip(CVPR'25)	51.7	57.4	48.6	44.7	69.4	45.6	56.3	53.4
Voxelize	50.8	**61.4**	46.2	47.9	71.5	**48.5**	55.7	54.6
DTC(CVPR'25)	55.5	58.2	45.5	43.2	74.1	47.8	55.0	54.3
3DZip (Ours)	**60.9**	59.7	**48.8**	**51.2**	69.6	48.1	**58.0**	**56.7**
Retain 32 Tokens ↓(2.3%)								
FastV(ECCV'24)	50.7	58.6	43.9	41.2	**73.6**	42.6	56.6	52.5
SparseVLM(ICML'25)	50.8	57.4	45.6	40.6	73.3	42.9	54.2	52.2
VisionZip(CVPR'25)	51.0	57.0	45.2	42.6	68.1	44.4	55.6	52.0
Voxelize	50.4	**60.3**	46.9	43.8	68.8	46.1	**57.7**	53.4
DTC(CVPR'25)	51.3	56.8	45.1	42.9	72.2	44.2	55.6	52.3
3DZip (Ours)	**56.3**	59.7	**47.4**	**50.5**	70.7	**47.7**	57.4	**55.7**

relative performance (Rel.), preserving 94.7% and 92.3% of uncompressed performance at 128 and 64 tokens, respectively, consistently outperforming the best baselines. For example, at 64 tokens, our method improves SQA3D from 50.2 (DTC) and 49.1 (VisionZip) to 52.8, and even under aggressive compression (32 tokens), it maintains strong performance (51.1), outperforming both 2D-based and existing 3D-aware baselines. These results indicate that jointly preserving feature-space diversity and geometric coherence is crucial for effective 3D token compression. Supp. Sec. A.1 further reports results on additional backbones (Video-3D-LLM [43] and SR-3D [44]), and Supp. Sec. A.3 reports additional ScanQA metrics, confirming the consistent gains of 3DZip.

Category-wise Analysis on OpenEQA Table 2 presents a category-wise breakdown on OpenEQA across seven capabilities. 3DZip achieves the highest average score at all token budgets. In *attribute recognition*, 3DZip scores 64.2 at 128 tokens, closely matching the uncompressed baseline (64.0) and outperforming the second-best method by +5.8 points. In *object recognition*, it retains 53.2 at 128 tokens and maintains 50.5 even at 32 tokens, yielding the largest margin (+6.7) at this budget.

3DZip also consistently improves performance in *object localization* and *spatial understanding*, the two most spatially demanding categories. Methods such as FastV and SparseVLM rely on LLM attention scores, which tend to emphasize globally salient regions while overlooking spatially distributed tokens that encode positional cues. VisionZip mitigates this through CLS-attention-based selection with token merging, and DTC applies spatial aggregation, but both rely on averaging operations that may dilute fine-grained geometric details. In con-

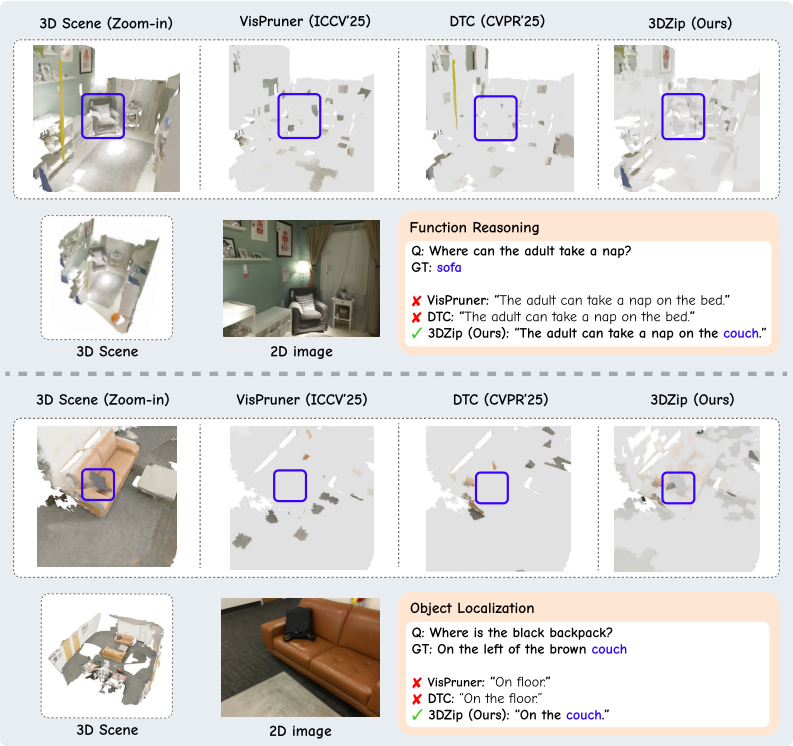


Fig. 4: Qualitative comparison of retained 3D tokens on OpenEQA. VisPruner tends to allocate many tokens to floor regions, while DTC may miss some key objects in the scene. In contrast, 3DZip retains tokens on multiple semantically important objects and further enriches them through spatially-constrained merging.

trast, 3DZip's DPP-based selection explicitly promotes feature diversity, helping preserve visually distinct and spatially informative tokens.

Furthermore, tasks reliant on prior knowledge and commonsense, such as *functional reasoning* and *world knowledge* prove robust to aggressive compression, with 3DZip maintaining stable performance across all budgets. An exception is *object state recognition*, where FastV slightly exceeds the uncompressed baseline (75.6 vs. 73.7). This category yields consistently high scores across all methods, indicating it is inherently easier—primarily involving binary-style judgments (e.g., open vs. closed) that require minimal spatial reasoning or geometric understanding of inter-object relationships. Moreover, since FastV and SparseVLM select tokens based on LLM attention guided by text-query relevance, they naturally excel at retaining tokens specific to such narrowly focused, object-centric state queries.

Qualitative Results Figure 4 visualizes the preserved 3D tokens and corresponding question answering results. 2D-attention-based methods like VisPruner tend to allocate many tokens to background and floor regions, missing critical

Table 3: Component-wise ablation results on SQA3D. (a) feature vs. spatial distance in anchor selection, (b) the effect of varying voxel sizes (δ) in coarse voxelization, and (c) the role of the merging step and spatial constraints.

Method	EM
All 1410 Tokens	
LLaVA-3D	55.7
Retain 64 Tokens	
Voxelization	49.8
DTC (CVPR'25)	50.2
Spatial distance	50.1
Feature distance	**52.8**
Retain 32 Tokens	
Voxelization	47.9
DTC (CVPR'25)	48.8
Spatial distance	49.0
Feature distance	**51.1**

(a)

Method	EM
All 1410 Tokens	
LLaVA-3D	55.7
Retain 64 Tokens	
w/o Coarse voxelize	51.8
δ=0.1m	52.6
δ=0.2m	**52.8**
δ=0.3m	52.1
Retain 32 Tokens	
w/o Coarse voxelize	50.2
δ=0.1m	50.9
δ=0.2m	**51.1**
δ=0.3m	50.6

(b)

Method	EM
All 1410 Tokens	
LLaVA-3D	55.7
Retain 64 Tokens	
DTC (CVPR'25)	50.2
w/o Merge	52.5
w/o Spatial-Const.	52.3
w/ All	**52.8**
Retain 32 Tokens	
DTC (CVPR'25)	48.8
w/o Merge	50.5
w/o Spatial-Const.	50.8
w/ All	**51.1**

(c)

local objects. Methods like DTC, which perform token matching within individual voxels, may struggle to capture object-level diversity across the entire scene, potentially leaving room for a relative concentration of tokens on large uniform structures. In contrast, 3DZip captures object-level diversity across the entire scene through feature-space DPP-based anchor selection, and further enriches each anchor with complementary context through spatially-constrained merging.

5.4 Ablations

Feature Distance vs. Spatial Distance Table 3a compares feature-space distance and spatial (XYZ) distance as the kernel used in DPP-based anchor selection. Across all token budgets, feature-space distance consistently outperforms spatial distance by a clear margin (e.g., 52.8 vs. 50.1 EM at 64 tokens). When spatial distance is used, DPP favors geometrically dispersed anchors, promoting broad spatial coverage. However, spatially distant regions may still contain visually similar or redundant content—for example, multiple anchors may be redundantly allocated to different parts of a large wall or floor. Such spatially biased selection leads to a long-tail distribution in token allocation, where tokens are disproportionately concentrated on a few large objects, hindering overall object coverage across the scene.

In contrast, feature-space distance directly captures semantic dissimilarity, encouraging the selection of tokens that represent distinct objects rather than repeated surface observations. This mitigates the long-tail bias in token allocation and improves object coverage across the scene. These results support our hypothesis that feature dispersion, rather than geometric dispersion, is the more effective criterion for 3D token compression. Beyond this comparison, Supp. Sec. B.3 and B.4 show that alternative diversity-based anchor selection methods are also effective, and that cosine similarity performs best among feature-space metrics.

Table 4: Efficiency and accuracy comparison on a single RTX 4090 using SQA3D.

Method	Retain Tokens	FLOPs (T)	Latency (ms/sample)	Cache Size (MB)	EM
LLaVA-3D	1410	9.18	342	722	55.7
FastV (ECCV'24)	128	1.41	191	139	50.9
DTC (CVPR'25)	128	0.90	196	101	51.3
3DZip (Ours)	128	**0.90**	**178**	**101**	**53.2**

Ablation on Coarse Voxelize Table 3b investigates the impact of the initial coarse voxelization step and the choice of voxel size δ. Removing the voxelization stage entirely (*w/o Coarse Voxelize*) leads to a noticeable performance drop (e.g., from 52.8 to 51.8 EM under the 64-token budget), confirming that the severe multi-view redundancy must be resolved before feature-diversity-based anchor selection. Without this stage, the DPP anchor selection operates purely in feature space, which can result in clustered anchors that fail to preserve the overall 3D geometry of the scene. However, overly coarse voxelization is also harmful: as shown with $\delta = 0.3$m, a larger voxel size prematurely merges distinct local objects, destroying fine-grained spatial cues essential for 3D reasoning. These results highlight the importance of balancing multi-view redundancy reduction and local geometric preservation when compressing 3D tokens. To complement this ablation, Supp. Sec. B.1 quantitatively shows that coarse voxelization reduces point-level redundancy caused by duplicate multi-view observations.

Effect of Spatially-Constrained Token Merging Table 3c ablates the merging strategy and spatial constraints under different token budgets. Removing the merging step (*w/o Merge*) leads to a performance drop (e.g., from 52.8 to 52.5 EM at 64 tokens), confirming that aggregating non-anchor tokens into nearby anchors enriches anchor representations with complementary context. Omitting spatial constraints (*w/o Spatial-Const.*) results in a similar decrease (e.g., from 52.8 to 52.3 at 64 tokens), indicating that geometrically constrained merging is essential for maintaining structural consistency. Combining both components consistently yields the best performance across all compression ratios. Additional results on the spatial threshold τ_g are provided in Supp. Sec. B.5.

5.5 Efficiency Analysis

We evaluate computational efficiency on a single NVIDIA RTX 4090 GPU using the SQA3D benchmark under identical settings. As shown in Table 4, our method reduces inference latency from 342ms/sample (LLaVA-3D) to 178ms/sample, achieving a 48.0% reduction. Under the same 128-token budget, our method outperforms DTC in practical speed (196ms vs. 178ms), yielding a 9.2% speedup. This efficiency gain stems from aggressive yet effective token reduction; compared to the LLaVA-3D baseline (1410 tokens), our method retains only 128 tokens (−90.9%), which reduces FLOPs from 9.18T to 0.90T (−90.2%) and KV cache size from 722MB to 101MB (−86.0%). Despite these substantial reductions, the EM score drops by only 2.5 points (55.7 to 53.2), demonstrating a highly favorable efficiency–accuracy trade-off. Notably, while DTC achieves similar theoretical FLOPs for a given token count, our method is more efficient in

practice due to our one-pass geometry-aware selection, which avoids the overhead of iterative matching and refinement. Furthermore, by removing redundant tokens before the LLM backbone, we achieve significant savings in both attention computation and KV cache footprint. Our implementation is also compatible with FlashAttention [35] for further acceleration.

6 Limitations and Future Work

While 3DZip achieves a strong efficiency–accuracy trade-off, it still has several limitations. The coarse voxelization in Stage 1 may attenuate fine-grained cues for small objects, since their tokens can be merged with nearby objects or background regions within the same voxel, as analyzed in Supp. Sec. B.8. In addition, the hyperparameters (δ, τ_g) are fixed across scenes rather than adapted to scene complexity. Since scene size can affect object density and inter-object distances, fixed hyperparameters may be suboptimal across different environments. Although their sensitivity to scene scale is examined in Supp. Sec. B.9, developing adaptive hyperparameter selection that accounts for scene complexity and small-object density remains a promising future direction.

7 Conclusion

In this work, we present 3DZip, a geometry-aware token compression framework for projection-based 3D VLMs. We identify two distinct forms of redundancy inherent to multi-view 3D representations—point-level redundancy caused by overlapping observations and object-level redundancy arising from repeated surface tokens of the same object—and demonstrate that spatial aggregation alone is insufficient to address the latter. To this end, 3DZip combines three complementary stages: coarse voxelization to reduce point-level redundancy, feature-diversity-guided anchor selection via DPP to capture semantically distinct objects across the scene, and spatially-constrained token merging to enrich anchors with complementary context while preserving geometric consistency. Extensive experiments on ScanQA, SQA3D, and OpenEQA demonstrate that 3DZip consistently outperforms both 2D-based and existing 3D-aware token compression methods across all token budgets, retaining 94.7% of the original accuracy with only 128 tokens and achieving a 1.92× speedup. We hope that our analysis of object-level token imbalance and the proposed feature-diversity-driven compression strategy provide useful insights for future work on efficient 3D VLMs.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korean government(MSIT) (No. RS-2024-00456152) and the "Advanced GPU Utilization Support Program" funded by the Government of the Republic of Korea(Ministry of Science and ICT), and the authors gratefully acknowledge the Cluster Server for Computational Science at Pusan National University for providing computational resources.

References

1. Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.
2. Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.
3. Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.
4. 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.
5. Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306, 2024.
6. Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d capabilities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4295–4305, 2025.
7. Jiajun Deng, Tianyu He, Li Jiang, Tianyu Wang, Feras Dayoub, and Ian Reid. 3d-llava: Towards generalist 3d lmms with omni superpoint transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3772–3782, 2025.
8. 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.
9. Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. In *International Conference on Machine Learning*, pages 20413–20451. PMLR, 2024.
10. Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. In *The Eleventh International Conference on Learning Representations*, 2023.
11. Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, et al. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16488–16498, 2024.
12. 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*, pages 19129–19139, 2022.
13. 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*, pages 2911–2921, 2023.

14. 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26428–26438, 2024.
15. Yunze Man, Liang-Yan Gui, and Yu-Xiong Wang. Situational awareness matters in 3d vision language reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13678–13688, 2024.
16. Zhangyang Qi, Ye Fang, Zeyi Sun, Xiaoyang Wu, Tong Wu, Jiaqi Wang, Dahua Lin, and Hengshuang Zhao. Gpt4point: A unified framework for point-language understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26417–26427, 2024.
17. Xiangxi Shi, Zhonghua Wu, and Stefan Lee. Aware visual grounding in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14056–14065, 2024.
18. Zehan Wang, Haifeng Huang, Yang Zhao, Ziang Zhang, and Zhou Zhao. Chat-3d: Data-efficiently tuning large language model for universal dialogue of 3d scenes. *arXiv preprint arXiv:2308.08769*, 2023.
19. 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*, pages 131–147. Springer, 2024.
20. Weitai Kang, Haifeng Huang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Robin3d: Improving 3d large language model via robust instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3905–3915, 2025.
21. Ruiyuan Lyu, Jingli Lin, Tai Wang, Shuai Yang, Xiaohan Mao, Yilun Chen, Runsen Xu, Haifeng Huang, Chenming Zhu, Dahua Lin, et al. Mmscan: A multi-modal 3d scene dataset with hierarchical grounded language annotations. *Advances in Neural Information Processing Systems*, 37:50898–50924, 2024.
22. Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 20482–20494. Curran Associates, Inc., 2023.
23. Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-llm: Extending language model for 3d visual reasoning. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, pages 2195–2206, February 2025.
24. 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*, pages 19–35. Springer, 2024.
25. Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. In *International Conference on Machine Learning*, 2025.
26. 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20857–20867, 2025.

27. Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19792–19802, 2025.
28. Weihao Ye, Qiong Wu, Wenhao Lin, and Yiyi Zhou. Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 22128–22136, 2025.
29. Zhihang Lin, Mingbao Lin, Luxi Lin, and Rongrong Ji. Boosting multimodal large language models with visual tokens withdrawal for rapid inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 5334–5342, 2025.
30. Changwoo Baek, Jouwon Song, Sohyeon Kim, and Kyeongbo Kong. Agilepruner: An empirical study of attention and diversity for adaptive visual token pruning in large vision-language models. In *The Fourteenth International Conference on Learning Representations*, 2026.
31. Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9392–9401, 2025.
32. Jouwon Song, Sohyeon Kim, and Kyeongbo Kong. Uncertainty-guided graph formulation via mwis for token pruning in lvlms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9510–9519, 2026.
33. Zhenglun Kong, Yize Li, Fanhu Zeng, Lei Xin, Shvat Messica, Xue Lin, Pu Zhao, Manolis Kellis, Hao Tang, and Marinka Zitnik. Token reduction should go beyond efficiency in generative models–from vision, language to multimodality. *arXiv preprint arXiv:2505.18227*, 2025.
34. Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*, 2024.
35. Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359, 2022.
36. Qizhe Zhang, Aosong Cheng, Ming Lu, Zhiyong Zhuo, Minqi Wang, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. [cls] attention is all you need for training-free visual token pruning: Make vlm inference faster. *arXiv e-prints*, pages arXiv–2412, 2024.
37. Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In *ICCV*, 2025.
38. 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19424–19434, June 2025.
39. Laming Chen, Guoxin Zhang, and Eric Zhou. Fast greedy map inference for determinantal point process to improve recommendation diversity. *Advances in neural information processing systems*, 31, 2018.
40. Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

41. Santhosh Kumar Ramakrishnan, Aaron Gokaslan, Erik Wijmans, Oleksandr Maksymets, Alexander Clegg, John M Turner, Eric Undersander, Wojciech Galuba, Andrew Westbury, Angel X Chang, Manolis Savva, Yili Zhao, and Dhruv Batra. Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.
42. Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.
43. Duo Zheng, Shijia Huang, and Liwei Wang. Video-3d llm: Learning position-aware video representation for 3d scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8995–9006, 2025.
44. An-Chieh Cheng, Yang Fu, Yukang Chen, Zhijian Liu, Xiaolong Li, Subhashree Radhakrishnan, Song Han, Yao Lu, Jan Kautz, Pavlo Molchanov, Hongxu Yin, Xiaolong Wang, and Sifei Liu. 3d aware region prompted vision language model. In *The Fourteenth International Conference on Learning Representations*, 2026.