

# Rethinking Visual Token Reduction in LVLMs Under Cross-Modal Misalignment

Rui Xu<sup>1</sup>, Yunke Wang<sup>2</sup>, Yong Luo<sup>1\*</sup>, Bo Du<sup>1\*</sup>

<sup>1</sup>School of Computer Science, Wuhan University

<sup>2</sup>School of Computer Science, The University of Sydney

xurui7943@gmail.com, yunke.wang@sydney.edu.au, {luoyong, dubo}@whu.edu.cn

## Abstract

Large Vision-Language Models (LVLMs) encode visual inputs as dense sequences of patch-level tokens to capture fine-grained semantics. These visual tokens often outnumber their textual counterparts by a large margin, leading to substantial computational overhead and limiting the scalability of LVLMs in practice. Previous efforts have explored visual token reduction either prior to or within the large language models (LLMs). However, most in-LLM reduction approaches rely on text-conditioned interactions, implicitly assuming that textual tokens can reliably capture the importance of visual tokens. In this work, we revisit this assumption and reveal causal, semantic, and spatial forms of cross-modal misalignment. These misalignments undermine the effectiveness of text-guided visual token reduction. To address this, we introduce VisionDrop, a training-free, visual-only pruning framework that selects informative visual tokens based on intra-modal (visual-to-visual) attention, without relying on textual signals. To further suppress redundancy throughout the model hierarchy, we treat the visual encoder and the LLM as a unified system and design a progressive pruning pipeline. Our method performs dominant token selection and lightweight contextual merging at multiple stages, enabling fine-grained visual information to be retained even under aggressive token budgets. Extensive experiments across diverse benchmarks show that VisionDrop achieves consistent improvements over existing approaches, despite requiring no additional training or complex modifications. Notably, when integrated with LLaVA-NeXT-7B, VisionDrop achieves a  $2.7\times$  reduction in inference latency and  $6\times$  in FLOPs, while retaining 95.71% of the original performance.

**Code** — <https://github.com/Ruixxxx/VisionDrop>

## Introduction

Large Vision-Language Models (LVLMs) have achieved remarkable progress across a wide range of multimodal tasks (Liu et al. 2023; Wang et al. 2024; Bai et al. 2025; Chen et al. 2025b; Zhu et al. 2025), such as visual question answering, image captioning, and visual reasoning. A key driver of this success is the use of dense patch-level tokenization, which encodes visual inputs as long sequences of

\*Corresponding authors: Yong Luo and Bo Du.  
Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

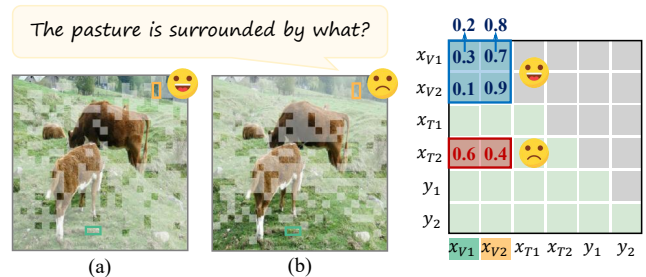


Figure 1: Comparison of visual token pruning strategies via attention maps from LLM decoding layers. Here,  $x_{V1}, x_{V2}$  are visual tokens,  $x_{T1}, x_{T2}$  are text instructions, and  $y_1, y_2$  are autoregressively generated outputs. (a) Our method identifies important visual tokens (e.g., the trees highlighted in orange) by leveraging image-to-image attention (blue box), which reflects intra-modal relevance and avoids interference from misaligned cross-modal signals. (b) In contrast, previous approaches rely on image-to-text attention (red box) to assess visual token importance, which can be overly sensitive to cross-modal noise, resulting in preservation of semantically redundant visual regions (e.g., the grass in green).

tokens to capture rich fine-grained semantics. However, this dense representation comes at a significant computational cost. Compared to textual tokens, visual tokens often dominate the input sequence. It is common for a single image to be represented by hundreds or even thousands of tokens (Liu et al. 2024a), leading to quadratic growth in attention computation (Vaswani et al. 2017). This limits the scalability of LVLMs for high-resolution (Liu et al. 2024b) or real-time applications (Team et al. 2024; Xu et al. 2025).

To address this inefficiency, recent studies have explored reducing the number of visual tokens either before or within the layers of the large language models (LLMs) (Yang et al. 2025; Zhang et al. 2025d; Chen et al. 2024b; Xing et al. 2025). Pruning visual tokens before entering the LLM enables effective compression of redundant features at the representation level, but may risk discarding subtle or task-relevant details. In contrast, performing token reduction within the LLM allows uninformative tokens to be identified after visual information has been integrated into the language context, though it risks discarding entangled or

contextually important features. Additionally, a more fundamental limitation of many in-LLM pruning strategies lies in their reliance on text-conditioned scoring mechanisms, which estimate the relevance between visual tokens and textual inputs. They implicitly assume strong and persistent alignment between modalities throughout LLM layers.

In this work, we revisit this critical assumption by asking: *Are visual and textual representations truly well-aligned within the LLM layers?* Our analysis uncovers three types of cross-modal misalignment that challenge this assumption and limit the effectiveness of text-guided visual token pruning: (1) Causal misalignment arises from the autoregressive nature of LLMs, where the last text token tends to focus on nearby tokens in the input sequence, introducing locality bias in visual token scoring. (2) Semantic misalignment emerges as visual and textual tokens become deeply entangled within LLM layers, weakening the distinctiveness and interpretability of textual queries for assessing visual importance. (3) Spatial misalignment stems from the flattening of positional embeddings across modalities and the absence of spatial priors in textual inputs. While visual encoders already struggle to align spatial structures with textual semantics, this issue worsens in the LLM, where visual and textual tokens are fused into a single sequence. Text-guided pruning under such conditions may discard spatially important regions that are not explicitly emphasized by the text. While CLIP-style encoders (Radford et al. 2021) and LVLM projection layers aim to align modalities at the input stage, such alignment degrades during multimodal token fusion in the LLM. As a result, pruning decisions based on text-guided signals may inadvertently discard visually salient or semantically important content.

To empirically validate this, we design a controlled study comparing visual-only and text-guided scoring strategies for visual token selection. Specifically, we replace image-to-text relevance scores with visual self-attention scores to assess the impact of modality alignment on pruning effectiveness across different compression ratios. Our findings reveal that, in general, visual-only scoring consistently outperforms text-guided scoring, especially under more aggressive pruning. This suggests that visual-textual alignment deteriorates as tokens propagate through the LLM, calling into question the reliability of text-guided reduction strategies.

Based on the analysis, we propose VisionDrop, a training-free framework for visual token reduction. Unlike prior approaches that rely on text-guided relevance, VisionDrop estimates token importance solely from visual self-attention, avoiding dependence on potentially misaligned textual cues. Our method performs stage-wise pruning across the full LVLM architecture, progressively reducing visual tokens in both the visual encoder and LLM layers. First, the dominant token selection identifies highly referenced visual tokens by the aforementioned visual-only importance score, ensuring that key semantic content is retained. Second, the lightweight contextual merging aggregates remaining tokens into contextual tokens by similarity, preserving complementary information. Together, these components are applied at multiple stages in the model, allowing VisionDrop to retain expressive visual representations under tight token budgets.

The contributions of our work are summarized as follows:

- We empirically investigate the misalignment between visual and textual representations within LLM layers and offer insights to the visual token importance scoring.
- We propose VisionDrop, a training-free pruning framework that progressively prunes visual tokens across both the visual encoder and the LLM.
- We propose a visual-only scoring method for dominant token selection and apply the contextual token merging to preserve complementary information at each stage.

Extensive evaluations on diverse benchmarks show that VisionDrop outperforms prior state-of-the-art approaches. Under only 5.6% token retention, it preserves 91.46% and 92.06% of the original LLaVA-1.5-7B (Liu et al. 2024a) and LLaVA-NeXT-7B (Liu et al. 2024b) performance, surpassing the best baseline by 0.96% and 1.71%, respectively.

## Related Works

### Large Vision-Language Models

The evolution of large vision-language models (LVLMs) has been driven by the impressive generalization and reasoning abilities of large language models (LLMs) (Touvron et al. 2023; Jiang et al. 2023; Bai et al. 2023; Qwen et al. 2025; Cai et al. 2024), extending their success to the multi-modal domain. Recent LVLMs integrate visual and textual modalities by projecting visual inputs into token sequences compatible with language models (Liu et al. 2023; Li et al. 2024; Wang et al. 2024; Bai et al. 2025; Chen et al. 2025b; Zhu et al. 2025), enabling them to leverage the full capacity of LLMs. However, the information density in visual data is often much lower than in textual data, resulting in an excessive number of visual tokens. For instance, LLaVA-1.5 (Liu et al. 2024a) encodes a  $336 \times 336$  image into 576 tokens, while LLaVA-NeXT (Liu et al. 2024b) expands this to 2,880 tokens for high-resolution input. This issue becomes even more critical in video understanding tasks. For example, LongVA (Zhang et al. 2024) transforms 2,000 frames into more than 200K visual tokens, while LongVILA (Chen et al. 2025a) supports inputs exceeding 6,000 frames and produces over a million visual tokens. The inflated visual token sequences place heavy demands on computation and memory, motivating the need for effective token reduction strategies to improve scalability and efficiency across tasks.

### Visual Token Reduction in LVLMs

To mitigate the computational burden in LVLMs, recent studies have explored reducing the number of visual tokens. Existing approaches to visual token reduction are primarily applied either in the visual encoder or within the LLM. They typically use either text-guided or text-agnostic scoring strategies to determine token importance. **Visual encoder pruning** approaches perform token reduction before tokens are fed into the LLM. Some approaches rely solely on visual information: FlowCut (Tong et al. 2025) leverages token-level information flow across layers to identify and prune redundant visual tokens in a progressive and structure-aware manner. VisionZip (Yang et al. 2025) and

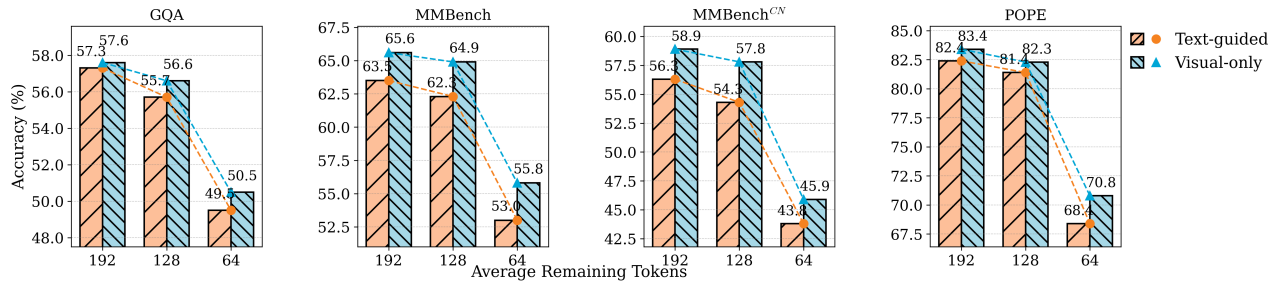


Figure 2: Accuracy comparisons between text-guided and visual-only scoring within the LLM across different average remaining token levels (192, 128, 64) on four benchmarks.



Figure 3: Visualization of text-guided visual token retention after shallow-layer pruning (specifically, the second layer) in the LLM. Pale translucent blocks indicate pruned tokens. Retained tokens consistently cluster at the bottom of the image, revealing a positional bias caused by the causal attention in autoregressive LLMs.

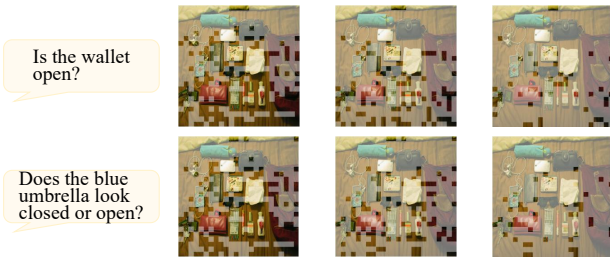


Figure 4: Visualization of text-guided visual token retention at the end of the first three LLM stages. While the wallet-related region is correctly preserved in response to the first question, the model fails to retain relevant tokens around the umbrella in the second case, reflecting the semantic misalignment caused by modal entanglement in LLM layers.

VisPruner (Zhang et al. 2025b) select dominant tokens based on attention scores and compress the rest using similarity-based merging or pruning. VScan (Zhang et al. 2025a) applies global-local scanning. Other works utilize textual guidance at this stage. CDPPruner (Zhang et al. 2025c) maximizes instruction-conditioned diversity of retained tokens. SparseVLM (Zhang et al. 2025d) computes token importance via cross-modal attention with text guidance. Although visual encoder pruning benefits from more structured token semantics, it may risk losing fine-grained cues critical for multimodal reasoning. **LLM decoding pruning** is an alternative direction that performs token reduction within the LLM, where visual and textual tokens are jointly processed.

Most of these approaches adopt text-guided token selection. FastV (Chen et al. 2024b) ranks tokens based on their attention from text during generation, and VScan also extends its pruning into LLM layers based on cross-modal attentions. PyramidDrop (Xing et al. 2025) selects visual tokens that are most relevant to the final instruction token, performing progressive pruning across LLM layers. However, these approaches rely on the assumption that visual and textual tokens remain semantically aligned within the LLM, a property that is not guaranteed due to modality entanglement introduced by joint self-attention (Venhoff et al. 2025).

## Methodology

### Rethinking Text-guided Visual Token Reduction

To alleviate the computational overhead in LLMs, recent studies have explored visual token reduction, either in the visual encoder (Yang et al. 2025; Zhang et al. 2025d) or during the LLM decoding phase (Chen et al. 2024b; Xing et al. 2025). In the latter case, mainstream approaches typically adopt text-guided strategies, using text relevance scores to determine which visual tokens to retain. However, this raises a critical question: *Are visual and textual representations truly well-aligned within the LLM layers?* Although visual inputs are projected into the textual embedding space before entering the LLM, using text-guided strategies for visual token reduction introduces three forms of **cross-modal misalignment**, due to the modality-specific biases in how visual and textual information are processed.

**Causal.** In text-guided pruning, the last text instruction token is commonly used to assess the importance of visual tokens. However, due to the causal nature of autoregressive LLMs, this token tends to focus disproportionately on later-positioned visual tokens in the input sequence (Zhang et al. 2025a). Unlike the [CLS] token or average-pooled visual features in bidirectional encoders, the instruction token does not serve as an explicit global aggregator. Figure 3 illustrates this issue, where shallow-layer pruning results show that the retained visual tokens consistently cluster near the end of the input sequence, regardless of their semantic relevance.

**Semantic.** As tokens propagate through the LLM, visual and textual representations become increasingly entangled. The final instruction token evolves into a hybrid embedding

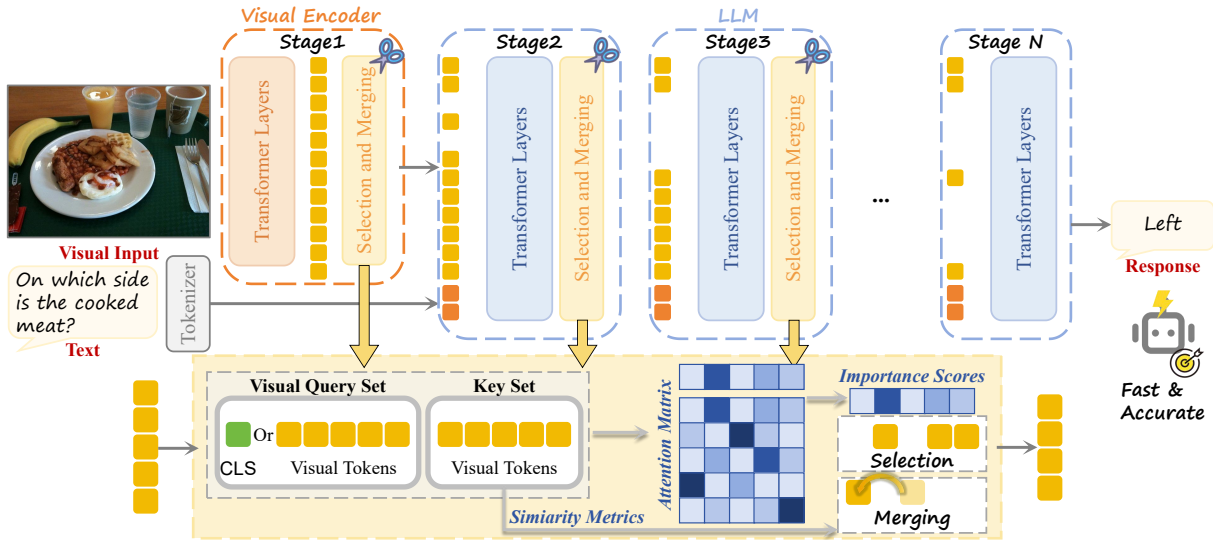


Figure 5: The visual encoder and LLM are partitioned into multiple pruning stages (e.g., Stage 1–N), where the number of visual tokens is progressively reduced. At the end of each stage, the importance of visual tokens is estimated via attention-based scores (values are denoted by color intensity, bottom), computed from self-attention among visual tokens (optionally utilizing the [CLS] as the query). Low-importance tokens are merged with each other using key-value similarity to form contextual representations, which are then propagated together with informative tokens to the next stage.

that no longer maintains a clear semantic alignment with individual visual tokens, limiting its effectiveness in identifying visually important regions. This misalignment is evident in Figure 4, where the retained visual tokens often fail to correspond to the semantics of the input question.

**Spatial.** Preserving spatially relevant visual information has long been a challenge in LVLMs (Chen et al. 2024a). Even before entering the LLM, visual encoders (e.g., CLIP-style models (Radford et al. 2021)) often struggle to align spatial structures with textual semantics, especially for tasks requiring fine-grained or spatial reasoning. This issue is further exacerbated within LLMs, where visual and textual tokens are flattened into a single sequence, and positional embeddings across modalities are merged, diluting the spatial priors of visual tokens. Moreover, textual inputs themselves lack spatial awareness and provide inherently incomplete representations of visual scenes. As a result, text-guided visual token pruning may amplify this spatial misalignment by discarding visually important regions that are not explicitly emphasized by the textual instruction.

These misalignments reveal that text-guided visual token pruning is fundamentally constrained by modality interaction biases. To empirically verify this, we conduct a controlled experiment by replacing text-guided scoring with visual-only scoring and observe the resulting performance changes. Specifically, we reproduce the PyramidDrop (Xing et al. 2025) framework. To ensure a fair comparison, we retain the same pruning configuration, including the pruning layers and token retention ratios, and replace the image-to-text relevance scores with image-to-image self-attention scores, thereby eliminating explicit textual guidance. We adopt LLaVA-v1.5-7B (Liu et al. 2024a) as the base LVLM

and conduct extensive evaluations on 4 standard image understanding benchmarks, including GQA (Hudson and Manning 2019), MMBench (Liu et al. 2024c), MMBench<sup>CN</sup> (Liu et al. 2024c), and POPE (Li et al. 2023), to compare the effectiveness of the two scoring strategies. As shown in Figure 2, the image-only scoring strategy consistently outperforms the text-guided approach when retaining 192 visual tokens, and in general, the performance gap widens as the number of retained tokens decreases to 128 and 64. This performance gap under higher compression ratios indicates that visual and textual tokens are not well aligned within the LLM. Consequently, text-guided scoring, which implicitly assumes reliable cross-modal alignment, becomes increasingly unreliable when fewer visual tokens remain. In contrast, visual-only scoring exhibits greater robustness under aggressive pruning, highlighting its potential as a more stable and generalizable strategy for visual token reduction.

## Preliminaries

LVLMs are commonly structured with three components: a vision encoder, a modality projector, and an LLM. Upon receiving the visual input, the vision encoder extracts patch-level features, which are subsequently mapped into a set of  $n$  visual tokens  $\mathbf{x}_V = \{x_V^i\}_{i=1}^n$  via the projector. These visual tokens are concatenated with the tokenized textual tokens  $\mathbf{x}_T$  and fed into the LLM for autoregressive generation. At each timestep  $t$ , the model predicts the next token  $y_t$  based on the conditional probability  $p_\theta(y_t | \mathbf{x}_V, \mathbf{x}_T, \mathbf{y} < t)$ , where  $\mathbf{y} < t$  denotes the previously generated tokens.

## VisionDrop

Visual token reduction in LVLMs has been explored at two primary stages: either within the visual encoder (Yang et al.

2025; Zhang et al. 2025d,b) or during the LLM decoding phase (Chen et al. 2024b; Xing et al. 2025). We observe that pruning at the encoder stage tends to be more stable, as visual tokens remain semantically coherent and well-structured. However, this may lead to the loss of fine-grained visual details that are crucial for downstream reasoning. In contrast, pruning in the LLM benefits from earlier visual-textual integration, where key visual cues may already be embedded in textual tokens, allowing more informed reduction. Yet as tokens propagate through LLM layers, visual information becomes increasingly entangled and diffused, making it difficult to reliably estimate token importance.

To address the cross-modal misalignment within LLMs and to leverage the complementary strengths of different pruning stages, we propose a training-free, visual-only pruning framework that performs progressive visual token reduction at both the encoder and LLM stages, balancing stability and expressiveness across the model hierarchy, as illustrated in Figure 5.

**Progressive Dominant Token Selection.** We partition the LVLM architecture into a sequence of stages  $\mathcal{S} = s_0, s_1, \dots, s_N$ , including both the visual encoder and layers in the LLM decoder. At the end of each stage  $s_n$ , where  $n = 1, 2, \dots, N$ , we apply a stage-specific pruning ratio  $\lambda_n$  to the current set of visual tokens  $\mathbf{x}_V^{(s_n)}$ , where  $|\mathbf{x}_V^{(s_n)}| = \lambda_n \cdot |\mathbf{x}_V|$ . Tokens are ranked based on visual-only attention-based importance scores, and the top-ranked subset is propagated to the next stage  $s_{n+1}$ . The selection is given by:

$$\mathbf{x}_V^{(s_{n+1})} = \left\{ x_V^i \in \mathbf{x}_V^{(s_n)} \mid S(i) \geq \tau_n \right\}, \quad (1)$$

where  $S(i)$  denotes the visual importance score of token  $x_V^i$ , and  $\tau_n$  is a threshold determined by  $\lambda_n$ .

To identify the most informative visual tokens, we compute importance scores by reusing the model’s self-attention maps. Specifically, we measure how frequently each token is attended to by visual query tokens, averaged over all query positions. For clarity, we present the single-head attention formulation and omit the pruning stage notation.

Formally, let  $\mathbf{x}_V^q \in \mathbb{R}^{L_1 \times D}$  denote the set of query visual tokens, and  $\mathbf{x} \in \mathbb{R}^{L_2 \times D}$  be the full input sequence, including both visual and textual tokens. The attention query and key matrices are computed as:

$$\mathbf{Q} = \mathbf{x}_V^q \mathbf{W}_Q, \quad \mathbf{K} = \mathbf{x} \mathbf{W}_K, \quad (2)$$

where  $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{D \times D}$  are the projection matrices. The single-head attention matrix is obtained by:

$$\mathbf{A} = \text{Softmax} \left( \frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{D}} \right), \quad (3)$$

where  $\mathbf{A} \in \mathbb{R}^{L_1 \times L_2}$  contains attention scores from each query visual token to all tokens in the sequence.

To compute the final importance score  $\mathbf{S} \in \mathbb{R}^{L_V}$  over the visual tokens, we first extract the attention weights  $\mathbf{A}_{\cdot, \mathcal{V}} \in \mathbb{R}^{L_1 \times L_V}$  corresponding to the visual key tokens from the full attention matrix  $\mathbf{A} \in \mathbb{R}^{L_1 \times L_2}$ . We then average across all visual queries:

$$\mathbf{S} = \frac{1}{L_1} \sum_{l=1}^{L_1} \mathbf{A}[l, \mathcal{V}], \quad (4)$$

where  $\mathcal{V}$  is the index set of visual tokens in the sequence.

**Visual Query Selection.** In LLM, where no explicit [CLS] token exists, we compute visual-to-visual attention by isolating attention maps within the visual subspace. Specifically, we extract attention scores from image queries to full token sequence, where  $\mathbf{Q}_V = \mathbf{x}_V \mathbf{W}_Q, \mathbf{K} = \mathbf{x} \mathbf{W}_K$ . We then keep only  $\mathbf{A}$  entries indexed by visual tokens.

In visual encoder, if a dedicated [CLS] token is present (e.g., CLIP (Radford et al. 2021)), we use attention from it to each visual token as token importance, where  $\mathbf{Q}_{\text{CLS}} = \mathbf{x}_{\text{CLS}} \mathbf{W}_Q$ . If no such token exists (e.g., SigLIP (Zhai et al. 2023)), we follow a consistent strategy with LLM and average the attention each token receives from all visual tokens.

This unified formulation enables modality-consistent and architecture-adaptive token importance estimation.

**Stage-wise Contextual Token Merging.** While dominant tokens capture the primary visual content, discarding the remaining ones may result in missing subtle or auxiliary visual cues. To address this, we introduce a lightweight merging step at each pruning stage to preserve such complementary information. While similar ideas have been proposed in prior work (Yang et al. 2025), our framework uniquely generalizes this operation across both the visual encoder and LLM decoder stages in a progressive fashion.

Specifically, we reuse the key embeddings from the attention module, which encode semantic content of tokens, to measure pairwise similarity. In the LLM, we explicitly extract the image-token portion of the key vectors before similarity computation, ensuring modality-pure merging. At the end of each stage, the non-dominant tokens are divided into candidate and reference sets, and each candidate token is matched to its most similar reference token based on a dot-product similarity. The matched tokens are then fused to produce enriched contextual tokens. This process ensures that visually redundant tokens are merged rather than dropped, preserving fine-grained details while keeping the token count under control.

## Experiments

### Experimental Setups

**Models and Baselines.** We evaluate the proposed VisionDrop on the widely adopted LLaVA-1.5-7B (Liu et al. 2024b) and LLaVA-NeXT-7B (Liu et al. 2024b) for image understanding, as well as Video-LLaVA-7B (Lin et al. 2024) for video understanding. We compare our method with several state-of-the-art visual token reduction baselines, including FastV (Chen et al. 2024b), PyramidDrop (Xing et al. 2025), SparseVLM (Zhang et al. 2025d), VisionZip (Yang et al. 2025), and VisPruner (Zhang et al. 2025b).

**Benchmarks.** We evaluate VisionDrop on 9 standard image understanding benchmarks. These include general visual question answering datasets such as GQA (Hudson and Manning 2019), VQAav2 (Goyal et al. 2017), and

Method	GQA	MMB	MMB <sup>CN</sup>	MME	POPE	SQA	VQA <sup>v2</sup>	VQA <sup>Text</sup>	VizWiz	Avg.
<i>Upper Bound, 576 Tokens (100%)</i>										
LLaVA-1.5-7B	61.92	66.31	58.63	1863	86.81	69.51	78.53	58.20	50.13	100%
<i>Retain 192 Tokens (↓66.7%)</i>										
FastV (ECCV24)	52.62	57.74	48.43	1540	75.59	68.07	70.51	52.77	46.76	88.45%
PDrop (CVPR25)	57.27	63.51	56.28	1778	82.40	<b>69.56</b>	75.57	56.10	48.95	96.11%
SparseVLM-v1.5 (ICML25)	59.44	<b>65.41</b>	<b>58.69</b>	1789	86.45	68.86	<u>77.01</u>	57.76	50.64	<u>98.64%</u>
VisionZip (CVPR25)	59.25	64.46	57.29	1767	86.39	68.86	76.78	<u>57.26</u>	<b>51.55</b>	98.11%
VisPruner (ICCV25)	<u>59.52</u>	64.91	57.62	1779	<u>86.57</u>	68.52	76.96	57.42	<u>51.46</u>	98.38%
VisionDrop (Ours)	<b>59.99</b>	<u>65.19</u>	<u>58.41</u>	<b>1801</b>	<b>87.23</b>	<u>69.06</u>	<b>77.28</b>	<b>57.81</b>	50.01	<b>98.76%</b>
<i>Retain 128 Tokens (↓77.8%)</i>										
FastV (ECCV24)	51.83	57.29	49.05	1502	75.55	68.37	66.61	51.09	49.41	87.88%
PDrop (CVPR25)	55.70	62.28	54.32	1656	81.43	<b>69.36</b>	74.21	55.32	49.01	94.04%
SparseVLM-v1.5 (ICML25)	58.43	<b>65.25</b>	<b>58.86</b>	1750	<b>86.37</b>	68.57	<u>76.21</u>	56.61	50.27	<u>97.76%</u>
VisionZip (CVPR25)	57.62	63.40	56.67	1768	84.69	68.82	75.60	56.81	<u>52.02</u>	97.16%
VisPruner (ICCV25)	58.48	63.57	56.67	1743	85.67	<u>69.01</u>	75.74	<u>57.00</u>	<b>52.60</b>	97.53%
VisionDrop (Ours)	<b>58.61</b>	<u>64.52</u>	<u>57.06</u>	<b>1777</b>	<u>85.92</u>	<u>68.52</u>	<b>76.24</b>	<u>57.63</u>	51.06	<b>97.80%</b>
<i>Retain 64 Tokens (↓88.9%)</i>										
FastV (ECCV24)	48.70	52.30	43.27	1409	67.21	69.36	61.08	49.69	50.55	83.13%
PDrop (CVPR25)	49.51	52.97	43.83	1429	68.42	<b>69.96</b>	63.04	50.80	50.02	84.23%
SparseVLM-v1.5 (ICML25)	53.77	61.27	52.13	1591	80.86	<u>69.61</u>	70.24	53.46	50.35	92.06%
VisionZip (CVPR25)	55.16	62.61	54.20	1687	80.45	69.01	72.42	55.48	<u>52.94</u>	94.62%
VisPruner (ICCV25)	<u>55.59</u>	<u>62.56</u>	54.48	1679	<u>81.29</u>	68.82	<u>72.64</u>	<b>55.83</b>	<b>53.00</b>	94.89%
VisionDrop (Ours)	<b>55.89</b>	<b>62.95</b>	<b>55.10</b>	<b>1698</b>	<b>81.58</b>	69.31	<b>73.16</b>	55.59	52.28	<b>95.22%</b>
<i>Retain 32 Tokens (↓94.4%)</i>										
VisionZip (CVPR25)	51.80	58.02	49.78	1592	75.11	68.72	67.12	53.17	<b>52.91</b>	89.92%
VisPruner (ICCV25)	51.98	59.02	50.78	<b>1593</b>	76.44	69.26	67.41	53.47	52.16	90.50%
VisionDrop (Ours)	<b>52.79</b>	<b>60.31</b>	<b>52.91</b>	1572	<b>77.19</b>	<b>69.41</b>	<b>68.55</b>	<b>53.56</b>	52.26	<b>91.46%</b>

Table 1: Performance comparison on LLaVA-1.5-7B (Liu et al. 2024a) under different token retention rates. The best results are marked in **bold**, and the second-best are underlined.

TextVQA (Singh et al. 2019); robustness and fairness benchmarks including MMBench (Liu et al. 2024c), MMBench-CN (Liu et al. 2024c), MME (Fu et al. 2024) and POPE (Li et al. 2023); as well as specialized datasets like VizWiz (Gurari et al. 2018), which features low-quality images, and ScienceQA (SQA) (Lu et al. 2022), which focuses on scientific reasoning. We also evaluate using 3 video understanding benchmarks, TGIF (Jang et al. 2017), MSVD (Chen and Dolan 2011), and MSRVT (Xu et al. 2016).

**Implementation Details.** We adopt the standard inference settings provided in the official implementations of each LLM. For progressive visual token pruning, we divide the model into five stages: the first stage concludes at the output of the visual encoder, while the remaining four end at decoding layers  $l = 8, 16, 24$ , and the final decoding layer of the LLM. The number of visual tokens is gradually reduced across these stages. In the first stage, all original visual tokens are retained. For each reduction configuration, the retention number in the second stage is set to  $1.5 \times$  the final count for image understanding and  $3 \times$  for video understanding, and tokens are pruned proportionally in subsequent stages to meet the overall budget.

## Results and Analyses

**Results on LLaVA-1.5.** As shown in Table 1, VisionDrop consistently outperforms all competing approaches across different token retention levels. Even with only 32

visual tokens (a 94.4% reduction), it preserves 91.46% of the full-token performance, demonstrating strong efficiency with minimal degradation. While SparseVLM (Zhang et al. 2025d) performs well at higher token counts but degrades under extreme compression, VisPruner (Zhang et al. 2025b) and VisionZip (Yang et al. 2025) show better robustness at low budgets, performing particularly well on VizWiz (Gurari et al. 2018), likely due to their early-stage pruning in low-quality images. Remarkably, VisionDrop maintains the best performance, especially under tighter token constraints.

**Results on LLaVA-NeXT.** LLaVA-NeXT (Liu et al. 2024b) enhances LLaVA-1.5 by supporting higher-resolution visual inputs, yielding more visual tokens and redundancy, making it a compelling testbed for evaluating token pruning strategies. As shown in Table 2, VisionDrop consistently outperforms all baselines across different token retention rates, demonstrating its strong ability to retain critical visual information. Notably, its advantage becomes more pronounced as the number of retained tokens decreases. At 160 tokens (a 94.4% reduction), VisionDrop achieves an average performance of 92.06%, which is 1.71% higher than the second-best approach.

**Results on Video-LLaVA.** We further evaluate VisionDrop on video understanding tasks. Following prior works (Chen et al. 2024b), we use the first 1,000 samples from each dataset for evaluation due to API usage limits. As shown in Table 3, VisionDrop achieves the highest average

Method	GQA	MMB	MMB <sup>CN</sup>	MME	POPE	SQA	VQA <sup>v2</sup>	VQA <sup>Text</sup>	VizWiz	Avg.
<i>Upper Bound, 2880 Tokens (100%)</i>										
LLaVA-NeXT-7B	64.27	69.39	62.00	1854	87.54	70.30	81.83	61.40	57.63	100%
<i>Retain 640 Tokens (↓77.8%)</i>										
FastV (ECCV24)	56.26	64.97	59.25	1646	83.98	68.27	73.90	54.80	55.00	92.62%
PDrop (CVPR25)	<b>61.80</b>	<b>68.11</b>	<b>61.10</b>	1782	86.44	<b>69.36</b>	<u>79.71</u>	59.44	55.44	<u>97.42%</u>
VisionZip (CVPR25)	61.31	66.14	<u>60.43</u>	1805	87.38	67.82	79.13	<u>60.18</u>	<b>57.32</b>	97.33%
VisionDrop (Ours)	<u>61.70</u>	<u>66.98</u>	<u>60.43</u>	<b>1807</b>	<b>87.85</b>	<u>68.32</u>	<b>79.76</b>	<b>60.45</b>	<u>56.78</u>	<b>97.72%</b>
<i>Retain 320 Tokens (↓88.9%)</i>										
FastV (ECCV24)	53.49	61.94	54.99	1495	76.16	66.83	68.24	51.40	52.07	86.82%
PDrop (CVPR25)	57.08	64.46	56.67	1672	79.85	<b>69.91</b>	74.81	54.57	53.66	91.93%
VisionZip (CVPR25)	<u>58.92</u>	63.40	<u>57.40</u>	<u>1712</u>	<u>84.63</u>	67.67	<u>76.32</u>	<u>58.86</u>	<u>56.64</u>	<u>94.26%</u>
VisionDrop (Ours)	<b>59.95</b>	<b>64.91</b>	<b>58.13</b>	<b>1789</b>	<b>85.21</b>	<u>67.97</u>	<b>77.67</b>	<b>59.24</b>	<b>56.97</b>	<b>95.71%</b>
<i>Retain 160 Tokens (↓94.4%)</i>										
VisionZip (CVPR25)	55.55	60.09	54.71	1626	79.63	<b>68.12</b>	71.72	56.24	55.95	90.35%
VisionDrop (Ours)	<b>57.02</b>	<b>62.00</b>	<b>55.61</b>	<b>1646</b>	<b>81.88</b>	67.53	<b>73.99</b>	<b>57.52</b>	<b>56.65</b>	<b>92.06%</b>

Table 2: Performance comparison on LLaVA-NeXT-7B (Liu et al. 2024b) under different token retention rates.

Method	TGIF		MSVD		MSRVTT		Avg.	
	Acc.	Score	Acc.	Score	Acc.	Score	Acc.	Score
Video-LLaVA	18.9	2.52	71.6	3.94	56.8	3.48	49.1	3.31
FastV	23.2	2.48	46.6	3.21	45.0	3.14	38.3	2.94
VisionZip	15.6	2.40	70.2	3.94	54.0	3.41	46.6	3.25
VisionDrop	18.8	2.48	68.4	3.87	54.6	3.42	<b>47.3</b>	<b>3.26</b>

Table 3: Performance comparison on Video-LLaVA-7B (Lin et al. 2024) under a 256 token retention.

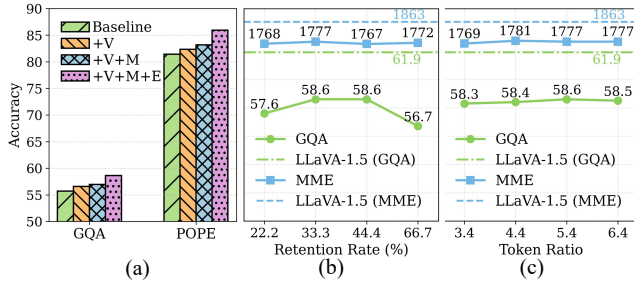


Figure 6: Ablation studies under 22.2% token retention. (a) Component-wise results. V: visual-only; M: Merging; E: Encoder pruning. (b) Token retention rates applied to the visual encoder. (c) Dominant-to-contextual token ratios.

performance among all compared approaches, obtaining an accuracy of 47.3% and a score of 3.26, while retaining only 12.5% of the visual tokens.

**Ablation Study.** We conduct ablation studies under a 22.2% token retention setting. Figure 6(a) presents the results of different component combinations of our method. The results demonstrate that all individual modules contribute positively to performance. Specifically, incorporating visual-only scoring, by removing dependence on text queries within the LLM, yields a notable improvement. The inclusion of stage-wise token merging further boosts performance by retaining fine-grained visual context that would otherwise be lost. Figure 6(b) examines the impact of different token

Method	# Token	FLOPs (T)	Memory (GB)	Time (ms)
LLaVA-1.5	576	9.06	14.52	237
VisionDrop	64	2.11	14.35	117 (×2.0)
LLaVA-NeXT	2880	46.25	16.56	593
VisionDrop	320	7.70	14.63	216 (×2.7)

Table 4: Efficiency analysis of VisionDrop on LLaVA-1.5-7B and LLaVA-NeXT-7B.

retention rates in the visual encoder. It is observed that a moderate retention rate of 33.3% achieves the best performance. Figure 6(c) further explores the effect of varying the dominant-to-contextual token ratios. The model maintains stable performance across ratios, with slight improvements when dominant tokens are emphasized.

**Efficiency Analysis.** To evaluate the efficiency of VisionDrop, we conduct a comparative analysis of FLOPs, memory usage, and inference latency on LLaVA-1.5 and LLaVA-NeXT. All experiments are conducted on a single 24GB NVIDIA GeForce RTX 3090 GPU. As shown in Table 4, by retaining only 64 and 320 tokens for LLaVA-1.5 and LLaVA-NeXT, respectively, and preserving over 95% of the original performance, VisionDrop achieves a 2.0× and 2.7× speedup, along with a 4.3× and 6.0× reduction in FLOPs.

## Conclusion

In this work, we introduce VisionDrop, a simple yet effective training-free, visual-only pruning framework designed to mitigate modality misalignment and reduce computational cost in LVLMS. By treating the visual encoder and the LLM as a unified processing pipeline, VisionDrop performs progressive dominant token selection and contextual merging without relying on text-conditioned signals. This allows the model to preserve critical visual cues under strict token budgets. Moreover, VisionDrop’s reliance on purely visual signals makes it especially advantageous in domains where language cues are sparse or weakly aligned with visual content, such as high-resolution medical or remote sensing images.

## Acknowledgments

This research was supported in part by the National Natural Science Foundation of China (Grant Nos. U23A20318, 62276195, and 62225113).

## References

- Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; et al. 2023. Qwen Technical Report. *arXiv preprint arXiv:2309.16609*.
- Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; et al. 2025. Qwen2.5-VL Technical Report. *arXiv preprint arXiv:2502.13923*.
- Cai, Z.; Cao, M.; Chen, H.; Chen, K.; Chen, K.; Chen, X.; et al. 2024. InternLM2 Technical Report. *arXiv preprint arXiv:2403.17297*.
- Chen, D. L.; and Dolan, W. B. 2011. Collecting Highly Parallel Data for Paraphrase Evaluation. In *ACL*, 190–200.
- Chen, G.; Shen, L.; Shao, R.; Deng, X.; and Nie, L. 2024a. LION : Empowering Multimodal Large Language Model with Dual-Level Visual Knowledge. In *CVPR*, 26530–26540.
- Chen, L.; Zhao, H.; Liu, T.; Bai, S.; Lin, J.; Zhou, C.; and Chang, B. 2024b. An Image is Worth 1/2 Tokens After Layer 2: Plug-and-Play Inference Acceleration for Large Vision-Language Models. In *ECCV*.
- Chen, Y.; Xue, F.; Li, D.; Hu, Q.; Zhu, L.; Li, X.; et al. 2025a. LongVILA: Scaling Long-Context Visual Language Models for Long Videos. In *ICLR*.
- Chen, Z.; Wang, W.; Cao, Y.; Liu, Y.; Gao, Z.; Cui, E.; et al. 2025b. Expanding Performance Boundaries of Open-Source Multimodal Models with Model, Data, and Test-Time Scaling. *arXiv preprint arXiv:2412.05271*.
- Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Zhang, M.; Lin, X.; et al. 2024. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2306.13394*.
- Goyal, Y.; Khot, T.; Summers-Stay, D.; Batra, D.; and Parikh, D. 2017. Making the V in VQA Matter: Elevating the Role of Image Understanding in Visual Question Answering. In *CVPR*, 6325–6334.
- Gurari, D.; Li, Q.; Stangl, A. J.; Guo, A.; Lin, C.; Grauman, K.; Luo, J.; and Bigham, J. P. 2018. VizWiz Grand Challenge: Answering Visual Questions From Blind People. In *CVPR*, 3608–3617.
- Hudson, D. A.; and Manning, C. D. 2019. GQA: A New Dataset for Real-World Visual Reasoning and Compositional Question Answering. In *CVPR*, 6700–6709.
- Jang, Y.; Song, Y.; Yu, Y.; Kim, Y.; and Kim, G. 2017. TGIF-QA: Toward Spatio-Temporal Reasoning in Visual Question Answering. In *CVPR*, 1359–1367.
- Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; de las Casas, D.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.
- Li, B.; Zhang, Y.; Guo, D.; Zhang, R.; Li, F.; Zhang, H.; Zhang, K.; Li, Y.; Liu, Z.; and Li, C. 2024. LLaVA-OneVision: Easy Visual Task Transfer. *arXiv preprint arXiv:2408.03326*.
- Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W. X.; and Wen, J. 2023. Evaluating Object Hallucination in Large Vision-Language Models. In *EMNLP*, 292–305.
- Lin, B.; Ye, Y.; Zhu, B.; Cui, J.; Ning, M.; Jin, P.; and Yuan, L. 2024. Video-LLaVA: Learning United Visual Representation by Alignment Before Projection. In *EMNLP*, 5971–5984.
- Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024a. Improved Baselines with Visual Instruction Tuning. In *CVPR*, 26286–26296.
- Liu, H.; Li, C.; Li, Y.; Li, B.; Zhang, Y.; Shen, S.; and Lee, Y. J. 2024b. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge.
- Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual Instruction Tuning. In *NeurIPS*.
- Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; Chen, K.; and Lin, D. 2024c. MMBench: Is Your Multi-modal Model an All-Around Player? In *ECCV*, volume 15064, 216–233.
- Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.; Zhu, S.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. In *NeurIPS*.
- Qwen; ; Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; et al. 2025. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115*.
- Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *ICML*, volume 139, 8748–8763.
- Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; and Rohrbach, M. 2019. Towards VQA Models That Can Read. In *CVPR*, 8317–8326.
- Team, G.; Mesnard, T.; Hardin, C.; Dadashi, R.; Bhupatiraju, S.; Pathak, S.; et al. 2024. Gemma: Open Models Based on Gemini Research and Technology. *arXiv preprint arXiv:2403.08295*.
- Tong, J.; Jin, W.; Qin, P.; Li, A.; Zou, Y.; Li, Y.; Li, Y.; and Li, R. 2025. FlowCut: Rethinking Redundancy via Information Flow for Efficient Vision-Language Models. *arXiv preprint arXiv:2505.19536*.
- Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; et al. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All you Need. In *NeurIPS*, 5998–6008.
- Venhoff, C.; Khakzar, A.; Joseph, S.; Torr, P.; and Nanda, N. 2025. How Visual Representations Map to Language Feature Space in Multimodal LLMs. *arXiv preprint arXiv:2506.11976*.

Wang, P.; Bai, S.; Tan, S.; Wang, S.; Fan, Z.; Bai, J.; et al. 2024. Qwen2-VL: Enhancing Vision-Language Model’s Perception of the World at Any Resolution. *arXiv preprint arXiv:2409.12191*.

Xing, L.; Huang, Q.; Dong, X.; Lu, J.; Zhang, P.; Zang, Y.; Cao, Y.; He, C.; Wang, J.; Wu, F.; and Lin, D. 2025. PyramidDrop: Accelerating Your Large Vision-Language Models via Pyramid Visual Redundancy Reduction. In *CVPR*.

Xu, J.; Mei, T.; Yao, T.; and Rui, Y. 2016. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *CVPR*, 5288–5296.

Xu, S.; Wang, Y.; Xia, C.; Zhu, D.; Huang, T.; and Xu, C. 2025. VLA-Cache: Efficient Vision-Language-Action Manipulation via Adaptive Token Caching. In *NeurIPS*.

Yang, S.; Chen, Y.; Tian, Z.; Wang, C.; Li, J.; Yu, B.; and Jia, J. 2025. VisionZip: Longer is Better but Not Necessary in Vision Language Models. In *CVPR*.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid Loss for Language Image Pre-Training. In *ICCV*, 11941–11952.

Zhang, C.; Ma, K.; Fang, T.; Yu, W.; Zhang, H.; Zhang, Z.; Xie, Y.; Sycara, K.; Mi, H.; and Yu, D. 2025a. VScan: Rethinking Visual Token Reduction for Efficient Large Vision-Language Models. *arXiv preprint arXiv:2505.22654*.

Zhang, P.; Zhang, K.; Li, B.; Zeng, G.; Yang, J.; Zhang, Y.; et al. 2024. Long Context Transfer from Language to Vision. *arXiv preprint arXiv:2406.16852*.

Zhang, Q.; Cheng, A.; Lu, M.; Zhang, R.; Zhuo, Z.; Cao, J.; Guo, S.; She, Q.; and Zhang, S. 2025b. Beyond Text-Visual Attention: Exploiting Visual Cues for Effective Token Pruning in VLMs. In *ICCV*.

Zhang, Q.; Liu, M.; Li, L.; Lu, M.; Zhang, Y.; Pan, J.; She, Q.; and Zhang, S. 2025c. Beyond Attention or Similarity: Maximizing Conditional Diversity for Token Pruning in MLLMs. *arXiv preprint arXiv:2506.10967*.

Zhang, Y.; Fan, C.-K.; Ma, J.; Zheng, W.; Huang, T.; Cheng, K.; Gudovskiy, D.; Okuno, T.; Nakata, Y.; Keutzer, K.; et al. 2025d. SparseVLM: Visual Token Sparsification for Efficient Vision-Language Model Inference. In *ICML*.

Zhu, J.; Wang, W.; Chen, Z.; Liu, Z.; Ye, S.; Gu, L.; et al. 2025. InternVL3: Exploring Advanced Training and Test-Time Recipes for Open-Source Multimodal Models. *arXiv preprint arXiv:2504.10479*.