DIPRUNE: TASK-AWARE TOKEN PRUNING WITH DUAL IMPORTANCE FOR EFFICIENT MULTIMODAL LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent training-free pruning approaches for Multimodal Large Language Models (MLLMs) effectively cut computational overhead by exploiting visual redundancy or text-vision attention. However, they frequently suffer from semantic degradation due to their task-agnostic design or unreliable attention estimates. Based on our empirical analysis, we have found that this issue arises because salient tokens in shallow layers persistently suppress emerging semantic ones through numerical inertia, leading to premature discarding of signals crucial for deep reasoning. To address the aforementioned issue, from the task-oriented aspects, we first reformulate training-free pruning as a minimization of the distortion in the final task loss and derive a tractable, token-wise upper bound to serve as a surrogate objective. Specifically, this formulation inherently reveals a previously neglected inter-layer term that accounts for gradients across layers. Accordingly, for the implementation, we propose **DIPrune**, a rank-based framework that employs a dual importance scoring mechanism to jointly optimize intra-layer static feature saliency and inter-layer dynamic semantic evolution. Extensive experiments on LLaVA and Qwen-VL demonstrate that DIPrune consistently achieves state-of-the-art results.

1 INTRODUCTION

Recent advances in Multimodal Large Language Models (MLLMs) have achieved strong performance but face a critical computational bottleneck: high-resolution images (Li et al., 2026) produce an excessive number of visual tokens, leading to quadratic growth in self-attention cost. Training-free token pruning (Bolya et al., 2023) has emerged to address this, with current methods falling into two categories. Visual-redundancy-based pruning (Yang et al., 2025; Wen et al., 2025; Zou et al., 2025; Zhang et al., 2026) removes statistically similar tokens but operates independently of the text instruction, lacking semantic guidance and risking the loss of task-relevant content. Text-vision-interaction-based pruning (Chen et al., 2024; Zhang et al., 2025; Xing et al., 2025) uses cross-modal attention for guidance, yet remains susceptible to inherent attention biases, such as position bias (Endo et al., 2025; Zhang et al., 2024), which can favor spatially proximate over semantically important tokens. Both paradigms therefore exhibit distinct limitations, either being task-agnostic or relying on unreliable attention estimates. It is, therefore, non-trivial to achieve task-aware and reliable attention estimates for training-free token pruning.

Empirically, we begin by conducting analysis and observation of visual tokens from the perspectives of different layers and tasks. Interestingly, we highlight an overlooked phenomenon for training-free token pruning: *the Early Inertial Bias in attention distributions*, i.e., tokens that acquire high attention weights in shallow layers exhibit strong numerical inertia, allowing them to persistently dominate in subsequent layers. This bias is exacerbated by the fact that the initial layers tend to focus on broad, generic visual features (Shi et al., 2025; Kang et al., 2025). Crucially, among these tokens, we distinguish a class that maintains high ranks solely due to inertia yet lacks task-driven dynamic evolution. We define these as **Inertial Tokens**, representing computational redundancy that should be discarded. Consequently, prevailing pruning strategies, whether based on redundancy (Shang et al., 2025; Wang et al., 2025; Arif et al., 2025; Yang et al., 2025; Zou et al., 2025) or snapshot attention (Xing et al., 2025; Chen et al., 2024; Zhang et al., 2025; Ye et al., 2025), could rely

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

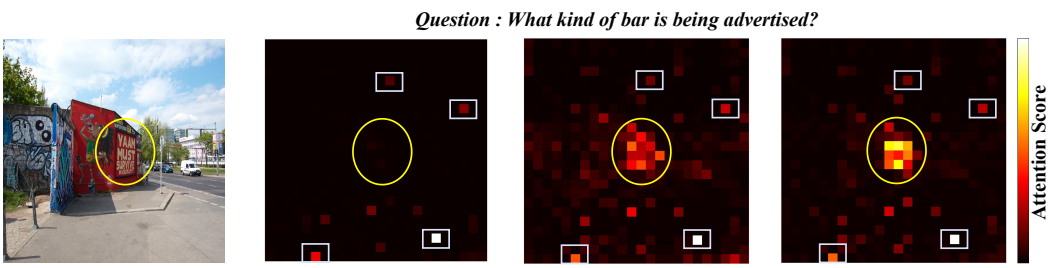


Figure 1: Cross-Layer Attention Evolution. Attention maps across depth (i.e., Layer 2/8/12) exhibit two behaviors: the answer-critical region (yellow circle) is weakly activated in early layers but gains attention with depth, while boxed tokens (white rectangle) remain highly attended across layers.

excessively on these early, static importance signals. This bias consistently leads to the premature removal of counterpart **Emerging Tokens**, i.e., tokens that appear less prominent in the shallow layers but increasingly capture semantically crucial information needed for task-specific reasoning in deeper layers. To address the aforementioned limitations, we first formulate training-free pruning as a budget-constrained visual token pruning optimization problem, which minimizes the distortion in the final task loss. Since the original objective is intractable, we further derive a tractable, token-wise upper bound that we use as a surrogate objective.

Crucially, this objective decouples into intra-layer and inter-layer terms, which respectively characterize Static Semantic Alignment through feature-salient importance and Dynamic Semantic Evolution through evolution-aware importance. Especially, it inherently reveals a previously neglected inter-layer term that accounts for gradients across layers.

Accordingly, we propose **DIPrune**, a rank-based framework for training-free token pruning. Based on our formulation, DIPrune instantiates a dual importance scoring mechanism that directly tackles the three core challenges: (1) To overcome the lack of textual guidance, it computes a Semantic Alignment Score (S_{align}) via cross-modal attention and inverse-rank normalization, prioritizing tokens semantically aligned with the instruction; (2) To mitigate reliance on biased attention estimates, it fuses S_{align} and the Semantic Evolution Score (S_{evol}) with weights, providing a robust, multi-perspective importance assessment that avoids over-reliance on any single, potentially biased signal; (3) To counteract the early inertial bias, it calculates S_{evol} by tracking rank displacement of tokens across adjacent layers, thereby preserving those gaining importance in deeper reasoning. By retaining only the top-K tokens based on this dual importance score, DIPrune reduces the self-attention complexity while maintaining high semantic fidelity under aggressive compression.

Extensive experiments show that DIPrune demonstrates exceptional robustness and generalizability. Evaluations across diverse MLLM architectures further verify that our method consistently establishes new state-of-the-art results. The contributions of our work can be summarized as follows:

- **Early Inertial Bias.** We systematically identify and validate the Early Inertial Bias in attention distributions, revealing that shallow Inertial Tokens persistently suppress deep Emerging Tokens.
- **DIPrune Framework.** We propose DIPrune, a rank-based pruning framework that employs a dual importance scoring mechanism to jointly optimize static semantic alignment and dynamic semantic evolution.
- **SOTA Performance.** DIPrune achieves new state-of-the-art results under high compression ratios across multiple mainstream MLLMs, demonstrating strong generalization capability and inference efficiency.

2 Related Work

Multimodal Large Language Models. In recent years, Multimodal Large Language Models (MLLMs) have achieved remarkable progress (Chen et al., 2023; Lin et al., 2024; Li et al., 2024; Tong et al., 2024; Zhu et al., 2024). Evolving from early paradigms based on frozen components,

such as BLIP (Li et al., 2022) and Flamingo (Alayrac et al., 2022), to end-to-end training architectures like LLaVA (Liu et al., 2024a) and Qwen-VL (Bai et al., 2025), MLLMs have demonstrated superior performance across a variety of vision-language tasks. However, these performance gains are accompanied by substantial computational costs. High-resolution visual inputs induce an explosive growth in the number of visual tokens; for instance, while LLaVA-v1.5 (Liu et al., 2024a) processes only 576 visual tokens, this number surges to 2,880 in LLaVA-NeXT (Li et al., 2024), with video models like Video-LLaVA (Lin et al., 2024) facing even more massive scales. Given that the computational complexity of the attention mechanism grows quadratically with sequence length, these massive visual sequences constitute a primary bottleneck for inference efficiency.

Vision Token Pruning. To alleviate computational bottlenecks, recent research has explored various paradigms for visual token compression and pruning, which can be broadly categorized into two streams. The first category evaluates token importance using attention maps within Transformer layers. FastV (Chen et al., 2024) pioneers the use of attention mechanisms to identify and discard ineffective visual tokens. Building on this, PyramidDrop (Xing et al., 2025) introduces a hierarchical pyramid structure, enabling progressive sequence compression. Furthermore, SparseVLM (Zhang et al., 2025) leverages key text queries to guide the pruning process. The second category focuses on eliminating inherent redundancy within visual features. VisionZip (Yang et al., 2025) merges spatially redundant tokens based on visual feature similarity. DART (Wen et al., 2025) proposes a de-duplication mechanism based on representative pivots, retaining tokens with unique information. HoloV (Zou et al., 2025) introduces a spatial-aware budget allocation strategy to address background loss. Constrained by attention bias or the lack of textual semantic guidance, existing methods struggle to reconcile critical information retention with complex task adaptation. Instead, we propose a strategy that enhances efficiency while maintaining high semantic fidelity.

3 Motivation and Analysis

In this section, we conduct an empirical analysis to investigate the impact of token importance across layers based on current attention-based pruning paradigms, and reveal the phenomenon of Early Inertial Bias inherent in attention distributions.

Observation I: Inertial Bias Prioritizes Irrelevant Inertial Tokens. To verify the existence of Early Inertial Bias, we tracked the averaged text-to-visual attention scores and rank evolution of the Top-20 tokens from the second layer to subsequent layers. As visualized in Figure 2, we can see that nearly half of the high-ranking tokens from the second layer maintain high ranks, while the rest gradually decrease in rank as the layer deepens. Interestingly, for those tokens whose rank evolves progressively across layers, their importance are influenced by previous layers, exhibiting a form of inertia that prevents abrupt changes. For example, the Top-K token sets significantly overlap between the second and the 10th layer, but not between the second and the 20th layer. Due to this inertial bias, existing attention-based pruning strategies tend to retain visually prominent yet potentially task-irrelevant generic features, leading to an inefficient allocation of the token budget toward Inertial Tokens.

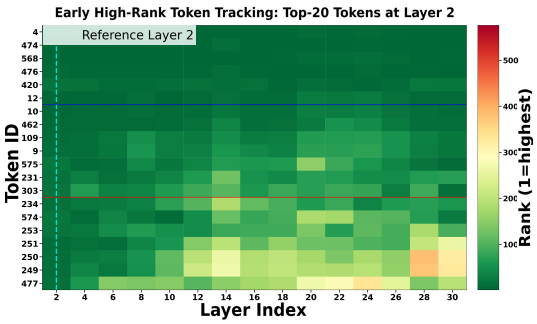


Figure 2: Analysis of Numerical Inertia. Visualization of the attention score and rank evolution of the Top-20 tokens from the second layer, please refer to Fig. 7 for more details.

Observation II: Inertial Bias Masks Critical Emerging Tokens. To assess the reliability of low attention scores in early layers, we compared the key token set from the final layer (32nd layer) against their rank distribution in early layers. As visualized in Figure 3, a rank-value discrepancy is reflected by the statistic that approximately 47% of the final-layer key tokens exhibit low scores in shallow layers. This discrepancy confirms that, due to numerical inertia, many tokens critical for final reasoning are masked in early layers, resulting in low scores. This suggests that the limita-

tion of the current static pruning perspective may lead to the information loss of these tokens with evolutionary potential.

This is also confirmed by the qualitative visualization of cross-layer attention (Figure 1). Tracking the evolution from the second to the 12th layer clearly reveals the coexistence of two distinct behavioral patterns: regions associated with generic backgrounds often maintain high attention scores due to inertia, whereas critical regions pointing to the answer are not significantly activated in early layers but gradually accumulate importance as network depth increases.

Remark. The discrepancy in layer-wise behaviors reveals the limitation of existing strategies: an evaluation based solely on a single-layer static perspective struggles to differentiate between Inertial Tokens and Emerging Tokens. This suggests that the core of the evaluation of token importance should not rely solely on static observations of a specific layer but incorporate dynamic evolutionary trends relative to the final objective of the task. This insight prompts us to extend our evaluation perspective to both intra-layer statistics and the inter-layer evolution. Consequently, in Section 4, we will theoretically derive a pruning metric from a task-aware perspective, aiming to identify an effective indicator capable of rectifying inertial bias and capturing dynamic semantics.

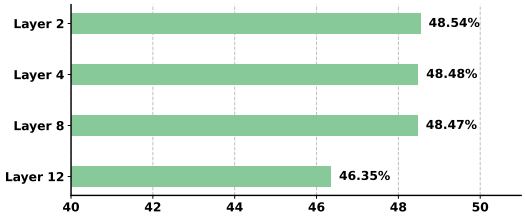


Figure 3: Analysis of Rank Discrepancy. Comparison between the final-layer key tokens and their rank distribution in early layers (2,4,8,12).

4 Theoretical Analysis

4.1 Optimization Objective under Budget Constraint

In a standard MLLM, the visual encoder and text tokenizer project inputs into a unified sequence $\mathbf{X} \in \mathbb{R}^{(N+M)\times D}$, comprising N visual tokens $\mathbf{V}$ and M text tokens $\mathbf{T}$. Since the self-attention mechanism entails $\mathcal{O}((N+M)^2)$ complexity dominated by massive visual tokens, pruning visual tokens becomes essential for inference efficiency.

Focusing on the l-th Transformer layer, let $\mathbf{X}^{(l)} = [\mathbf{V}^{(l)}; \mathbf{T}^{(l)}]$ denote the multimodal input.

Since our pruning target is restricted to the visual modality, we focus on optimizing the binary mask $\mathbf{m} \in \{0,1\}^N$ applied to $\mathbf{V}^{(l)}$. Our goal is to find a mask $\mathbf{m}$ with a strict sparsity constraint $\|\mathbf{m}\|_0 = K$ that minimizes the distortion in the final task loss $\mathcal{L}$. Let $\mathcal{F}(\cdot)$ denote the mapping from layer l to the final loss. The optimization objective is formulated as:

$$\min_{\mathbf{m}} \Delta\mathcal{L} = \left|\mathcal{F}([\mathbf{m} \odot \mathbf{V}^{(l)}; \mathbf{T}^{(l)}]) - \mathcal{F}([\mathbf{V}^{(l)}; \mathbf{T}^{(l)}])\right| \quad \text{s.t.} \quad \|\mathbf{m}\|_0 = K. \tag{1}$$

Remark: While formulated as a masking operation for theoretical derivation, in practice, unselected tokens are physically removed to accelerate inference.

4.2 Theoretical Error Decomposition

Direct optimization of Eq. 1 faces two primary challenges: 1) the discrete constraints imposed by the binary mask result in a complex discrete optimization problem; and 2) the exact gradient $\nabla\mathcal{L}$ is inaccessible during the forward pass. Consequently, we derive a tractable, token-wise upper bound for Eq. 1 to serve as a surrogate objective.

Derivation. Let the pruning perturbation for the i-th token be $\Delta\mathbf{v}_i = -(1-m_i)\mathbf{v}_i$. With the assumption that $\mathcal{F}$ is locally $L_{\mathcal{F}}$-smooth (Nesterov, 2004), we apply the Descent Lemma for Eq. 1 and get the upper bound:

$$\Delta\mathcal{L} \leq |\langle \nabla_{\mathbf{V}^{(l)}}\mathcal{F}, \Delta\mathbf{V}\rangle| + \frac{L_{\mathcal{F}}}{2}\|\Delta\mathbf{V}\|_F^2, \tag{2}$$

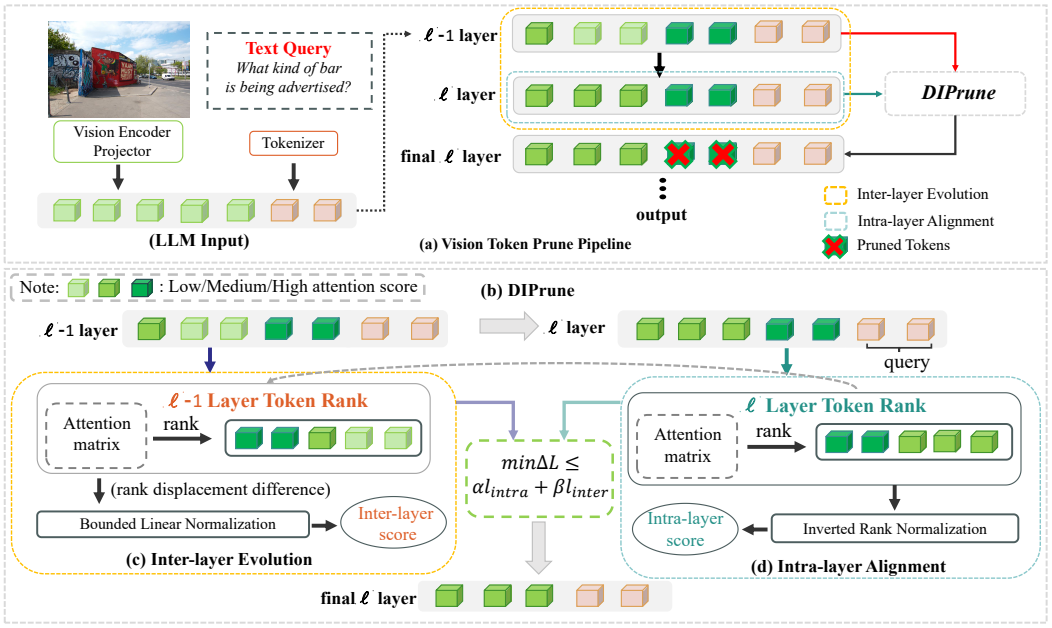


Figure 4: Overview of the DIPrune Framework. (a) The pipeline illustrates the multimodal encoding process (left) and the pruning operation at specific layers (right). (b-d) The DIPrune Mechanism: A dual importance scoring design that captures (c) Inter-layer Dynamic Semantic Evolution and evaluates (d) Intra-layer Static Feature Saliency. The two scores are jointly used to select the optimal top-K tokens.

where $\Delta\mathbf{V} = (\mathbf{m} - \mathbf{1}) \odot \mathbf{V}^{(l)}$ denotes the aggregate perturbation matrix, $\nabla_{\mathbf{V}^{(l)}}\mathcal{F}$ denotes the gradient matrix, and $L_{\mathcal{F}}$ bounds the Hessian spectral norm. To derive a selection criterion for individual tokens, we further exploit the linearity of the inner product and the additivity of the Frobenius norm to decompose the upper bound at the level of individual tokens. Let $\mathbf{g}_i = \nabla_{\mathbf{v}_i}\mathcal{F}$ denote the gradient w.r.t. token $\mathbf{v}_i$. The error bound can be written as:

$$\Delta\mathcal{L} \leq \sum_{i=1}^{N}(1 - m_i)\left(|\langle \mathbf{g}_i, \mathbf{v}_i\rangle| + \frac{L_{\mathcal{F}}}{2}\|\mathbf{v}_i\|^2\right). \tag{3}$$

Based on the Cauchy-Schwarz inequality, we bound the interaction term by the product of the gradient and perturbation norms, thus removing any dependence on their relative orientation. Moreover, we employ Young's Inequality with a positive coefficient μ to decouple these coupled factors into independent additive constraints. The bound is relaxed to:

$$\Delta\mathcal{L} \leq \sum_{i=1}^{N}(1 - m_i)\left(\underbrace{\frac{1}{2\mu}\|\mathbf{g}_i\|^2}_{\mathcal{I}_{\text{Inter}}} + \underbrace{\frac{\mu + L_{\mathcal{F}}}{2}\|\mathbf{v}_i\|^2}_{\mathcal{I}_{\text{Intra}}}\right). \tag{4}$$

To render the tractable global gradient energy $\|\mathbf{g}_i\|^2$ in $\mathcal{I}_{\text{Inter}}$ computable during inference, we utilize a local truncated horizon δ to approximate the inter-layer dependencies. By the chain rule, the gradient flow can be approximated as $\mathbf{g}_i^{(l)} \approx (\frac{\partial \mathbf{V}^{(l+\delta)}}{\partial \mathbf{v}_i^{(l)}})^\top \mathbf{g}^{(l+\delta)}$. Here, the magnitude of the Jacobian $\|\frac{\partial \mathbf{V}^{(l+\delta)}}{\partial \mathbf{v}_i^{(l)}}\|$ quantifies the token's influence on the evolution of intermediate representations.

Eq. 4 effectively transforms the combinatorial global minimization problem into a tractable local selection task. Under the sparsity budget K, minimizing the error upper bound is achieved by retaining the K tokens with the largest combined importance scores $w \triangleq \mathcal{I}_{\text{Inter}} + \mathcal{I}_{\text{Intra}}$. Notably, this decomposition inherently reveals two distinct importances, serving as the theoretical foundation for our dual importance framework:

- **Inter-layer Importance ($\mathcal{I}_{\text{Inter}}$):** This term is governed by the per-token gradient energy $\|\mathbf{g}_i\|^2$. Deriving from the sensitivity of the global loss to the current token through subsequent layers, it inherently captures the inter-layer dependencies.
- **Intra-layer Importance ($\mathcal{I}_{\text{Intra}}$):** This term is determined by the per-token feature magnitude $\|\mathbf{v}_i\|^2$. As it depends solely on the information content of the removed features within the current layer, it represents the intra-layer information.

This theoretical decomposition forms the basis of the design principles underlying our framework, which we describe in detail in the next section.

5 DIPRUNE FRAMEWORK

Inspired by Eq. 4, which describes the pruning error bound in terms of two distinct importance, we introduce DIPrune, a rank-based framework as illustrated in Figure 4. To minimize the error bound, we formulate the token importance scoring function of l-th layer $S^{(l)}_{\text{DIPrune}}$ as a weighted fusion:

$$S^{(l)}_{\text{DIPrune}}(\mathbf{v}_i) = \underbrace{\alpha \cdot S^{(l)}_{\text{align}}(\mathbf{v}_i)}_{\mathcal{I}_{\text{Intra}}} + \underbrace{\beta \cdot S^{(l)}_{\text{evol}}(\mathbf{v}_i)}_{\mathcal{I}_{\text{Inter}}}, \tag{5}$$

where $S^{(l)}_{\text{align}}$ and $S^{(l)}_{\text{evol}}$ are the scores for feature-salient importance (i.e., Semantic Alignment Score) and dynamic evolution importance (i.e., Semantic Evolution Score), respectively, and α, β are hyperparameters with $\alpha + \beta = 1$, balancing the trade-off between minimizing static information loss and preserving dynamic evolution potential. By retaining the Top-K tokens with the highest $S^{(l)}_{\text{DIPrune}}$, we effectively minimize the theoretical error upper bound derived in Section 4.2.

5.1 SEMANTIC ALIGNMENT SCORE

To mitigate the intra-layer information loss ($\mathcal{I}_{\text{Intra}}$), we prioritize tokens with high feature saliency. We utilize the cross-modal attention as a direct proxy for this saliency. Specifically, we define the saliency of visual tokens based on the attention scores where text tokens act as Queries ($\mathbf{Q}_T$) and visual tokens as Keys ($\mathbf{K}_V$). Let $\mathbf{A} \in \mathbb{R}^{M \times N}$ be the cross-modal attention matrix averaged across all heads. The raw importance a_i for each visual token $\mathbf{v}_i$ is computed by aggregating attention weights across all text queries:

$$a_i = \frac{1}{M} \sum_{j=1}^{M} \mathbf{A}_{j,i}. \tag{6}$$

To ensure scale invariance across layers, we employ Inverted Rank Normalization on $\{a_i\}_{i=1}^{N}$. Defining $\mathcal{R}^{(l)}(\mathbf{v}_i) \in \{1, \ldots, N\}$ as the rank of token $\mathbf{v}_i$ based on a_i, the Semantic Alignment Score is formulated as:

$$S^{(l)}_{\text{align}}(\mathbf{v}_i) = \frac{N - \mathcal{R}^{(l)}(\mathbf{v}_i) + 1}{N}. \tag{7}$$

This metric maps importance to the $(0, 1]$ interval, providing a robust measure of static alignment independent of absolute attention magnitudes.

5.2 SEMANTIC EVOLUTION SCORE

To capture the Dynamic Semantic Evolution ($\mathcal{I}_{\text{Inter}}$), we focus on tokens exhibiting significant semantic shifts across layers. Instead of relying on noisy feature distances, we quantify the inter-layer rank displacement to capture the trajectory of importance. Defining the rank displacement over a step δ as $\Delta\mathcal{R}^{(l)}_i \triangleq \mathcal{R}^{(l-\delta)}(\mathbf{v}_i) - \mathcal{R}^{(l)}(\mathbf{v}_i)$, we calculate the Semantic Evolution Score via a Bounded Linear Normalization:

$$S^{(l)}_{\text{evol}}(\mathbf{v}_i) = \frac{1}{2}\left(\frac{\text{Clamp}(\Delta\mathcal{R}^{(l)}_i, -M_{\max}, M_{\max})}{M_{\max}} + 1\right). \tag{8}$$

By clamping the rank displacement within $[-M_{\max}, M_{\max}]$, this metric identifies tokens with rapidly rising priority while suppressing spurious fluctuations; the unselected tokens and their KV caches are then removed, reducing attention cost to $\mathcal{O}((K+M)^2)$ (see Appendix A.3).

Table 1: Performance Comparison on LLaVA-v1.5-7B across Diverse Benchmarks. Results are reported under different pruning ratios. Gray shading denotes our method, and **bold** indicates the best results. Gray-text rows (†) apply baselines at the same pruning layer as DIPrune (L6/L10); DIPrune(Ours) prunes at L10 by default.

Methods	GQA	MMB	MMB_{CN}	MME	POPE	SQA	VQA_{v2}	VQA_{Text}	Average
Upper Bound, 576 Tokens	61.9	64.7	58.1	1862	85.9	69.5	78.4	58.2	100%
LLaVA-1.5 7B	**Retain 192 Tokens** (↓ 66.7%)								
FastV (ECCV'24)	52.7	61.2	57.0	1612	64.8	67.3	67.1	52.5	89.4%
PDrop (CVPR'25)	57.1	63.2	56.8	1766	82.3	68.8	75.1	56.1	96.4%
VisionZip (CVPR'25)	59.3	64.5	57.3	1767	86.4	68.9	76.8	57.3	98.6%
SparseVLM (ICML'25)	57.6	62.5	53.7	1721	83.6	69.1	75.6	56.1	96.0%
DART (EMNLP'25)	58.9	63.6	57.0	1856	82.8	**69.8**	76.7	57.4	97.8%
HoloV (NeurIPS'25)	58.7	**65.4**	58.0	1771	85.0	67.7	76.4	55.9	97.6%
DIPrune(Ours)	**61.2**	64.2	**58.7**	**1856**	**85.8**	69.6	**78.0**	**58.5**	**99.9%**
LLaVA-1.5 7B	**Retain 128 Tokens** (↓ 77.8%)								
FastV (ECCV'24)	49.6	56.1	56.4	1490	59.6	60.2	61.8	51.3	83.8%
PDrop (CVPR'25)	56.0	61.1	56.6	1644	82.3	68.3	72.9	55.1	94.9%
VisionZip (CVPR'25)	57.6	63.4	56.7	1768	84.7	68.8	75.6	56.8	97.2%
SparseVLM (ICML'25)	56.0	60.0	51.1	1696	80.5	67.1	73.8	54.9	92.8%
DART (EMNLP'25)	57.9	63.2	57.0	1845	80.1	69.1	75.9	56.4	96.5%
HoloV (NeurIPS'25)	57.6	63.9	56.9	1771	82.2	69.3	75.4	55.6	96.5%
DIPrune(Ours)	**60.5**	**64.1**	**58.9**	**1857**	**86.2**	**69.4**	**77.2**	**58.0**	**99.5%**
LLaVA-1.5 7B	**Retain 64 Tokens** (↓ 88.9%)								
FastV (ECCV'24)	46.1	48.0	52.7	1256	48.0	51.1	55.0	47.8	74.4%
PDrop (CVPR'25)	41.9	33.3	50.5	1092	55.9	68.6	69.2	45.9	76.7%
VisionZip (CVPR'25)	55.1	60.1	55.4	1690	77.0	69.0	72.4	55.5	93.4%
SparseVLM (ICML'25)	52.7	56.2	46.1	1505	75.1	62.2	68.2	51.8	86.3%
DART (EMNLP'25)	55.9	60.6	53.2	1765	73.9	**69.8**	72.4	54.4	92.6%
HoloV (NeurIPS'25)	55.1	**63.3**	55.1	1703	76.9	69.2	72.7	54.9	93.7%
$\text{VisionZip}^{\dagger}_{\text{L6}}$	56.0	61.2	55.2	1729	78.2	69.1	73.2	55.0	93.9%
$\text{HoloV}^{\dagger}_{\text{L6}}$	56.1	61.9	56.2	1717	75.6	69.1	73.1	56.1	94.0%
$\text{DIPrune}_{\text{L6}}$	56.8	61.8	55.3	1744	81.1	67.7	73.3	55.3	94.6%
$\text{VisionZip}^{\dagger}_{\text{L10}}$	57.3	62.3	55.8	1759	81.8	68.3	73.9	55.8	95.4%
$\text{HoloV}^{\dagger}_{\text{L10}}$	57.5	62.5	56.4	1729	77.4	69.5	73.9	56.6	95.2%
DIPrune(Ours)	**58.9**	62.9	**56.7**	**1807**	**86.2**	69.4	**75.0**	**57.2**	**97.7%**

Table 2: Performance Comparison on LLaVA-NeXT-7B across Diverse Benchmarks, retaining only 320 tokens. Gray shading denotes our method, and **bold** indicates the best results.

Methods	GQA	MMB	MMB_{CN}	MME	POPE	SQA	VQA_{v2}	VQA_{Text}	Average
Upper Bound, 2880 Tokens	64.2	67.4	60.6	1851	86.5	70.1	81.8	64.9	100%
LLaVA-NeXT 7B	**Retain 320 Tokens** (↓ 88.9%)								
FastV (ECCV'24)	55.9	61.6	51.9	1661	71.7	62.8	71.9	55.7	87.2%
PDrop (CVPR'25)	56.4	63.4	56.2	1663	77.6	67.5	73.5	54.4	90.6%
MustDrop (2024.11)	57.3	62.8	55.1	1641	82.1	68.0	73.7	**59.9**	92.5%
FasterVLM (arXiv'24)	56.9	61.6	53.5	1701	83.6	66.5	74.0	56.5	91.1%
SparseVLM (ICML'25)	56.1	60.6	54.5	1533	82.4	66.1	71.5	58.4	90.6%
DART (EMNLP'25)	61.7	65.3	58.2	1710	84.1	68.4	79.1	58.7	95.9%
HoloV (NeurIPS'25)	61.7	65.3	57.5	1738	83.9	68.9	79.5	58.7	95.8%
DIPrune (ours)	**63.1**	**67.9**	**60.4**	**1788**	**87.7**	**69.3**	**80.3**	58.7	**98.2%**

6 EXPERIMENTS

6.1 SETUP

Models. We evaluate DIPrune across diverse and representative MLLMs to ensure its generalizability and scalability under diverse architectural settings. Specifically, we adopt LLaVA-v1.5 (7B/13B) as the standardized multimodal benchmark backbone. We further include the high-resolution variant

LLaVA-NeXT to examine performance under extremely dense visual token conditions. To verify cross-architecture universality and temporal adaptability, we extend our evaluation to Qwen2.5-VL, Video-LLaVA, and the latest Qwen3-VL series models, respectively. Due to space constraints, detailed descriptions of the evaluation benchmarks and specific implementation details are provided in the Appendix A.1 and A.2.

6.2 Main Results

As reported in Tab. 1, DIPrune achieves the highest average performance across all budgets on LLaVA-v1.5-7B. It retains 99.9% of the unpruned performance at 192 tokens (↓66.7%) and exceeds the best baseline by 2.3% at 128 tokens (↓77.8%). Even at 64 tokens (↓88.9%), it retains 97.7%, comparable to HoloV with 192 tokens (97.6%). Under layer-aligned settings (gray rows), DIPrune maintains consistent advantages at both L6 and L10, with the margin widening at the deeper layer, consistent with the inter-layer evolution signal becoming more informative with depth.

6.3 Generalization and Robustness

While the results on LLaVA-1.5-7B validate the effectiveness of DIPrune, we further conduct a systematic evaluation under more challenging settings to assess its generalization across model scales, visual sequence lengths, architectures, and modalities. Specifically, we examine high-resolution inputs on LLaVA-NeXT-7B and cross-architecture transfer on the Qwen-VL series below, while results on model scaling (LLaVA-1.5-13B), video understanding (Video-LLaVA-7B), visual grounding, and generative tasks are deferred to the Appendix B.1.

High-Resolution Adaptation: LLaVA-NeXT-7B. To evaluate DIPrune under ultra-long visual token streams, we test it on LLaVA-NeXT-7B with an extreme compression setting (↓88.9%). As shown in Table 2, DIPrune achieves 98.2% average performance retention, substantially outperforming FastV (87.2%, +11.0%) and SparseVLM (90.6%, +7.6%). These results demonstrate strong sequence-length robustness: DIPrune remains highly effective even with thousands of visual tokens under stringent budgets.

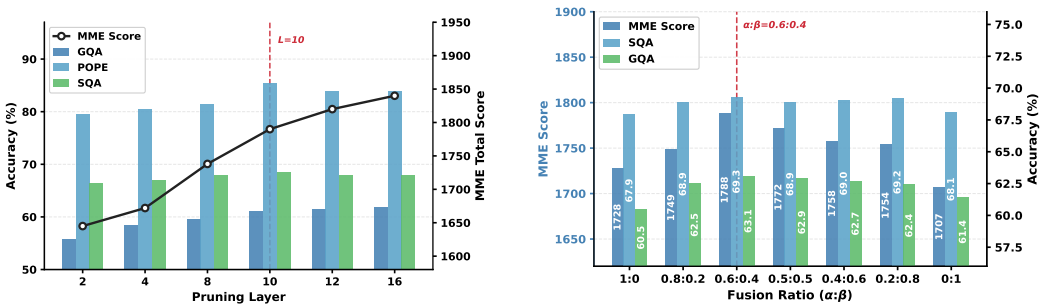
Table 3: Performance Comparison on Qwen2.5-VL-7B.

Methods	MMB	MME	POPE	SQA	Avg.
Upper Bound	82.8	2304	86.1	84.7	100%
Token Pruning Rate = 66.7%					
FastV	75.7	2072	82.2	78.5	92.6%
HoloV	78.3	2093	85.0	79.8	94.6%
DIPrune (Ours)	**81.4**	**2193**	**87.5**	**80.8**	**97.6%**
Token Pruning Rate = 77.8%					
FastV	74.9	2036	80.7	78.0	91.2%
HoloV	76.5	2043	82.3	79.8	92.7%
DIPrune (Ours)	**80.2**	**2114**	**85.4**	**80.5**	**95.7%**
Token Pruning Rate = 88.9%					
FastV	69.2	1940	78.6	77.4	87.6%
HoloV	72.4	2006	80.7	79.5	90.5%
DIPrune (Ours)	**76.1**	**2053**	**84.2**	**79.9**	**93.3%**

Cross-Architecture Generalization: Qwen-VL Series. To verify that DIPrune does not rely on LLaVA-specific designs, we apply the same pruning pipeline to Qwen2.5-VL-7B, which differs substantially in its native dynamic-resolution visual encoding and multimodal positional scheme. As shown in Table 3, DIPrune consistently outperforms FastV and HoloV across all token budgets, surpassing HoloV by about 3% in average performance at each pruning ratio (e.g., 93.3% vs. 90.5% at 88.9%), with notable gains on MMBench (+3.7) and POPE (+3.5). Moreover, consistent improvements on Qwen3-VL Dense-8B and MoE-30B-A3B (Appendix B.1) further confirm that DIPrune generalizes across both dense and MoE architectures.

6.4 Ablation Studies

Effectiveness of Pruning Layers. The pruning depth dictates the trade-off between semantic preservation and computational efficiency. We evaluate DIPrune by varying the pruning layer $L \in \{2, 4, 8, 10, 12, 16\}$. As illustrated in Figure 5a, performance consistently improves as the pruning stage moves deeper. Specifically, early-stage pruning (e.g., $L \leq 4$) results in substantial degradation, as multimodal embeddings require sufficient layers for stable feature abstraction and cross-modal integration. While deeper layers (e.g., $L \geq 12$) offer marginal gains, they yield smaller computational savings due to the massive visual token count. Consequently, we select the 10-th layer



(a) Impact of Pruning Layer L on LLaVA-NeXT-7B. (b) Sensitivity of Fusion Ratio(α: β).

Figure 5: **Ablation studies** on LLaVA-NeXT-7B.

Table 4: **Efficiency analysis** on LLaVA-1.5-7B (64 tokens, 88.9% pruning). TTFT (ms) and ITT (ms/token) are measured on POPE. (a) Default configurations; (b) baselines pruned at the same layer as DIPrune. Acc: average relative performance.

(a) Default configurations

Method	TTFT↓	ITT↓	Acc↑
Baseline	73.23	16.21	100%
FastV	32.43	14.95	74.4%
SparseVLM	33.43	14.98	86.3%
HoloV	31.83	14.94	93.7%
DIPrune	32.50	14.96	**97.7%**
Overhead	0.24 ms (0.74% of TTFT)		

(b) Layer-aligned comparison

Method	TTFT↓	Speedup↑	Acc↑
VisionZip$^{\dagger}_{L6}$	29.04	2.52×	93.9%
HoloV$^{\dagger}_{L6}$	32.13	2.28×	94.0%
DIPrune$_{L6}$	31.58	2.32×	**94.6%**
VisionZip$^{\dagger}_{L10}$	30.57	2.40×	95.4%
HoloV$^{\dagger}_{L10}$	33.83	2.16×	95.2%
DIPrune$_{L10}$	32.50	2.25×	**97.7%**

as the default pruning layer to strike an optimal balance between semantic integrity and inference throughput. Please refer to Appendix B.2 for more details.

Effectiveness of Dual Importance. On LLaVA-NeXT-7B, we observe optimal performance with the configuration of $\alpha = 0.6$ and $\beta = 0.4$ (Figure 5b). This equilibrium precisely highlights the functional trade-off: an excessively high α ignores the inter-layer semantic dynamics, while an over-reliance on β introduces instability due to the lack of saliency anchoring from the current layer. The consistent optimality of this ratio across diverse architectures confirms the robustness of our design. Furthermore, we conduct additional sensitivity analysis experiments on Qwen2.5-VL-7B, which consistently validate the stability of the dual importance parameter configuration.

6.5 Efficiency Analysis

As shown in Tab. 4(a), DIPrune reduces TTFT by 55.6% while ITT remains nearly identical across pruning methods. At comparable TTFT, it retains notably higher accuracy (97.7% vs. 93.7% of HoloV) with only 0.24 ms overhead (0.74% of TTFT). To control for the pruning position, Tab. 4(b) evaluates VisionZip, HoloV, and DIPrune under identical pruning layers, where DIPrune still leads by +0.6% at L6 and +2.3% at L10, indicating that the gain stems from the token selection criterion (Appendix B.3).

7 Conclusion & Limitations

In this work, we identify the Early Inertial Bias in attention-based pruning for MLLMs and propose DIPrune, a rank-based framework with rigorous theoretical error bound decomposition. By evaluating both intra-layer static alignment and inter-layer dynamic evolution, DIPrune effectively preserves task-critical tokens. We note that in highly cluttered scenes or when text queries are extremely short, the margin of alignment-based pruning may shrink; furthermore, as its importance estimation is grounded in the semantic evolution within the LLM backbone, latency benefits may be constrained in scenarios that favor pre-LLM token compression.

AI USE STATEMENT

Generative AI tools were employed solely to improve the language of this manuscript, including grammar correction, wording refinement, and LaTeX table formatting. They played no role in developing the method, conducting experiments, analyzing data, or forming scientific claims. The authors carried out all technical work independently and have carefully reviewed and verified every part of the paper.

ETHICS STATEMENT

This work proposes a training-free token pruning method to improve the inference efficiency of MLLMs. All experiments use publicly available models and benchmarks under their original licenses, and involve no human subjects or private data. By reducing computational cost, DIPrune may lower the hardware barrier for deploying MLLMs; we do not foresee direct negative societal impacts beyond those of the underlying models.

REPRODUCIBILITY STATEMENT

We provide comprehensive details to facilitate reproduction of our results. The theoretical derivation and the scoring formulation are presented in Sec. 4 and Sec. 5, respectively. The appendix further specifies the benchmarks, implementation details, and default hyperparameters (Appendix A.1 and A.2), the complexity analysis and decoupled attention implementation (Appendix A.3 and A.5), and the efficiency measurement setup (Appendix B.3). We plan to release our code to support future research.

REFERENCES

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Kazi Hasan Ibn Arif, JinYi Yoon, Dimitrios S Nikolopoulos, Hans Vandierendonck, Deepu John, and Bo Ji. Hired: Attention-guided token dropping for efficient inference of high-resolution vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 1773–1781, 2025.

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

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023.

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

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *CoRR*, abs/2312.14238, 2023.

Hongyuan Dong, Jiawen Li, Bohong Wu, Jiacong Wang, Yuan Zhang, and Haoyuan Guo. Benchmarking and improving detail image caption. *arXiv preprint arXiv:2405.19092*, 2024.

Mark Endo, Xiaohan Wang, and Serena Yeung-Levy. Feather the throttle: Revisiting visual token pruning for vision-language model acceleration. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22826–22835. IEEE, 2025.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. MME: A comprehensive evaluation benchmark for multimodal large language models. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6904–6913, 2017.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6693–6702. IEEE, 2019.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. See what you are told: Visual attention sink in large multimodal models. In *International Conference on Learning Representations*, volume 2025, pp. 87676–87703, 2025.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022.

Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-gemini: Mining the potential of multi-modality vision language models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 48(3):3530–3543, 2026.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 292–305, 2023.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 5971–5984, 2024.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 26286–26296. IEEE, 2024a.

Shuo Liu, Kaining Ying, Hao Zhang, Yue Yang, Yuqi Lin, Tianle Zhang, Chuanhao Li, Yu Qiao, Ping Luo, Wenqi Shao, and Kaipeng Zhang. Convbench: A multi-turn conversation evaluation benchmark with hierarchical ablation capability for large vision-language models. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024b.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten

Sattler, and Gül Varol (eds.), *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part VI*, volume 15064 of *Lecture Notes in Computer Science*, pp. 216–233. Springer, 2024c.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024d.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 35:2507–2521, 2022.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq R. Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, volume ACL 2022 of *Findings of ACL*, pp. 2263–2279. Association for Computational Linguistics, 2022.

Yurii E. Nesterov. *Introductory Lectures on Convex Optimization - A Basic Course*, volume 87 of *Applied Optimization*. Springer, 2004. ISBN 978-1-4613-4691-3.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. Object hallucination in image captioning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 4035–4045, 2018.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22857–22867. IEEE, 2025.

Cheng Shi, Yizhou Yu, and Sibei Yang. Vision function layer in multimodal llms. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8309–8318. IEEE, 2019.

Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Iyer, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, Xichen Pan, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

Haicheng Wang, Zhemeng Yu, Gabriele Spadaro, Chen Ju, Victor Quétu, Shuai Xiao, and Enzo Tartaglione. Folder: Accelerating multi-modal large language models with enhanced performance. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23614–23625. IEEE, 2025.

Zichen Wen, Yifeng Gao, Shaobo Wang, Junyuan Zhang, Qintong Zhang, Weijia Li, Conghui He, and Linfeng Zhang. Stop looking for "important tokens" in multimodal language models: Duplication matters more. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9972–9991, 2025.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, and Dahua Lin. Conical visual concentration for efficient large vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 14593–14603. Computer Vision Foundation / IEEE, 2025.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802. IEEE, 2025.

Xubing Ye, Yukang Gan, Yixiao Ge, Xiao-Ping Zhang, and Yansong Tang. Atp-llava: Adaptive token pruning for large vision language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24972–24982. IEEE, 2025.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling (eds.), *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II*, volume 9906 of *Lecture Notes in Computer Science*, pp. 69–85. Springer, 2016.

Ce Zhang, Kaixin Ma, Tianqing Fang, Wenhao Yu, Hongming Zhang, Zhisong Zhang, Haitao Mi, and Dong Yu. Vscan: Rethinking visual token reduction for efficient large vision-language models. *Trans. Mach. Learn. Res.*, 2026.

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. *CoRR*, abs/2412.01818, 2024.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis A. Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. Sparsevlm: Visual token sparsification for efficient vision-language model inference. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, volume 267 of *Proceedings of Machine Learning Research*. PMLR / OpenReview.net, 2025.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024.

Xin Zou, Di Lu, Yizhou Wang, Yibo Yan, Yuanhuiyi Lyu, Xu Zheng, Linfeng Zhang, and Xuming Hu. Don't just chase "highlighted tokens" in mllms: Revisiting visual holistic context retention. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025.

Supplementary Material

This appendix is organized as follows:

- **Appendix A** details the experimental settings, including benchmarks, implementation details, computational complexity, the robustness of the query-window strategy, and the implementation of decoupled attention.
- **Appendix B.1** extends the evaluation to model scaling, video understanding, visual grounding, and the Qwen3-VL series, together with generative tasks and a stratified analysis on query length and scene density.
- **Appendix B.2** presents further ablations, including the empirical validation of proxy scores, hyperparameter sensitivity, pruning-layer selection, and extended token budgets.
- **Appendix B.3** provides a detailed efficiency analysis, including the measurement setup, overhead breakdown, layer-aligned comparison, and end-to-end deployment efficiency.
- **Appendix B.4** visualizes the retained visual tokens under different pruning ratios.
- **Appendix C** discusses the broader social impacts.

A Detailed Experiments Settings

A.1 Benchmarks

To comprehensively assess the impact of DIPrune on the core capabilities of MLLMs, we employ a diverse suite of eight widely recognized public benchmarks. Following standard evaluation protocols (Shang et al., 2025; Singh et al., 2019), we categorize these benchmarks into four distinct competency domains: 1) General Visual Question Answering, evaluated via VQAv2 (Goyal et al., 2017) and GQA (Hudson & Manning, 2019); 2) Fine-grained Perception and Reasoning, assessed using TextVQA (Singh et al., 2019) and ScienceQA (Lu et al., 2022); 3) Hallucination Evaluation, gauged by POPE (Li et al., 2023); and 4) Comprehensive Instruction Following, measured through MME (Fu et al., 2025), MMBench, and MMBench-CN (Liu et al., 2024c). Beyond these benchmarks, we further evaluate DIPrune on (i) visual grounding with the RefCOCO series datasets (Yu et al., 2016) (RefCOCO, RefCOCO+, and RefCOCOg); (ii) video question answering with MSVD-QA and MSRVTT-QA; and (iii) generative and fine-grained tasks, including detailed captioning (DetailCaps (Dong et al., 2024)), open-ended hallucination (CHAIR (Rohrbach et al., 2018)), multi-turn reasoning (ConvBench (Liu et al., 2024b)), OCR (OCRBench (Liu et al., 2024d)), and chart understanding (ChartQA (Masry et al., 2022)).

A.2 Implementation Details

We implement DIPrune across a diverse set of MLLMs to validate its generalizability, including LLaVA-v1.5-7B/13B, LLaVA-NeXT-7B, Video-LLaVA-7B, and the Qwen-VL series (including Qwen2.5-VL-7B, Qwen3-VL-Dense-8B, and Qwen3-VL-MoE-30B), which span the two most representative multimodal architectures. Unless otherwise stated, all experiments strictly follow the official evaluation protocols and default scripts released with each model to ensure reproducibility and fair comparison.

We default to performing one-shot pruning at the 10th layer of the LLM backbone. During the forward pass, we introduce a manual text-to-vision attention calculation (as formulated in Eq. 6). This design ensures that our pruning strategy is fully compatible with high-performance attention kernels such as SDPA and FlashAttention-2, thereby enabling more efficient inference computation without compromising the effectiveness of the strategy. For DIPrune, hyperparameters are set based on comprehensive ablation studies, with default values of $\delta = 1$, $M_{max} = 200$, $\alpha = 0.6$, and $\beta = 0.4$. This configuration yields consistent performance gains across various MLLMs and evaluation benchmarks, demonstrating robust cross-architecture applicability and stability.

A.3 Computational Complexity Analysis

Theoretical FLOPs Reduction. Adopting the evaluation framework from PyramidDrop (Xing et al., 2025), we estimate the theoretical floating-point operations (FLOPs) incurred by the LLM decoding layers when processing visual features. The computational load across K layers is primarily distributed between the causal self-attention blocks and the Feed-Forward Networks (FFN). The aggregate FLOPs for the pre-filling stage are calculated as follows:

$$\text{FLOPs}_{\text{LLM}} = \sum_{k=1}^{K} \left(4n_k d^2 + 2n_k^2 d + 3n_k dm\right), \tag{9}$$

where n_k represents the number of visual tokens retained at layer k, while d and m denote the hidden state dimension and the FFN intermediate dimension, respectively. Crucially, the term $2n_k^2 d$ highlights the quadratic dependency of the attention mechanism on the token count. Therefore, strategically pruning n_k provides a direct pathway to significantly alleviate the inference burden, offering substantial efficiency gains particularly for long-context multimodal inputs.

Computing Budget: Detailed Estimation of DIPrune. To rigorously evaluate the efficiency of our proposed method, we provide a fine-grained estimation of the additional computational overhead introduced by DIPrune. The pruning process operates via three distinct stages:

Lightweight Attention Construction. In this stage, we generate a targeted importance map by computing the interaction between visual keys $\mathbf{K}_{vis}$ and a selected subset of text queries $\mathbf{Q}_{sub}$. To further minimize computational overhead, we implement a text query windowing strategy. Specifically, rather than performing a full-scale attention pass over the entire text sequence, we restrict the computation to a strategic window of the most recent text queries, typically set to $0.8\times$ the original text length (i.e., $L_{win} \approx 0.8L_{total}$). Our empirical validation confirms that this truncated window yields performance virtually identical to the full-text baseline, while offering a tangible reduction in FLOPs (see Appendix A.4 for robustness analysis, and Appendix A.5 for implementation details of decoupled attention analysis).

This operation corresponds to a dense matrix multiplication between projections of shape $\mathbb{R}^{N_v \times D}$ and $\mathbb{R}^{D \times L_{win}}$. Consequently, the FLOPs for this tailored construction are reduced to:

$$\text{FLOPs}_{\text{attn}} \approx 2 \cdot N_v \cdot L_{win} \cdot D \tag{10}$$

where N_v denotes the number of visual tokens and D is the hidden state dimension. This formulation ensures that the overhead remains minimal and strictly proportional to the reduced query window size. Crucially, this decoupled design enables full compatibility with hardware-optimized attention kernels such as FlashAttention-2 and SDPA, which typically do not expose intermediate attention weights. By isolating the scoring computation, DIPrune retains the high throughput benefits of these efficient kernels for the primary forward pass.

Dual Importance Scoring. This stage transforms raw attention values into the final token importance score $S_{DIPrune}^{(l)}$. The computation consists of three sub-steps: (1) *Score Aggregation*: Averaging attention scores across H heads and summing over L_{win} queries involves approximately $N_v \cdot L_{win} \cdot H$ additions. (2) *Rank Calculation*: Deriving the rank $\mathcal{R}$ for N_v tokens requires sorting, with a complexity governed by $N_v \log N_v$. (3) *Displacement & Fusion*: Calculating the rank displacement ($\Delta\mathcal{R} = \mathcal{R}^{(l-\delta)} - \mathcal{R}^{(l)}$, consistent with Eq. 8) and fusing it with the alignment score ($\alpha S_{align} + \beta S_{evol}$) involves element-wise operations on N_v tokens, totaling approximately $5N_v$ FLOPs. Therefore, the total FLOPs for the scoring stage can be approximated as:

$$\text{FLOPs}_{\text{score}} \approx N_v \cdot (L_{win} \cdot H + \log N_v + 5) \tag{11}$$

Given that N_v (e.g., 576) is relatively small compared to the hidden dimension D (e.g., 4096), this term is computationally negligible compared to the matrix multiplication in the previous stage.

Pruning Execution. The final stage involves selecting the top-k indices and gathering the retained tokens. The selection process on an N_v-dimensional score vector has a linear complexity of $O(N_v)$. While the subsequent token gathering involves memory copy operations, the arithmetic FLOPs are nominally zero. Thus, the cost is bounded by:

$$\text{FLOPs}_{\text{exec}} \approx N_v \tag{12}$$

Summary of Overhead. Aggregating the components above, the total computational overhead introduced by DIPrune at the pruning layer is dominated by the lightweight attention construction:

$$\begin{aligned} \text{FLOPs}_{\text{total}} &= \text{FLOPs}_{\text{attn}} + \text{FLOPs}_{\text{score}} + \text{FLOPs}_{\text{exec}} \\ &\approx 2 \cdot N_v \cdot L_{win} \cdot D + O(N_v \cdot H) \\ &\approx 2 \cdot N_v \cdot L_{win} \cdot D \end{aligned} \tag{13}$$

This derivation demonstrates that the overhead of DIPrune scales linearly with the visual token count N_v and the query window size L_{win}, rather than the quadratic complexity $O(N_v^2)$ associated with full self-attention. This efficiency ensures that DIPrune imposes minimal latency while enabling substantial acceleration in subsequent layers.

A.4 Robustness of Query Window Strategy

To further validate the claim that the text query windowing strategy achieves performance comparable to full-text attention, we evaluate DIPrune across query-window ratios ranging from 0.3 to 1.0 on LLaVA-v1.5-7B. As shown in Table 5, performance saturates around a ratio of 0.8, which is therefore adopted as our default setting.

Table 5: Performance under different query-window ratios on LLaVA-v1.5-7B.

Window Ratio	0.3	0.5	0.7	**0.8**	0.9	1.0
TextVQA	56.11	56.83	57.11	**57.21**	57.13	57.19
GQA	57.74	57.74	58.01	**58.88**	58.68	58.72

Furthermore, comparing the default ratio ($L_{win}/L_{total} = 0.8$) against the full-attention baseline ($L_{win}/L_{total} = 1.0$) across different prompt lengths on TextVQA, the performance gap is -0.15 / $+0.44$ / -0.18 for short, medium, and long prompts respectively, with an overall difference of $+0.06$. These results confirm that the windowing strategy maintains robust performance across a wide range of ratios and prompt lengths, validating its use as an efficient approximation without sacrificing accuracy.

A.5 Implementation Details of Decoupled Attention

In this section, we provide a detailed description of the implementation specifications of the decoupled attention computation.

Detailed Implementation. At the pruning layer l, the hidden states are projected onto queries and keys. We extract the visual key block $K_{\text{vis}} \in \mathbb{R}^{N_v \times D}$ and the recent text query window $Q_{\text{sub}} \in \mathbb{R}^{L_{\text{win}} \times D}$, and compute per-head attention as follows:

$$A^{(h)} = \text{softmax}\left(\frac{Q_{\text{sub}}^{(h)} \left(K_{\text{vis}}^{(h)}\right)^\top}{\sqrt{d_h}} \right), \quad h = 1, \ldots, H \tag{14}$$

where $d_h = D/H$ is the per-head dimension. The per-head attention maps are then aggregated via mean pooling across all H heads to obtain a unified attention map:

$$\bar{A} = \frac{1}{H} \sum_{h=1}^{H} A^{(h)} \in \mathbb{R}^{L_{\text{win}} \times N_v} \tag{15}$$

The raw saliency score for each visual token i is obtained by averaging over all query positions in the window:

$$a_i = \frac{1}{L_{\text{win}}} \sum_{j=1}^{L_{\text{win}}} \bar{A}[j, i], \quad i = 1, \ldots, N_v \tag{16}$$

This yields a scalar importance score a_i for each visual token, reflecting its overall relevance to the text queries within the current window. The final dual importance score is then computed following Eq. 7 and Eq. 8 in the main paper, identical to the standard DIPrune scoring pipeline.

Table 6: Performance Comparison on LLaVA-v1.5-13B across Diverse Benchmarks. Results are reported under different pruning ratios. Gray shading denotes our method, and **bold** indicates the best results.

Methods	GQA	MMB	MMB_{CN}	MME	POPE	SQA	VQA_{Text}	Average
Upper Bound, 576 Tokens	63.3	68.9	62.3	1818	85.9	72.8	61.3	100%
LLaVA-1.5 13B	**Retain 192 Tokens** ($\downarrow 66.7\%$)							
FastV (ECCV'24)	59.1	54.0	51.2	1641	82.3	56.4	51.6	86.0%
VisionZip (CVPR'25)	59.1	66.9	-	1754	85.1	73.5	59.5	97.3%
SparseVLM (ICML'25)	58.7	67.4	61.0	1768	82.2	73.1	55.4	96.0%
DART (EMNLP'25)	62.1	68.2	61.4	**1855**	84.0	73.6	60.2	99.3%
DIPrune(Ours)	**62.7**	**68.4**	**63.4**	1817	**87.0**	72.8	**61.1**	**100.1%**
LLaVA-1.5 13B	**Retain 128 Tokens** ($\downarrow 77.8\%$)							
FastV (ECCV'24)	57.7	57.9	48.8	1673	79.3	57.0	56.0	86.8%
VisionZip (CVPR'25)	57.9	66.7	-	1743	85.2	74.0	58.7	96.8%
SparseVLM (ICML'25)	57.9	65.8	55.8	1774	81.1	69.9	49.9	92.3%
DART (EMNLP'25)	60.9	67.4	60.7	**1839**	81.8	**74.3**	59.0	98.0%
DIPrune(Ours)	**62.3**	**67.5**	**63.8**	1815	**87.0**	72.7	**60.8**	**99.9%**
LLaVA-1.5 13B	**Retain 64 Tokens** ($\downarrow 88.9\%$)							
FastV (ECCV'24)	53.7	50.9	42.1	1567	69.3	56.8	47.1	78.3%
VisionZip (CVPR'25)	56.2	64.9	-	1676	76.0	**74.4**	57.4	93.3%
DivPrune (CVPR'25)	57.9	64.3	-	1779	84.7	72.4	58.7	96.1%
SparseVLM (ICML'25)	50.6	61.3	54.8	1502	65.0	69.0	47.2	83.9%
DART (EMNLP'25)	57.1	65.4	59.3	1722	75.4	74.1	55.9	96.1%
DIPrune(Ours)	**60.5**	**66.8**	**63.5**	**1795**	**87.3**	72.9	**59.5**	**98.9%**

B Additional Experiments and Analysis

B.1 More Experimental Validation on Generalization and Scalability

Model Scaling to Larger Models: LLaVA-v1.5-13B. To assess cross-scale generalization, we evaluate DIPrune on LLaVA-v1.5-13B across various pruning intensities. As shown in Table 6, DIPrune demonstrates exceptional resilience on the larger 13B architecture. Specifically, under aggressive compression (128 tokens, $\downarrow 77.8\%$), our method achieves 99.9% average performance retention compared to the unpruned upper bound. This significantly outperforms the 99.5% retention observed on the 7B counterpart under the same budget, delivering consistently positive gains across nearly all competency dimensions. Even under extreme compression conditions (retaining only 64 tokens, $\downarrow 88.9\%$), DIPrune maintains 98.9% performance, substantially leading the best-performing baseline method, DART, by $+2.8\%$. Overall, these results demonstrate the robust cross-scale transferability of DIPrune and support a plausible scale-dynamics synergy interpretation: larger-capacity models may exhibit more pronounced inter-layer attention reallocation, rendering the rank-displacement-based ΔRank trend signal more discriminative. This enables DIPrune to more effectively preserve emergent evidence that is critical for later-stage reasoning under tight token budgets.

Cross-Modality Generalization: Video-LLaVA-7B. We further apply DIPrune to the video understanding model Video-LLaVA-7B to assess its applicability to spatiotemporal token sequences. Compared to static images, video inputs typically exhibit stronger temporal redundancy and more complex dynamic disturbances. As shown in Table 7, experimental results demonstrate that DIPrune substantially reduces the token budget while maintaining superior overall performance, demonstrating strong cross-modality generalization.

Fine-Grained Task Generalization: Qwen2.5-VL-7B. To further verify the efficacy of DIPrune on tasks requiring precise spatial reasoning, we conduct experiments on the challenging visual grounding benchmarks: RefCOCO, RefCOCO+, and RefCOCOg. Unlike general VQA tasks that rely heavily on global semantic abstraction, visual grounding demands the retention of fine-grained spatial features to accurately regress bounding box coordinates for specific textual descriptions. As

Table 7: Video QA Evaluations on Video-LLaVA-7B under 50% Token Retention.

Methods	MSVD-QA		MSRVTT-QA		Average	
	Acc.	Score	Acc.	Score	Acc.	Score
Upper Bound (Video-LLaVA)	70.2	3.9	57.3	3.5	63.8	3.7
Retain 50% Tokens ($\downarrow 50\%$)						
FastV (ECCV'24)	71.0	3.9	55.0	3.5	63.0	3.7
FasterVLM (arXiv'24)	70.5	3.9	56.2	3.5	63.4	3.7
DART (EMNLP'25)	71.0	4.0	56.7	3.6	63.8	3.8
HoloV (NeurIPS'25)	71.0	4.0	56.5	3.6	63.7	3.8
DIPrune (Ours)	**71.3**	**4.0**	**56.9**	**3.6**	**64.1**	**3.8**

illustrated in Table 8, DIPrune consistently achieves superior performance compared to baseline methods. This success highlights a key advantage of our framework: the Semantic Alignment (S_{align}) component effectively anchors tokens corresponding to the textual referring expressions (e.g., "the cat on the left"), while the Semantic Evolution (S_{evol}) component ensures the preservation of tokens critical for subsequent coordinate regression, particularly those that emerge only in deeper layers. This demonstrates that DIPrune does not merely retain salient objects but actively preserves the spatial-semantic integrity necessary for dense localization tasks.

Table 8: Performance Comparisons on Qwen2.5-VL-7B across Three Referring Expression Comprehension Benchmarks. Results are reported under the aggressive pruning ratio (retaining 25% tokens). Gray shading denotes our method, and **bold** indicates the best results.

Method	RefCOCO			RefCOCO+			RefCOCOg		Average
	val	testA	testB	val	testA	testB	val	test	
Upper Bound (Qwen2.5-VL-7B)	89.45	92.56	85.16	83.50	89.02	79.15	86.76	87.24	100%
Qwen2.5-VL-7B	**Retain 25% Tokens** ($\downarrow 75\%$)								
FastV (ECCV'24)	43.57	46.81	40.86	39.47	43.78	36.02	43.04	42.69	48.5%
PyramidDrop (CVPR'25)	46.46	53.83	37.23	42.29	47.76	32.81	45.32	44.91	50.4%
VScan (TMLR)	74.32	79.05	68.22	67.22	73.72	**58.95**	69.42	69.43	80.7%
DIPrune (Ours)	**74.88**	**79.76**	**68.26**	**68.16**	**74.54**	58.80	**73.47**	**73.27**	**82.3%**

Cross-Architecture Scalability: Qwen3-VL Series. To further demonstrate the scalability and architecture-agnostic nature of DIPrune, we extend our evaluation to two additional models from the Qwen3-VL series: Qwen3-VL-8B (Dense) and Qwen3-VL-30B-A3B (MoE). These models represent distinct architectural paradigms—a standard dense transformer and a Mixture-of-Experts (MoE) architecture—and are evaluated against the recent strong baseline HoloV across three pruning ratios. Results are reported on MME, MMBench, and POPE in Table 9.

Table 9: Performance comparison on Qwen3-VL series under different pruning ratios (MME / MMBench / POPE). Gray shading denotes our method, and **bold** indicates the best results.

Model	Method	66.7%	77.8%	88.9%
Qwen3-VL-8B (Dense)	HoloV	2007 / 78.6 / 83.5	1956 / 76.8 / 80.5	1798 / 67.7 / 74.6
	DIPrune (Ours)	**2152 / 81.3 / 88.3**	**2029 / 78.6 / 87.8**	**1918 / 75.8 / 85.9**
Qwen3-VL-30B-A3B (MoE)	HoloV	2076 / 80.4 / 84.5	1941 / 76.3 / 81.2	1724 / 67.1 / 71.7
	DIPrune (Ours)	**2278 / 85.1 / 89.1**	**2089 / 83.8 / 86.9**	**1877 / 78.8 / 75.2**

As shown in Table 9, DIPrune consistently outperforms HoloV across all pruning ratios on both models. Notably, the performance advantage becomes more pronounced under aggressive compression (88.9% pruning ratio), where DIPrune surpasses HoloV by up to **+11.3** on POPE for Qwen3-VL-8B and **+11.7** on MMBench for Qwen3-VL-30B-A3B. Furthermore, the consistent gains observed from the 8B dense model to the 30B-A3B MoE model confirm that DIPrune is not restricted to specific backbone design, and generalizes effectively across both model scales and architectural variants of the latest generation MLLMs.

Generative Task Generalization. We further evaluate DIPrune on detailed captioning (DetailCaps (Dong et al., 2024)), open-ended hallucination (CHAIR (Rohrbach et al., 2018)), multi-turn reasoning (ConvBench (Liu et al., 2024b)), and fine-grained perception (OCRBench (Liu et al., 2024d), ChartQA (Masry et al., 2022)). As shown in Tab. 10, DIPrune consistently achieves superior performance across both models and pruning ratios, with the captioning gain growing steadily as the pruning ratio increases, confirming its effectiveness beyond discriminative tasks.

Table 10: Results on generative and fine-grained benchmarks. Cap.: DetailCaps (CAPTURE); CHAIR: CHAIR_i; Conv.: ConvBench; OCR: OCRBench; Chart: ChartQA (%).

Ratio	Method	LLaVA-1.5-7B					Qwen2.5-VL-7B				
		Cap.↑	CHAIR↓	Conv.↑	OCR↑	Chart↑	Cap.↑	CHAIR↓	Conv.↑	OCR↑	Chart↑
66.7%	FastV	.496	13.45	4.27	300	17.4	.543	8.68	4.31	426	63.4
	HoloV	.489	13.55	4.29	299	15.4	.599	10.13	6.32	481	57.6
	DIPrune	**.501**	**12.91**	**4.32**	**307**	**17.5**	**.613**	**8.66**	**6.40**	**492**	**71.8**
77.8%	FastV	.485	13.20	4.07	287	16.5	.520	**8.23**	4.24	332	57.8
	HoloV	.475	14.00	4.16	291	17.0	.581	10.12	6.21	360	44.7
	DIPrune	**.493**	**11.54**	**4.28**	**293**	**17.5**	**.604**	8.68	**6.25**	**378**	**63.4**
88.9%	FastV	.455	17.51	3.77	236	14.8	.446	9.68	3.80	178	41.8
	HoloV	.453	14.98	4.08	247	16.2	.552	10.36	5.76	218	30.4
	DIPrune	**.482**	**14.79**	**4.16**	**270**	**16.4**	**.582**	**8.12**	**5.82**	**220**	**42.9**

Stratified Analysis on Query Length and Scene Density. To further quantify the limitations discussed in Sec. 7, we stratify GQA by query length and the above CHAIR results by the number of ground-truth instances on LLaVA-1.5-7B (88.9% pruning; Tab. 11). DIPrune leads HoloV across all query lengths, while the margin shrinks for the shortest queries (+2.0% at 3–5 words vs. +4.1% at 6–11 words), consistent with a weaker alignment signal. On CHAIR_i, DIPrune performs best in sparse and dense scenes, whereas HoloV is slightly better in medium-density scenes, indicating that cluttered scenes remain a challenging regime.

Table 11: Stratified analysis on LLaVA-1.5-7B (88.9% pruning). (a) GQA accuracy (%) by query length; (b) CHAIR_i (↓) by the number of ground-truth instances. Bold: best among pruning methods.

(a) Query length

Length	#Samples	Full	FastV	HoloV	DIPrune
3–5	2,071	51.1	41.7	46.2	**48.2**
6–11	8,425	62.9	44.9	56.0	**60.1**
12–25	2,082	68.8	55.3	60.3	**64.7**
All	12,578	61.9	46.1	55.1	**58.9**

(b) Scene density

#Inst.	#Images	Full	FastV	HoloV	DIPrune
1–2	217	14.41	20.83	16.38	**14.91**
3–4	187	13.14	14.09	**12.76**	14.06
≥5	93	14.44	16.64	16.17	**15.98**
All	497	13.94	17.51	14.98	**14.79**

B.2 More Ablation Analysis

Empirical Validation of Proxy Scores. As S_{align} and S_{evol} are tractable proxies of $\mathcal{I}_{\text{Intra}}$ and $\mathcal{I}_{\text{Inter}}$, we validate them on LLaVA-1.5-7B (64 tokens; Tab. 12). (a) Since the attention logit couples projected feature saliency with query-directional alignment, S_{align} acts as a task-conditioned instantiation of $\mathcal{I}_{\text{Intra}}$: replacing it with the raw feature magnitude $\|v_i\|^2$ consistently degrades performance, with or without S_{evol}. (b) Using the gradient energy $G_i = \|\partial \mathcal{L}_{\text{ans}} / \partial h_i^{(10)}\|_2^2$ of the unpruned model

as an offline reference, we report GradMass, i.e., the fraction of $\sum_i G_i$ captured by the retained tokens. S_{evol} alone achieves the highest GradMass (61.8% vs. 16.0–16.5%), supporting it as an effective proxy for $\mathcal{I}_{\text{Inter}}$, while combining both terms yields the best accuracy, confirming their complementarity.

Table 12: Validation of proxy scores on LLaVA-1.5-7B (64 tokens). (a) Intra-layer instantiation; (b) GradMass: fraction of gradient energy captured by retained tokens.

(a) Intra-layer term

Configuration	GQA↑	MME↑
$\|v_i\|^2$ only	53.4	1698
$\|v_i\|^2$ + S_{evol}	55.7	1763
S_{align} only	58.1	1722
DIPrune	**58.9**	**1807**

(b) Inter-layer term

Method	GradMass↑	GQA↑
HoloV	16.5%	55.1
S_{align} only	16.0%	58.1
S_{evol} only	**61.8%**	57.7
DIPrune	59.7%	**58.9**

Impact of Rank Displacement Step δ. We evaluate the rank displacement step $\delta \in \{1, 2, 3, 4\}$ to determine the optimal observation scope. As shown in Table 13, LLaVA-NeXT yield optimal performance at $\delta = 1$ and degrade as the window widens. This observation suggests that critical evolutionary signals are likely derived from adjacent layers, whereas wider windows may introduce irrelevant historical noise. Consequently, we fix $\delta = 1$ as the default setting to maximize capture precision with minimal memory overhead.

Impact of Rank Displacement Clipping Threshold (M_{max}). The clipping threshold M_{max} in Eq. 8 is designed to balance the sensitivity toward semantic emergence against robustness to stochastic fluctuations. As summarized in Table 14, varying $M_{max} \in \{50, 100, 200, 300, 400\}$ on LLaVA-NeXT-7B reveals three distinct behavioral regimes. Initially, in the under-sensitivity regime ($M_{max} \leq 100$), restrictive thresholds constrain performance on fine-grained tasks such as TextVQA (58.3%). The model achieves a balanced trade-off at $M_{max} = 200$, reaching superior performance on TextVQA (58.7%) while maintaining high stability on GQA and SQA. This reflects that the threshold effectively encapsulates the typical magnitude of rank displacement for valid emerging tokens, allowing the DIPrune framework to accurately anchor information that becomes critical only in deeper layers of reasoning. However, further increasing the threshold into the noise interference regime ($M_{max} \geq 300$) leads to performance degradation on reasoning-intensive benchmarks like GQA, even as generalized perception metrics such as MME continue to scale (reaching 1807). We hypothesize that excessively high thresholds inadvertently incorporate non-semantic-related noise. Consequently, we adopt $M_{max} = 200$ as the default configuration to strike a favorable trade-off between capturing semantic emergence and suppressing non-stationary noise.

Table 13: Impact of δ on LLaVA-NeXT-7B. $\delta = 1$ is the default setting.

Metric	Step (δ)			
	1	2	3	4
MME	**1788**	1767	1782	1780
GQA	**63.1**	62.8	62.2	62.1
SQA	**69.3**	69.1	69.2	69.2
POPE	**87.7**	87.3	87.4	87.2

Table 14: Ablation Study on M_{max} on LLaVA-NeXT-7B. $M_{max} = 200$ is the default setting.

Metric	Clipping Threshold (M_{max})				
	50	100	200	300	400
TextVQA	58.3	58.3	58.7	**59.3**	59.3
GQA	63.1	63.1	**63.1**	63.1	62.6
SQA	**69.5**	69.5	69.3	69.1	69.0
MME	1785	1780	1788	1796	**1807**

Extended Sensitivity Analysis on Dual Importance Weights. To evaluate the generalizability of the proposed dual importance scoring mechanism, we extend the sensitivity analysis of hyperparameters α and β to the Qwen2.5-VL-7B architecture. Following the default experimental protocol, we perform one-shot pruning at the 10th transformer layer and evaluate the performance trajectory across a spectrum of fusion ratios $\alpha : \beta$ under the constraint $\alpha + \beta = 1$.

As shown in Table 15, the empirical results exhibit a consistent peak performance at $\alpha = 0.6$ and $\beta = 0.4$. This persistent optimality across diverse architectures suggests that DIPrune effectively

captures architectural invariants within MLLMs. By maintaining a precise equilibrium between dynamic semantic evolving and stable saliency anchoring, our framework minimizes the cumulative approximation error induced by pruning while substantially reducing the tuning overhead for future MLLM deployments.

Table 15: Extended Sensitivity Analysis on Qwen2.5-VL-7B. We evaluate the performance trajectory across a spectrum of fusion ratios $\alpha : \beta$. The default setting ($\alpha = 0.6, \beta = 0.4$) consistently achieves high performance. Underlined ratios correspond to using only S_{align} or S_{evol}.

Metrics	1.0 : 0.0	0.8 : 0.2	**0.6 : 0.4**	0.5 : 0.5	0.4 : 0.6	0.2 : 0.8	0.0 : 1.0
MME	1943	2046	**2193**	2134	2097	2019	1984
MMB	79.3	80.7	**81.4**	81.2	80.9	80.6	78.8
POPE	84.4	85.9	**87.5**	86.4	86.2	85.6	83.7

Analysis on Pruning Layer Selection. The ablation study in Figure 5a demonstrates that the 10th layer serves as the optimal pruning point for LLaVA-NeXT-7B. Here we provide a deeper discussion on whether and how this choice generalizes across models of different depths, visual token counts, and task types.

We do not treat the 10th layer as a universal constant. Rather, the optimal pruning point reflects a fundamental trade-off: the model must accumulate sufficient semantic information for reliable token ranking, while retaining enough remaining layers for pruning to yield meaningful efficiency gains. This trade-off is precisely captured by our layer-wise ablation (Figure 5a): pruning too early degrades accuracy because multimodal representations are not yet mature, whereas pruning too late offers diminishing accuracy improvements at the cost of reduced computational savings.

More generally, the optimal pruning layer depends on three interacting factors:

- **Model depth.** Deeper models naturally develop stable representations at later absolute layers; yet the relative pruning point, expressed as a fraction of total depth, tends to remain consistent across architectures.
- **Visual token count.** Models processing denser visual token sequences (e.g., LLaVA-NeXT with 2880 tokens) benefit more from earlier pruning to amortize the quadratic attention cost, whereas models with fewer tokens can afford to prune slightly later without significant latency overhead.
- **Task type.** Tasks requiring fine-grained spatial reasoning (e.g., visual grounding) may favor deeper pruning to preserve spatially precise token signals, while general VQA tasks are more tolerant of earlier pruning.

In practice, rather than relying on a fixed absolute layer index, we recommend selecting the pruning point based on the relative stage of representation maturity—typically within the early-to-middle portion of the LLM backbone. For a given model, an effective operating point can be efficiently identified by scanning a small set of candidate layers on a lightweight validation subset, which incurs negligible additional cost. This procedure is consistent with standard hyperparameter selection in training-free methods and requires no gradient computation or model fine-tuning.

Performance Across Extended Token Budgets. The performance advantage of DIPrune over competing methods becomes increasingly pronounced as the token budget decreases. This trend stems from its core design: under generous budgets, most informative tokens are commonly retained across methods, leaving limited room for differentiation. Under high compression ratios, however, the ability to effectively identify and preserve tokens with semantic evolutionary potential becomes the decisive factor. By jointly modeling intra-layer semantic alignment and inter-layer semantic evolution, DIPrune demonstrates a more substantial competitive advantage precisely under resource-constrained, extreme compression settings. To further substantiate this, we report results across four additional token budgets on LLaVA-v1.5-7B, presented as TextVQA / GQA / MME in Table 16.

Table 16: Performance comparison across extended token budgets on LLaVA-v1.5-7B (TextVQA / GQA / MME). **Bold** indicates the best results.

Method	432 Tokens	288 Tokens	32 Tokens	16 Tokens
VisionZip	58.08 / 61.57 / 1836	57.92 / 60.32 / 1811	53.03 / 52.15 / 1580	49.76 / 46.95 / 1350
HoloV	52.24 / 55.83 / 1605	54.63 / 57.83 / 1620	53.55 / 52.37 / 1596	50.40 / 48.60 / 1430
DIPrune (Ours)	**58.32 / 62.00 / 1862**	**58.25 / 61.93 / 1828**	**55.96 / 56.20 / 1756**	**53.84 / 54.60 / 1712**

As shown in Table 16, DIPrune achieves the best results across all four token budgets. Notably, the performance gap widens progressively as the budget decreases: under the most aggressive setting (16 tokens), DIPrune outperforms VisionZip by **+4.08** on TextVQA and **+7.65** on GQA, and surpasses HoloV by **+3.44** on TextVQA and **+6.00** on GQA. These results confirm that the joint modeling of intra-layer semantic alignment and inter-layer semantic evolution confers increasing advantages under high compression ratios, where identifying semantically evolving tokens is most critical.

B.3 More Efficiency Analysis

Setup. All efficiency experiments are conducted on a single NVIDIA H800 80GB GPU with FP16 precision, CUDA 12.1, PyTorch 2.1.2, Transformers 4.37.2, and batch size 1. For TTFT/ITT (Tab. 4 and Tab. 17), GPU-side latency is measured with `torch.cuda.Event` over 500 POPE samples (after 5 warm-up samples) with a fixed generation length of 32 tokens. Unless otherwise stated, baselines follow their default pruning configurations.

Cross-Architecture Latency and Overhead. Tab. 17 extends Tab. 4(a) to LLaVA-NeXT-7B and Qwen2.5-VL-7B. DIPrune reduces TTFT by 55.6%/56.9%/35.8% on LLaVA-1.5/LLaVA-NeXT/Qwen2.5-VL while ITT remains nearly identical across pruning methods, and it achieves the highest accuracy on all three models at comparable TTFT. The overhead of DIPrune, comprising proxy scoring (attention aggregation and rank displacement) and selection (score fusion, Top-K selection, and token gathering), accounts for at most 0.74% of TTFT and grows only marginally with the visual token count, consistent with the linear complexity in Appendix A.3.

Table 17: Efficiency across architectures (88.9% pruning ratio). TTFT (ms) and ITT (ms/token) are measured on POPE. *Overhead*: DIPrune's pruning cost, split into proxy scoring and selection, and its ratio to TTFT. –: not reported.

Method	LLaVA-1.5-7B			LLaVA-NeXT-7B			Qwen2.5-VL-7B		
	TTFT↓	ITT↓	Acc↑	TTFT↓	ITT↓	Acc↑	TTFT↓	ITT↓	Acc↑
Baseline	73.23	16.21	100%	102.10	17.44	100%	84.28	16.96	100%
FastV	32.43	14.95	74.4%	43.82	15.43	87.2%	53.78	15.46	87.6%
SparseVLM	33.43	14.98	86.3%	45.25	15.46	90.6%	54.35	15.44	–
HoloV	31.83	14.94	93.7%	43.47	15.44	95.8%	53.52	15.47	90.5%
DIPrune	32.50	14.96	**97.7%**	43.98	15.40	**98.2%**	54.08	15.46	**93.3%**
Scoring / Selection	0.149 / 0.091 ms			0.182 / 0.122 ms			0.163 / 0.102 ms		
Overhead (total)	0.24 ms (0.74%)			0.30 ms (0.69%)			0.27 ms (0.49%)		

Layer-Aligned Comparison. For Tab. 4(b), we apply the selection criteria of VisionZip and HoloV at the same pruning layer as DIPrune (L6 and L10), so that the compared methods share identical per-layer token counts. Moving the pruning layer from L6 to L10 improves DIPrune by +3.1%, compared with +1.5% for VisionZip and +1.2% for HoloV, suggesting that DIPrune benefits more from the richer inter-layer evolution signal at deeper layers, which corroborates the analysis in Sec. 3.

End-to-End Deployment Efficiency. We further report peak memory, total wall-clock time, and average per-sample wall-clock time under a 66.7% pruning ratio (Tab. 18). DIPrune achieves the lowest memory on all benchmarks (up to 22.9% less than SparseVLM), as its decoupled attention computation keeps the main forward pass compatible with efficient kernels (Appendix A.5), while maintaining comparable latency and clearly higher accuracy.

Table 18: End-to-end efficiency on LLaVA-v1.5-7B (66.7% pruning ratio). Mem: peak GPU memory; Time: total wall-clock time; Lat.: average wall-clock time per sample.

(a) GQA

Method	Mem (MiB)	Time (m:s)	Lat. (ms)	Acc (%)
SparseVLM	19,458	11:56	56.9	57.6
PDrop	15,616	11:56	56.9	57.1
DIPrune	**15,174**	11:57	57.0	**61.2**

(b) MME

Method	Mem (MiB)	Time (m:s)	Lat. (ms)	Score (Sum)
SparseVLM	19,458	2:09	54.3	1721
PDrop	15,616	2:07	53.5	1766
DIPrune	**15,174**	**2:02**	**51.4**	**1856**

(c) TextVQA

Method	Mem (MiB)	Time (m:s)	Lat. (ms)	Acc (%)
SparseVLM	20,170	7:06	85.2	56.1
PDrop	15,980	7:16	87.2	56.1
DIPrune	**15,544**	7:10	86.0	**58.5**

B.4 More Sparsification Visualization

We provide detailed visualizations of the retained visual patches under varying pruning ratios (66.7%, 77.8%, and 88.9%) to intuitively demonstrate the effectiveness of DIPrune. As illustrated in Figure 6, the black regions denote discarded visual tokens, whereas the colored regions highlight key semantic areas that are highly aligned with the text description, showcasing how DIPrune strategically retains content with high information density. Compared to FastV, a representative pruning method based on static attention, DIPrune significantly outperforms it in preserving relevant visual cues, even under extremely high pruning ratios (e.g., 88.9%). This superiority is achieved through our dual importance scoring mechanism. Rather than relying on isolated static attention scores, this strategy comprehensively integrates task-aware semantic alignment with dynamic inter-layer evolution. By jointly optimizing Semantic Evolution (S_{evol}) and Semantic Alignment (S_{align}), DIPrune effectively safeguards the critical features required for deep reasoning while filtering out redundant tokens.

C Broader Social Impacts

This paper proposes a task-aware token pruning framework for Multimodal Large Language Models (MLLMs), with the primary goal of advancing the inference efficiency and scalability of large-scale vision-language models. By reducing computational overhead and memory requirements, this work may help lower the hardware barriers for deploying high-resolution MLLMs, potentially making such powerful multimodal capabilities more accessible to a broader research and practitioner community.

We do not anticipate any significant negative societal or ethical impacts arising directly from this work. The methods introduced are general-purpose and focus on improving computational efficiency and model throughput, without targeting sensitive application domains. As with many advances in machine learning, downstream applications should be deployed responsibly in accordance with existing ethical guidelines.

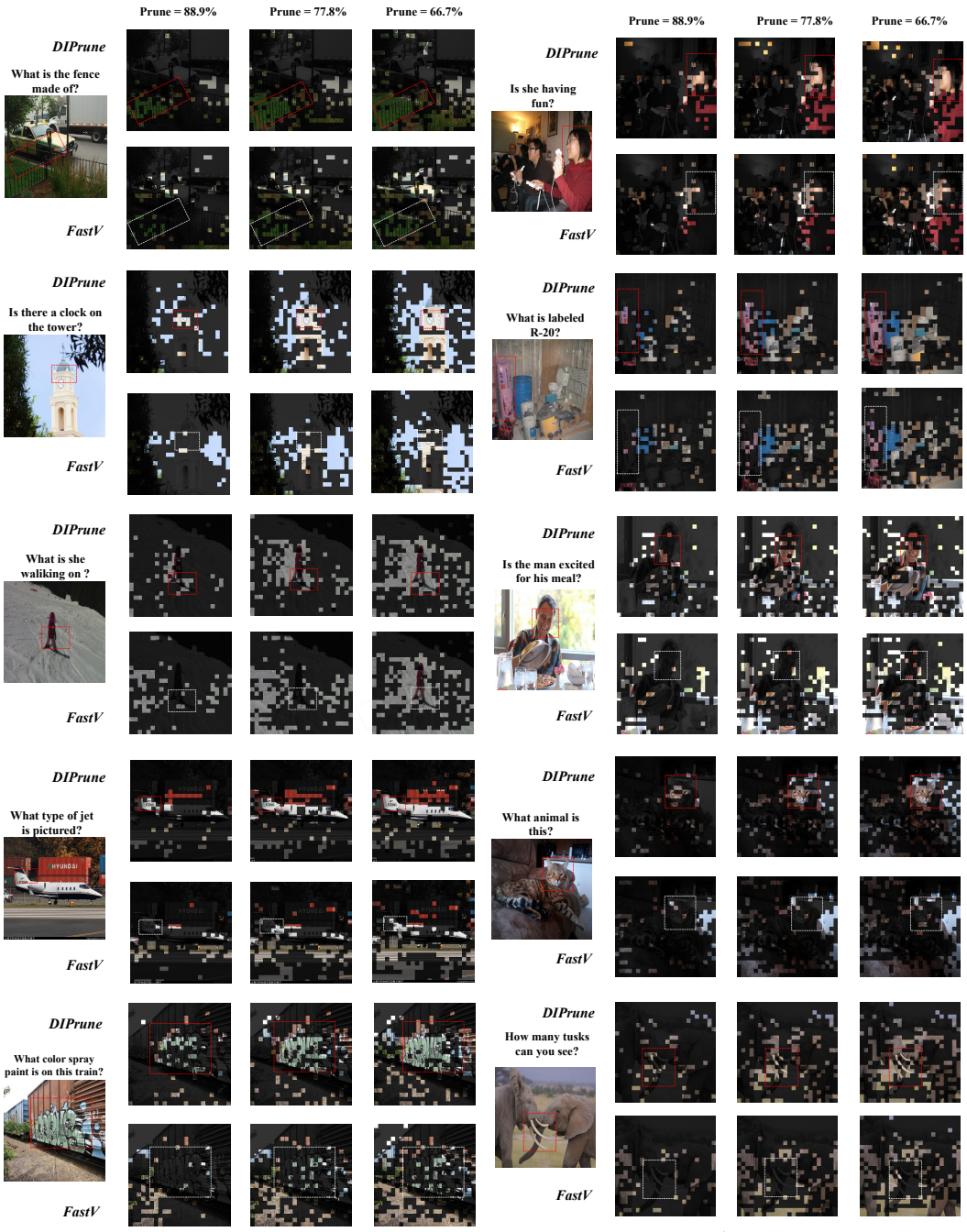


Figure 6: Visualization of visual tokens retained by LLaVA-v1.5-7B on VQAv2 samples. We present comparison results under three progressive pruning ratios (66.7%, 77.8%, and 88.9%). The red bounding boxes in the original images highlight the task-critical visual regions directly relevant to the text query. In the pruning results, the red and white dashed boxes indicate the retention status of these key regions by the DIPrune and FastV strategies, respectively.

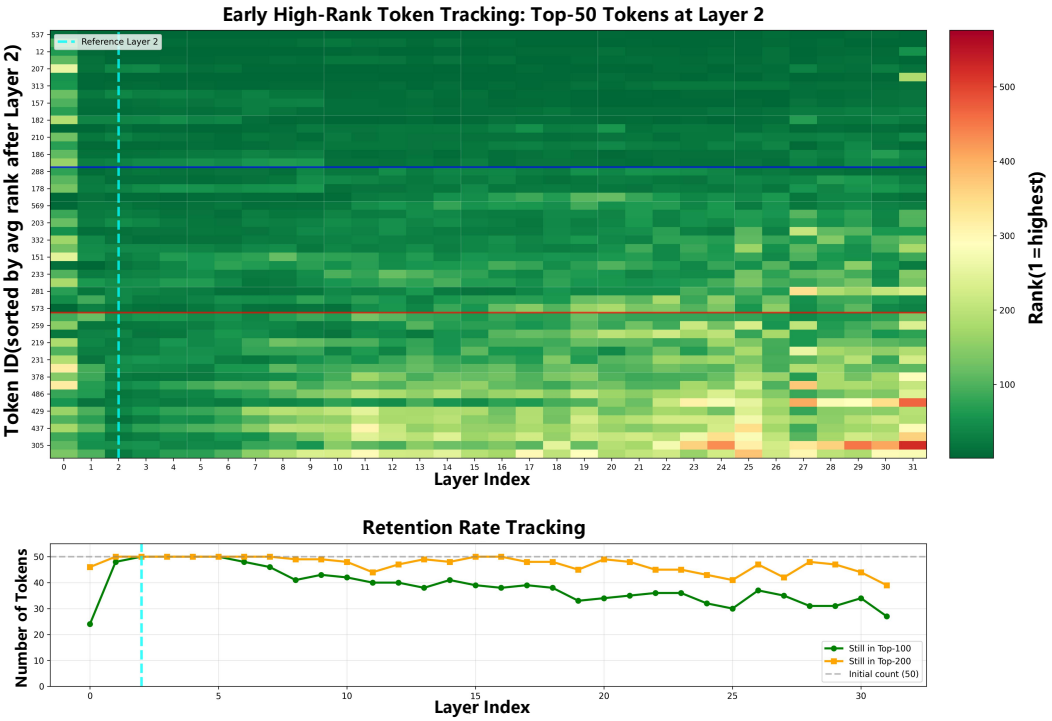


Figure 7: Analysis of Numerical Inertia. Visualization of the attention score and rank evolution of the Top-50 tokens from the second layer. The heatmap shows that high-ranking tokens (green regions) in shallow layers tend to maintain their dominance throughout the network.