PosPrune: Visual Token Pruning with Positional Bias Correction for Efficient Large Vision-Language Models

Ziyang Wang*, Mengwei Li*, Hao Yin, Wenhao Liu, Zilei Wang†

University of Science and Technology of China
{ziyang_w, curry, yinhnavi, wenhaoliu}@mail.ustc.edu.cn, zlwang@ustc.edu.cn

Abstract

Large Vision-Language Models (LVLMs) enhance performance on vision-language tasks by integrating visual features from pre-trained vision encoders into large language models (LLMs). However, the large number of visual tokens introduces significant computational overhead. Existing token pruning methods either perform global selection via [CLS]-based attention in the vision encoder or prune within LLM decoding layers. These approaches face two key challenges: (1) [CLS]-based attention primarily focuses on visually salient regions across the entire image, often overlooking semantically important tokens essential for reasoning; and (2) strong positional bias in the shallow decoder layers causes the model to favor later-positioned tokens, while neglecting earlier ones that may carry critical reasoning cues. To address these issues, we propose PosPrune, a training-free, two-stage visual token pruning framework. At the vision encoder, we introduce an Asymmetric Region-aware Pruning (ARP) strategy that retains more tokens in semantically rich regions while discarding more tokens from semantically less informative regions, thus preserving spatial diversity and task-relevant details. In the LLM decoding stage, we find that the positional bias in shallow layers is primarily driven by model architecture rather than task semantics. Based on this insight, we propose a novel Positional Bias Correction (PBC) mechanism to mitigate this bias. To further reduce redundancy, we apply Maximal Marginal Relevance (MMR) to select tokens that best balance textual relevance and diversity. Extensive experiments on various LVLMs and benchmarks demonstrate the general effectiveness of our approach. Notably, when applied to LLaVA-1.5-7B, PosPrune achieves a reduction of 85% in FLOPs while preserving 98.5% of the original performance.

Introduction

In recent years, Large Vision-Language Models (LVLMs) have emerged as prominent multimodal learning frameworks that effectively integrate visual and textual information (Liu et al. 2024a; Chen et al. 2024b; Zhu et al. 2023; Li et al. 2024b; Bai et al. 2025). These models typically employ a vision encoder to convert images into discrete visual tokens, which are then jointly processed with text tokens by a Large Language Model (LLM) (Touvron et al. 2023; Chiang et al. 2023; Brown et al. 2020). As model capabilities advance, the input length of LVLMs has increased substantially—particularly in scenarios involving high-resolution images, multi-image inputs, or videos—where the number of visual tokens often far exceeds that of textual ones. This massive number of visual tokens imposes significant computational and memory overhead, making it challenging to deploy LVLMs in resource-constrained or latency-sensitive environments (Yao et al. 2025).

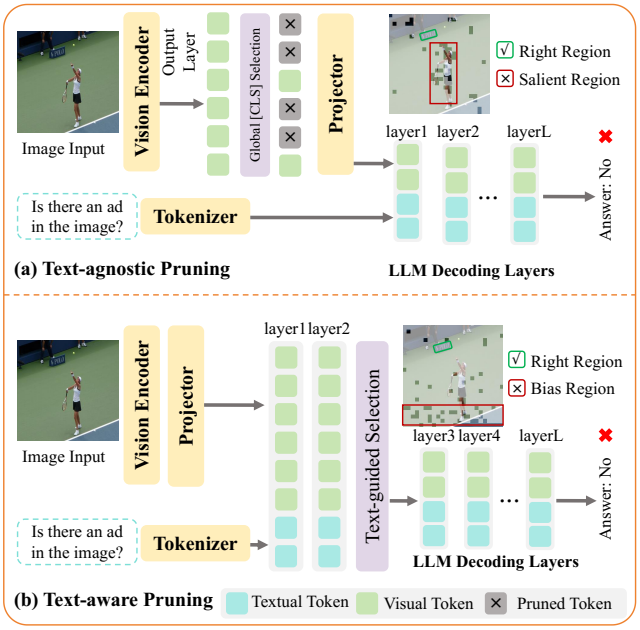


Figure 1: Limitations of existing methods: (a) Using the [CLS] token's attention for global selection tends to overfocus on highly salient image regions. (b) Shallow decoder layers in LLMs exhibit positional bias, favoring visual tokens close to the final instruction token.

Previous work has shown that visual tokens exhibit substantial redundancy (Chen et al. 2024a; Lin et al. 2025). To improve efficiency without sacrificing model performance, recent training-free approaches have explored two main directions for pruning redundant visual tokens. One approach

*These authors contributed equally.
†Corresponding author.

is text-agnostic pruning (Zhang et al. 2024; Yang et al. 2025; Wang et al. 2025), typically conducted at the output of the vision encoder based on token importance—measured via self-attention scores or [CLS]-based attention. The other is text-aware pruning (Yin, Si, and Wang 2025; Chen et al. 2024a; Xing et al. 2024), which is performed within the language model decoder by leveraging cross-modal attention as a criterion to assess token importance.

This work identifies two major limitations of these methods. Firstly, when employing the attention mechanism of the [CLS] token in the vision encoder for global selection over visual tokens, the model tends to focus on highly salient regions of the image (detailed in Fig. 1(a)), while neglecting semantically critical tokens that are essential for vision-language reasoning. Secondly, previous studies (Vaswani et al. 2017; Wen et al. 2025) have shown that the final instruction token tends to attend disproportionately to nearby visual tokens, resulting in a strong positional bias in the shallow layers of the decoder. This bias causes the model to overlook semantically relevant tokens appearing earlier in the sequence (detailed in Fig. 1(b)). Although this positional bias becomes negligible in the deeper layers of the decoder (Zhang et al. 2025), pruning at these stages yields limited improvements in inference efficiency. Therefore, how to accurately and efficiently prune visual tokens based on the attention behaviors in both the vision encoder and the LLM decoder remains a critical and unsolved challenge.

To address these limitations, we propose a novel two-stage visual token pruning method (illustrated in Figure 5) that efficiently identifies essential visual tokens across both the vision encoder and the language decoder. In the vision encoder stage, we observe that the central regions of an image exhibit higher entropy than the edges, suggesting that they contain richer semantic information. To this end, we introduce an Asymmetric Regional Pruning (ARP) strategy, which divides the visual tokens into the $n \times n$ grid and applies lower pruning ratios in the semantically dense central regions and higher pruning ratios in edge regions. This design preserves spatial diversity and local details, which are essential for effective downstream text-guided pruning. In the decoder stage, we find that the positional bias in shallow layers is primarily driven by the inherent model architecture rather than task-specific semantics. Therefore, this architectural bias can be explicitly modeled and corrected during the decoding process. Based on this insight, we design a Position Bias Correction (PBC) mechanism that adjusts shallow-layer attention scores to mitigate positional bias. Finally, we apply the Maximal Marginal Relevance (MMR) strategy to select a subset of visual tokens that balances text relevance and token-level diversity, effectively reducing redundancy while preserving model performance.

We extensively evaluate our approach on eight benchmark datasets using two LVLMs: LLaVA-1.5 (Liu et al. 2024a) and Qwen-2.5-VL (Bai et al. 2025). Our method, PosPrune, achieves state-of-the-art performance with significant gains in efficiency. For example, on LLaVA-1.5-7B, PosPrune reduces FLOPs by 85% while maintaining 98.5% of the original performance and improving inference speed by 1.8×.

Our key contributions are as follows:

- We find that the positional bias present in the shallow layers of the LLM decoder is related to the model architecture rather than to the task semantics.
- We propose PosPrune, a training-free, two-stage visual token pruning framework for accurate and efficient pruning in both vision encoder and LLM decoder stages.
- We validate the generality and effectiveness of PosPrune across 8 image understanding benchmarks, showing substantial improvements in efficiency with minimal performance degradation even under aggressive pruning.

Related Work

Large Vision-Language Models

Large Vision-Language Models (LVLMs) (Li et al. 2023a; Zhu et al. 2023; Liu et al. 2024a; Bai et al. 2023, 2025) have made significant strides in multimodal understanding and generation by integrating pre-trained vision encoders with large language models. Notable examples such as LLaVA (Liu et al. 2024a), MiniGPT-4 (Zhu et al. 2023), and Qwen-VL (Bai et al. 2023) enable tasks like image captioning (Agrawal et al. 2019; Plummer et al. 2015), visual question answering (Hudson and Manning 2019; Lu et al. 2022), and multimodal reasoning (Fu et al. 2024; Yue et al. 2024) through end-to-end training that jointly leverages visual and textual inputs. Recent works (Bai et al. 2025; Li et al. 2024b) further extend LVLMs to handle high-resolution images by increasing the number of visual tokens, enhancing semantic understanding but also increasing computational and memory costs. However, the substantial computational and memory overhead incurred by directly deploying such large-scale LVLMs poses significant challenges in real-world applications, making efficient token pruning strategies essential for balancing resource consumption and model performance.

Visual Token Pruning

To improve inference efficiency, various studies have attempted to prune visual tokens at different stages of LVLMs (Zhang et al. 2024; Yin, Si, and Wang 2025; Xing et al. 2024; Wang et al. 2025). These methods can be categorized as follows: (1) Text-agnostic pruning methods (Wang et al. 2025; Yang et al. 2025; Zhang et al. 2024), which prunes redundant visual tokens during the vision encoding stage. For instance, VisionZip (Yang et al. 2025) selects dominant tokens based on the attention distribution of the [CLS] token, then merges semantically similar tokens from the remaining ones. (2) Text-aware pruning methods (Xing et al. 2024; Yin, Si, and Wang 2025), which uses cross-modal attention in the LLM decoding layers to select tokens based on textual context. For instance, HiMAP (Yin, Si, and Wang 2025) first leverages text-image cross-attention in the shallow layers of the decoder to guide pruning, and then utilizes image-to-image attention in the intermediate layers to further refine the pruning process. (3) Two-stage pruning methods (Liu et al. 2024b; Zhang et al. 2025), which first prune redundant tokens during the vision encoding stage and further refines pruning in the LLM decoding layers. MustDrop (Liu et al. 2024b) combines local spatial merging and dual-attention filtering for progressive pruning. This work

analyzes token redundancy and positional bias in both stages of LVLMs, proposing PosPrune, a two-stage framework that reduces computational overhead while maintaining strong task performance.

Empirical Analysis

Rethinking Token Pruning in the Vision Encoder

Using the attention scores of the [CLS] token globally for token pruning in the vision encoder can reduce computation. However, this approach often prioritizes highly salient regions (detailed in Fig. 1(a)), which may overlook critical visual details essential for the textual task, potentially lowering accuracy. This motivates us to rethink the core objective of token pruning in the vision encoder: beyond merely reducing the token count, it is crucial to preserve the semantic diversity of token representations and retain key visual cues aligned with the input text. We therefore examine the semantic information density of visual tokens across different spatial regions and observe that the density varies significantly. In particular, edge regions often contain redundant background or low-texture content, consistent with prior findings in vision research (Huang et al. 2022; Min et al. 2022).

To quantify semantic density, we propose a proxy metric defined by the log-determinant of the covariance matrix of regional features. Specifically, in the LLaVA-1.5-7B model, the input image is first divided by the visual encoder into a 24×24 grid of patches. We then group these patches into 36 region blocks arranged in a 6×6 layout, where each block contains adjacent 4×4 patches. The outermost 20 blocks (in the first and last rows and columns) are defined as the edge region, while the central 16 blocks are the central region. For each region block, we collect its token feature representations into a matrix $X \in \mathbb{R}^{n \times d}$, where n is the number of tokens in the block and d is the feature dimension. We compute the covariance matrix Σ of the feature representations in X and define the proxy semantic information entropy as:

$$H = \log |\Sigma + \epsilon I|, \tag{1}$$

where ϵI ensures numerical stability.

As shown in Figure 2, based on our analysis on the COCO-val-2014 dataset (Lin et al. 2014), the central regions consistently exhibit higher entropy than the edge regions, indicating richer semantic content centrally and more redundancy peripherally. These findings suggest that token pruning should not solely rely on visual saliency. Instead, it should dynamically retain tokens based on regional semantic density, thereby maximizing the preservation of informative content while effectively reducing redundancy.

Bias Consistency Across Tasks

Previous studies (Vaswani et al. 2017; Wen et al. 2025) have found that the final instruction token primarily attends to nearby visual tokens, inducing strong positional bias in the shallow decoder layers. As the model layers deepen, this positional bias gradually diminishes until it disappears (Zhang et al. 2025). While this layer-dependent bias is consistently observed, its underlying cause remains unclear. Previous work has primarily described its presence, but does not answer a key question: Is the degree of positional bias in visual tokens determined by task semantics, or is it solely a consequence of the model architecture?

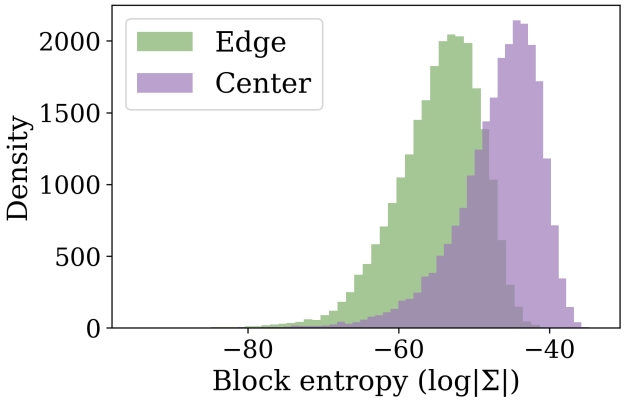
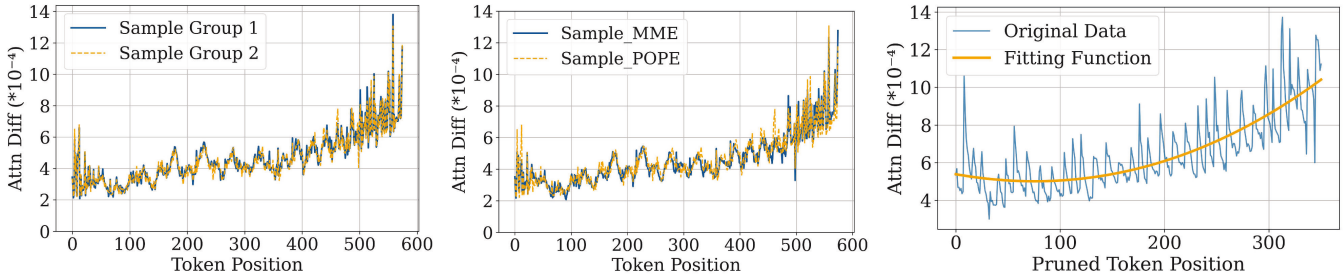


Figure 2: Entropy distribution across image regions. Central regions exhibit higher entropy than edge regions, indicating richer semantic content and lower redundancy.

To investigate this, we performed both qualitative and quantitative analyses on the LLaVA-1.5-7B model. Specifically, we extracted attention scores of visual tokens from shallow (2nd) and deep (20th) layers. The attention scores in shallow layers reflect a combination of task semantics and positional bias, while those in deep layers mainly capture the influence of task semantics. However, since attention on the same token differs between shallow and deep layers, directly subtracting the deep-layer attention scores from the shallow-layer scores on a single image cannot effectively isolate semantic effects. To mitigate this, we averaged the differences between shallow-layer and deep-layer attention scores across multiple samples, thereby estimating positional bias while smoothing out sample-specific semantic variations. We randomly sampled two non-overlapping sets of images from the MME dataset (Fu et al. 2023), and one additional set from the POPE dataset (Li et al. 2023b) for cross-dataset analysis. The sample sizes were kept consistent across groups. We then performed statistical analyses on the attention score differences within each group.

As shown in Figure 3(a), the two sampled sets of 800 images each from the MME dataset exhibit highly consistent trends in average positional bias. Figure 3(b) further compares the average positional bias of equally sized samples between the MME and POPE datasets, revealing similar alignment. To quantify the correlation of positional bias, we computed Pearson correlation coefficients at varying sample sizes (detailed in Figure 4). As the sample size increases, semantic variability is gradually smoothed out, and Pearson correlation coefficients rise significantly, indicating a strong architecture-induced positional bias. Based on these comprehensive analyses, we conclude that the positional bias in shallow layers is primarily driven by model architecture rather than task semantics. This architectural bias can thus be explicitly modeled and corrected during decoding.



(a) MME Dataset (b) MME and POPE Datasets (c) Positional Bias Fitting Function

Figure 3: (a) Two non-overlapping groups of 800 samples randomly selected from the MME dataset show consistent positional bias trends. (b) One group from the MME dataset and another from the POPE dataset exhibit similar bias patterns. (c) 800 MME samples are used to compute positional bias after fixed-ratio pruning in the vision encoder. A polynomial regression is fitted to derive a bias correction function for use during LLM decoding.

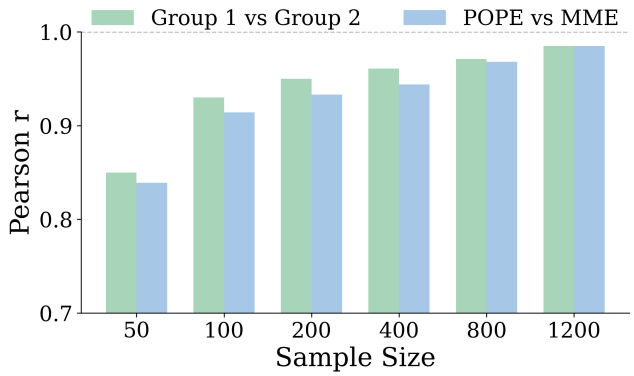


Figure 4: Pearson correlation at different sample sizes. Group 1 and Group 2 are non-overlapping random samples from MME, while POPE and MME each provide one sample group of equal size from their datasets.

Method

We propose PosPrune, a training-free, two-stage visual token pruning framework applied during both the vision encoder and LLM decoding stages, as illustrated in Figure 5.

Asymmetric Regional Pruning

Motivated by the observation that edge regions exhibit lower semantic density than central ones, we propose a region-aware token pruning strategy called Asymmetric Regional Pruning (ARP), which compresses the visual token sequence by selecting representative tokens within local region blocks guided by attention scores. This approach reduces the token count while preserving critical spatial structure and local details essential for subsequent text-guided pruning.

Formally, let the output of the vision encoder be denoted as $\mathbf{X} \in \mathbb{R}^{B\times N\times D}$, where B is the batch size, N is the total number of tokens (including the [CLS] token), and D is the feature dimension. Excluding the [CLS] token, the remaining $N-1$ tokens correspond to visual tokens. These tokens are then reshaped into a two-dimensional spatial grid. To balance accuracy and efficiency, we partition them into 36 region blocks in a 6×6 layout. The outermost 20 blocks (those in the first and last rows and columns) are defined as the edge region, while the remaining 16 blocks form the central region. Tokens in these regions are pruned with different ratios, guided by the attention scores from the [CLS] token.

Specifically, given the multi-head self-attention output $\mathbf{A} \in \mathbb{R}^{B\times H_a\times N\times N}$, where H_a is the number of attention heads, we compute the average attention from the [CLS] token to each visual token as:

$$\boldsymbol{\alpha} = \frac{1}{H_a}\sum_{h=1}^{H_a} \mathbf{A}_{[:,h,0,1:N]} \in \mathbb{R}^{B\times(N-1)}. \tag{2}$$

A lower pruning ratio R_c is applied to the central regions, while a higher pruning ratio R_e is applied to the edge regions. Note that, for LVLMs without a [CLS] token (e.g., Qwen-2.5-VL (Bai et al. 2025)), we follow VisionZip (Yang et al. 2025), using the average attention each visual token receives from all others as the importance metric for pruning.

Positional Bias-Corrected MMR Token Pruning

After pruning in the vision encoder, we propose a text-guided pruning method during decoding that corrects positional bias and applies Maximal Marginal Relevance to balance relevance and diversity, effectively reducing redundancy while preserving essential visual information.

Positional Bias Correction. As demonstrated in the previous section through extensive empirical and structural analysis, shallow-layer cross-attention exhibits positional bias due to architectural design rather than task semantics. If left uncorrected, this positional bias can lead the model to favor retaining visual tokens appearing later in the sequence during shallow-layer pruning, thereby overlooking earlier tokens that are more critical for downstream reasoning. To mitigate this issue, we propose an effective Positional Bias Correction (PBC) mechanism operating on shallow-layer attention. Formally, for an input visual token sequence of length T, let $A^{(\text{shallow})}(x) \in \mathbb{R}^T$ and $A^{(\text{deep})}(x) \in \mathbb{R}^T$ denote the cross-attention score vectors from a shallow and a deep decoder layer, respectively, for a given input sample x. Let $A_i^{(\cdot)}(x)$

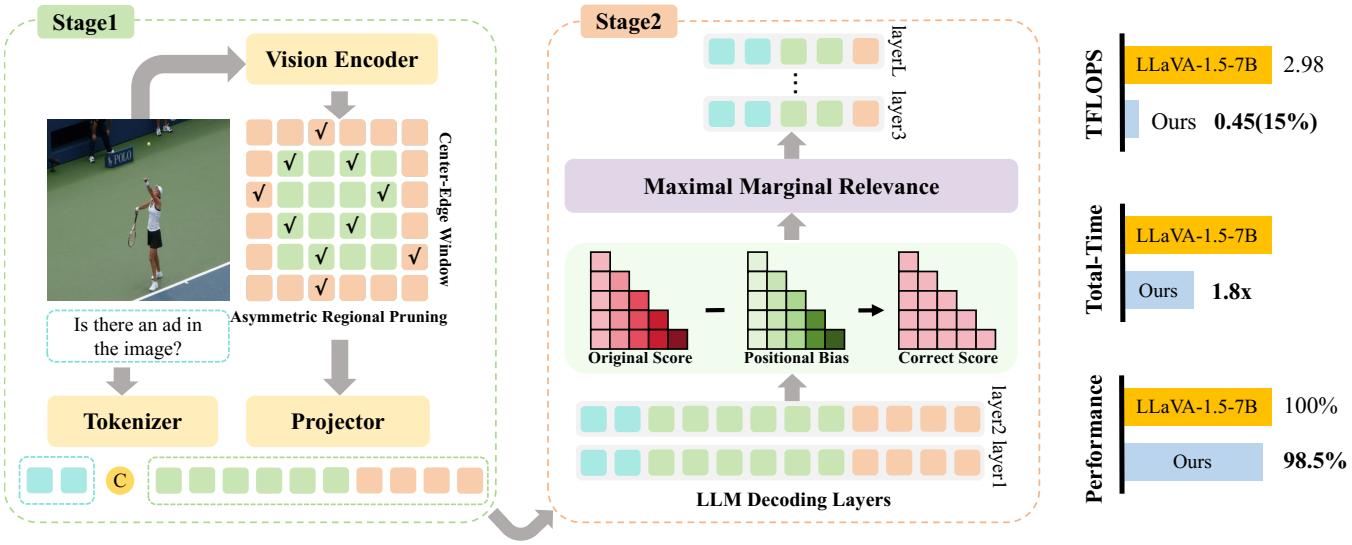


Figure 5: Overview of PosPrune. In the vision encoder, we employ Asymmetric Regional Pruning to prune redundant visual tokens while preserving spatial diversity and task-relevant details. During the LLM decoding stage, we first correct the positional bias in the shallow layers of the decoding process, then apply Maximal Marginal Relevance for further efficient pruning.

represent the attention score of the i-th visual token. The positional bias at token position i is defined as:

$$\Delta A_i(x) = A_i^{(\text{shallow})}(x) - A_i^{(\text{deep})}(x). \tag{3}$$

In theory, computing the expectation of this difference over a sufficiently large dataset $\mathcal{D}$ would suppress the inherent semantic variance and yield a much cleaner and more robust estimate of the structural positional bias:

$$B = \mathbb{E}_{x\sim\mathcal{D}}\left[A^{(\text{shallow})}(x) - A^{(\text{deep})}(x)\right]. \tag{4}$$

However, due to the high variance in semantic attention across individual samples, obtaining a bias-free estimate of B_i requires a prohibitively large number of diverse and representative inputs, as implied by the law of large numbers. In practice, this makes exact computation infeasible and highly impractical. To address this, we approximate the positional bias $\Delta A(i)$ using polynomial regression on a small validation set. The position-dependent bias is modeled as:

$$\Delta A(i) \approx \sum_{k=0}^{n} a_k i^k, \tag{5}$$

where i denotes the 1D token index and a_k are regression coefficients. The corrected shallow-layer attention becomes:

$$A_i^{\text{corrected}}(x) = A_i^{(\text{shallow})}(x) - \Delta A(i). \tag{6}$$

We randomly sampled 800 samples from the MME dataset for fitting, and the results are shown in Figure 3(c). This correction ensures that token pruning decisions in shallow layers are based on true semantic relevance rather than architectural artifacts. Empirically, we find this method significantly improves pruning precision and demonstrates robust generalization across different tasks and datasets under the same LVLM backbone. Results are reported in Table 1.

Maximal Marginal Relevance Pruning. To prune visual tokens under textual guidance while avoiding redundant selection, we adopt the Maximal Marginal Relevance (MMR) strategy (Carbonell and Goldstein 1998). Originally proposed for text summarization and information retrieval, MMR balances a candidate's relevance to a given query with its dissimilarity to previously selected items, encouraging both strong semantic alignment and high information diversity. Formally, the MMR score is defined as:

$$\text{MMR}(x_i) = \lambda \cdot \text{Rel}(x_i) - (1-\lambda) \cdot \max_{x_j \in S} \text{Sim}(x_i, x_j), \tag{7}$$

where x_i is a candidate token, S is the set of already selected tokens, $\text{Rel}(x_i)$ denotes relevance to the text, $\text{Sim}(x_i, x_j)$ is the similarity between tokens, and $\lambda \in [0, 1]$ controls the relevance-diversity trade-off.

In our setting, $\text{Rel}(x_i)$ is defined as the corrected cross-attention score $A_i^{\text{corrected}}$, capturing the token's importance conditioned on the text. Token features for computing $\text{Sim}(x_i, x_j)$ are taken from shallow decoder hidden states and L2-normalized for cosine similarity computation. The diversity term penalizes redundant selections by measuring each candidate's maximum similarity with selected tokens. We implement MMR pruning via a greedy selection process: the token with the highest relevance is chosen first, followed by tokens selected based on their MMR scores until the target number is reached. This approach ensures the retained tokens are both semantically aligned with the text and diverse in content.

Experiments

This section present the performance of our method on various benchmarks and compare it with several state-of-the-art (SOTA) approaches. We also conduct ablation studies to validate the effectiveness of each component in our method.

Method	TFLOPS	AOKVQA	SQA	POPE	MME	T-VQA	VizWiz	MMB	GQA	Avg.
LLaVA-1.5-7B										
Original	2.98	75.6	67.9	85.9	1785	58.0	54.0	64.2	61.9	100%
FastV	0.80	73.9	67.8	64.8	1490	50.6	54.9	56.4	55.8	90.4%
MustDrop	0.83	73.3	67.4	67.9	1594	54.5	51.2	60.0	53.1	91.6%
HiMAP	0.71	73.4	67.5	79.1	1714	55.9	54.1	60.9	57.3	96.0%
VisionZip	0.49	74.2	68.0	80.7	1737	55.9	54.4	60.8	57.2	96.7%
PosPrune	**0.45**	**74.3**	**68.2**	**84.8**	**1792**	**56.1**	**55.3**	**61.0**	**59.5**	**98.5%**
LLaVA-1.5-13B										
Original	5.81	81.9	71.6	86.1	1792	61.2	54.9	68.7	63.1	100%
FastV	1.50	76.4	71.3	69.1	1697	56.4	55.9	61.9	59.2	93.2%
MustDrop	1.43	73.5	71.2	62.3	1626	55.9	53.7	63.8	55.3	90.2%
HiMAP	1.23	78.4	72.4	81.9	1758	**58.3**	56.6	65.9	59.3	97.3%
VisionZip	0.95	79.8	72.4	80.5	1738	58.2	55.2	65.4	57.7	96.4%
PosPrune	**0.85**	**80.0**	**72.9**	**85.5**	**1770**	58.0	**56.7**	**66.1**	**59.7**	**98.4%**
Qwen2.5-VL-7B										
Original	4.05	87.0	76.7	87.4	2316	74.1	66.9	79.8	57.9	100%
FastV	1.40	84.4	74.0	83.9	2109	66.2	66.3	76.5	52.5	94.4%
HiMAP	1.20	79.8	76.7	79.8	1980	66.0	62.9	76.6	52.5	92.3%
VisionZip	1.19	83.4	77.1	83.5	2083	66.9	66.5	76.5	54.3	95.3%
PosPrune	**1.16**	**85.1**	**81.1**	**84.4**	**2170**	**71.3**	**66.6**	**76.9**	**56.0**	**97.8%**

Table 1: Comparison of PosPrune with FastV, MustDrop, HiMAP and VisionZip across different models and datasets.

Experimental Settings

Baselines and Models. We evaluate our method on three representative models: LLaVA-1.5-7B (Liu et al. 2024a), LLaVA-1.5-13B (Liu et al. 2024a), and Qwen2.5-VL-7B-Instruct (Bai et al. 2025). Furthermore, we compare our approach with four SOTA visual token pruning methods: FastV (Chen et al. 2024a), HiMAP (Yin, Si, and Wang 2025), MustDrop (Liu et al. 2024b), and VisionZip (Yang et al. 2025). It is important to note that MustDrop selects a set of indispensable tokens based on the attention to the [CLS] token in the vision encoder. This design conflicts with the architecture of Qwen2.5-VL-7B-Instruct, hence we compare with MustDrop only on the other two models.

Benchmarks and Evaluation. We conduct experiments on eight widely used image understanding benchmarks: ScienceQA (Lu et al. 2022), AOKVQA (Schwenk et al. 2022), MME (Fu et al. 2023), POPE (Li et al. 2023b), VizWiz (Gurari et al. 2018), TextVQA (Singh et al. 2019), MMBench (Liu et al. 2024c), and GQA (Hudson and Manning 2019). All experiments follow the LLaVA evaluation standard (Liu et al. 2024a) and utilize the LMMs-Eval framework (Li et al. 2024a). Additionally, we evaluate all methods in terms of FLOPs, total inference time, and GPU memory usage. For FLOPs calculation, we follow the computation method used in HiMAP, i.e., the FLOP of the l-th layer's attention and MLP module is calculated as:

$$\text{FLOP} = 4nd^2 + 2n^2d + 2ndm, \tag{8}$$

where n is the number of visual tokens, d is the hidden dimension, and m is the intermediate size of the FFN layer.

Method	Token	TFLOPS	Total-Time	GPU	MME
Original	576	2.98	634s	14.7	1785
FastV	128	0.8	450s	14.2	1490
MustDrop	64	0.83	458s	14.2	1594
HiMAP	64	0.71	349s	14.1	1714
VisionZip	96	0.49	332s	13.9	1737
PosPrune	72	0.45	349s	**13.8**	**1792**
PosPrune*	64	0.42	**320s**	**13.8**	1768

Table 2: Efficiency comparison using LLaVA-1.5-7B.

Implementation Details. During the vision encoder stage, the pruning ratios are set to $R_c = 25\%$ and $R_e = 50\%$. In the decoder stage, pruning is applied at the second layer with $R = 87.5\%$. For the MMR strategy, $\lambda = 0.9$ balances text-guided accuracy and token diversity. Note that, for LVLMs with dynamic image resolutions, the varying number of visual tokens across tasks leads to inconsistent positional bias when using absolute positions. Normalized relative positions (i/N) are thus adopted to ensure consistent bias correction. Other pruning baselines use default parameters from their codebases.

Results and Analysis

As shown in Table 1, we conduct comprehensive experiments across different models and datasets. The results demonstrate that our method consistently outperforms existing SOTA methods across all benchmarks and models, achieving better performance at lower FLOPs. Furthermore,

Method	Token	TFLOPS	MME	SQA
Original	576	2.98	1785	67.9
+ Global [CLS] Selection	352	1.80	1687	67.4
+ ARP	352	1.80	1751	67.7
+ ARP MMR	72	0.45	1746	67.8
+ ARP MMR PBC	72	0.45	**1792**	**68.2**

Table 3: Ablation study of individual components using LLaVA-1.5-7B on the MME and ScienceQA datasets.

Polynomial Degree	1	2	3	4	5
MME	1765	1786	**1792**	1774	1663
SQA	67.7	68.1	**68.2**	**68.2**	67.5

Table 4: Investigation of the polynomial degree using LLaVA-1.5-7B on the MME and ScienceQA datasets.

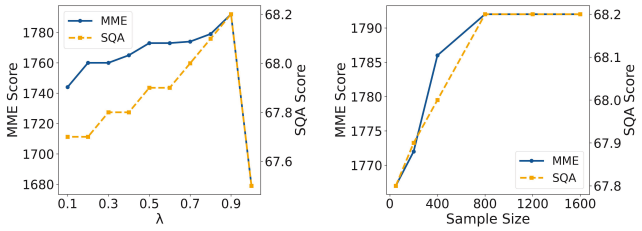
although the correction function is fitted using a small number of samples, it generalizes well across all tasks under the same model configuration. These results confirm that our method effectively reduces the number of visual tokens while maintaining high task performance.

In Table 2, we compare the efficiency of our method with other approaches. All experiments are conducted on a server equipped with a single 48GB NVIDIA A40 GPU. At a pruning ratio of 87.5%, our method achieves a 3% improvement in benchmark performance compared to VisionZip, while incurring only a 17-second increase in total inference time. When the pruning ratio is increased to 88.9%, our method outperforms all baselines across all evaluation metrics.

Ablation Studies

Ablations on the Individual Modules of PosPrune. We perform ablation studies on the visual token pruning modules in the LLaVA-1.5-7B model, as shown in Table 3. Introducing ARP alone outperforms global [CLS] selection at equal FLOPs, showing that non-uniform sampling reduces local detail loss. When adding MMR, the performance remains stable even at higher pruning ratios, suggesting that MMR successfully balances token diversity with semantic relevance to the text. Finally, incorporating position bias correction (PBC) further improves results, supporting the hypothesis that positional bias stems from the model architecture rather than task-specific semantic factors.

Ablations on Polynomial Degree. As shown in Table 4, we investigate the impact of varying the degree of the polynomial used to fit the positional bias curve on model performance. When the polynomial degree is too low, it fails to adequately capture the underlying positional bias, leading to underfitting. Performance is maximized with a degree-3 polynomial, which most effectively approximates the positional bias curve. However, increasing the polynomial degree beyond this point results in overfitting, where the model captures oscillatory artifacts caused by residual semantic noise, thereby degrading performance. Therefore, we select



(a) Different λ Settings (b) Different Sample Sizes

Figure 6: (a) Investigation of λ parameter settings. (b) Performance of the fitting function with different sample sizes.

a degree-3 polynomial for fitting the positional bias curve.

Ablations on Sample Size. Figure 6(b) further analyzes the effect of sample size on the quality of the fitted curve, using LLaVA-1.5-7B on the MME and ScienceQA datasets. With a limited number of samples, residual semantic variations among visual tokens introduce noise, hindering accurate modeling of the positional bias. As the sample size increases, the fitting quality steadily improves and stabilizes when the number of samples reaches 800. Based on this observation, we adopt the polynomial fitted with 800 samples for positional bias correction in our model.

Ablations on Parameter λ. We study the impact of the parameter λ in the MMR algorithm using LLaVA-1.5-7B on the MME and ScienceQA datasets, as shown in Figure 6(a). When $\lambda = 1$, token selection relies solely on relevance scores, completely ignoring the diversity term. This results in a significant performance degradation, especially at higher pruning ratios, indicating that relevance-only pruning tends to select redundant tokens. The optimal performance occurs at $\lambda = 0.9$, where selection is mainly driven by relevance while incorporating moderate diversity, effectively reducing redundancy. As λ decreases further, the emphasis on relevance weakens, causing another drop in performance. These results demonstrate that effective token pruning during decoding requires a relevance-focused strategy complemented by appropriate diversity control; relying exclusively on either relevance or diversity is insufficient.

Conclusion

In this work, we analyzed the limitations of existing visual token pruning methods in both the vision encoder and LLM decoding stages, and proposed PosPrune, a training-free, two-stage pruning framework for accelerating LVLM inference. PosPrune combines asymmetric region-aware pruning in the vision encoder with positional bias-corrected MMR pruning in the decoder, effectively reducing computational cost while preserving task-critical information. Extensive experiments across multiple LVLMs and vision-language benchmarks demonstrate its strong efficiency-performance trade-off. We hope this work will inspire future research into visual token redundancy and promote the development of more efficient and scalable LVLM architectures.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant 62176246. This work is also supported by Anhui Province Key Research and Development Plan (202304a05020045) and Anhui Province Natural Science Foundation (2208085UD17). This work is also supported by National Natural Science Foundation of China under Grant 62406098 and 62376256, and The Joint Fund for Medical Artificial Intelligence under Grant MAI2022Q011.

References

Agrawal, H.; Desai, K.; Wang, Y.; Chen, X.; Jain, R.; Johnson, M.; Batra, D.; Parikh, D.; Lee, S.; and Anderson, P. 2019. Nocaps: Novel object captioning at scale. In *Proceedings of the IEEE/CVF international conference on computer vision*, 8948–8957.

Bai, J.; Bai, S.; Yang, S.; Wang, S.; Tan, S.; Wang, P.; Lin, J.; Zhou, C.; and Zhou, J. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 1(2): 3.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Carbonell, J.; and Goldstein, J. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 335–336.

Chen, L.; Zhao, H.; Liu, T.; Bai, S.; Lin, J.; Zhou, C.; and Chang, B. 2024a. 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*, 19–35. Springer.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024b. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 24185–24198.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3): 6.

Fu, C.; Chen, P.; Shen, Y.; Qin, Y.; Zhang, M.; Lin, X.; Yang, J.; Zheng, X.; Li, K.; Sun, X.; et al. 2023. MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models. *arXiv preprint arXiv:2306.13394*.

Fu, X.; Hu, Y.; Li, B.; Feng, Y.; Wang, H.; Lin, X.; Roth, D.; Smith, N. A.; Ma, W.-C.; and Krishna, R. 2024. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, 148–166. Springer.

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 *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Huang, G.; Wang, Y.; Lv, K.; Jiang, H.; Huang, W.; Qi, P.; and Song, S. 2022. Glance and focus networks for dynamic visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 45(4): 4605–4621.

Hudson, D. A.; and Manning, C. D. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6700–6709.

Li, B.; Zhang, P.; Zhang, K.; Pu, F.; Du, X.; Dong, Y.; Liu, H.; Zhang, Y.; Zhang, G.; Li, C.; et al. 2024a. Lmms-eval: Accelerating the development of large multimoal models.

Li, F.; Zhang, R.; Zhang, H.; Zhang, Y.; Li, B.; Li, W.; Ma, Z.; and Li, C. 2024b. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, Y.; Du, Y.; Zhou, K.; Wang, J.; Zhao, W. X.; and Wen, J.-R. 2023b. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Lin, Z.; Lin, M.; Lin, L.; and Ji, R. 2025. Boosting multimodal large language models with visual tokens withdrawal for rapid inference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 5334–5342.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 26296–26306.

Liu, T.; Shi, L.; Hong, R.; Hu, Y.; Yin, Q.; and Zhang, L. 2024b. Multi-stage vision token dropping: Towards efficient multimodal large language model. *arXiv preprint arXiv:2411.10803*.

Liu, Y.; Duan, H.; Zhang, Y.; Li, B.; Zhang, S.; Zhao, W.; Yuan, Y.; Wang, J.; He, C.; Liu, Z.; et al. 2024c. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, 216–233. Springer.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35: 2507–2521.

Min, J.; Zhao, Y.; Luo, C.; and Cho, M. 2022. Peripheral vision transformer. *Advances in Neural Information Processing Systems*, 35: 32097–32111.

Plummer, B. A.; Wang, L.; Cervantes, C. M.; Caicedo, J. C.; Hockenmaier, J.; and Lazebnik, S. 2015. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, 2641–2649.

Schwenk, D.; Khandelwal, A.; Clark, C.; Marino, K.; and Mottaghi, R. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, 146–162. Springer.

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 *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, H.; Yu, Z.; Spadaro, G.; Ju, C.; Quétu, V.; Xiao, S.; and Tartaglione, E. 2025. Folder: Accelerating multi-modal large language models with enhanced performance. *arXiv preprint arXiv:2501.02430*.

Wen, Z.; Gao, Y.; Li, W.; He, C.; and Zhang, L. 2025. Token Pruning in Multimodal Large Language Models: Are We Solving the Right Problem? *arXiv preprint arXiv:2502.11501*.

Xing, L.; Huang, Q.; Dong, X.; Lu, J.; Zhang, P.; Zang, Y.; Cao, Y.; He, C.; Wang, J.; Wu, F.; et al. 2024. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*.

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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, 19792–19802.

Yao, Z.; Xu, Y.; Xu, H.; Liao, Y.; and Xie, Z. 2025. Efficient deployment of large language models on resource-constrained devices. *arXiv preprint arXiv:2501.02438*.

Yin, H.; Si, G.; and Wang, Z. 2025. Lifting the Veil on Visual Information Flow in MLLMs: Unlocking Pathways to Faster Inference. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 9382–9391.

Yue, X.; Ni, Y.; Zhang, K.; Zheng, T.; Liu, R.; Zhang, G.; Stevens, S.; Jiang, D.; Ren, W.; Sun, Y.; et al. 2024. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9556–9567.

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

Zhang, Q.; Cheng, A.; Lu, M.; Zhuo, Z.; Wang, M.; Cao, J.; Guo, S.; She, Q.; and Zhang, S. 2024. [CLS] Attention is All You Need for Training-Free Visual Token Pruning: Make VLM Inference Faster. *arXiv e-prints*, arXiv–2412.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.