EvoPrune: Early-Stage Visual Token Pruning for Efficient MLLMs

Anonymous ACL submission

Abstract

Multimodal Large Language Models (MLLMs) have achieved remarkable performance on vision–language tasks, yet their inference efficiency remains constrained by the rapid growth of visual tokens in high-resolution images and videos. Existing visual token pruning methods typically operate after visual encoding, neglecting the substantial computational overhead incurred during the encoding process. To address this limitation, we propose EvoPrune, an early-stage visual token pruning framework that performs token reduction directly within visual encoding. EvoPrune adopts a layer-wise pruning strategy that jointly leverages token similarity, diversity, and attention-based importance to preserve the most informative visual tokens at selected encoder layers. Extensive experiments on image and video benchmarks demonstrate the effectiveness of EvoPrune. Notably, on VideoMME, EvoPrune achieves a 2× inference speedup with less than 1% performance degradation, highlighting its suitability for latency-sensitive MLLM deployment.

1 Introduction

Multimodal Large Language Models (MLLMs) (Wang et al., 2024; Li et al., 2024a; Liu et al., 2023; Bai et al., 2025; Achiam et al., 2023; Team et al., 2024; Chen et al., 2024c; Lu et al., 2024; Guo et al., 2025; Liu et al., 2024a) have achieved remarkable progress across a wide range of vision–language tasks, substantially expanding their practical applications. However, high-resolution image processing and long-video understanding introduce a massive number of visual tokens, leading to severe inference inefficiency. The resulting token explosion imposes substantial computational and memory overhead on both visual encoding and LLM prefilling, creating a major bottleneck for deploying MLLMs in latency-sensitive scenarios such as real-time video understanding and edge computing.

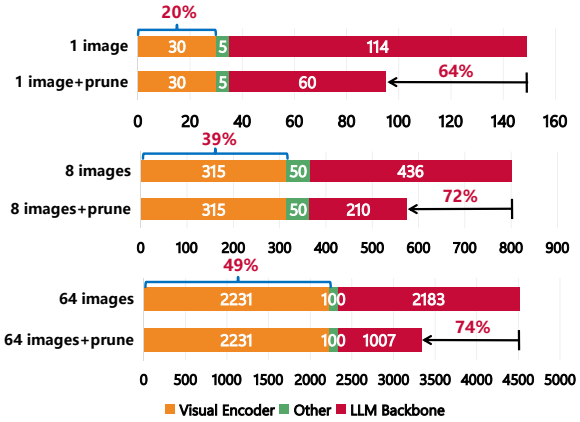

Figure 1: **Inference time of different components in MLLMs with and without visual Pruning under different input scales.**

Existing approaches for mitigating visual token explosion can be broadly divided into two paradigms. The first (Li et al., 2024a; Dai et al., 2023; Li et al., 2023) employs learned projectors trained on large-scale datasets for token selection, achieving moderate token reduction but requiring retraining and exhibiting limited generalization. The second performs training-free token pruning during inference based on visual token representations. Within this paradigm, attention-based methods (Chen et al., 2024a; Zhang et al., 2024d; Xing et al., 2024; Yang et al., 2025) preserve salient tokens using attention scores, whereas similarity-based approaches (Yang et al., 2025; Alvar et al., 2025; Zhang et al., 2025b; Shang et al., 2025) remove redundant tokens through unsupervised similarity analysis, such as determinantal point processes (Chen et al., 2018). These methods achieve substantial inference acceleration with minimal performance degradation.

Despite their effectiveness, existing visual token pruning methods share a key limitation: pruning is performed only after visual encoding. As shown in

Fig. 1, the computational cost of the vision encoder increases substantially as inputs scale from single images to multi-frame videos (8 and 64 frames), eventually approaching the cost of the LLM backbone. However, most prior methods apply pruning at intermediate stages or within the LLM, primarily reducing language-model computation while leaving the dominant encoder overhead largely unaffected. As a result, their acceleration gains diminish with increasing input scale. Specifically, the remaining inference time increases from 64% for single-image inputs to 72% and 74% for 8-frame and 64-frame inputs, respectively, revealing a clear scalability bottleneck under large visual inputs.

To address these limitations, we propose **EvoPrune**, an early-stage visual token pruning framework for MLLMs. EvoPrune removes redundant tokens during the initial stages of visual encoding, before expensive feature extraction, thereby reducing the computational overhead of both the vision encoder and the downstream LLM. This design enables substantial end-to-end inference acceleration across diverse input resolutions.

Unlike existing methods that prune tokens only after full visual encoding, **EvoPrune** introduces progressive token reduction directly within the vision encoder. At designated layers, it dynamically merges redundant tokens by jointly leveraging token similarity, diversity, and intermediate attention signals. This multi-factor, hierarchical strategy preserves highly informative visual tokens while significantly lowering the computational overhead of both the vision encoder and downstream multimodal modules, enabling highly efficient end-to-end inference across diverse visual inputs.

We evaluate EvoPrune on multiple image and video understanding benchmarks. Experimental results demonstrate state-of-the-art efficiency–performance trade-offs. Notably, on the challenging VideoMME benchmark, EvoPrune achieves a 50% reduction in inference latency with less than 1% performance degradation. Our contributions are summarized as follows:

1. We propose a novel early-stage pruning paradigm, which acts in the visual encoding stage to address the long-neglected encoding overhead in existing works.

2. We propose a layer-wise pruning strategy with multi-factor guidance that jointly leverages token similarity, diversity, and attention-based importance to maximize information retention during token selection.

3. We validate EvoPrune across diverse vision–language tasks, showing superior inference efficiency while maintaining competitive performance.

2 Related work

2.1 Multimodal Large Language models

Large-scale multimodal models (Wang et al., 2024; Li et al., 2024a; Lu et al., 2024; Liu et al., 2023; Bai et al., 2025; Achiam et al., 2023; Team et al., 2024; Wang et al., 2025a; Li et al., 2023; Liu et al., 2024a) have achieved remarkable progress, facilitating their expansion to more complex tasks. These models feed visual-encoder representations into Large Language Models (LLMs) after specific processing (e.g., via a projector). With the growing prevalence of applications, community researchers have found that real-world scenarios frequently demand the comprehension of large-scale images, videos, or sequential image frames.

Recent studies (Liu et al., 2024b,e; Shi et al., 2025; Zhang et al., 2024c,b) have focused on improving MLLM performance on large-scale image understanding, while others (Zhang et al., 2024e; Chen et al., 2024b; Shen et al., 2024; Shu et al., 2025; Wang et al., 2025b) extend MLLMs to long-video understanding. However, increasing task complexity substantially enlarges the number of visual tokens, particularly in high-resolution image and video scenarios. The resulting computational overhead has become a critical bottleneck for efficient MLLM deployment.

2.2 Visual Token Pruning

Visual token pruning has been widely studied to mitigate the quadratic complexity of Vision Transformers (ViTs). Early methods (Bolya et al., 2022; Rao et al., 2021; Li et al., 2024c) primarily target standalone vision models, focusing on image classification and recognition by reducing spatial redundancy within ViTs.

With the emergence of MLLMs, recent work extends visual token pruning to long-context multimodal settings involving high-resolution images and long videos, where preserving fine-grained semantics and cross-modal alignment is critical. Existing approaches (Wang et al., 2024; Guo et al., 2025; Li et al., 2024d; Wang et al., 2025a; Cha et al., 2024) can be broadly categorized into four

groups. Transformation-based methods compress visual representations through multi-scale feature fusion. Similarity-based methods (Yang et al., 2025; Alvar et al., 2025; Zhang et al., 2025b; Shen et al., 2024; Shao et al., 2025; Song et al., 2024; Zhang et al., 2024a) merge semantically or spatially redundant tokens. Attention-based methods (Shang et al., 2025; Zhang et al., 2025c,a, 2024d, 2025b) retain salient tokens based on attention importance, while query-based methods (Liu et al., 2024c; Wang et al., 2025a; Li et al., 2023; Dai et al., 2023; Song et al., 2025) perform task-aware pruning using query-guided token selection. For video understanding, additional temporal strategies (Shao et al., 2025; Chen et al., 2018; Shen et al., 2024; Song et al., 2024; Shen et al., 2025), such as keyframe sampling and cross-frame token merging, are introduced to reduce temporal redundancy. Overall, existing methods aim to balance compression efficiency, information preservation, and multimodal compatibility, providing a foundation for efficient long-context MLLMs.

3 Method

3.1 Overall

Visual token pruning effectively reduces MLLM inference costs. However, most existing methods prune tokens only after full visual encoding. This delayed pruning yields limited acceleration and leaves the vision encoder as a major computational bottleneck, particularly for high-resolution images and long videos. To address this limitation, we propose **EvoPrune**, an *early-stage* visual token pruning framework that integrates progressive token merging directly into the vision encoder. By hierarchically merging redundant tokens across encoder layers, EvoPrune minimizes computation throughout multimodal inference while preserving task-critical visual information.

Figure 2 illustrates the overall architecture of EvoPrune. At selected vision encoder layers, tokens are first processed by standard multi-head self-attention, followed by token merging guided by a composite score matrix integrating three complementary criteria:

- **Semantic Similarity**, which merges visually and semantically redundant tokens;
- **Information Diversity**, which preserves representational richness by preventing the merging of distinct tokens;

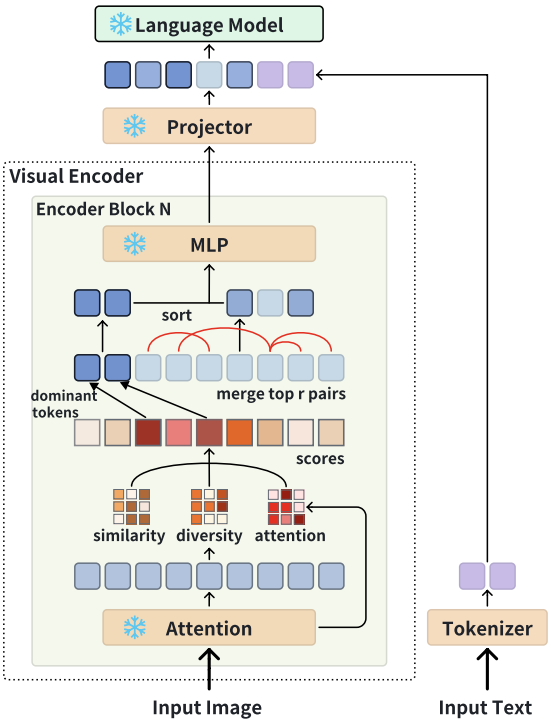

Figure 2: **Overview of the EvoPrune framework.** Token merging operations are applied at selected visual encoder layers. A composite score matrix integrating semantic similarity, diversity, and attention-based importance, guides token pair selection.

- **Attention-Based Importance**, which retains critical tokens to safeguard downstream reasoning.

Token pairs with the highest composite scores are merged and propagated to subsequent layers, enabling progressive token reduction while preserving discriminative visual representations.

By embedding pruning into the visual encoder, EvoPrune alleviates the computational bottleneck of post-encoder pruning and achieves efficient multimodal inference without sacrificing visual fidelity.

3.2 Layer-wise Pruning Budget Allocation

EvoPrune adopts a layer-wise pruning formulation that explicitly allocates token merging budgets across selected visual encoder layers.

Algorithm 1 summarizes the EvoPrune pipeline. Given an input image or video, a global pruning target, and a predefined merging strategy, EvoPrune determines the number of tokens to merge at each encoder layer during inference.

Formally, for a visual encoder with L transformer layers, let $\mathcal{T}_0$ denote the initial token set containing N_0 tokens, and let R denote the global

Algorithm 1 Layer-wise Token Pruning in EvoPrune

Require: Initial visual tokens $\mathcal{T}_0$; visual encoder with L transformer blocks; total pruning budget R; layer-wise merging strategy $\mathcal{S}$

Ensure: Encoded visual tokens $\mathcal{T}_L$

Allocate the total pruning budget R across encoder layers using strategy $\mathcal{S}$, yielding a layer-wise schedule $\mathbf{r} = \{r_1, r_2, \ldots, r_L\}$, where $\sum_{l=1}^{L} r_l = R$
$\mathcal{T} \leftarrow \mathcal{T}_0$
for $l = 1$ to L **do**
  $\mathcal{T} \leftarrow \text{Attention}_l(\mathcal{T})$
  if $r_l > 0$ **then**
    $\mathcal{T} \leftarrow \text{ScoreGuidedMerge}(\mathcal{T}, r_l)$
  end if
  $\mathcal{T} \leftarrow \text{MLP}_l(\mathcal{T})$
end for
Return $\mathcal{T}$

pruning target. EvoPrune computes a layer-wise pruning schedule:

$$\mathbf{r} = \{r_1, r_2, \ldots, r_L\}, \quad \text{s.t.} \sum_{l=1}^{L} r_l = R,$$

where $r_l \geq 0$ denotes the number of tokens merged at layer l. During inference, token merging is applied whenever $r_l > 0$ using the score-guided strategy described in Section 3.3.

Merging Strategy Design Due to space constraints, detailed layer-wise merging strategies are deferred to the appendixB. We systematically investigate multiple budget allocation strategies and analyze their trade-offs between efficiency and accuracy. Complete formulations and experimental results are provided in the appendix.

3.3 Score-Guided Token Merging

We propose a score-guided token merging strategy tailored for in-encoder pruning in MLLMs. The method introduces a score matrix that evaluates the benefit of merging token pairs using multiple complementary criteria, making it well suited for multimodal, information-sensitive settings.

Given a token set:

$$\mathcal{T} = \{t_1, t_2, \ldots, t_N\}$$

we partition the tokens into two disjoint groups based on index parity:

$$a = \{t_i \in \mathcal{T} \mid i \bmod 2 = 0\},$$
$$b = \{t_i \in \mathcal{T} \mid i \bmod 2 = 1\}.$$

We then construct a score matrix:

$$Score \in R^{|a| \times |b|},$$

where each entry $Score_{ij}$ quantifies the utility of merging token $t_i \in a$ into token $t_j \in b$. Scores are computed only for cross-group pairs.

For each token $t_i \in a$, the merge target is selected as:

$$j^* = \arg\max_{j \in b} Score_{ij}.$$

ensuring that each token in a participates in at most one merge. From all candidate pairs $\{(i, j^*) \mid t_i \in a\}$, we select the top-$r$ pairs with the highest scores:

$$\mathcal{M} = \text{Top-}r\{(i, j^*) \mid Score_{ij^*}\}.$$

The selected token pairs are then merged from group a into b, producing the updated token set $\mathcal{T}'$ for subsequent layers.

This strategy provides a principled and flexible framework for visual token merging, enabling explicit control over merge operations while preserving token structure for downstream processing. The detailed formulation of the score matrix is presented in the following subsection.

3.4 Similarity, Diversity, and Attention Integration

To guide token merging within the vision encoder, we define the matching score between tokens i and j as:

$$\text{Score}_{ij} = \underbrace{\text{Sim}_{ij}}_{\text{Similarity Attraction}} \times \underbrace{w_{ij}^{\text{div}}}_{\text{Diversity Penalty}} \times \underbrace{w_{ij}^{\text{attn}}}_{\text{Attention Preservation}}.$$

Here, Sim_{ij} measures semantic similarity between visual tokens, w_{ij}^{div}: penalizes redundant merges to preserve token diversity, and w_{ij}^{attn} incorporates attention importance learned by the encoder. Together, these three components balance token compactness and information preservation during pruning.

Based on the composite score, EvoPrune performs hierarchical token merging across encoder layers using bipartite soft matching to iteratively aggregate token pairs. Unlike prior methods that rely solely on pairwise similarity, our formulation explicitly incorporates diversity and attention constraints, producing more discriminative and information-preserving visual representations.

Similarity Attraction The similarity attraction term measures the visual correspondence between tokens i and j via cosine similarity of their feature embeddings a_i and b_j:

$$\text{Sim}_{ij} = \frac{a_i \cdot b_j}{\|a_i\| \, \|b_j\|}.$$

This score lies in Sim_{ij} after normalization and captures angular proximity in the embedding space. For numerical stability and cross-token comparability, the similarity can optionally be further normalized using a sigmoid or softmax function before being incorporated into the overall matching score.

Diversity Penalty To promote structural diversity during token merging, we quantify the local distinctiveness of each token embedding $\mathbf{v}_i \in R^d$ via a density-based measure. Let $\mathcal{N}_i$ denote the set of its K nearest neighbors under squared Euclidean distance. The local density is defined as

$$d_i = 1 - \frac{1}{K} \sum_{k \in \mathcal{N}_i} \exp\left(-\frac{\|\mathbf{v}_i - \mathbf{v}_k\|^2}{\tau^2} \right).$$

where $\tau > 0$ denotes the Gaussian bandwidth. A larger d_i indicates that token i resides in a sparser region of the embedding space, thereby reflecting higher diversity.

Based on this measure, we define the diversity penalty for a token pair (i, j) as

$$w_{ij}^{\text{div}} = \exp(-\lambda_d (d_i + d_j)),$$

where $\lambda_d > 0$ balances the diversity constraint. This formulation suppresses merges in redundant regions while prioritizing tokens with higher informational diversity during matching.

Attention Preservation To prevent the loss of critical visual and contextual information, the attention preservation module leverages attention-derived importance scores to identify and retain salient tokens throughout the pruning process.

To quantify token importance, we compute a score s_i for each token i by averaging attention weights across all heads and positions. Formally, letting $\mathbf{A}^h \in R^{N \times N}$ denote the attention matrix for head h, where N is the total token count, the importance score is defined as:

$$s_i = \frac{1}{H} \sum_{h=1}^{H} \frac{1}{N} \sum_{j=1}^{N} A_{ij}^h,$$

where H is the number of attention heads and A_{ij}^h denotes the attention weight from token i to token j in head h. A larger s_i indicates higher global attention and thus greater semantic importance.

To control the proportion of preserved tokens, we introduce a **Critical Token Ratio (CTR)**, which determines the number of protected tokens as:

$$N_{\text{protect}} = \min(\text{CTR} \times T, \, T - 2r),$$

where T is the current number of tokens and r is the number of tokens merged per pruning step. This adaptive formulation reduces the protected set as pruning progresses, avoiding excessive constraints in later stages.

We then select the top N_{protect} tokens according to s_i to form the critical set $\mathcal{S}$, which is preserved across subsequent pruning stages. Finally, the attention preservation weight for a token pair (i, j) is defined as:

$$w_{ij}^{\text{attn}} = \begin{cases} -\infty, & \text{if } i \in \mathcal{S} \text{ or } j \in \mathcal{S}, \\ 1, & \text{otherwise.} \end{cases}$$

Setting $w_{ij}^{\text{attn}} = -\infty$ for pairs involving critical tokens prevents their merging, preserving salient information throughout the pruning process.

4 Experiment

4.1 Experimental Settings

Model Architectures. To ensure fair and reproducible comparison, we follow standard experimental settings from prior work on multimodal efficiency and token pruning. For image-based evaluation, we adopt the LLaVA-1.5-7B configuration (Liu et al., 2024a), while video-based experiments use the LLaVA-Video-7B setting (Zhang et al., 2024e) with a Qwen2 language backbone (Yang et al., 2024). These complementary setups enable consistent evaluation of pruning methods across both image and video modalities.

Benchmarks. To comprehensively evaluate efficiency and generalization, we conduct experiments on a diverse suite of vision–language benchmarks. For image understanding, we include VQAv2 (Jia et al., 2025), MME (Fu et al., 2023), MMBench (MMB^{EN} and MMB^{CN}) (Liu et al., 2024d), and MMVet (Yu et al., 2024), covering perception, reasoning, multilingual understanding, and fine-grained visual recognition. For video understanding, we evaluate on MVBench (Li et al., 2024b),

Table 1: **Performance comparison on image benchmarks. Relative Latency** denotes the execution time of the pre-LLM, LLM, and overall stages normalized to the unpruned baseline (Native). **Acc.** represents the average performance across all benchmarks, and **Rel.** indicates the mean relative accuracy (%) compared to the baseline.

Method	Benchmarks ↑					Relative Latency ↓			Avg. ↑	
Metric	VQA^{V2}	MME	MMB^{EN}	MMB^{CN}	MMVet	pre LLM	LLM	Overall	Acc.	Rel.
Upper Bound, All 576 Tokens (100%)										
Native	78.5	1508	61.8	58.1	32.2	1.00	1.00	1.00	76.5	100.0%
Retain 128 Tokens (↓ 77.8%)										
FasterVLM	75.5	1353	58.3	54.8	33.1	1.15	0.77	0.85	72.3	94.6%
VisPruner	75.8	1405	58.8	55.2	32.8	1.32	0.77	0.90	73.2	95.7%
DART	74.7	1344	57.5	52.9	31.3	1.20	0.77	0.88	70.9	92.7%
DivPrune	76.0	1400	59.3	54.8	31.5	1.16	0.77	0.85	72.9	95.3%
CDPruner	76.6	1433	60.6	55.0	31.7	1.41	0.77	0.91	73.9	96.6%
EvoPrune (Ours)	76.8	1441	59.7	57.3	33.7	**1.10**	**0.77**	**0.84**	**74.9**	**97.9%**
Retain 64 Tokens (↓ 88.9%)										
FasterVLM	72.7	1294	56.7	51.9	26.8	1.09	0.73	0.79	68.2	89.2%
VisPruner	73.9	1321	57.7	52.2	29.5	1.35	0.73	0.86	69.9	91.3%
DART	71.3	1250	53.1	49.1	28.8	1.13	0.73	0.80	66.2	86.6%
DivPrune	74.1	1342	58.1	52.4	29.9	1.12	0.73	0.79	70.4	92.0%
CDPruner	75.4	1418	59.1	53.3	30.3	1.38	0.73	0.86	72.2	94.4%
EvoPrune (Ours)	75.6	1414	58.3	54.6	32.4	**1.07**	**0.73**	**0.77**	**72.9**	**95.3%**
Retain 32 Tokens (↓ 94.4%)										
FasterVLM	70.9	1190	53.7	47.6	26.5	1.09	0.71	0.77	64.5	84.4%
VisPruner	72.1	1196	53.9	48.5	26.6	1.35	0.71	0.84	65.2	85.2%
DART	67.1	1119	49.1	43.3	23.7	1.09	0.71	0.76	59.8	78.1%
DivPrune	71.2	1287	55.7	49.1	26.9	1.10	0.71	0.77	66.8	87.4%
CDPruner	73.6	1384	57.4	49.6	28.6	1.22	0.71	0.80	**69.6**	**91.0%**
EvoPrune (Ours)	73.1	1337	56.8	51.3	26.6	**1.06**	**0.71**	**0.75**	68.7	89.8%

LongVideoBench (Wu et al., 2024), and Video-MME (Fu et al., 2025), which assess temporal reasoning, motion perception, and long-range consistency. Together, these benchmarks provide a comprehensive and balanced evaluation of pruning performance across diverse multimodal scenarios.

Compared Methods. We compare our method against state-of-the-art visual token pruning approaches spanning major methodological paradigms: attention-based (FasterVLM (Zhang et al., 2024a)), hybrid attention- and similarity-based (VisPruner (Zhang et al., 2025b)), similarity-based (DART (Wen et al., 2025), DivPrune (Alvar et al., 2025)), and query- and similarity-based (CDPruner (Zhang et al., 2025c)). This diverse suite of baselines ensures a comprehensive evaluation, demonstrating that our improvements are robust across distinct architectural paradigms and task settings.

Implementation Details. All experiments are conducted on NVIDIA H20 GPUs to ensure efficient and consistent inference across models. In the *Attention Preservation* module, we set the **Critical Token Ratio (CTR)** to 0.25, preserving the top 25% most important tokens according to attention scores prior to pruning.

For layer-wise token merging, we adopt a *skip* strategy, performing merging at every other visual encoder layer. The total number of merged tokens is uniformly distributed across all merge steps, with a fixed number r of tokens merged per operation. This yields a stable and consistent compression schedule, avoiding overly aggressive early-stage pruning while preserving sufficient information flow in deeper layers. In addition to the skip strategy, we evaluate alternative merging schedules to analyze efficiency–accuracy trade-offs, with detailed results provided in the Appendix.

4.2 Main Results

EvoPrune for image understanding We evaluate EvoPrune on image-based benchmarks using LLaVA-1.5-7B. We report end-to-end (overall) latency alongside its components: *pre-LLM* (visual encoding and token pruning) and *LLM* (language model inference). As shown in Table 1, EvoPrune consistently achieves the optimal efficiency-accuracy trade-off across all token retention levels. At a 77.8% token reduction (128 retained tokens), EvoPrune yields the highest average accuracy (74.9) and relative performance (97.9%) while achieving the lowest overall latency (0.84×). Compared to the strongest baseline, CDPruner, EvoPrune improves average accuracy by **1.0%** and reduces latency by **7.7%**. Even under aggressive compression (64 and 32 tokens), EvoPrune main-

Table 2: **Performance comparison on video benchmarks. Relative Latency** denotes the execution time of the pre-LLM, LLM, and overall stages normalized to the unpruned baseline (Native). **Acc.** represents the average performance across all benchmarks, and **Rel.** indicates the mean relative accuracy (%) compared to the baseline.

Method	MVBench ↑	LongVideoBench ↑			Video-MME ↑				Relative Latency ↓			Avg. ↑	
Metric	test	val	perception	relation	w/o sub	short	medium	long	pre-LLM	LLM	Overall	Acc.	Rel.
Upper Bound, All 64 × 169 Tokens (100%)													
Native	58.0	56.0	64.0	48.9	59.7	71.3	58.0	49.7	1.00	1.00	1.00	57.9	100.0%
Retain 64 × 64 Tokens (↓ 62.1%)													
FasterVLM	58.3	54.9	61.6	49.0	59.4	70.1	57.4	50.6	1.01	0.45	0.77	57.5	99.4%
VisPruner	58.3	54.9	61.6	49.0	59.4	70.1	57.4	50.6	1.01	0.45	0.77	57.5	99.4%
DART	57.8	54.2	61.3	47.9	59.3	70.6	57.3	50.1	1.04	0.45	0.78	57.1	98.7%
DivPrune	57.8	54.8	62.4	48.2	60.0	71.2	58.0	50.8	1.01	0.45	0.77	57.5	99.4%
CDPruner	58.1	55.1	61.3	49.7	59.6	70.8	57.0	51.0	1.01	0.45	0.77	57.6	99.5%
EvoPrune (Ours)	56.9	55.5	61.9	49.7	60.6	70.9	58.9	51.9	**0.54**	**0.45**	**0.55**	**57.7**	**99.7%**
Retain 64 × 32 Tokens (↓ 81.1%)													
FasterVLM	56.8	52.7	58.9	47.2	58.1	67.1	56.6	50.6	1.01	0.30	0.71	55.8	96.5%
VisPruner	57.2	53.5	60.0	47.8	58.3	69.4	56.2	49.2	1.01	0.30	0.71	56.3	97.3%
DART	57.3	53.5	60.8	47.2	59.0	69.3	57.7	50.3	1.02	0.30	0.71	56.6	97.8%
DivPrune	56.9	54.4	60.8	48.0	58.6	68.4	56.9	50.5	1.01	0.30	0.71	56.6	97.8%
CDPruner	57.2	54.5	61.8	48.0	58.6	68.8	56.3	50.8	1.01	0.30	0.71	56.7	97.9%
EvoPrune (Ours)	56.1	55.1	62.7	48.5	58.7	68.3	57.2	50.6	**0.50**	**0.30**	**0.47**	**56.7**	**98.0%**
Retain 64 × 16 Tokens (↓ 90.5%)													
FasterVLM	54.5	51.2	57.0	46.1	55.8	63.8	54.1	49.6	1.01	0.24	0.68	53.8	93.0%
VisPruner	55.2	51.4	58.9	44.8	56.3	65.9	54.0	49.1	1.01	0.24	0.68	54.3	93.8%
DART	55.5	53.0	61.1	45.9	57.6	67.6	56.0	49.3	1.02	0.24	0.68	55.4	95.7%
DivPrune	56.0	53.1	60.3	46.8	56.9	66.1	55.3	49.2	1.01	0.24	0.68	55.3	95.6%
CDPruner	55.9	52.7	60.5	45.8	57.2	66.7	55.4	49.6	1.01	0.24	0.68	55.3	95.5%
EvoPrune (Ours)	55.8	53.2	58.9	48.2	57.4	65.8	56.0	50.3	**0.46**	**0.24**	**0.43**	**55.4**	**95.8%**

tains robust performance and the fastest inference times across all metrics. These results demonstrate that EvoPrune preserves semantic richness in highly compressed representations and generalizes effectively across diverse image-understanding tasks, validating it as a scalable and efficient pruning framework for MLLMs.

EvoPrune for video understanding The benefits of EvoPrune's early-stage visual token pruning are particularly pronounced in the video domain. Following the standard LLaVA-Video protocol, we uniformly sample 64 frames per video and resize them to 384×384, yielding 64×169 visual tokens after intermediate pooling. Crucially, baseline methods prune tokens after this pooling step, operating on an already reduced token set. Conversely, EvoPrune prunes earlier within the visual encoder, handling a substantially larger token space. This introduces a stricter reliability challenge, as the model must identify and preserve salient tokens prior to pooling without degrading performance.

As shown in Table 2, EvoPrune consistently outperforms existing methods across multiple video benchmarks. Retaining 64×64 tokens (62.1% reduction) yields 57.7% average accuracy while reducing overall latency to 0.55×. Under aggressive pruning to 64×32 tokens (81.1% reduction), EvoPrune maintains 98.0% relative accuracy, reducing pre-LLM and overall latency to 0.50× and 0.47×, respectively. Even under extreme compression to 64×16 tokens (90.5% reduction), it preserves 95.8% relative accuracy (55.4% average) with an overall latency of 0.43×, whereas baselines suffer from larger accuracy drops or inferior speedups (0.68× latency). These results demonstrate that EvoPrune successfully executes early-stage pruning within the visual encoder, offering a superior efficiency–accuracy trade-off despite a larger initial token budget.

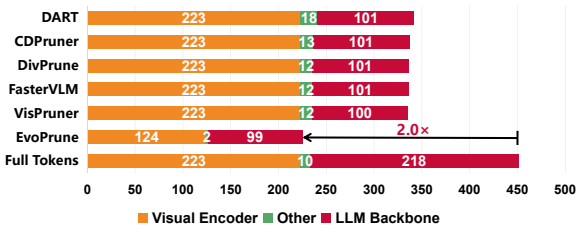

Figure 3: **Breakdown of Time-To-First-Token (TTFT, 1 unit = 10 ms).** The Visual Encoder, Other, and LLM Backbone correspond to the visual encoder, intermediate processing modules (e.g., pooling), and the language model backbone, respectively.

4.3 Efficiency Results

We evaluate EvoPrune against state-of-the-art visual token pruning methods on Video-MME. Figure 3 breaks down the **Time-To-First-Token (TTFT)** across the visual encoder, intermediate modules, and LLM backbone. Existing approaches (e.g., VisPruner, FasterVLM, DivPrune, CDPruner, DART) exhibit notable efficiency bottlenecks in video scenarios; while they reduce LLM computation, the unoptimized visual encoder dominates

Method	MVBench	LongVideoBench	VideoMME	Avg.	Rel.
Native	58.0	56.0	59.7	57.9	100.0%
EvoPrune (w/o Attn&Div)	54.6	54.3	59.0	55.9	96.5%
EvoPrune (w/o Attn)	55.5	54.9	59.7	56.7	97.9%
EvoPrune (w/o Div)	56.2	55.2	60.4	57.3	98.9%
EvoPrune	**56.9**	**55.5**	**60.6**	**57.7**	**99.7%**

Table 3: **Ablation study on score factors.** The **Avg.** denotes the mean score across benchmarks, and **Rel.** indicates the accuracy relative to the unpruned model.

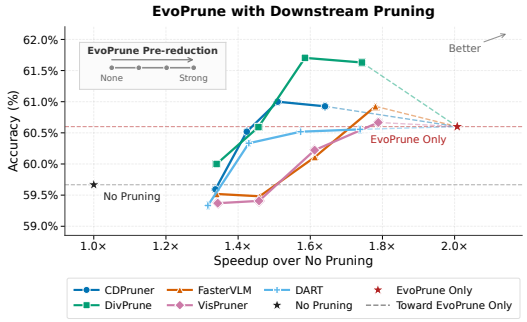

Figure 4: **EvoPrune with downstream pruning.** Each solid curve denotes a downstream pruner with increasingly stronger EvoPrune pre-reduction from left to right. Dashed segments connect the strongest setting to EvoPrune-only.

overall latency.

Consequently, at equivalent pruning rates, these methods yield limited and comparable end-to-end speedups (1.3×–1.4×). Conversely, EvoPrune introduces early-stage pruning within the visual encoder, achieving a 1.8× speedup at this stage, while accelerating intermediate processing by 5× and LLM computation by approximately 2×. Overall, EvoPrune delivers a **2**× TTFT reduction over the full-token baseline, demonstrating holistic pipeline acceleration. These results confirm that EvoPrune preserves accuracy (Section 4.2) while providing substantial hardware deployment speedups, making it highly suitable for real-time video understanding.

4.4 Synergy with Downstream Pruners

As illustrated in Fig. 4, EvoPrune delivers remarkable standalone efficacy, achieving a 2.0× inference speedup and improved accuracy over the unpruned baseline by acting as a semantic denoising filter that removes background redundancies.

Furthermore, EvoPrune exhibits strong orthogonality and synergy when integrated as a pre-reduction module with existing downstream pruning algorithms; as shown by the solid curves, moderate EvoPrune pre-reduction consistently expands the Pareto frontier, yielding superior efficiency and accuracy compared to downstream pruners in isolation (the leftmost points).

This synergy validates that EvoPrune provides a optimized, information-dense token subset for subsequent stages. By filtering out structural noise early in the pipeline, it enables downstream pruners to operate more precisely on a refined feature space, surpassing both standalone settings. Ultimately, these results establish EvoPrune as a versatile, plug-and-play module that enhances the efficiency-accuracy trade-off in multimodal architectures.

4.5 Ablation Study

To analyze the contribution of each component in **EvoPrune**, we conduct ablation studies across three video understanding benchmarks (Table 3), evaluating variants without *Attention Preservation*, the *Diversity Penalty*, or both. Removing both components (w/o AttnDiv) incurs the largest performance drop—reducing the average score from **57.7** to **55.9**—which indicates that naive pruning lacks the constraints necessary for long video understanding. Individually, omitting attention preservation (w/o Attn) causes a greater degradation than omitting the diversity penalty (w/o Div), suggesting that attention-critical tokens are more vital for maintaining spatiotemporal semantics. Ultimately, the full **EvoPrune** model achieves peak performance (**57.7** average score, **99.7%** relative accuracy), demonstrating that attention-guided preservation and diversity-aware selection provide complementary benefits crucial for robust token pruning.

5 Conclusion

We propose EvoPrune, a training-free, plug-and-play visual token pruning method that accelerates MLLMs via early-stage pruning and layer-wise token fusion within the vision encoder. Evaluations demonstrate that EvoPrune consistently yields a superior efficiency-accuracy Pareto frontier over state-of-the-art baselines, particularly on high-resolution inputs. Future work will extend this framework to video via temporal-aware pruning to exploit cross-frame redundancies.

Limitations

Although EvoPrune achieves strong efficiency–performance trade-offs, several limitations remain. First, the current framework primarily focuses on spatial redundancy within the visual encoder and does not explicitly exploit temporal redundancy across video frames. Incorporating temporal-aware token merging may further improve efficiency for long-video understanding. Second, EvoPrune adopts a predefined layer-wise pruning schedule shared across inputs. More adaptive pruning strategies conditioned on input complexity could potentially yield better efficiency–accuracy trade-offs. We leave these directions for future work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. 2025. Divprune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9392–9401.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. 2022. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*.

Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. 2024. Honeybee: Locality-enhanced projector for multimodal llm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13817–13827.

Laming Chen, Guoxin Zhang, and Eric Zhou. 2018. Fast greedy map inference for determinantal point process to improve recommendation diversity. *Advances in neural information processing systems*, 31.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. 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*, pages 19–35. Springer.

Yukang Chen, Fuzhao Xue, Dacheng Li, Qinghao Hu, Ligeng Zhu, Xiuyu Li, Yunhao Fang, Haotian Tang, Shang Yang, Zhijian Liu, and 1 others. 2024b. Longvila: Scaling long-context visual language models for long videos. *arXiv preprint arXiv:2408.10188*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024c. 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*, pages 24185–24198.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in neural information processing systems*, 36:49250–49267.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, and 1 others. 2023. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, and 2 others. 2025. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *Preprint*, arXiv:2405.21075.

Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang, Jianyu Jiang, Jiawei Wang, and 1 others. 2025. Seed1. 5-vl technical report. *arXiv preprint arXiv:2505.07062*.

Ziheng Jia, Zicheng Zhang, Jiaying Qian, Haoning Wu, Wei Sun, Chunyi Li, Xiaohong Liu, Weisi Lin, Guangtao Zhai, and Xiongkuo Min. 2025. Vqa2: visual question answering for video quality assessment. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 6751–6760.

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

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Lou, Limin Wang, and Yu Qiao. 2024b. Mvbench: A comprehensive multi-modal video understanding benchmark. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 22195–22206. IEEE.

Xirui Li, Chao Ma, Xiaokang Yang, and Ming-Hsuan Yang. 2024c. Vidtome: Video token merging for zero-shot video editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7486–7495.

Yanwei Li, Chengyao Wang, and Jiaya Jia. 2024d. Llama-vid: An image is worth 2 tokens in large language models. In *European Conference on Computer Vision*, pages 323–340. Springer.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024b. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Ting Liu, Liangtao Shi, Richang Hong, Yue Hu, Quanjun Yin, and Linfeng Zhang. 2024c. Multi-stage vision token dropping: Towards efficient multimodal large language model. *arXiv preprint arXiv:2411.10803*.

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. 2024d. Mmbench: Is your multi-modal model an all-around player? In *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*, pages 216–233. Springer.

Yuliang Liu, Biao Yang, Qiang Liu, Zhang Li, Zhiyin Ma, Shuo Zhang, and Xiang Bai. 2024e. Textmonkey: An ocr-free large multimodal model for understanding document. *arXiv preprint arXiv:2403.04473*.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. 2024. Deepseek-vl: Towards real-world vision-language understanding. *Preprint*, arXiv:2403.05525.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. 2021. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949.

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

Kele Shao, Keda Tao, Can Qin, Haoxuan You, Yang Sui, and Huan Wang. 2025. Holitom: Holistic token merging for fast video large language models. *arXiv preprint arXiv:2505.21334*.

Leqi Shen, Guoqiang Gong, Tao He, Yifeng Zhang, Pengzhang Liu, Sicheng Zhao, and Guiguang Ding. 2025. Fastvid: Dynamic density pruning for fast video large language models. *arXiv preprint arXiv:2503.11187*.

Xiaoqian Shen, Yunyang Xiong, Changsheng Zhao, Lemeng Wu, Jun Chen, Chenchen Zhu, Zechun Liu, Fanyi Xiao, Balakrishnan Varadarajan, Florian Bordes, and 1 others. 2024. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*.

Baifeng Shi, Boyi Li, Han Cai, Yao Lu, Sifei Liu, Marco Pavone, Jan Kautz, Song Han, Trevor Darrell, Pavlo Molchanov, and 1 others. 2025. Scaling vision pre-training to 4k resolution. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9631–9640.

Yan Shu, Zheng Liu, Peitian Zhang, Minghao Qin, Junjie Zhou, Zhengyang Liang, Tiejun Huang, and Bo Zhao. 2025. Video-xl: Extra-long vision language model for hour-scale video understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26160–26169.

Dingjie Song, Wenjun Wang, Shunian Chen, Xidong Wang, Michael X Guan, and Benyou Wang. 2025. Less is more: A simple yet effective token reduction method for efficient multi-modal llms. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7614–7623.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, and 1 others. 2024. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18221–18232.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

An-Lan Wang, Bin Shan, Wei Shi, Kun-Yu Lin, Xiang Fei, Guozhi Tang, Lei Liao, Jingqun Tang, Can Huang, and Wei-Shi Zheng. 2025a. Pargo: Bridging vision-language with partial and global views. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7491–7499.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Yi Wang, Xinhao Li, Ziang Yan, Yinan He, Jiashuo Yu, Xiangyu Zeng, Chenting Wang, Changlian Ma, Haian Huang, Jianfei Gao, and 1 others. 2025b. Internvideo2. 5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*.

Zichen Wen, Yifeng Gao, Shaobo Wang, Junyuan Zhang, Qintong Zhang, Weijia Li, Conghui He, and Linfeng Zhang. 2025. Stop looking for important tokens in multimodal language models: Duplication matters more. *arXiv preprint arXiv:2502.11494*.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. 2024. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Preprint*, arXiv:2407.15754.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, and 1 others. 2024. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. *arXiv preprint arXiv:2410.17247*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. 2025. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19792–19802.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2024. Mm-vet: Evaluating large multimodal models for integrated capabilities. In *International conference on machine learning*. PMLR.

Ce Zhang, Kaixin Ma, Tianqing Fang, Wenhao Yu, Hongming Zhang, Zhisong Zhang, Yaqi Xie, Katia Sycara, Haitao Mi, and Dong Yu. 2025a. Vscan: Rethinking visual token reduction for efficient large vision-language models. *arXiv preprint arXiv:2505.22654*.

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

Qizhe Zhang, Aosong Cheng, Ming Lu, Zhiyong Zhuo, Minqi Wang, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. 2024a. [cls] attention is all you need for training-free visual token pruning: Make vlm inference faster. *arXiv e-prints*, pages arXiv–2412.

Qizhe Zhang, Mengzhen Liu, Lichen Li, Ming Lu, Yuan Zhang, Junwen Pan, Qi She, and Shanghang Zhang. 2025c. Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. *Preprint*, arXiv:2506.10967.

Yi-Fan Zhang, Qingsong Wen, Chaoyou Fu, Xue Wang, Zhang Zhang, Liang Wang, and Rong Jin. 2024b. Beyond llava-hd: Diving into high-resolution large multimodal models. *arXiv preprint arXiv:2406.08487*.

Yipeng Zhang, Yifan Liu, Zonghao Guo, Yidan Zhang, Xuesong Yang, Chi Chen, Jun Song, Bo Zheng, Yuan Yao, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2024c. Llava-uhd v2: an mllm integrating high-resolution feature pyramid via hierarchical window transformer. *arXiv preprint arXiv:2412.13871*.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and 1 others. 2024d. Sparsevlm: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*.

Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. 2024e. Llava-next: A strong zero-shot video understanding model.

A Qualitative Visualization of Token Merging Dynamics

To provide further insight into the spatial behavior of EvoPrune, Fig. 5 visualizes the token merging process by comparing the original high-resolution image, the standard uncompressed patch grid, and the merged patch layout produced by EvoPrune. The visualization demonstrates how EvoPrune balances token reduction with semantic preservation through spatially adaptive merging.

Specifically, regions with low semantic variation, such as the glowing area inside the fireplace (green box), exhibit more aggressive merging of visually similar patches. In contrast, regions containing fine-grained textures and high-frequency details, such as the Christmas tree (red box), retain a denser token distribution. These observations suggest that EvoPrune selectively removes redundant visual information while preserving structurally and semantically important regions.

Overall, the merged patches largely preserve scene layout and object boundaries despite substantial token reduction, indicating that EvoPrune maintains the core visual structures required for downstream multimodal reasoning while significantly reducing computational overhead.

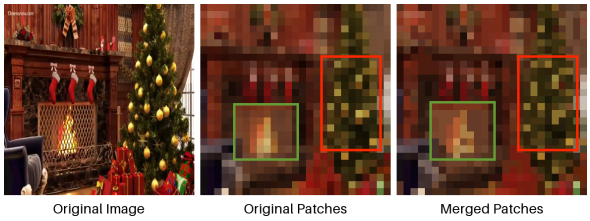

Figure 5: **Qualitative visualization of token merging dynamics.** The green box indicates a tendency for broader merging in visually redundant regions (e.g., the fireplace), while the red box shows the relative preservation of fine-grained tokens in semantically denser regions (e.g., the Christmas tree).

B Analysis of Layer-wise Merging Strategies

This appendix provides a detailed analysis of the layer-wise token merging strategies explored in EvoPrune. Our goal is to understand how different distributions of the pruning budget across encoder layers affect the trade-off between inference efficiency and model performance.

All strategies are normalized to ensure that the total pruning budget equals R. This design space enables flexible control over where pruning is applied during visual encoding, allowing us to study the trade-offs between early-stage acceleration and representational capacity under different layer-wise pruning patterns.

B.1 Layer-wise Merging Strategy Design

We explore a set of layer-wise token merging strategies to investigate how different distributions of the pruning budget $\mathbf{r}$ across encoder layers affect model efficiency and performance. Each strategy specifies how the total pruning budget R is allocated among selected layers.

- **Mean:** The pruning budget is uniformly distributed across all selected encoder layers, assigning equal weights to each layer.
- **Skip:** Token merging is applied at alternating layers, where only every other layer participates in pruning.
- **First[N]:** Token merging is restricted to the first N encoder layers, while the remaining layers are left unpruned.
- **Last[N]:** Token merging is applied only to the last N encoder layers, concentrating the pruning budget on deeper representations.
- **Increasing[α]:** The pruning budget increases with layer depth, assigning progressively larger budgets to deeper layers. The parameter α controls the growth rate, with $\alpha = 1$ by default.
- **Decreasing[α]:** The pruning budget decreases with layer depth, emphasizing stronger pruning in earlier layers. The parameter α similarly controls the decay rate, with $\alpha = 1$ by default.

B.2 Experimental Results and Analysis

We evaluate the above strategies on representative image and video benchmarks under the same overall pruning budget. The results reveal clear differences in efficiency–accuracy trade-offs among the strategies. For clarity, we denote each experimental setting as *merging strategy* + *token budget*. For example, *First16* indicates the *First* merging strategy with a token budget of 16.

Our results indicate that the stage at which token pruning is applied significantly influences both performance and inference speed. Strategies that

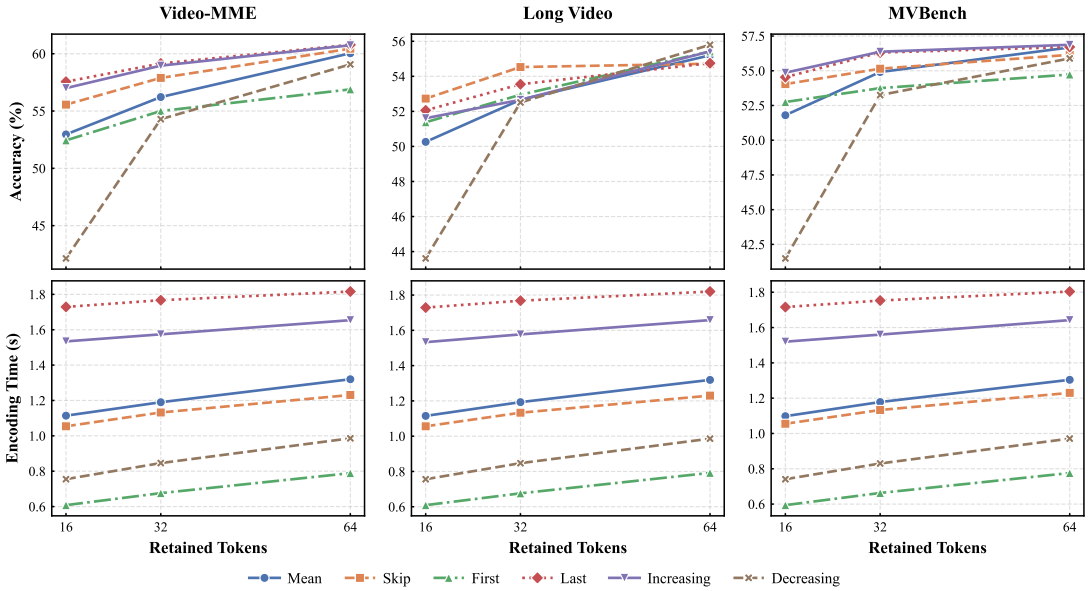

Figure 6: **Performance comparison of different layer-wise merging strategies across video benchmarks** We present the Accuracy (top row) and Encoding Time (bottom row) as functions of the retained token budget $B \in \{16, 32, 64\}$. Each curve represents a specific layer-wise allocation pattern. Specifically, we set the window size $N = 13$ for *First* and *Last* (out of 26 total layers), and the growth/decay rate $\alpha = 1$ for *Increasing* and *Decreasing*.

concentrate pruning at the early stages, such as *First* and *Decreasing*, exhibit the lowest inference times (e.g., *First16* reaches 0.608s on Video-MME). However, they suffer from a drastic degradation in accuracy, particularly in *Decreasing16* where the accuracy on MVBench drops to 41.49%. This suggests that aggressive pruning in early layers destroys fine-grained spatial-temporal features before they can be effectively integrated. Conversely, *Last* and *Increasing* strategies maintain high accuracy (up to 60.78% on Video-MME) by preserving full token density in earlier layers, but they are computationally expensive, with *Last64* being nearly 1.5× slower than the average.

Among all configurations, the **Skip** strategy provides the most robust Pareto frontier. By applying token merging at alternating layers, it prevents the accumulation of pruning error in any single stage of the visual encoder. Quantitatively, *Skip64* achieves a superior accuracy of 60.44% on Video-MME with a latency of 1.231s, outperforming the *Mean* strategy (60.04%, 1.320s) in both metrics. Even at the most aggressive budget ($B = 16$), *Skip* maintains 55.56% accuracy on Video-MME, significantly higher than *Mean* (52.96%) and *First* (52.44%). This demonstrates that a staggered pruning pattern effectively balances the preservation of representational capacity with computational efficiency. Consequently, we adopt the *Skip* strategy as our default allocation pattern for all experiments.

C Analysis of Spatial Token Reduction Strategies

To systematically validate the efficacy of our proposed merging approach, Fig. 7 presents a controlled comparison against alternative token reduction paradigms across three video comprehension benchmarks. Specifically, we evaluate EvoPrune's unconstrained merging strategy against random token dropping (Random), saliency-based direct discarding (Discard), and locality-constrained merging (LocalMerge) under strictly matched token budgets (e.g., 16, 32, and 64 tokens). As illustrated, all merging-based methods consistently outperform both the random and direct discard baselines across varying compression ratios. This empirical trend robustly demonstrates that simply discarding tokens inevitably leads to irreversible information loss, whereas merging mechanisms effectively preserve semantic integrity by aggregating visual evidence. It confirms that the performance stability of EvoPrune is intrinsically derived from its feature preservation capability rather than merely from the reduction of sequence length.

Furthermore, a detailed comparison between locality-constrained strategies and our global merg-

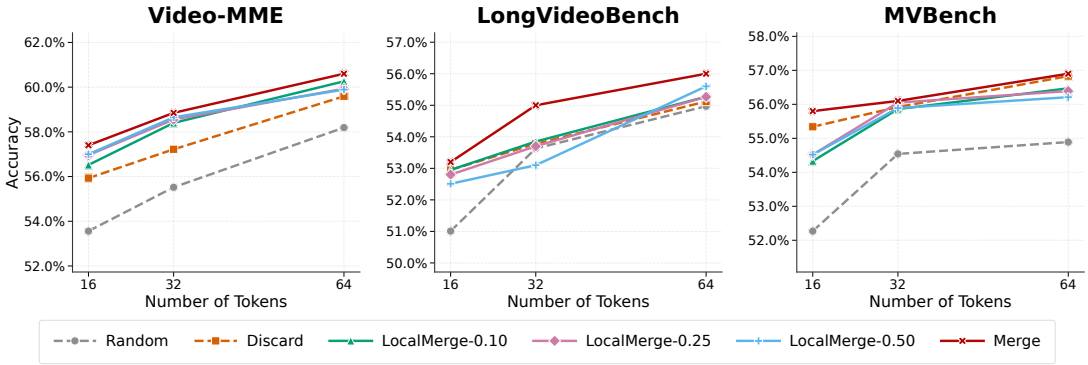

Figure 7: **Accuracy comparisons under different pruning strategies across three benchmarks.** The numerical suffix in 'LocalMerge' indicates the predefined range constraint for local token merging. Unconstrained global merging ('Merge') consistently yields the most optimal performance.

ing approach highlights the inherent limitations of spatial restrictions. The *LocalMerge* variants, which restrict token clustering to predefined neighborhood ranges (denoted by the thresholds 0.10, 0.25, and 0.50), generally exhibit sub-optimal accuracy compared to the unconstrained global *Merge* strategy. This performance gap subtly indicates that visually redundant information—such as static background elements or temporally repeated object textures—is frequently distributed across distant spatial or temporal regions within the video representations.

By eschewing strict locality constraints, EvoPrune is capable of identifying and fusing highly similar tokens regardless of their physical distance within the feature map. Consequently, this globally adaptive aggregation prevents the inadvertent separation or loss of scattered but semantically identical features. These findings substantially validate our design choice: prioritizing global semantic similarity over local spatial proximity is a more optimal approach for token reduction in complex multimodal reasoning tasks.

D Compatibility with FlashAttention-style Kernels

EvoPrune does not require materializing the full $N_l \times N_l$ attention matrix. Instead, it only relies on lightweight column-wise attention statistics, which can be accumulated in a streaming manner during attention computation.

Therefore, EvoPrune remains compatible with IO-aware attention backends such as FlashAttention, while preserving the same $O(N_l D)$ memory/IO complexity. The additional reductions introduce only minor overhead compared with standard attention computation.Moreover, since EvoPrune progressively reduces the token count across layers, the downstream savings in both computation and memory traffic quickly dominate the small cost of statistic accumulation.

E Benchmarks

E.1 Image benchmarks

VQAv2 (Jia et al., 2025) A large-scale Visual Question Answering benchmark that emphasizes true visual reasoning by mitigating language priors inherent in earlier VQA datasets. Built from images in MS-COCO with paired natural language questions and 10 human-annotated answers, VQAv2 collects "complementary" image pairs where the same question yields different answers, forcing models to rely on visual content rather than statistical shortcuts. It contains 1.1M image–question pairs across train/val/test and is widely used to evaluate open-ended VQA performance using standard accuracy metrics.

MME (Fu et al., 2023) A comprehensive evaluation benchmark for Multimodal Large Language Models (MLLMs), designed to jointly assess both perception and high-level cognition across 14 diverse subtasks. To ensure fair comparison and avoid leakage from public datasets, instruction–answer pairs are manually crafted with concise prompts, minimizing prompt engineering bias. MME provides quantitative metrics and a systematic leaderboard of 30 advanced MLLMs, revealing persistent gaps in current multimodal understanding and guiding future model improvements.

MMBench (Liu et al., 2024d) A finely-designed evaluation suite that systematically assesses the

multimodal perception and reasoning skills of vision-language models across diverse visual understanding tasks. It comprises a large, meticulously curated set of bilingual multiple-choice questions spanning dozens of fine-grained abilities, with rigorous quality control and a novel CircularEval strategy to convert free-form model outputs into standardized choices. Integrated into VLMEvalKit, MMBench offers robust, scalable, and objective benchmarking that reveals distinct capability profiles of contemporary multimodal models.

MMVet (Yu et al., 2024) A evaluation benchmark for large multimodal models, designed to assess their integrated vision–language reasoning beyond isolated tasks. It defines six core vision–language capabilities—recognition, knowledge, spatial awareness, language generation, OCR, and mathematics—and examines their 16 pairwise integrations through open-ended image–question pairs. A unified LLM-based evaluator scores free-form responses, enabling consistent comparison across diverse question types and answer styles. MM-Vet thus provides insight into models' holistic multimodal competence, revealing strengths and weaknesses beyond aggregate scores.

E.2 Video benchmarks

MVBench (Li et al., 2024b) A benchmark emphasizing dynamic temporal reasoning beyond static frame analysis to rigorously assess multimodal large language models' video understanding capabilities. It defines 20 temporally-oriented tasks by systematically transforming traditional spatial vision tasks into sequence-dependent challenges, with each task instantiated as multiple-choice QA derived from public video annotations over diverse sources. This design enables fair, scalable evaluation of perception and cognition across actions, interactions, motion, and event reasoning, revealing substantial gaps in current model performance. The metric of interest is task accuracy over the full suite of temporal tests.

LongVideoBench (Wu et al., 2024) A long-context question-answering benchmark that probes the ability of multimodal models to comprehend interleaved video and language inputs spanning up to an hour, requiring detailed retrieval and reasoning over extended temporal sequences rather than isolated frames. It comprises 3,763 web-sourced videos and 6,678 human-annotated multiple-choice questions across 17 fine-grained referring-reasoning categories, explicitly designed to assess long-form multimodal understanding. Evaluations show significant challenges for current LMMs, with performance gains tied to processing denser frame inputs, positioning it as a rigorous testbed for future long-context LMMs.

Video-MME (Fu et al., 2025) A full-spectrum evaluation benchmark for multimodal large language models on video analysis, designed to stress temporal, domain, and multimodal comprehension beyond static frames. It comprises 900 manually curated videos totaling 254 hours across six diverse visual domains and 30 subcategories, with 2,700 expert-annotated multiple-choice QA pairs probing recognition, temporal reasoning, and event understanding using both frames and auxiliary modalities (subtitles, audio). Videos span short (<2min), medium (4–15min) and long (30–60min) durations, enabling rigorous assessment of long-context and multimodal capabilities. Standardized accuracy metrics facilitate comparison across state-of-the-art models.

F Compared Methods

FasterVLM (Zhang et al., 2024a) Existing visual token pruning in VLMs often relies on text–visual cross-attention, which misaligns with true token importance and degrades accuracy at high pruning rates. FasterVLM proposes a training-free method that uses [CLS] attention from the visual encoder to more accurately rank and prune visual tokens before they enter the multimodal projection, reducing redundant tokens without LLM interaction. This early pruning significantly speeds up inference while retaining the majority of performance across various VLM architectures and reduction ratios.

VisPruner (Zhang et al., 2025b) The first work to demonstrate that text–visual attention scores are a suboptimal criterion for visual token pruning in large vision–language models. Instead, it exploits intrinsic visual cues to identify and retain a compact set of informative tokens and removes redundant ones via similarity filtering, preserving diverse visual information. This plug-and-play approach sustains performance across VLM architectures without any training, achieving substantial FLOPs and latency reduction while maintaining comparable accuracy to full models.

DART (Wen et al., 2025) The first framework to explicitly enforce disease-relevant alignment between visual findings and textual content in radiology report generation. It formulates a two-stage pipeline where disease-aware image–text retrieval based on contrastive learning is used to obtain initial reports that closely match pathological patterns, followed by a self-correcting re-alignment module that refines the generated text for clinical consistency. Evaluated on MIMIC-CXR and IU X-ray benchmarks, DART achieves state-of-the-art performance in both report quality and disease classification, demonstrating improved trustworthiness of automated radiology reports.

DivPrune (Alvar et al., 2025) Formulates visual-token pruning as a Max–Min Diversity Problem to explicitly maximize representational diversity among retained tokens, rather than relying on attention scores or calibration. It greedily selects a subset of visual tokens with maximal pairwise distances, pruning the rest to reduce redundancy and preserve broad visual information without fine-tuning. DivPrune operates as a plug-and-play module for large multimodal models, achieving state-of-the-art accuracy across 16 image/video language benchmarks while significantly reducing inference latency and memory usage.

CDPruner (Zhang et al., 2025c) A novel training-free strategy that departs from traditional attention- or similarity-based pruning by maximizing the conditional diversity of retained visual tokens conditioned on the instruction. CDPruner defines conditional similarity between tokens and formulates pruning as a determinantal point process (DPP) to select a diverse subset that aligns with the user query. This model-agnostic method significantly reduces FLOPs and latency while preserving task accuracy across diverse vision-language benchmarks, establishing new state-of-the-art performance.