Combating Textual Noise and Redundancy: Entropy-Aware Dense Visual Token Pruning

Xuehui Wang[†], Xuankun Yang[†], and Wei Shen[✉]

Shanghai Jiao Tong University, Shanghai, China
{wangxuehui,kk-dao,wei.shen}@sjtu.edu.cn
Codes: https://github.com/SJTU-DeepVisionLab/EADP

Abstract. Visual token pruning is a crucial strategy for accelerating VLMs by compressing redundant image patches, yet existing methods often fail to preserve critical cues under dense instructions and fine-grained queries. In this paper, we investigate this failure and identify two underlying bottlenecks: the widespread dispersion of textual noise that corrupts dense cross-modal scoring, and the feature fragmentation inherent to standard token selection. To address these issues, we propose **E**ntropy-**A**ware **D**ense **P**runing (EADP), a framework that reformulates pruning as a structured compression problem. EADP first leverages statistical entropy to quantify and filter out textual noise, yielding a robust, fine-grained instruction relevance score. Subsequently, instead of naive Top-K selection, EADP casts token selection as a submodular maximization problem with a spatial prior, explicitly ensuring a holistic and non-redundant visual representation. Extensive experiments demonstrate that EADP improves the accuracy-efficiency trade-off of VLMs, robustly preserving fine-grained visual cues under strict token budgets while achieving SoTA performance on challenging multimodal benchmarks.

Keywords: Visual token pruning · Efficient VLM

1 Introduction

Vision-language models (VLMs) [4,6,17,18,25,33,36,37] are foundational for multimodal understanding, yet their deployment is hindered by the high computational cost of long visual contexts. In particular, modern VLMs represent images as dense grids of patch tokens, directly driving up quadratic self-attention costs and end-to-end latency [10, 53, 70]. This makes *visual token pruning* an appealing and increasingly necessary strategy for real-time and resource-constrained applications: retaining only a compact subset of informative tokens significantly accelerates inference with minimal performance degradation [2, 5, 12, 35, 49, 52, 56, 60, 63, 67].

In practice, however, things rarely go that smoothly. When we push pruning to the regimes that actually matter for deployment, tight budgets, dense instructions, fine-grained cues, and challenging negative queries, existing pruning systems often become brittle, easily discarding small yet decisive visual cues. This raises a

† Equal contribution. ✉ Corresponding author.

deceptively simple question: *what exactly goes wrong when we prune?* To answer this, we dissect the standard visual pruning pipeline, which typically consists of two stages: scoring token importance and selecting the Top-K tokens [38, 73]. In the scoring stage, the prevailing recipe relies on a single global text feature (e.g., the CLIP [47] `EOS` token) to evaluate each visual token [31, 65, 66, 69]. While cheap and stable, it often acts as a highly compressed summary, missing fine-grained details. The intuitive fix for such coarse global guidance is dense guidance, computing cross-modal similarity between *every* text token and visual token. Surprisingly, this "more informative" strategy often refuses to help. We observe that functional words and punctuation typically produce broadly dispersed, near-uniform similarity responses [9, 54]. Together, they accumulate into an indiscriminate "noise floor" that drowns out the sparse, localized peaks of truly meaningful semantic entities. Moreover, even if we succeed in obtaining a clean, fine-grained relevance score, a second surprise awaits at the selection stage. The standard Top-K rule, though universally used, is a flawed compression strategy under strict budgets. It tends to over-concentrate on a few highly discriminative regions, wasting the limited token budget on redundant local maxima (e.g., repeatedly sampling the head of a target object) while leaving other integral semantic parts entirely unrepresented.

These observations reveal that visual pruning is not merely a scoring task, but a structured compression problem requiring both noise-robust saliency and holistic visual coverage. To address this, we propose **Entropy-Aware Dense Pruning (EADP)**, a lightweight framework designed to make instruction relevance fine-grained and token selection non-redundant. Concretely, EADP first reformulates the instruction relevance scoring into a dual-stream process. We begin by computing dense cross-modal similarities between all non-`EOS` CLIP text tokens and VLM visual tokens. To eliminate the dispersed textual noise identified earlier, we introduce an entropy-guided denoising mechanism that directly calculates the entropy of each text token's spatial similarity distribution. By filtering out high-entropy tokens and weighting the retained low-entropy ones, we obtain a clean dense guidance score. Finally, we fuse this denoised dense signal with the global `EOS` semantic context, yielding a comprehensive instruction relevance map that possesses both local precision and macroscopic stability.

With a robust instruction relevance map in hand, EADP proceeds to tackle the feature fragmentation and redundancy caused by standard Top-K selection. We first refine the score map to preserve spatial integrity: a lightweight Gaussian smoothing is applied to incorporate local structural context, followed by a score polarization step that exponentially re-amplifies the core visual entities against the smoothed background. Armed with these structurally aware scores, we abandon the naive Top-K heuristic and cast the final token selection as a facility location submodular maximization problem [24, 34, 46]. This mathematically principled objective shifts the focus from mere peak-chasing to holistic representativeness. It guarantees that all semantic parts of the original image are well-covered by the compressed subset, naturally penalizing redundant tokens and yielding a compact, highly informative visual representation for the downstream LLM.

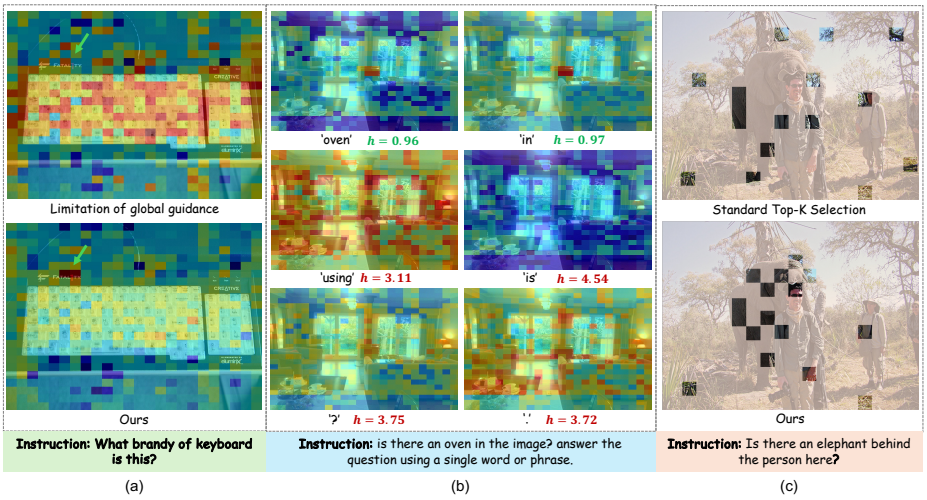

Fig. 1: (a) illustrates a limitation of global guidance: it tends to attend to background regions. (b) highlights the dispersion phenomenon caused by textual noise. (c) reveals the issues of feature fragmentation and selection redundancy.

In summary, our main contributions are three-fold:

- Crucial Insights into Pruning Bottlenecks: We systematically analyze the failure modes of existing visual token pruning paradigms. We identify the dispersion phenomenon of textual noise in dense guidance, and reveal feature fragmentation and redundancy inherent to standard Top-K selection.
- Entropy-Aware Dense Scoring: We propose a novel dispersion-aware scoring mechanism that elegantly quantifies and filters textual noise using statistical entropy. Without relying on external NLP parsers, this approach extracts highly precise, fine-grained instruction relevance and fuses it with global semantic context for robust guidance.
- Submodular Token Selection with Spatial Priors: We reformulate visual token selection as a facility location submodular maximization problem. Coupled with a spatial smoothing and score polarization prior, this objective explicitly guarantees holistic feature representativeness and penalizes local redundancy, fundamentally overcoming the limitations of greedy Top-K sampling.

2 Related Works

Large Vision-Language Models. Recent Vision-Language Models (VLMs) extend LLMs with visual encoders to enable robust multimodal instruction following [1, 26, 47]. Building upon early contrastive pretraining and unified architectures [8, 21, 30, 47, 53], current paradigms predominantly align pretrained vision backbones with LLMs via lightweight connectors and multimodal instruction tuning [29, 37, 62, 72]. Further scaling and advanced prompting techniques

(e.g., Chain-of-Thought) have unlocked emergent capabilities in fine-grained and compositional reasoning [1, 3, 7, 23, 55]. However, the pursuit of higher-resolution perception inevitably produces massive visual token sequences. The resulting quadratic computational overhead poses a severe bottleneck for practical inference and deployment [26, 37], urgently motivating research into compact representations and visual token pruning.

Visual Token Pruning. Visual token pruning aims to drop redundant patches while preserving task-relevant information, broadly following three paths. *(i) Score-/saliency-based pruning* [5, 38, 49, 52, 63, 69, 70] evaluates token importance to retain the top-k subset, utilizing heuristics, cross-modal dependency, or lightweight predictors. *(ii) Structure-aware pruning* [2, 12, 35, 60, 67] exploits spatial layouts by grouping patches or selecting informative regions to maintain structural coverage and diversity. *(iii) Training-/policy-based pruning* [31, 56, 65] learns dynamic, context-adaptive token-dropping decisions end-to-end. Despite promising speedups, existing methods heavily rely on coarse global guidance and heuristic Top-K sampling, suffering from textual noise and feature fragmentation. In contrast, our EADP resolves these bottlenecks by filtering dispersed dense textual noise via Information Entropy and mathematically guaranteeing holistic feature representativeness through Submodular Maximization.

3 Observations and Insights

Recent methods [12,65,69] tend to combine feature similarity among visual tokens with instruction relevance to derive the importance score for each visual token and then sample a subset of high score tokens as the final visual tokens. In this section, we conduct pilot studies to revisit existing SoTA pruning paradigms, revealing critical bottlenecks in both importance scoring and token selection.

3.1 The Limitation of Global Guidance

We utilize CDPruner, which adopts the `EOS` token from the CLIP text encoder as the global guidance token to compute instruction relevance, to investigate the efficacy of global guidance. In our experiment, we visualize the instruction relevance maps generated by CDPruner on dense visual tasks, as shown in Fig. 1(a). We notice that the global guidance token tends to highlight broad, semantically vague background regions and does not consistently concentrate on fine-grained critical clues. This observation suggests that the global guidance token acts as a highly compressed semantic summary, which inherently lacks the resolution required to capture fine-grained textual instructions [28, 61]. Consequently, global-guided pruning can be sub-optimal for fine-grained recognition and challenging negative queries where the referenced object is absent from the image.

3.2 The Dispersion Phenomenon of Textual Noise

An intuitive solution to the aforementioned limitation is dense guidance: calculating the cross-modal similarity between every text token in the prompt

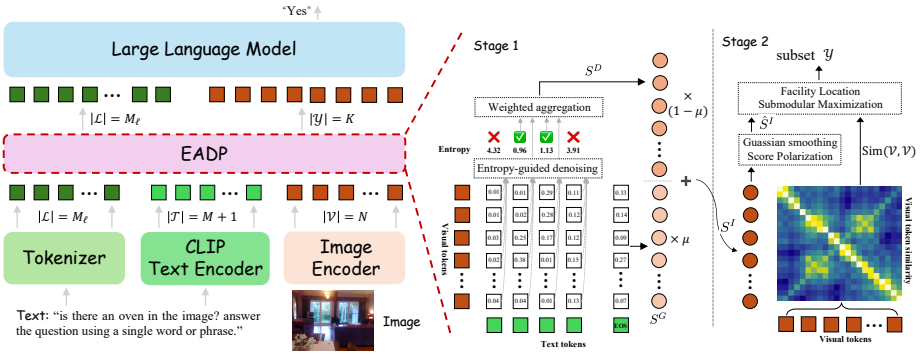

Fig. 2: Overview of the EADP. EADP acts as a plug-and-play module compressing N visual tokens into a highly informative subset of K tokens for the downstream LLM. **Stage 1**: An entropy-guided denoising mechanism filters out high-entropy textual noise to get the dense guidance score S^D. This is fused with the global `EOS` score S^G to yield a robust instruction relevance score S^I. **Stage 2**: After refining S^I via gaussian smoothing and score polarization to prevent feature fragmentation, a submodular maximization objective uses inter-token visual similarity to select the final subset $\mathcal{Y}$.

and the visual tokens. However, our empirical results reveal a counter-intuitive phenomenon: naive dense aggregation yields no performance improvement. To understand why dense guidance fails, we visualize the spatial similarity distribution induced by individual text tokens on visual tokens and we uncover a distinct Dispersion Phenomenon in cross-modal similarity: as illustrated in Fig. 1(b), semantic entity words (e.g., "oven", "cat") exhibit a highly concentrated similarity distribution, accurately localizing their corresponding visual patches. In stark contrast, functional words [9, 48, 58] (e.g., "using", "is") and punctuation marks (e.g., "?") exhibit a uniformly dispersed similarity map across the entire image.

While an individual dispersed text token produces uniformly low response values across visual patches, their cumulative effect is devastating. Since functional words and punctuation typically constitute the majority of a text prompt, aggregating their similarities creates a dominating, indiscriminate "noise floor". This textual noise inflates the scores of irrelevant background regions, diluting and submerging the localized activation peaks generated by the few semantic entity words. Identifying and filtering out these dispersed text tokens, without relying on NLP parsers, is a prerequisite for effective dense pruning. We later show that statistical entropy can quantify this dispersion degree in Sec. 4.2.

3.3 Feature Fragmentation and Selection Redundancy

Even if we could perfectly filter the textual noise and obtain accurate importance scores, the final step, selecting a compact subset of visual tokens, remains non-trivial. To isolate the selection issue, we perform a controlled analysis: we manually choose the prompt words that refer to the target object and use their induced similarity scores as instruction relevance, combined with inter-token feature

similarity, to derive the final importance scores. Subsequently, we apply the standard Top-K selection based on these scores.

As illustrated in Fig. 1(c), we observe a severe "feature fragmentation" issue. Rather than capturing the holistic structure of the referring object, the Top-K selection tends to over-fit to the most highly discriminative local regions (e.g., the tail of a elephant), leaving other integral parts (e.g., the body and head) entirely unrepresented. Furthermore, this naive selection inevitably leads to severe redundancy, where selected tokens heavily cluster within a few high-score peaks, wasting the limited token budget on repetitive local features while completely missing the broader semantic context. These observations highlight three requirements for a robust pruning framework: (1) a spatial smoothing mechanism to propagate activations and incorporate local context, (2) a contrast sharpening operation to highlight core visual entities and suppress smoothed background noise, and (3) a principled selection paradigm that guarantees holistic feature representativeness, ensuring all semantic parts are covered, while penalizing local redundancy. This finding motivates our adoption of a gaussian spatial prior, score polarization, and submodular maximization in Sec. 4.3.

4 Methodology

In this section, we present our Entropy-Aware Dense Pruning (EADP) framework. Building upon the insights from Sec. 3, EADP explicitly quantifies textual dispersion to filter out noisy text tokens, fuses dense and global guidance for robust instruction relevance scoring, and reformulates token selection as a submodular maximization problem to preserve spatial integrity and eliminate redundancy. The overview pipeline of EADP is shown in Fig. 2.

4.1 Preliminaries and Problem Formulation

Given an input image I and a task-specific text prompt P, a VLM typically employs a vision encoder to extract a sequence of visual tokens $\mathcal{V} = \{v_1, v_2, ..., v_N\} \in \mathbb{R}^{N \times d_v}$, where N is the number of visual patches and d_v is the feature dimension. Meanwhile, the LLM tokenizer converts the text prompt into language tokens for generation; we denote these LLM tokens as $\mathcal{L} = \{l_1, l_2, \ldots, l_{M_\ell}\}$. These LLM tokens are concatenated with projected visual tokens and fed into the LLM to autoregressively generate the output.

We additionally encode the same prompt P by a CLIP text encoder to obtain a sequence of CLIP text features $\mathcal{T} = \{t_1, t_2, \ldots, t_M, t_{\texttt{EOS}}\} \in \mathbb{R}^{(M+1)\times d}$, where $t_{\texttt{EOS}}$ is the `EOS` token feature and d is the CLIP embedding dimension. To compute cross-modal cosine similarities between $\mathcal{T}$ and visual tokens from the VLM vision tower, we project visual tokens to the CLIP embedding space via a lightweight projector (implemented identically to CDPruner), producing $\tilde{\mathcal{V}} = \{\tilde{v}_j\}_{j=1}^{N}$ with $\tilde{v}_j \in \mathbb{R}^d$. For simplicity, we use v_j to denote the projected visual token when computing cross-modal similarities below.

The goal of visual token pruning is to identify a compact subset $\mathcal{Y} \subset \mathcal{V}$ with size $|\mathcal{Y}| = K \ll N$ that preserves the most informative visual cues required for

correct generation. This requires (i) an instruction relevance scoring function $\mathcal{F} : v_j \mapsto \mathbb{R}$ and (ii) a selection strategy Φ:

$$\mathcal{Y} = \Phi(\mathcal{F}(\mathcal{V}); K) \in \mathbb{R}^{K \times d_v}. \tag{1}$$

We detail the formulation of $\mathcal{F}$ and Φ in Sec. 4.2 and Sec. 4.3, respectively.

4.2 Dispersion-Aware Dense Scoring and Fusion

In EADP, instruction relevance is computed from two complementary components: a global guidance score following prior work [12, 69], and our proposed dispersion-aware dense guidance score. Both are derived from cross-modal cosine similarities between CLIP text features $\mathcal{T}$ and projected visual tokens $\mathcal{V}$.

Global Guidance Score. We compute similarities between the `EOS` token and visual tokens as in CDPruner [69], yielding a global guidance score $S^G \in \mathbb{R}^N$:

$$s_j^G = \frac{v_j \cdot t_{\texttt{EOS}}}{\|v_j\| \cdot \|t_{\texttt{EOS}}\|}, \quad v_j \in \mathcal{V}. \tag{2}$$

Dispersion-aware Dense Guidance Score. As observed in Secs. 3.1 and 3.2, relying solely on the global token t_{EOS} neglects fine-grained details, while naive dense guidance introduces severe textual dispersion noise from functional words. To address this, we introduce an entropy-guided denoising mechanism. We first compute the dense cosine similarity matrix $C \in \mathbb{R}^{M \times N}$ between *non-`EOS`* CLIP text tokens $\{t_i\}_{i=1}^M$ and all visual tokens $\{v_j\}_{j=1}^N$:

$$c_{i,j} = \frac{t_i \cdot v_j}{\|t_i\| \cdot \|v_j\|}, \quad i \in \{1, \ldots, M\},\ j \in \{1, \ldots, N\}. \tag{3}$$

Since $M \leq 77$ in CLIP, calculating C requires just one lightweight matrix multiplication, adding negligible overhead compared to the LLM attention.

To measure the dispersion degree of a text token's attention, we convert its similarity scores over all visual tokens into a probability distribution via a temperature-controlled Softmax:

$$p_{i,j} = \frac{\exp(c_{i,j}/\tau)}{\sum_{k=1}^{N} \exp(c_{i,k}/\tau)}. \tag{4}$$

We then calculate the Information Entropy $H \in \mathbb{R}^M$ for the tokens:

$$h_i = -\sum_{j=1}^{N} p_{i,j} \log p_{i,j}, \quad H = \{h_1, h_2, ..., h_M\} \tag{5}$$

A higher entropy h_i indicates that the attention of t_i is uniformly dispersed across the image (i.e., textual noise, such as punctuation or functional words), whereas a lower h_i signifies a highly concentrated attention on specific visual entities.

We retain a proportion $q \in (0, 1]$ of text tokens with the *lowest* entropy. Specifically, let $\lambda = Q_q(H)$ denote the q-quantile (lower tail) of entropy values; we keep tokens satisfying $h_i \leq \lambda$:

$$\mathcal{T}' = \{t_i \in \mathcal{T} | h_i \leq \lambda\}, \quad \lambda = Q_q(H), \tag{6}$$

Next, we assign each retained text token in $\mathcal{T}'$ an entropy-based weight:

$$\alpha_i = \texttt{softmax}_{i \in \mathcal{T}'}(-h_i/\gamma) \tag{7}$$

where $\gamma > 0$ controls how strongly low-entropy tokens are emphasized. The dispersion-aware dense guidance score $S^D \in \mathbb{R}^N$ is computed as a weighted aggregation of similarities:

$$s_j^D = \sum_{i \in \mathcal{T}'} \alpha_i \cdot c_{i,j} \tag{8}$$

Instruction Relevance Score. While the dense guidance score S^D excels at capturing fine-grained details, the global guidance score S^G still provides a valuable macroscopic semantic context. Therefore, we fuse them to obtain the instruction relevance score $S^I \in \mathbb{R}^N$:

$$s_j^I = \mu s_j^G + (1 - \mu) s_j^D \tag{9}$$

where $\mu \in [0, 1]$ is a balancing hyper-parameter. Furthermore, the instruction relevance score S^I is applied to `min-max` normalization to ensure that it lies in $[0, 1]$, and then served as the foundational metric for our subsequent spatial smoothing and submodular selection stages:

$$s_j^I = \frac{s_j^I - \min_{k \in \{1,\ldots,N\}} s_k^I}{\max_{k \in \{1,\ldots,N\}} s_k^I - \min_{k \in \{1,\ldots,N\}} s_k^I + \varepsilon}, \quad S^I = \{s_1^I, \ldots, s_N^I\}, \tag{10}$$

where ε is a small constant for numerical stability.

4.3 Spatial Smoothing Prior and Submodular Token Selection

To elegantly resolve the issues of feature fragmentation and selection redundancy stated in Sec. 3.3, we introduce a spatial smoothing prior coupled with score polarization, followed by a submodular maximization sampling strategy.

Spatial Continuity and Score Polarization. The fused relevance score S^I is computed token-wise. To explicitly inject a spatial continuity prior, we reshape the 1D score vector $S^I \in \mathbb{R}^N$ back into its corresponding 2D spatial map $S_{\text{2D}}^I \in \mathbb{R}^{H \times W}$, where $N = H \times W$. Subsequently, we apply a lightweight 3×3 Gaussian convolutional kernel G to smooth the score map, aggregating local structural context:

$$S_{\text{smooth}}^I = S_{\text{2D}}^I * G. \tag{11}$$

While spatial smoothing effectively propagates high activations to adjacent structural parts (e.g., providing local context around a highly discriminative

region), it inherently acts as a low-pass filter that diminishes the peak scores of those discriminative tokens. To prevent critical features from being submerged, we introduce a non-linear score polarization step. We flatten S^I_{smooth} back to a 1D vector $\{s^I_{\text{smooth},j}\}_{j=1}^N$ and apply an exponential sharpening filter:

$$\hat{s}^I_j = \left(s^I_{\text{smooth},j}\right)^\beta, \quad \forall j \in \{1, \ldots, N\}, \tag{12}$$

where $\beta > 1$ is a polarization factor. This operation exponentially amplifies the most salient regions while suppressing the smoothed background noise, yielding the final robust instruction relevance score $\hat{S}^I$.

Facility Location Submodular Maximization. Given $\hat{S}^I$, our final objective is to sample a compact, non-redundant subset $\mathcal{Y} \subset \mathcal{V}$. To fundamentally align token selection with the essence of visual compression, i.e., representing the holistic with a few tokens, we reformulate the process as a classic Facility Location Submodular Maximization problem:

$$\arg\max_{\mathcal{Y}\subset\mathcal{V}} \sum_{j=1}^{N} \hat{s}^I_j \cdot \max_{v_i\in\mathcal{Y}} \text{Sim}(v_i, v_j), \tag{13}$$

where $\text{Sim}(v_i, v_j)$ denotes the cosine similarity between the visual features of token v_i and v_j. Unlike determinantal point processes (DPP) [42], which are used in CDPruner and primarily model diversity, the inner maximization term explicitly quantifies representativeness. It encourages that every single token v_j in the original image is well-represented by its most similar counterpart in the selected subset $\mathcal{Y}$. This explicitly prevents over-fitting to localized high-score regions (e.g., redundantly sampling only the head of a target object) by ensuring holistic feature coverage across all semantic parts. Weighted by the highly polarized instruction relevance score $\hat{S}^I$, this objective forces the selection to prioritize salient regions, while the submodular maximization naturally penalizes selecting redundant tokens within those localized high-score areas.

While finding the exact optimal subset is NP-Hard, the facility location objective is inherently submodular, allowing us to employ a highly efficient greedy algorithm with a theoretical $1 - 1/e$ lower bound approximation. At each step, we iteratively add the candidate token $u \in \mathcal{V} \setminus \mathcal{Y}$ that yields the maximum marginal gain, computed as:

$$\text{Gain}(u) = \sum_{j=1}^{N} \hat{s}^I_j \times \left(\max\left(\text{Sim}(u, v_j), \text{Curr}(v_j)\right) - \text{Curr}(v_j)\right) \tag{14}$$

where $\text{Curr}(v_j) = \max_{v_i\in\mathcal{Y}} \text{Sim}(v_i, v_j)$ caches the maximum similarity between v_j and the currently selected subset $\mathcal{Y}$. This caching allows efficient incremental updates of marginal gains. In practice, the runtime depends on whether we precompute token-to-token similarities. Computing cosine similarities on-the-fly yields a cost dominated by similarity evaluations, while optional precomputation of an $N \times N$ similarity matrix can accelerate greedy updates at the expense of $\mathcal{O}(N^2)$ memory. *We provide proofs and implementation details in the supplementary.*

Table 1: Performance comparison on LLaVA-1.5-7B. Avg. denotes the average over 9 benchmarks (excluding VizWiz). * indicates results reproduced on the VizWiz validation set, since the official test challenge is no longer available.

Method	VQA^{V2}	GQA	VizWiz	SQA^{IMG}	VQA^{Text}	POPE	MME	MMB^{EN}	MMB^{CN}	MMVet	Avg.
Upper Bound, All 576 Tokens (**100**%)											
LLaVA-1.5-7B	78.5	61.9	50.1/55.6*	69.6	58.2	85.9	1513.4	64.7	58.1	31.3	64.9
Retain 128 Tokens (↓ **77.8**%)											
FastV(ECCV24)	71.3	55.3	51.9	68.7	55.6	70.0	1343.6	62.5	54.8	26.5	59.1
PDrop(CVPR25)	73.7	56.8	49.4	69.5	56.1	77.9	1421.9	62.1	55.0	27.1	61.0
SparseVLM(ICML25)	74.9	57.0	49.7	69.3	56.0	83.3	1408.3	62.2	56.1	30.1	62.1
PruMerge+(2024.05)	74.8	58.0	53.7	68.6	54.3	83.2	1411.9	62.0	55.1	29.9	61.8
TRIM(COLING25)	75.5	58.7	51.6	68.3	52.2	85.5	1418.5	61.9	52.9	29.5	61.7
VisionZip(CVPR25)	75.5	57.9	51.6	68.9	55.9	84.8	1423.7	61.5	55.9	31.3	62.5
DART(EMNLP25)	74.9	58.3	52.8	69.1	55.8	81.2	1415.1	60.7	56.0	31.1	62.0
DivPrune(CVPR25)	76.0	59.2	57.5*	68.2	55.3	86.8	1405.1	61.5	54.8	30.6	62.5
CDPruner(NIPS25)	76.2	59.3	57.3*	69.0	56.0	87.3	1431.4	62.4	54.7	32.1	63.2
HiPrune(AAAI26)	74.9	57.3	55.4*	68.3	56.2	82.8	1364.4	62.2	56.4	31.2	61.9
EADP(Ours)	**76.7**	**60.0**	**58.0***	**69.0**	**56.5**	**87.2**	**1439.0**	**62.7**	**55.2**	**32.5**	**63.5**
Retain 64 Tokens (↓ **88.9**%)											
FastV(ECCV24)	57.2	48.0	49.1	68.7	52.1	37.2	992.1	49.6	43.3	20.1	47.3
PDrop(CVPR25)	58.1	49.2	46.3	68.3	51.1	40.1	1004.4	48.3	37.8	19.7	47.0
SparseVLM(ICML25)	67.3	51.5	49.4	69.0	52.5	70.2	1192.9	58.0	49.3	25.1	55.8
PruMerge+(2024.05)	71.2	55.1	53.7	69.2	52.1	76.3	1310.5	59.2	51.6	28.0	58.7
TRIM(COLING25)	72.8	56.8	51.1	69.1	50.8	85.3	1356.1	60.5	49.7	26.5	59.9
VisionZip(CVPR25)	72.1	56.0	52.9	69.0	55.0	77.9	1362.2	60.3	55.0	29.0	60.3
DART(EMNLP25)	71.9	55.2	53.5	68.7	54.3	74.3	1371.3	59.3	54.0	26.5	59.2
DivPrune(CVPR25)	74.1	57.5	57.8*	68.0	54.5	85.5	1356.2	60.1	52.3	28.1	60.9
CDPruner(NIPS25)	75.0	58.6	58.0*	68.1	54.7	87.0	1399.1	61.1	53.2	29.5	61.9
HiPrune(AAAI26)	69.2	53.6	55.4*	68.9	54.2	73.0	1236.0	59.5	53.4	27.9	57.9
EADP(Ours)	**75.6**	**59.4**	**59.5***	**68.9**	**55.0**	**86.5**	**1403.6**	**61.9**	**53.1**	**29.4**	**62.2**
Retain 32 Tokens (↓ **94.4**%)											
PruMerge+(2024.05)	65.3	51.3	53.5	68.1	48.7	68.9	1255.2	54.6	45.2	25.5	54.5
TRIM(COLING25)	68.9	53.7	50.7	68.0	46.5	84.2	1259.3	57.1	39.5	22.8	56.0
VisionZip(CVPR25)	67.4	52.2	52.4	68.7	52.0	70.6	1257.1	56.8	48.9	24.8	56.0
DART(EMNLP25)	67.3	52.6	52.5	69.1	52.2	71.1	1288.4	58.5	49.3	25.5	56.7
DivPrune(CVPR25)	71.2	54.9	57.4*	68.6	52.4	81.5	1284.9	57.6	49.1	26.3	58.4
CDPruner(NIPS25)	73.6	57.0	57.9*	69.5	52.1	87.1	1353.0	59.1	49.6	27.8	60.4
EADP(Ours)	**74.0**	**58.2**	**59.3***	**69.3**	**52.7**	**86.7**	**1347.0**	**59.4**	**50.1**	**30.6**	**60.9**

5 Experiments

5.1 Experimental setup

Model. To comprehensively validate the effectiveness and generalizability of our method, we conduct experiments on multiple representative VLMs. Specifically, we adopt several variants from the LLaVA family, including LLaVA-1.5 [37] for standard image understanding, LLaVA-NEXT [36] designed to handle high-resolution visual inputs, and LLaVA-Video [71] for video-based reasoning. In addition, we further evaluate EADP on Qwen2.5-VL [4], an advanced VLM. Results on more architectures are available in supplementary material.

Benchmarks. For LLaVA models (LLaVA-1.5 and LLaVA-NeXT), we evaluate on 10 image-based VQA benchmarks: VQAv2 [15], GQA [20], VizWiz [19], ScienceQA-IMG [41], TextVQA [51], POPE [32], MME [13], MMBench [39], MMBench-CN [39], and MM-Vet [64]. For Qwen-series advanced VLMs, we follow their standard evaluation protocols and report results on a diverse set of benchmarks. Specifically, Qwen2.5-VL is evaluated on 8 benchmarks: TextVQA, ChartQA [43], AI2D [22], OCRBench [40], HallBench [16], MME, MMBench, and

Table 2: Performance comparison on LLaVA-NeXT-7B. Avg. denotes the average performance across 9 benchmarks (excluding VizWiz). Boldface indicates the results of our method only.

Method	VQA^{V2}	GQA	VizWiz	SQA^{IMG}	VQA^{Text}	POPE	MME	MMB^{EN}	MMB^{CN}	MMVet	Avg.
Upper Bound, All 2,880 Tokens (**100**%)											
LLaVA-NeXT-7B	81.3	62.5	55.2/60.9*	67.6	60.3	86.8	1510.9	65.8	57.3	40.0	66.3
Retain 640 Tokens (↓ **77.8**%)											
FastV (ECCV24)	76.6	58.4	54.9	67.2	57.5	80.0	1433.2	62.4	54.1	37.1	62.8
PDrop (CVPR25)	78.7	59.5	53.8	66.9	57.1	83.2	1456.3	63.3	54.8	35.4	63.5
SparseVLM (ICML25)	79.5	61.4	53.6	67.3	58.7	85.7	1464.8	64.8	57.1	35.7	64.8
PruMerge+ (2024.05)	78.0	61.1	57.9	67.4	55.2	85.5	1476.9	64.0	56.6	33.6	63.9
TRIM (COLING25)	77.9	61.7	54.8	67.1	55.5	85.9	1473.5	66.0	55.9	36.9	64.5
VisionZip (CVPR25)	79.2	61.4	57.1	67.5	58.5	86.2	1492.1	65.8	58.3	39.2	65.6
DART (EMNLP25)	78.7	61.0	57.0	68.0	59.0	85.7	1444.9	65.1	57.5	37.3	64.9
DivPrune (CVPR25)	79.3	61.9	60.6*	67.8	57.0	86.9	1469.7	65.8	57.3	38.0	65.3
CDPruner (NeurIPS25)	79.9	62.3	60.8*	67.9	58.4	87.1	1474.5	66.2	57.5	40.7	66.0
HiPrune (AAAI26)	78.8	60.7	61.2*	68.0	48.6	85.3	1475.3	67.0	58.8	38.4	64.4
EADP (Ours)	**80.0**	**62.7**	**60.6***	**68.0**	**59.2**	**87.4**	**1494.7**	**66.5**	**57.9**	**40.1**	**66.3**
Retain 320 Tokens (↓ **88.9**%)											
FastV (ECCV24)	63.4	52.1	51.3	66.3	52.9	59.9	1134.3	55.8	46.4	21.1	52.7
PDrop (CVPR25)	67.7	53.3	49.7	66.5	50.7	63.5	1180.7	56.9	48.9	23.7	54.5
SparseVLM (ICML25)	73.9	57.8	54.2	67.0	56.0	77.8	1383.5	63.3	53.4	31.9	61.1
PruMerge+ (2024.05)	74.8	58.4	57.7	67.5	54.2	79.1	1423.7	62.6	54.8	32.3	61.7
TRIM (COLING25)	75.3	60.2	53.5	66.4	50.9	85.7	1445.8	62.1	52.2	33.5	62.1
VisionZip (CVPR25)	76.3	59.3	56.2	67.3	56.4	82.9	1402.7	62.7	55.6	35.2	62.9
DART (EMNLP25)	75.1	59.1	56.8	67.1	56.3	81.4	1433.5	63.8	55.4	36.3	62.9
DivPrune (CVPR25)	77.2	61.1	60.1*	67.5	56.2	84.7	1423.3	63.9	55.7	34.8	63.6
CDPruner (NeurIPS25)	77.9	61.6	59.9*	67.7	56.4	86.8	1453.0	64.1	55.7	37.9	64.5
HiPrune (AAAI26)	74.4	57.6	61.3*	67.3	56.5	78.9	1406.8	64.2	57.0	34.7	62.3
EADP (Ours)	**78.6**	**62.2**	**60.4***	**67.6**	**57.0**	**86.9**	**1491.3**	**64.6**	**55.9**	**39.4**	**65.2**
Retain 160 Tokens (↓ **94.4**%)											
PruMerge+ (2024.05)	69.1	55.6	57.2	66.7	50.9	73.7	1298.3	58.3	49.1	27.6	57.3
TRIM (COLING25)	70.3	57.1	52.9	65.7	47.4	84.4	1281.6	61.2	45.4	30.1	58.4
VisionZip (CVPR25)	70.8	55.8	55.5	67.7	53.4	76.0	1319.4	59.2	51.2	31.6	59.1
DART (EMNLP25)	73.1	56.3	56.7	67.5	53.6	77.2	1332.8	61.5	53.3	32.0	60.1
DivPrune (CVPR25)	75.0	60.2	60.7*	67.1	53.7	80.0	1376.3	62.3	53.4	32.4	61.4
CDPruner (NeurIPS25)	76.4	60.6	59.7*	67.5	53.2	86.3	1402.3	63.6	53.8	35.0	62.9
HiPrune (AAAI26)	67.3	53.7	59.5*	68.7	48.6	67.7	1200.7	59.8	50.7	29.2	56.2
EADP (Ours)	**77.0**	**61.6**	**60.2***	**67.4**	**54.2**	**86.0**	**1419.5**	**63.4**	**54.6**	**35.6**	**63.4**

MMBench-CN. For Qwen3-VL, we further include DocVQA [45] and InfoVQA [44]. We also evaluate LLaVA-Video on 3 video benchmarks: LongVideoBench [57], MVBench [27], and Video-MME [14]. All experiments use default settings and metrics; task details are provided in the supplementary material.

Comparisons. We compare EADP with recent representative visual token pruning methods spanning different design philosophies, including FastV [5], PyramidDrop [59], SparseVLM [70], LLaVA-Prumerge [49], TRIM [52], VisionZip [60], DART [56], DivPrune [2], HiPrune [38] and CDPruner [69].

Implementation Details. For image-based experiments on LLaVA, we build our implementation upon the official release. For video tasks, we evaluate using `lmms-eval` toolkit [68]. As for Qwen-series models, we rely on `VLMEvalKit` [11] to perform evaluation, following its official configuration to ensure fair comparison with prior work. All experiments are conducted on NVIDIA RTX 3090 GPUs.

5.2 Results on LLaVA series

Standard-resolution inputs. We evaluate EADP on LLaVA-1.5-7B and report results in Tab. 1 under 3 common budgets of retained tokens. EADP consistently

achieves the highest average accuracy across all pruning methods at each budget. With 77.8% tokens pruned, it reaches 63.5 average, closest to the full-token upper bound, surpassing CDPruner and DivPrune. Under heavier pruning, it remains strong (62.2 at 64 tokens; 60.9 at 32 tokens), beating DivPrune by 2.5 points on average and showing robustness to extreme reduction. EADP also excels on GQA and MMVet, highlighting its fine-grained vision–instruction modeling.

High-resolution inputs. While higher visual resolution improves fine-grained reasoning in VLMs [4, 6, 37], it also increases redundancy, making token pruning essential. We evaluated EADP on LLaVA-NeXT-7B with 672×672 input resolution, producing 2880 visual tokens per image. As shown in Tab. 2, EADP is highly robust in this high-density regime: with 640 tokens, it matches the unpruned upper bound at 66.3 average. With 320/160 tokens, it achieves 65.2/63.4, respectively, outperforming strong recent baselines. These results confirm EADP's ability to preserve fine-grained semantics under high-resolution inputs.

5.3 Results on advanced VLMs

Qwen2.5-VL. We further evaluate EADP on the recent open-source VLM Qwen2.5-VL [4] to assess generalization across heterogeneous architectures. Following CDPruner [69], we fix the input resolution to 1008×1008, yielding 1,296 visual tokens per image. As Qwen2.5-VL uses a different visual encoder and projection design, methods requiring a dedicated `[CLS]` token are inapplicable; thus we compare only with DivPrune [2], HiPrune [38], and CDPruner. Empirically, pruning causes larger degradation on Qwen2.5-VL than on LLaVA, suggesting its visual tokens are more tightly coupled with downstream reasoning. As shown in Tab. 3, when retaining 256 tokens, EADP maintains 74.3 average accuracy, outperforming all baselines by a clear margin. Even under the highly compressed 128-token setting, EADP preserves 68.4 accuracy, demonstrating robustness under extreme token constraints. Specifically, EADP's advantage is particularly evident on hallucination-sensitive and reasoning-intensive benchmarks such as HallBench.

Qwen3-VL. We also evaluate EADP on Qwen3-VL-8B and additionally report results on DocVQA and InfoVQA, which require preserving fine-grained textual and layout information in document images. As shown in Tab. 4, EADP consistently achieves best average performance under all token budgets. Specifically, when retaining 128 tokens, EADP obtains 62.7 average scores, outperforming the strongest baseline by 3.5. Gains are particularly evident on document-oriented benchmarks: at 256 tokens, EADP improves DocVQA by 7.0 and 9.8 over DivPrune and CDPruner, respectively, while also bringing improvements on InfoVQA. These results demonstrate that EADP can effectively preserve fine-grained textual evidence and holistic visual structures under strict token budgets.

5.4 Results on Video Understanding

Video understanding poses a more challenging scenario for visual token efficiency, as redundancy naturally accumulates across both spatial and temporal dimensions [50]. To examine EADP in this high-redundancy setting, we conduct

Table 3: Performance comparison on Qwen2.5-VL-7B. Avg. denotes the average performance across 8 benchmarks. Boldface indicates the results of our method only.

Method	TextVQA	ChartQA	AI2D	OCRBench	HallBench	MME	MMB-EN	MMB-CN	Avg.
Upper Bound, All 1296 Tokens (**100**%)									
Qwen2.5-VL-7B	84.5	86.4	84.2	869	47.7	2303.4	83.9	82.8	84.0
Retain 512 Tokens (↓ **60.5**%)									
DivPrune(CVPR25)	78.8	77.1	79.6	760	42.2	2188.4	81.7	80.4	78.1
CDPruner(NeurIPS25)	66.2	67.5	72.9	625	39.9	2122.4	79.6	78.3	71.6
HiPrune(AAAI26)	75.8	74.2	79.3	679	40.5	2176.5	80.3	80.1	75.9
EADP(Ours)	**78.6**	**76.2**	**79.5**	**744**	**42.2**	**2213.3**	**81.6**	**80.8**	**78.0**
Retain 256 Tokens (↓ **80.2**%)									
DivPrune(CVPR25)	74.0	67.0	77.4	665	36.4	2173.0	80.5	78.3	73.6
CDPruner(NeurIPS25)	55.1	56.8	69.8	509	34.4	2017.6	77.6	74.7	65.0
HiPrune(AAAI26)	64.2	56.7	74.1	579	37.3	2153.2	78.4	79.0	69.4
EADP(Ours)	**73.8**	**67.4**	**78.7**	**668**	**38.7**	**2202.0**	**80.2**	**78.5**	**74.3**
Retain 128 Tokens (↓ **90.1**%)									
DivPrune(CVPR25)	66.1	51.1	73.7	516	31.2	2054.9	77.4	77.3	66.4
CDPruner(NeurIPS25)	44.7	47.2	68.3	389	27.8	1876.7	73.3	71.7	58.2
HiPrune(AAAI26)	51.1	41.3	72.3	497	31.2	2026.8	75.0	76.2	62.3
EADP(Ours)	**65.8**	**51.9**	**75.7**	**556**	**36.7**	**2137.9**	**78.4**	**76.4**	**68.4**

Table 4: Performance comparison on Qwen3-VL-8B. Avg. denotes the average performance across 10 benchmarks. Boldface indicates the results of our method only.

Method	TextVQA	ChartQA	AI2D	OCRBench	HallBench	MME	MMB-EN	MMB-CN	DocVQA	InfoVQA	Avg.
Upper Bound, All 1024 Tokens (**100**%)											
Qwen3-VL-8B	83.8	82.3	86.3	855	51.2	2440.2	86.3	84.6	94.5	66.4	84.3
Retain 512 Tokens (↓ **50.0**%)											
DivPrune(CVPR25)	75.0	59.8	78.6	658	41.0	2217.5	82.1	79.4	74.6	46.0	71.3
CDPruner(NeurIPS25)	74.5	62.0	76.9	660	41.6	2199.8	81.0	79.3	72.3	47.3	71.1
EADP(Ours)	**75.2**	**65.9**	**77.7**	**673**	**42.5**	**2261.1**	**81.6**	**79.7**	**77.7**	**49.9**	**73.1**
Retain 256 Tokens (↓ **75.0**%)											
DivPrune(CVPR25)	69.5	47.5	75.2	595	38.7	2185.3	80.0	78.8	55.8	38.3	65.3
CDPruner(NeurIPS25)	68.5	51.9	74.1	601	41.3	2200.6	79.8	80.0	53.0	37.6	65.6
EADP(Ours)	**71.4**	**55.5**	**75.7**	**625**	**42.8**	**2202.4**	**80.5**	**78.9**	**62.8**	**42.3**	**68.2**
Retain 128 Tokens (↓ **87.5**%)											
DivPrune(CVPR25)	62.6	37.0	72.4	499	36.7	2136.3	77.0	76.6	39.5	33.8	59.2
CDPruner(NeurIPS25)	60.6	40.4	71.7	515	34.7	2153.8	77.9	77.5	37.8	32.5	59.2
EADP(Ours)	**66.1**	**43.6**	**73.6**	**550**	**40.2**	**2200.2**	**79.9**	**78.4**	**45.3**	**35.3**	**62.7**

experiments on LLaVA-Video [71], a VLM tailored for video reasoning. As shown in Tab. 5, EADP consistently achieves the best performance across all configuration. When the pruning ratio reaches 81.1%, our method achieves 57.0 average accuracy, only 4.2 lower than original performance and outperforms all baselines. Importantly, the performance gap becomes more pronounced as the pruning ratio increases, suggesting that EADP better identifies temporally and semantically critical tokens, preventing severe information loss under high compression. *More detailed results are provided in the supplementary materials.*

5.5 Efficiency Analysis

We evaluate the computational efficiency of EADP in terms of prefill time, end-to-end inference latency and FLOPs. All metrics are averaged across 2,000 instances equally sampled from VizWiz, TextVQA, POPE, and MME. Tab. 6 presents our main results on LLaVA-1.5-7B under three visual token budgets. *Additional results on different architectures are detailed in the supplementary material.* Compared to the unpruned LLaVA-1.5-7B, EADP consistently delivers significant acceleration.

Table 5: Performance comparison on LLaVA-Video-7B with 64 frames per video.

Method	MVBench	LongVideoBench	Video-MME	Avg.
Upper Bound, All 64 × 169 Tokens (**100**%)				
LLaVA-Video-7B	60.4	58.7	64.4	61.2
Retain 64 × 64 Tokens (↓ **62.1**%)				
DivPrune(CVPR25)	55.2	58.7	61.2	58.4
CDPruner(NeurIPS25)	54.1	57.4	60.9	57.5
HiPrune(AAAI26)	54.0	56.0	59.0	56.3
EADP(Ours)	**55.7**	**57.9**	**62.0**	**58.5**
Retain 64 × 32 Tokens (↓ **81.1**%)				
DivPrune(CVPR25)	53.7	55.7	59.2	56.2
CDPruner(NeurIPS25)	53.2	55.4	58.3	55.6
HiPrune(AAAI26)	51.1	54.9	56.0	54.0
EADP(Ours)	**54.5**	**56.6**	**60.4**	**57.2**
Retain 64 × 16 Tokens (↓ **90.5**%)				
DivPrune(CVPR25)	52.1	52.7	57.2	54.0
CDPruner(NeurIPS25)	50.2	53.8	56.3	53.4
HiPrune(AAAI26)	48.8	52.0	53.8	51.5
EADP(Ours)	**52.6**	**54.5**	**57.3**	**54.8**

Table 6: Efficiency analysis on LLaVA-1.5-7B.

Method	Prefill(ms)	Latency(ms)	FLOPs(G)
Upper Bound, All 576 Tokens (**100**%)			
LLaVA-1.5-7B	207.4	256.8	4489.8
Retain 128 Tokens (↓ **77.8**%)			
DivPrune(CVPR25)	109.6 (×1.9)	158.0 (×1.6)	1529.9 (×2.9)
CDPruner(NeurIPS25)	135.9 (×1.5)	183.3 (×1.4)	1538.3 (×2.9)
HiPrune(AAAI26)	99.4 (×2.0)	150.1 (×1.7)	1529.9 (×2.9)
EADP(Ours)	**118.8 (×1.7)**	**169.3 (×1.5)**	**1538.4 (×2.9)**
Retain 64 Tokens (↓ **88.9**%)			
DivPrune(CVPR25)	92.7 (×2.2)	140.4 (×1.8)	1107.0 (×4.1)
CDPruner(NeurIPS25)	105.9 (×2.0)	157.3 (×1.6)	1115.5 (×4.0)
HiPrune(AAAI26)	85.7 (×2.4)	137.6 (×1.9)	1107.0 (×4.1)
EADP(Ours)	**97.0 (×2.1)**	**147.7 (×1.7)**	**1115.6 (×4.0)**
Retain 32 Tokens (↓ **94.4**%)			
DivPrune(CVPR25)	79.8 (×2.6)	128.3 (×2.0)	895.6 (×5.0)
CDPruner(NeurIPS25)	85.8 (×2.4)	137.46 (×1.9)	904.1 (×5.0)
HiPrune(AAAI26)	74.8 (×2.8)	126.54 (×2.0)	895.6 (×5.0)
EADP(Ours)	**81.9 (×2.5)**	**129.63 (×2.0)**	**904.1 (×5.0)**

Table 7: Ablation studies on core design choices. We evaluate the impact of various hyperparameters and algorithmic variants across five benchmarks. All experiments are conducted using LLaVA-1.5-7B with a fixed budget of 128 visual tokens.

	VizWiz	SQA	TextVQA	POPE	MME	Avg.
global-dense balancing coefficient						
μ =0.0	57.8	68.8	56.2	84.6	1421.9	67.7
μ =0.2	58.2	68.9	56.3	85.9	1422.0	68.1
μ =0.5	57.9	69.0	56.5	87.2	1439.0	68.5
μ =0.8	58.0	68.9	56.5	86.4	1440.6	68.3
μ =1.0	57.5	68.7	56.1	87.3	1422.2	68.1
keep proportion						
q =0.3	58.2	69.0	56.6	87.2	1437.2	68.6
q =0.5	58.0	69.0	56.5	87.2	1439.0	68.5
q =0.7	57.4	68.3	56.0	86.7	1420.1	67.9
q =0.9	57.1	68.2	55.8	86.4	1411.0	67.6
aggregation method						
(1)	58.0	69.0	56.5	87.2	1439.0	68.5
(2)	57.1	68.3	55.9	87.0	1430.2	68.0
(3)	58.0	69.2	56.5	87.2	1434.3	68.5
(4)	58.3	68.9	56.3	87.4	1441.9	68.6

	VizWiz	SQA	TextVQA	POPE	MME	Avg.
polarization value						
β =0.0	58.0	68.8	55.9	86.7	1395.4	67.8
β =0.5	57.9	69.1	56.0	87.0	1394.9	67.9
β =2.0	57.8	69.1	56.5	87.2	1412.4	68.2
β =5.0	58.0	69.0	56.6	87.3	1414.1	68.3
β =10.0	56.8	68.8	56.4	87.2	1439.0	68.2
spatial smoothing size						
w.o.	57.9	68.6	55.7	86.9	1413.4	67.9
size=3	58.0	69.0	56.5	87.2	1439.0	68.5
size=5	57.9	69.2	56.5	87.2	1435.0	68.5
size=7	57.9	69.1	56.3	87.2	1435.3	68.4
how to measure dispersion degree						
(A)	58.0	69.0	56.5	87.2	1439.0	68.5
(B)	57.9	69.3	56.4	87.2	1429.8	68.4
(C)	57.9	69.2	56.4	87.3	1435.1	68.5

For instance, retaining 128 tokens reduces FLOPs by nearly 66% while achieving a 1.75× speedup in prefill time and a 1.50× speedup in end-to-end latency. When compared to existing approaches such as DivPrune and HiPrune, EADP exhibits a marginal computational overhead which originates from the dense scoring. However, considering the superior performance, this minimal overhead represents a highly worthwhile trade-off. *We further report a wall-clock time analysis for the different components of our method in supplementary material.*

5.6 Ablation Studies

Ablation on the global-dense balancing coefficient. The coefficient μ balances the global and dense guidance score. Relying purely on dense guidance score ($\mu = 0.0$) yields the lowest performance. As we increase μ, the performance steadily improves and peak at $\mu = 0.5$. This trend demonstrates that global scene representation provides an indispensable foundation for holistic understanding, while fine-grained dense alignment serves as a crucial supplement for details.

Ablation on the polarization value. The hyperparameter β is designed to sharpen the score map, distinctively prioritizing salient region. Disabling polarization ($\beta = 0.0$) significantly degrades performance. Interestingly, the performance reaches a plateau when $\beta \geq 1.0$, indicating that the submodular maximization is highly robust.
Ablation on the keep proportion. We analyze the impact of preserving only a proportion q of text tokens with lowest entropy while calculating the dense guidance score. Notably, aggressively discarding 70% of text tokens ($q = 0.3$) achieves the highest score, which strongly corroborates our core motivation: high-entropy text tokens correspond to dispersed, uninformative visual attention.
Ablation on spatial smoothing size. Spatial smoothing is crucial for obtaining contiguous scores maps. Compared to the case applying a 3×3 or 5×5 kernel, removing the smoothing ("w.o.") leads to a distinct performance degradation. This confirms that enforcing spatial coherence is essential, which could mitigate the impact of localized noise and guide the submodular optimizer to select structurally complete semantic regions.
Ablation on aggregation method. For weighting the semantic relevance scores of the filtered texts (see Eq. (7)), we consider four styles: (1) using the weighted inverse entropy, (2) averaging over the top-k scores, (3) using the gated L1 norm, and (4) our proposed method. As shown in Tab. 7, after removing noisy texts, effectively weighting each visual score computed from the remaining text tokens is beneficial, whereas the simple mean operation (Style 2) yields smaller gains than the other three. For simplicity and robustness, we ultimately adopt style 4.
Ablation on dispersion measurement. We explored three statistics to quantify the degree of dispersion: (A) entropy, (B) central mass ratio, and (C) variance. As shown in Tab. 7, we found that these schemes achieve comparable performance, demonstrating that the noise-filtering process is robust to the choice of statistic.
More details of the ablation studies can be referred to the supplementary material.

6 Conclusion

In this work, we investigate the failure modes of visual token pruning in VLMs and identify textual noise dispersion and feature fragmentation as critical bottlenecks. To address these issues, we introduce EADP, which reformulates visual pruning as a structured compression task. EADP elegantly quantifies and filters textual noise via statistical entropy to yield robust instruction relevance scores. Furthermore, by casting token selection as a facility location submodular maximization problem with spatial priors, EADP explicitly guarantees holistic and non-redundant visual coverage. Extensive experiments across diverse VLM architectures demonstrate that EADP achieves state-of-the-art performance, robustly preserving fine-grained visual cues under strict token budgets and significantly advancing the accuracy-efficiency trade-off for practical VLM deployment.

Acknowledgements

This work was supported by the NSFC under Grant 62322604 and 62576207.

References

1. Alayrac, J., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., van den Driessche, G., Mugford, K., Sifre, L., Soyer, H., Doersch, C., Gupta, A., Stanczyk, P., Noh, H., Gontijo Lopes, R., Smith, A., Vinyals, O., Zisserman, A., Simonyan, K.: Flamingo: a visual language model for few-shot learning. arXiv preprint arXiv:2204.14198 (2022)
2. Alvar, S.R., Singh, G., Akbari, M., Zhang, Y.: Divprune: Diversity-based visual token pruning for large multimodal models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 9392–9401 (2025)
3. Bai, J., et al.: Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. arXiv preprint arXiv:2308.12966 (2023)
4. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report (2025), https://arxiv.org/abs/2502.13923
5. Chen, L., Zhao, H., Liu, T., Bai, S., Lin, J., Zhou, C., Chang, B.: An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In: European Conference on Computer Vision. pp. 19–35. Springer (2024)
6. Chen, Z., Wang, W., Cao, Y., Liu, Y., Gao, Z., Cui, E., Zhu, J., Ye, S., Tian, H., Liu, Z., et al.: Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. arXiv preprint arXiv:2412.05271 (2024)
7. Chen, Z., et al.: InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. arXiv preprint arXiv:2312.14238 (2023)
8. Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
9. Clark, K., Khandelwal, U., Levy, O., Manning, C.D.: What does BERT look at? an analysis of BERT's attention. In: Linzen, T., Chrupała, G., Belinkov, Y., Hupkes, D. (eds.) Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Association for Computational Linguistics, Florence, Italy (Aug 2019). https://doi.org/10.18653/v1/W19-4828, https://aclanthology.org/W19-4828/
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale (2021), https://arxiv.org/abs/2010.11929
11. Duan, H., Yang, J., Qiao, Y., Fang, X., Chen, L., Liu, Y., Dong, X., Zang, Y., Zhang, P., Wang, J., et al.: Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 11198–11201 (2024)
12. Duan, Y., Li, A., Li, Y., Li, L., Wang, P.: Gridprune: From" where to look" to" what to select" in visual token pruning for mllms. arXiv preprint arXiv:2511.10081 (2025)
13. Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., Ji, R., Shan, C., He, R.: Mme: A comprehensive evaluation

benchmark for multimodal large language models (2025), https://arxiv.org/abs/2306.13394
14. Fu, C., Dai, Y., Luo, Y., Li, L., Ren, S., Zhang, R., Wang, Z., Zhou, C., Shen, Y., Zhang, M., Chen, P., Li, Y., Lin, S., Zhao, S., Li, K., Xu, T., Zheng, X., Chen, E., Shan, C., He, R., Sun, X.: Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis (2025), https://arxiv.org/abs/2405.21075
15. Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6325–6334 (2017). https://doi.org/10.1109/CVPR.2017.670
16. Guan, T., Liu, F., Wu, X., Xian, R., Li, Z., Liu, X., Wang, X., Chen, L., Huang, F., Yacoob, Y., Manocha, D., Zhou, T.: Hallusionbench: An advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14375–14385 (June 2024)
17. Guan, T., Wang, Z., Fu, P., Guo, Z., Shen, W., Zhou, K., Yue, T., Duan, C., Sun, H., Jiang, Q., et al.: A token-level text image foundation model for document understanding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23210–23220 (2025)
18. Guan, T., Yang, Z., Wan, J., Yang, M., Guo, Z., Hu, Z., Luo, R., Chen, R., Jiang, S., Wang, P., et al.: Codepercept: Code-grounded visual stem perception for mllms. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 33542–33552 (2026)
19. Gurari, D., Li, Q., Stangl, A.J., Guo, A., Lin, C., Grauman, K., Luo, J., Bigham, J.P.: Vizwiz grand challenge: Answering visual questions from blind people (2018), https://arxiv.org/abs/1802.08218
20. Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6693–6702 (2019). https://doi.org/10.1109/CVPR.2019.00686
21. Jia, C., Yang, Y., Xia, Y., Chen, Y., Parekh, Z., Pham, H., Le, Q.V., Sung, Y., Li, Z., Duerig, T.: ALIGN: Scaling up visual and vision-language representation learning with noisy text supervision. arXiv preprint arXiv:2102.05918 (2021)
22. Kembhavi, A., Salvato, M., Kolve, E., Seo, M., Hajishirzi, H., Farhadi, A.: A diagram is worth a dozen images (2016), https://arxiv.org/abs/1603.07396
23. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are zero-shot reasoners. arXiv preprint arXiv:2205.11916 (2022)
24. Krause, A., Golovin, D.: Submodular function maximization. In: Tractability (2014), https://api.semanticscholar.org/CorpusID:6107490
25. Li, B., Zhang, Y., Guo, D., Zhang, R., Li, F., Zhang, H., Zhang, K., Zhang, P., Li, Y., Liu, Z., et al.: Llava-onevision: Easy visual task transfer. arXiv preprint arXiv:2408.03326 (2024)
26. Li, J., Li, D., Xiong, C., Hoi, S.C.H.: BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: Proceedings of the 40th International Conference on Machine Learning (ICML) (2023)
27. Li, K., Wang, Y., He, Y., Li, Y., Wang, Y., Liu, Y., Wang, Z., Xu, J., Chen, G., Lou, P., Wang, L., Qiao, Y.: Mvbench: A comprehensive multi-modal video understanding benchmark. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 22195–22206 (2024). https://doi.org/10.1109/CVPR52733.2024.02095

28. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., Chang, K.W., Gao, J.: Grounded language-image pre-training (2022), https://arxiv.org/abs/2112.03857
29. Li, W., Chen, L., Dai, D., Zhu, Z., Tan, M., Yuan, L., Li, L., Wang, J., Liu, J.: InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)
30. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., Choi, Y., Gao, J.: OSCAR: Object-semantics aligned pre-training for vision-language tasks. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
31. Li, Y., Yang, J., Shen, Z., Han, L., Xu, H., Tang, R.: Catp: Contextually adaptive token pruning for efficient and enhanced multimodal in-context learning. arXiv preprint arXiv:2508.07871 (2025)
32. Li, Y., Du, Y., Zhou, K., Wang, J., Zhao, X., Wen, J.R.: Evaluating object hallucination in large vision-language models. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.emnlp-main.20, https://aclanthology.org/2023.emnlp-main.20/
33. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. In: Proceedings of the 2024 conference on empirical methods in natural language processing. pp. 5971–5984 (2024)
34. Lin, H.C., Bilmes, J.A.: A class of submodular functions for document summarization. In: Annual Meeting of the Association for Computational Linguistics (2011), https://api.semanticscholar.org/CorpusID:320371
35. Lin, Z., Lin, M., Lin, L., Ji, R.: Boosting multimodal large language models with visual tokens withdrawal for rapid inference. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 5334–5342 (2025)
36. Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., Lee, Y.J.: Llava-next: Improved reasoning, ocr, and world knowledge (January 2024), https://llava-vl.github.io/blog/2024-01-30-llava-next/
37. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)
38. Liu, J., Du, F., Zhu, G., Lian, N., Li, J., Chen, B.: Hiprune: Training-free visual token pruning via hierarchical attention in vision-language models. arXiv preprint arXiv:2508.00553 (2025)
39. Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., Chen, K., Lin, D.: Mmbench: Is your multi-modal model an all-around player? (2024), https://arxiv.org/abs/2307.06281
40. Liu, Y., Li, Z., Huang, M., Yang, B., Yu, W., Li, C., Yin, X.C., Liu, C.L., Jin, L., Bai, X.: Ocrbench: on the hidden mystery of ocr in large multimodal models. Science China Information Sciences **67**(12) (Dec 2024). https://doi.org/10.1007/s11432-024-4235-6, http://dx.doi.org/10.1007/s11432-024-4235-6
41. Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.W., Zhu, S.C., Tafjord, O., Clark, P., Kalyan, A.: Learn to explain: Multimodal reasoning via thought chains for science question answering. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems. vol. 35, pp. 2507–2521. Curran Associates, Inc. (2022)
42. Macchi, O.: The coincidence approach to stochastic point processes. Advances in Applied Probability **7**(1), 83–122 (1975)

43. Masry, A., Long, D.X., Tan, J.Q., Joty, S., Hoque, E.: ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.) Findings of the Association for Computational Linguistics: ACL 2022. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.findings-acl.177, https://aclanthology.org/2022.findings-acl.177/
44. Mathew, M., Bagal, V., Tito, R.P., Karatzas, D., Valveny, E., Jawahar, C.V.: Infographicvqa (2021), https://arxiv.org/abs/2104.12756
45. Mathew, M., Karatzas, D., Jawahar, C.V.: Docvqa: A dataset for vqa on document images (2021), https://arxiv.org/abs/2007.00398
46. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions—i. Mathematical Programming **14**, 265–294 (1978), https://api.semanticscholar.org/CorpusID:206800425
47. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Proceedings of the 38th International Conference on Machine Learning (ICML) (2021)
48. Rogers, A., Kovaleva, O., Rumshisky, A.: A primer in BERTology: What we know about how BERT works. Transactions of the Association for Computational Linguistics **8** (2020). https://doi.org/10.1162/tacl_a_00349, https://aclanthology.org/2020.tacl-1.54/
49. Shang, Y., Cai, M., Xu, B., Lee, Y.J., Yan, Y.: Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22857–22867 (2025)
50. Shao, K., Tao, K., Qin, C., You, H., Sui, Y., Wang, H.: Holitom: Holistic token merging for fast video large language models (2025), https://arxiv.org/abs/2505.21334
51. Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8309–8318 (2019). https://doi.org/10.1109/CVPR.2019.00851
52. Song, D., Wang, W., Chen, S., Wang, X., Guan, M.X., Wang, B.: 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. pp. 7614–7623 (2025)
53. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NeurIPS) (2017)
54. Vig, J., Belinkov, Y.: Analyzing the structure of attention in a transformer language model. In: Linzen, T., Chrupała, G., Belinkov, Y., Hupkes, D. (eds.) Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. Association for Computational Linguistics, Florence, Italy (Aug 2019), https://aclanthology.org/W19-4808/
55. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q.V., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)
56. Wen, Z., Gao, Y., Wang, S., Zhang, J., Zhang, Q., Li, W., He, C., Zhang, L.: Stop looking for "important tokens" in multimodal language models: Duplication matters more. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 9972–9991 (2025)

57. Wu, H., Li, D., Chen, B., Li, J.: LongVideoBench: A benchmark for long-context interleaved video-language understanding. In: Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (eds.) Advances in Neural Information Processing Systems. vol. 37, pp. 28828–28857. Curran Associates, Inc. (2024). https://doi.org/10.52202/079017-0907
58. Xiao, G., Tian, Y., Chen, B., Han, S., Lewis, M.: Efficient streaming language models with attention sinks (2024), https://arxiv.org/abs/2309.17453
59. Xing, L., Huang, Q., Dong, X., Lu, J., Zhang, P., Zang, Y., Cao, Y., He, C., Wang, J., Wu, F., Lin, D.: Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction (2025), https://arxiv.org/abs/2410.17247
60. Yang, S., Chen, Y., Tian, Z., Wang, C., Li, J., Yu, B., Jia, J.: Visionzip: Longer is better but not necessary in vision language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19792–19802 (2025)
61. Yao, L., Huang, R., Hou, L., Lu, G., Niu, M., Xu, H., Liang, X., Li, Z., Jiang, X., Xu, C.: Filip: Fine-grained interactive language-image pre-training (2021), https://arxiv.org/abs/2111.07783
62. Ye, R., Jin, W., Wang, H., Wu, Y., Xu, J., Liu, Y., Luo, P.: mplug-owl: Modularization empowers large language models with multimodality. arXiv preprint arXiv:2304.14178 (2023)
63. Ye, W., Wu, Q., Lin, W., Zhou, Y.: Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 22128–22136 (2025)
64. Yu, W., Yang, Z., Li, L., Wang, J., Lin, K., Liu, Z., Wang, X., Wang, L.: Mm-vet: Evaluating large multimodal models for integrated capabilities (2024), https://arxiv.org/abs/2308.02490
65. Zhang, H., Lyu, M., He, C., Ao, Y., Lin, Y.: Trimtokenator: Towards adaptive visual token pruning for large multimodal models. arXiv preprint arXiv:2509.00320 (2025)
66. Zhang, H., Lyu, M., Huang, B., Ao, Y., Lin, Y.: Trimtokenator-lc: Towards adaptive visual token pruning for large multimodal models with long contexts (2025), https://arxiv.org/abs/2512.22748
67. Zhang, H., Ou, C., Yan, D., Wang, P., Yan, Q., Li, Y., Xiao, R., Shen, C.: Piofvlm: Rethinking training-free visual token reduction for vlm acceleration from an inference-objective perspective. arXiv preprint arXiv:2602.04657 (2026)
68. Zhang, K., Li, B., Zhang, P., Pu, F., Cahyono, J.A., Hu, K., Liu, S., Zhang, Y., Yang, J., Li, C., Liu, Z.: Lmms-eval: Reality check on the evaluation of large multimodal models (2024), https://arxiv.org/abs/2407.12772
69. Zhang, Q., Liu, M., Li, L., Lu, M., Zhang, Y., Pan, J., She, Q., Zhang, S.: Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. arXiv preprint arXiv:2506.10967 (2025)
70. Zhang, Y., Fan, C.K., Ma, J., Zheng, W., Huang, T., Cheng, K., Gudovskiy, D., Okuno, T., Nakata, Y., Keutzer, K., et al.: Sparsevlm: Visual token sparsification for efficient vision-language model inference. arXiv preprint arXiv:2410.04417 (2024)
71. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: Llava-video: Video instruction tuning with synthetic data (2025), https://arxiv.org/abs/2410.02713
72. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: MiniGPT-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023)

73. Zou, X., Lu, D., Wang, Y., Yan, Y., Lyu, Y., Zheng, X., Zhang, L., Hu, X.: Don't just chase "highlighted tokens" in mllms: Revisiting visual holistic context retention (2025), https://arxiv.org/abs/2510.02912