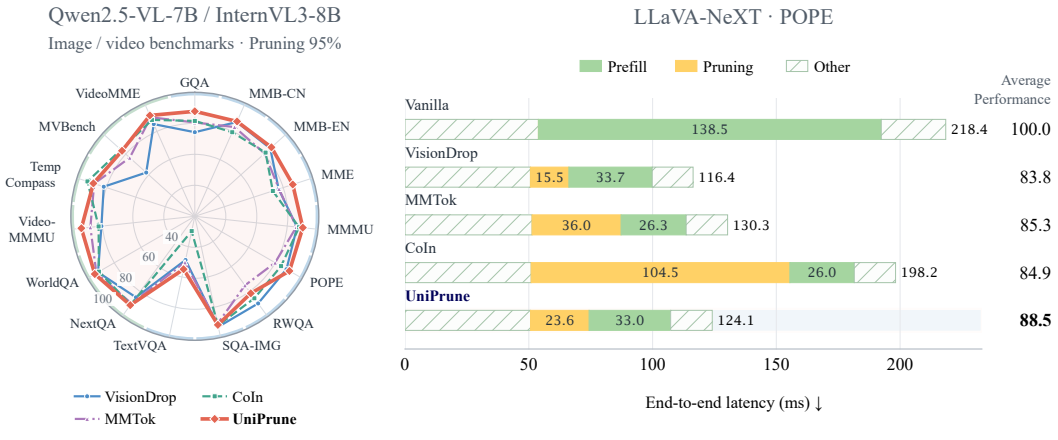
ONE OBJECTIVE, ONE STAGE: UNIFYING RELEVANCE, COVERAGE, AND REDUNDANCY IN VISUAL TOKEN PRUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal large language models (MLLMs) process large numbers of visual tokens alongside text, incurring substantial computation and inference latency. Existing training-free visual token pruning methods typically rely on single heuristics such as relevance, coverage, or redundancy. Although some approaches mitigate the limitations of single criteria through multi-stage or multi-module combinations, they still lack a principled, unified measure of the retained set value. Therefore, we propose **UniPrune**, a unified visual token pruning method that operates under a single objective in a single stage. UniPrune first utilizes cross-modal relevance operators to measure how visual tokens support text semantics. Subsequently, it uses Log-Sum-Exp (LSE) aggregation to combine the complementary support provided by different visual tokens, while its logarithmic form reduces the marginal benefit of redundant support. This objective jointly and explicitly captures relevance, semantic coverage, and redundancy suppression. We prove that this objective is monotone submodular, allowing the classical greedy algorithm to provably achieve an approximately optimal solution with low complexity. Across multiple benchmarks on three MLLM base models, UniPrune consistently achieves the highest average performance retention among all compared methods, including under the most aggressive 95% visual token pruning. UniPrune is independent of the vision encoder, requires minimal architectural changes, and remains compatible with acceleration methods such as FlashAttention.

1 INTRODUCTION



(a) Performance on 15 benchmarks **(b) Inference efficiency**

Figure 1: **Performance and efficiency overview.** (a) performance across fifteen image and video benchmarks. (b) end-to-end latency on LLaVA-NeXT-7B (POPE), with latency decomposed into prefill, pruning, and other processing.

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

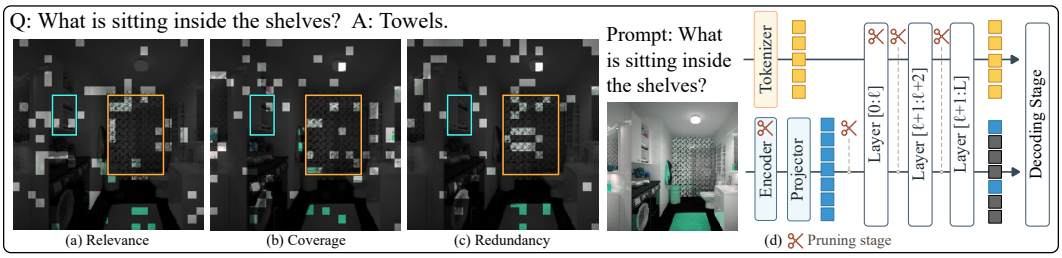


Figure 2: **Limitations of visual token pruning principles and an overview of pruning positions.** (a) Relevance-based pruning preserves relevant evidence in the blue box, but selection is locally concentrated in blocks. (b) Coverage-based pruning distributes tokens more evenly while also covering irrelevant regions in the orange box. (c) Redundancy-based pruning prioritizes less repetitive tokens but misses the relevant evidence in the blue box. (d) Existing pruning positions include the vision encoder, the projector output, and locations within or between Transformer layers, with single-stage or multi-stage pruning.

The success of large language models (LLMs) has established multimodal large language models (MLLMs/VLMs) as a common multimodal inference paradigm. A vision encoder converts an image into visual tokens, which are processed by the LLM together with text tokens (Liu et al., 2023; 2024a). This substantially increases computation and latency: a 336×336 image produces 576 visual tokens in LLaVA-1.5, while high-resolution models such as Qwen2.5-VL can produce up to 2880 tokens (Liu et al., 2024b; Bai et al., 2025). Prior work shows that many low-contribution tokens can be removed with limited degradation (Chen et al., 2024). However, pruning decisions depend jointly on the prompt and the visual tokens already retained. This raises a central challenge: how to preserve prompt-relevant and complementary visual tokens under a limited budget.

Existing visual token pruning methods mainly follow three principles. (1) **Relevance-based selection** retains tokens related to the prompt using attention or cosine similarity, as in FastV and SparseVLM (Chen et al., 2024; Zhang et al., 2025b). As shown in Figure 2(a), such methods may concentrate on redundant local regions. (2) **Coverage-based selection** seeks representative tokens: CoIn uses spatial-volume coverage, while MMTok maximizes coverage over visual and text tokens (Du et al., 2026; Dong et al., 2026). These methods distribute tokens more broadly, but may allocate budget to irrelevant regions, as shown in Figure 2(b). (3) **Redundancy-based selection** favors non-repetitive tokens; DART identifies repeated information using visual and text anchors (Wen et al., 2025). Low redundancy, however, does not guarantee task relevance, as illustrated in Figure 2(c). These principles can overlap, and recent methods combine them through multiple modules, token merging, or information recovery (Huang et al., 2026; Fang et al., 2026). Such pipelines may prune at different locations, including the vision encoder, projector, or Transformer layers, and may require multiple stages, as shown in Figure 2(d). Their differing inductive biases make principled comparison and consistent budget allocation difficult. Together, these challenges highlight two issues: the criteria lack a unified measure of set value, while combining them increases processing, configuration, and architectural complexity. This motivates asking whether relevance, coverage, and redundancy suppression can be captured by one set objective optimized in one stage.

We propose **UniPrune** (Unified Visual Token Pruning), a training-free method that unifies relevance, semantic coverage, and redundancy suppression through a single set objective. A cross-modal relevance operator quantifies how strongly each visual token supports each text token. For any retained subset, Log-Sum-Exp (LSE) aggregates support across visual tokens, while summation over text tokens rewards coverage of distinct semantic aspects. Because LSE exhibits diminishing returns, redundant evidence contributes progressively less to the objective. We prove that this objective is a monotone submodular function, so classical greedy selection achieves a $(1 - 1/e)$-approximation guarantee with low computational complexity. UniPrune selects tokens by maximizing marginal gain at a single pruning position, without requiring additional stages or modules. On LLaVA-1.5-7B, UniPrune retains 95.2% and 91.0% of full-model performance at extreme pruning ratios of 90% and 95%, respectively. Across three base models, fifteen image and video benchmarks, and multiple pruning ratios, UniPrune achieves the highest average performance retention, outperforming the strongest competing method by 0.7 to 3.0 percentage points, while delivering a $1.76\times$ end-to-end speedup and a $2.45\times$ prefill speedup on LLaVA-NeXT-7B.

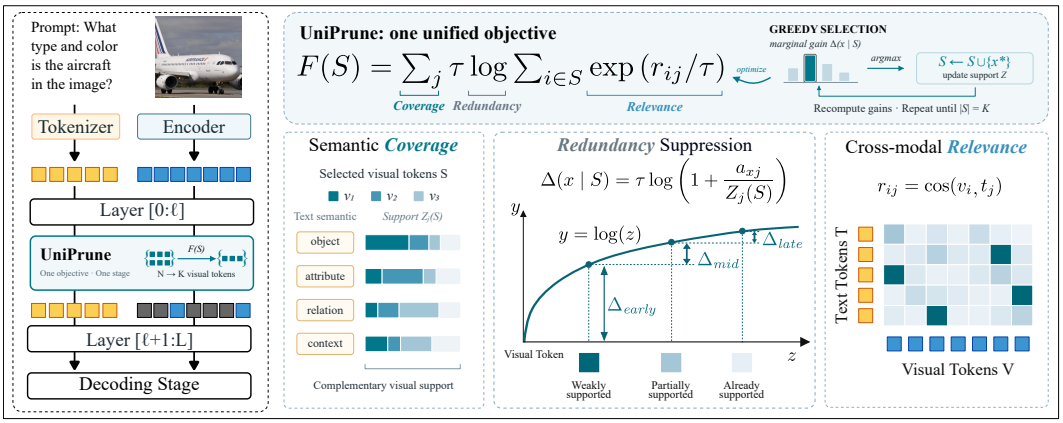


Figure 3: **UniPrune's pruning framework and unified set objective.** After an early LLM layer, visual tokens are pruned and passed to subsequent layers. Cross-modal relevance defines visual support for text semantics, logarithmic aggregation suppresses repeated evidence through diminishing returns, and summing over text tokens encourages semantic coverage. Greedy selection adds the token with the largest marginal gain and updates cumulative support.

Our contributions are threefold:

- **Method.** We introduce a unified visual token pruning objective and a single-stage framework that jointly captures relevance, semantic coverage, and redundancy suppression. By assigning value to retained token sets rather than scoring tokens independently, UniPrune supports set-aware marginal-gain selection without additional stages or modules.
- **Theory.** We derive the closed-form marginal gain of the proposed LSE set objective and prove that it is normalized, monotone, and submodular. Greedy selection therefore achieves a $(1-1/e)$-approximation guarantee under a fixed token budget with computational cost $O(NMd+KNM)$.
- **Experiments.** We systematically evaluate UniPrune on three base models and fifteen image and video benchmarks under multiple pruning ratios, including extreme compression settings. Ablation studies and visualizations further validate the benefits of set-level optimization and the single-objective pruning design.

2 Visual Token Pruning and Compression

Early visual token pruning builds on importance filtering, token merging, and representation reorganization in vision Transformers (Liang et al., 2022; Bolya et al., 2023; Cao et al., 2023), gradually incorporating language instructions. Methods differ in compression position, information preservation, and pipeline design. Relevance-based methods such as FastV and SparseVLM (Chen et al., 2024; Zhang et al., 2025b) may miss low-relevance tokens that still provide background, spatial relations, or context. VisionZip and LLaVA-PruMerge use vision-encoder features for compression, combining representative-token retention with feature aggregation (Yang et al., 2025; Shang et al., 2025), but may favor salient regions at the expense of background or complementary information. Nüwa performs staged pruning (Huang et al., 2026), while VisionTrim adopts a multi-module token compression framework (Yu et al., 2026). However, such designs require additional token processing, resulting in higher computational overhead and architectural complexity.

Another line of work formulates compression as subset-selection optimization, considering relationships among retained tokens and their ability to represent the original information. CoIn models informativeness and coverage to evaluate subsets (Du et al., 2026); MMTok preserves cross-modal information through maximum coverage (Dong et al., 2026). These methods formulate set objectives and offer a perspective on pruning design. However, existing set-optimization objectives require weighted combinations of heuristic criteria and do not unify them within a single objective. To address this limitation, UniPrune integrates cross-modal relevance, semantic coverage, and redundancy suppression into a set-value function. It selects tokens through a single iterative process without weighting multiple criteria, balancing performance and efficiency under a unified objective.

3 METHOD

We formulate token pruning as a set optimization problem under a fixed token budget. As Figure 3 illustrates, UniPrune prunes at an early LLM layer after an interaction between visual and text tokens. It computes cross-modal relevance, semantic coverage, and redundancy suppression through set aggregation, and performs single-stage greedy selection by marginal gain.

3.1 PROBLEM DEFINITION

Given visual and text inputs, let their representations at a model layer be

$$V = \{v_i\}_{i=1}^{N}, \quad v_i \in \mathbb{R}^d, \tag{1}$$

$$T = \{t_j\}_{j=1}^{M}, \quad t_j \in \mathbb{R}^d. \tag{2}$$

Here, N and M denote the numbers of visual and text tokens, respectively, d is their shared feature dimension, and i and j index their sequence positions.

Following the subset-selection formulation of CoIn (Du et al., 2026), we seek a subset $S \subseteq V$ with $|S| = K$ that preserves as much support for the text semantics as possible. The token retention budget satisfies $1 \leq K \ll N$. We next define an objective for this subset-selection problem.

3.2 A UNIFIED SET OBJECTIVE FOR RELEVANCE, COVERAGE, AND REDUNDANCY

Overall objective. For a visual subset S, UniPrune assigns the following LSE-based set value:

$$F(S) = \tau \sum_{j=1}^{M} \log\left(\sum_{v_i \in S} \exp(r_{ij}/\tau)\right). \tag{3}$$

For each text token, the inner exponential sum collects visual support and the logarithm controls the marginal value of additional support. The outer sum combines support across text tokens, so the objective rewards semantic coverage while reducing the contribution of redundant evidence. The terms in Equation 3 are constructed as follows.

Cross-modal relevance and positive support. We introduce a relevance operator $\mathcal{U}$ to measure the relationship between visual and text tokens. In our main formulation, this operator is cosine similarity:

$$r_{ij} = \mathcal{U}(v_i, t_j) = \frac{v_i^\top t_j}{\|v_i\|_2 \|t_j\|_2}. \tag{4}$$

Here, r_{ij} measures the relevance of visual token v_i to text token t_j. Alternative relevance operators are compared in Section 4.2 and Appendix D.6. We map relevance to positive support with

$$a_{ij} = \exp(r_{ij}/\tau),$$

where $\tau > 0$ is the temperature. This mapping preserves relevance ordering and gives stronger support to more relevant matches. The temperature controls how sharply relevance differences affect support: a smaller τ concentrates support on the strongest matches, whereas a larger τ makes contributions more uniform. We compute it from each sample's relevance distribution and token budget, as described in Appendix B.

Set-dependent support and diminishing returns. For a visual subset S, the support accumulated for text token t_j is

$$Z_j(S) = \sum_{v_i \in S} a_{ij}. \tag{5}$$

This inner sum accumulates evidence from multiple visual tokens for one text semantic. The term associated with each text token in *Equation* 3 is the LSE of these support values:

$$G_S(t_j) = \tau \log Z_j(S) = \tau \log\left(\sum_{v_i \in S} \exp(r_{ij}/\tau)\right). \tag{6}$$

The logarithm is increasing, so additional support remains beneficial. However, its strict concavity makes the same support increment less valuable when $Z_j(S)$ is already large. Candidate tokens that repeat well-supported semantics therefore receive smaller gains, which suppresses redundancy.

Semantic coverage and the unified objective. The support for each text token represents one semantic aspect. The outer sum in Equation 3 aggregates these aspects across the text sequence. It is this sum over text tokens, rather than the inner sum used by the LSE for one text token, that provides semantic coverage.

Thus, relevance determines the support magnitude, coverage is encouraged by summing support scores for each text token, and redundancy is suppressed by logarithmic diminishing returns.

3.3 Monotone Submodularity and Theoretical Guarantee

The LSE form in Equation 3 is defined for non-empty sets, but the token selection process over subsets starts from $S_0 = \varnothing$. To extend the objective to the empty set and obtain $F(\varnothing) = 0$, we introduce an infinitesimal ϵ. We use the same symbol F for this regularized form:

$$G_S(t_j) = \tau \log\left(1 + \frac{Z_j(S)}{\epsilon}\right), \qquad F(S) = \sum_{j=1}^{M} G_S(t_j). \tag{7}$$

Equivalently, $G_S(t_j) = \tau \log(\epsilon + Z_j(S)) - \tau \log \epsilon$. Thus, the baseline normalizes the LSE score at the empty set while retaining the same support accumulation and diminishing-return structure.

Using the complete visual set V, the subset loss caused by pruning is

$$D_{\text{set}}(S) = \sum_{j=1}^{M} \big[G_V(t_j) - G_S(t_j)\big] \geq 0. \tag{8}$$

Since $G_V(t_j)$ is constant with respect to S, minimizing this loss is equivalent to maximizing $F(S)$ under the fixed budget.

To solve Equation 7, we consider a candidate's marginal contribution to the current set. For any $S \subseteq V$ and an unselected visual token $x \in V \setminus S$, define

$$\Delta(x \mid S) = F(S \cup \{x\}) - F(S) = \tau \sum_{j=1}^{M} \log\left(1 + \frac{a_{xj}}{\epsilon + Z_j(S)}\right). \tag{9}$$

Here, a_{xj} is candidate x's support for text token t_j. The denominator $\epsilon + Z_j(S)$ makes the candidate's marginal contribution depend on existing support and remains positive even when $S = \varnothing$. As support for a text token accumulates, further support for that token yields smaller gains, encouraging selection of visual tokens that support less-covered semantics.

Theorem 1 (Normalization, monotonicity, and diminishing returns)**.** *For positive a_{ij}, τ, and ϵ, the objective F is normalized, nonnegative, monotone, and submodular over all subsets of V. In particular, $F(\varnothing) = 0$, and for any $A \subseteq B \subseteq V$ and $x \in V \setminus B$,*

$$F(A) \leq F(B), \qquad \Delta(x \mid A) \geq \Delta(x \mid B). \tag{10}$$

Proof. Since $Z_j(\varnothing) = 0$ and $Z_j(S) \geq 0$, the objective satisfies $F(\varnothing) = 0$ and $F(S) \geq 0$ for every $S \subseteq V$. For $A \subseteq B$, positive support values give $Z_j(A) \leq Z_j(B)$, so $F(A) \leq F(B)$. Moreover, for a fixed candidate x, each term in Equation 9 decreases as $Z_j(S)$ increases. Thus $\Delta(x \mid A) \geq \Delta(x \mid B)$, establishing submodularity, including when $A = \varnothing$. □

Theorem 1 shows that adding visual tokens cannot decrease the objective, while the marginal contribution of the same token cannot increase as the selected set grows. For a fixed budget $1 \leq K \leq |V|$, greedy selection starts from $S_0 = \varnothing$ and repeatedly adds an unselected token with the largest marginal gain until K tokens are retained. Because F is normalized, nonnegative, monotone, and submodular, the classical greedy guarantee (Nemhauser et al., 1978) gives

$$F(S_{\text{greedy}}) \geq (1 - 1/e) F(S^*). \tag{11}$$

Here, S^* is an optimal subset under the fixed budget, as defined in Equation 7. For a more detailed proof, please refer to Appendix C.1. The next section details the marginal gain computation and greedy selection procedure.

3.4 Greedy Selection and Complexity

We solve Equation 7 greedily by selecting the visual token with the largest marginal contribution at each step. Starting from the empty set $S_0 = \varnothing$, we initialize the cumulative support as $Z_j(S_0) = 0$ for $j = 1, \ldots, M$. For $k = 0, \ldots, K-1$, we select the candidate with the largest marginal gain and update the retained set:

$$x_{k+1} \in \arg\max_{x \in V \setminus S_k} \Delta(x \mid S_k), \qquad S_{k+1} = S_k \cup \{x_{k+1}\}. \tag{12}$$

Here, S_k contains the k visual tokens selected so far. At the first iteration, $S_0 = \varnothing$ and $Z_j(S_0) = 0$, so the marginal gain reduces to

$$\Delta(x \mid S_0) = F(\{x\}) = \tau \sum_{j=1}^{M} \log\Big(1 + \frac{a_{xj}}{\epsilon}\Big).$$

After selecting x_{k+1}, we update the cumulative support for each text token:

$$Z_j(S_{k+1}) = Z_j(S_k) + a_{x_{k+1}j}.$$

The updated support is used to compute the marginal gains in the next iteration, allowing candidate rankings to change as the selected set grows. After K iterations, the algorithm returns $S_{\text{greedy}} = S_K$.

Computing all visual–text relevance scores takes $O(NMd)$ time, and constructing the support matrix takes $O(NM)$ time. The K rounds of candidate evaluation take $O(KNM)$ time, giving a total complexity of $O(NMd + KNM)$. The implementation stores the support matrix and the M cumulative support values, requiring $O(NM)$ storage without constructing an $N \times N$ visual–visual similarity matrix.

Overall, UniPrune performs single-stage greedy token selection using the unified set objective after the projector or an early LLM layer, by default after the second LLM layer in this paper. The retained visual tokens are restored to their original order and passed to subsequent layers. Thus, UniPrune requires no attention maps, token merging, or reconstruction and is compatible with FlashAttention (Dao et al., 2022; Dao, 2024).

4 Experiments

4.1 Main Results

The existing methods can be roughly divided into four categories: **(i) single-stage pruning**, including FastV and DART; **(ii) multi-stage pruning**, including SparseVLM, VisionDrop, and PruneSID; **(iii) multi-module frameworks**, including VisionTrim, Nüwa, VLM-Pruner, HoloV, and Zoo-Prune; and **(iv) set-optimization methods**, including CoIn, MMTok, and UniPrune.

The experimental results in Tables 1–3 show that, although single-stage methods are simple and efficient, they typically rank tokens independently using a single heuristic criterion, which can lead to inferior performance. Multi-stage and multi-module methods introduce additional processing to capture spatial, semantic, or cross-layer information, and generally achieve better performance than single-stage methods at the cost of greater pipeline complexity. Set-optimization methods instead evaluate the value of a retained subset and explicitly model interactions among visual tokens. This design allows CoIn and MMTok to outperform many methods that do not optimize token subsets, especially when the token budget is tight.

On LLaVA-1.5-7B, UniPrune achieves the best average performance at both budgets, retaining 95.2% and 91.0% of full-model performance with 64 and 32 visual tokens, respectively. This comparison shows that UniPrune improves not only over single-stage and multi-module baselines, but also over existing set-optimization methods. Based on the average results in Table 1, we therefore compare VisionDrop, CoIn, MMTok, and UniPrune on the other two base models. On Qwen2.5-VL-7B, UniPrune achieves 88.6% and 84.0% retention at 10% and 5% token budgets, exceeding the strongest competitors by 1.8 and 3.0 percentage points, respectively. On InternVL3-8B, it reaches 93.1% and 90.4%, improving over the best competing methods by 1.3 and 2.2 points. Overall, UniPrune's consistent gains across multiple benchmarks and all three base models indicate that jointly modeling relevance, coverage, and redundancy yields stable and broadly applicable improvements rather than gains limited to a particular architecture or compression ratio.

Table 1: **LLaVA-1.5-7B results at 90%/95% pruning rates, corresponding to retention budgets of 64/32 visual tokens.** **Blue**, **bold**, and underline mark the best, second-best, and third-best pruned values within each budget and metric column. The same notation applies to the following tables.

Method	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
Upper bound, 576 tokens	62.0	55.8	64.1	1867	35.9	85.9	56.0	69.5	46.0	100.0%
LLaVA-1.5-7B	*Retain 64 tokens* (↓ 90%)									
FastV [ECCV24]	53.6	50.3	59.5	1664	35.7	68.3	48.6	**69.3**	36.9	89.4%
SparseVLM [ICML25]	53.7	49.0	59.8	1608	35.6	77.8	48.5	**69.4**	34.0	89.3%
DART [EMNLP25]	56.5	50.5	61.1	**1752**	35.0	73.9	51.9	68.9	38.4	92.2%
HoloV [NeurIPS25]	55.1	51.4	60.2	1696	35.8	76.9	51.5	68.7	41.2	92.9%
VisionDrop [AAAI26]	55.8	**52.0**	61.2	1666	35.9	78.9	51.1	69.0	**41.8**	93.5%
VLM-Pruner [CVPR26]	**59.2**	50.3	60.7	1741	35.4	81.6	52.3	68.4	40.7	**94.2%**
CoIn [CVPR26]	57.9	49.5	60.1	1663	35.0	**86.4**	50.3	68.5	41.2	93.5%
Zoo-Prune [CVPR26]	58.5	50.6	60.1	1657	34.8	**86.0**	49.7	68.2	**41.4**	93.5%
VisionTrim [ICLR26]	53.1	**51.6**	59.4	1612	35.8	70.3	50.5	68.7	40.8	90.7%
MMTok [ICLR26]	58.3	49.6	60.8	1713	35.6	85.6	50.3	68.8	41.1	94.1%
Nüwa [ICLR26]	**58.6**	49.7	**62.0**	1679	**36.2**	79.8	**52.9**	67.9	39.9	93.7%
PruneSID [ICLR26]	57.4	50.6	59.9	1650	34.9	83.6	50.9	68.2	41.1	93.1%
UniPrune	58.2	50.8	**62.2**	**1830**	**36.1**	80.9	**53.1**	69.2	40.5	**95.2%**
LLaVA-1.5-7B	*Retain 32 tokens* (↓ 95%)									
FastV [ECCV24]	49.7	44.0	54.3	1419	**35.7**	54.7	45.2	**69.2**	30.1	81.0%
DART [EMNLP25]	53.1	47.4	58.2	1625	34.4	65.7	47.8	69.1	32.7	86.3%
HoloV [NeurIPS25]	52.5	**49.2**	58.6	1608	35.0	70.3	49.7	68.8	**37.9**	88.9%
VisionDrop [AAAI26]	52.7	**49.2**	**59.1**	1559	35.0	72.1	50.2	**69.4**	**38.0**	89.2%
VLM-Pruner [CVPR26]	54.9	47.9	57.7	**1644**	**35.1**	70.0	49.7	**69.2**	33.7	88.2%
CoIn [CVPR26]	**55.5**	47.4	57.7	1598	33.2	**83.4**	48.6	68.3	36.0	89.3%
MMTok [ICLR26]	**56.2**	46.9	58.9	1638	34.4	**83.1**	48.0	69.0	36.9	**90.3%**
Nüwa [ICLR26]	52.8	45.5	58.5	1435	**35.7**	77.8	**54.0**	67.0	37.1	88.7%
PruneSID [ICLR26]	54.5	46.7	56.8	1581	34.3	75.5	47.8	67.9	33.9	87.3%
UniPrune	55.4	**49.0**	**60.1**	**1682**	34.8	77.3	**53.5**	68.8	35.2	**91.0%**

Table 2: **Qwen2.5-VL-7B results at 90%/95% pruning rates.**

Method	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
Upper bound (full)	60.8	80.1	84.2	2329	50.3	86.1	69.7	87.9	82.9	100.0%
Qwen2.5-VL-7B	*Retain 10% of tokens* (↓ 90%)									
FastV [ECCV24]	46.6	59.7	62.4	1880	**45.9**	67.5	58.4	80.3	**72.1**	82.0%
VisionDrop [AAAI26]	49.5	**74.7**	**76.0**	2005	45.8	78.7	**65.4**	81.2	51.0	**86.8%**
CoIn [CVPR26]	53.6	69.9	73.5	1829	43.8	**79.4**	**62.6**	**83.6**	34.3	83.0%
MMTok [ICLR26]	**54.0**	72.0	75.1	**2009**	45.8	77.6	58.7	80.4	**57.3**	86.7%
UniPrune	**55.0**	**73.9**	**76.6**	**2103**	**46.7**	**82.2**	59.9	**81.2**	55.5	**88.6%**
Qwen2.5-VL-7B	*Retain 5% of tokens* (↓ 95%)									
FastV [ECCV24]	39.7	45.5	50.8	1570	43.9	35.4	51.2	78.1	**60.3**	68.1%
VisionDrop [AAAI26]	45.1	**69.3**	**72.0**	1791	43.4	**76.0**	**62.4**	**81.7**	40.5	**81.0%**
CoIn [CVPR26]	**49.6**	63.6	68.6	1699	**44.1**	72.6	**59.5**	**80.9**	24.7	77.2%
MMTok [ICLR26]	49.0	66.5	67.9	**1797**	43.2	68.7	52.3	79.8	41.8	78.1%
UniPrune	**53.3**	**69.7**	**72.9**	**2014**	**45.3**	**78.0**	56.7	80.5	**45.5**	**84.0%**

Table 3: **InternVL3-8B video understanding results at 90%/95% pruning rates.**

Method	NExT-QA	WorldQA	Video-MMMU	TempCompass	MVBench	Video-MME	Avg.
Upper bound (full)	79.8	58.1	41.8	69.8	71.4	58.6	100.0%
InternVL3-8B	*Retain 10% of tokens* (↓ 90%)						
FastV [ECCV24]	**76.2**	**56.6**	35.0	62.0	61.7	**55.1**	91.0%
VisionDrop [AAAI26]	74.3	55.7	33.8	62.6	55.0	52.9	87.8%
CoIn [CVPR26]	**76.3**	54.6	35.0	**65.8**	**65.4**	53.7	**91.8%**
MMTok [ICLR26]	75.6	**56.0**	**36.1**	63.9	61.1	**55.1**	91.4%
UniPrune	75.5	55.1	**39.2**	**64.0**	**64.5**	**54.8**	**93.1%**
InternVL3-8B	*Retain 5% of tokens* (↓ 95%)						
FastV [ECCV24]	**72.6**	**55.1**	34.3	56.3	53.1	**53.0**	85.6%
VisionDrop [AAAI26]	67.6	53.5	33.7	57.1	44.4	49.9	81.1%
CoIn [CVPR26]	72.1	53.2	34.4	**64.8**	**60.1**	51.6	**88.2%**
MMTok [ICLR26]	71.5	54.2	**36.7**	61.6	54.7	52.7	87.6%
UniPrune	**72.5**	**54.9**	**39.1**	**62.0**	**59.4**	**53.4**	**90.4%**

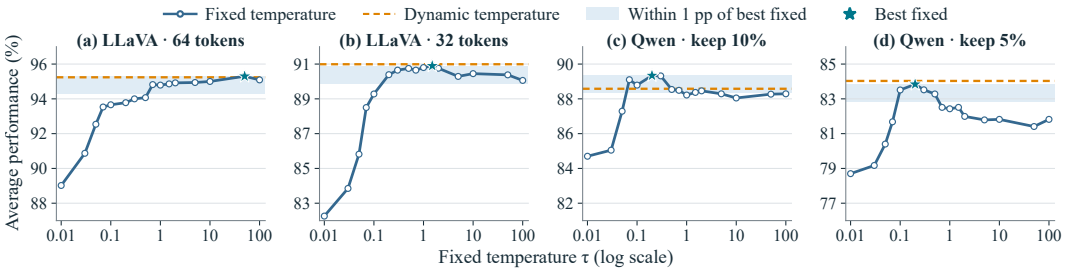


Figure 4: **Temperature sensitivity.** The temperature axis uses a logarithmic scale. Solid lines represent fixed temperatures and dashed lines represent dynamic temperatures. Stars identify the best fixed settings. Shaded bands indicate settings within one percentage point of the best fixed result.

4.2 Ablation Study

We analyze UniPrune's key designs from three perspectives: the unified objective, temperature settings, and relevance operators.

Unified Objective Ablation. To evaluate the unified objective, we compare four variants: **Frozen-Marginal Top-K** (fixes the initial ranking throughout selection); **Single-Prototype LSE** (pools all text tokens into one prototype); **Linear Support** (sums support without logarithmic aggregation); and **Hard-Max Coverage** (retains only the strongest match). Table 4 reports the average retention of all four variants across nine benchmarks. The results show that updating marginal gains from the selected set and preserving fine-grained text conditioning are both important. Linear Support likewise underperforms the full method. Removing the logarithmic transformation eliminates diminishing returns and makes marginal gains independent of the selected set, reducing the selection to a static top-K procedure. This ablation confirms that the complete objective requires both logarithmic aggregation and set-dependent marginal gains. In addition, under tighter budgets, the gap between Hard-Max Coverage and UniPrune narrows, consistent with the $\tau \log |S|$ bound in Equation 21. See Appendix C for the corresponding theoretical derivation.

Table 4: **Structural ablation of the unified objective on LLaVA-1.5-7B.**

Variant	64 tokens	32 tokens
Frozen-Marginal Top-K	67.2	63.1
Single-Prototype LSE	75.1	69.8
Linear Support	89.8	85.5
Hard-Max Coverage	91.6	90.0
UniPrune	**95.2**	**91.0**

Table 5: **Comparison of relevance operators on LLaVA-1.5-7B with fixed $\tau = 1$.**

Operator	Formula	64 tokens	32 tokens
Negative Euclidean	$-\|v_i - t_j\|_2$	94.3	89.4
Negative squared L2	$-\|v_i - t_j\|_2^2$	94.0	89.3
RBF (Gaussian)	$\exp\big(-\|v_i - t_j\|_2^2/(2\sigma^2)\big)$	94.1	89.3
Inner product	$v_i^\top t_j$	90.4	84.1
Cosine	$\cos(v_i, t_j)$	**94.8**	**90.8**

Temperature Importance and Sensitivity. We compare fixed and per-sample dynamic temperatures on LLaVA-1.5-7B and Qwen2.5-VL-7B under both budgets. Figure 4 sweeps 16 fixed temperatures: solid curves denote fixed-temperature results, dashed lines denote dynamic-temperature results, and shaded bands mark settings within one percentage point of the best fixed result. The default $\tau = 1$ is not consistently optimal, showing the importance of temperature scaling. Broad effective temperature ranges exist for both models and budgets, indicating low temperature sensitivity. Fixed temperatures can provide comparable or even better accuracy, so we do not treat dynamic temperature as a core innovation or contribution. Detailed comparisons of fixed and dynamic temperatures are provided in Appendix B.3.

Comparison of Relevance Operators. To test the unified objective's dependence on the relevance operator, we fix $\tau = 1$ to avoid the temperature mechanism's effects across operators, and change only the relevance operator. Table 5 compares five representative operators covering distances, kernels, inner products, and cosine similarity. Here, $\sigma > 0$ is the Gaussian kernel bandwidth. Additional operators and results appear in Table D6 in Appendix D.6. Cosine similarity gives the best result under both budgets in Table 5, supporting its use as an effective relevance measure. Cosine similarity compares the directions of visual and text embeddings rather than their magnitudes, making the relevance scores more robust to differences in feature scale.

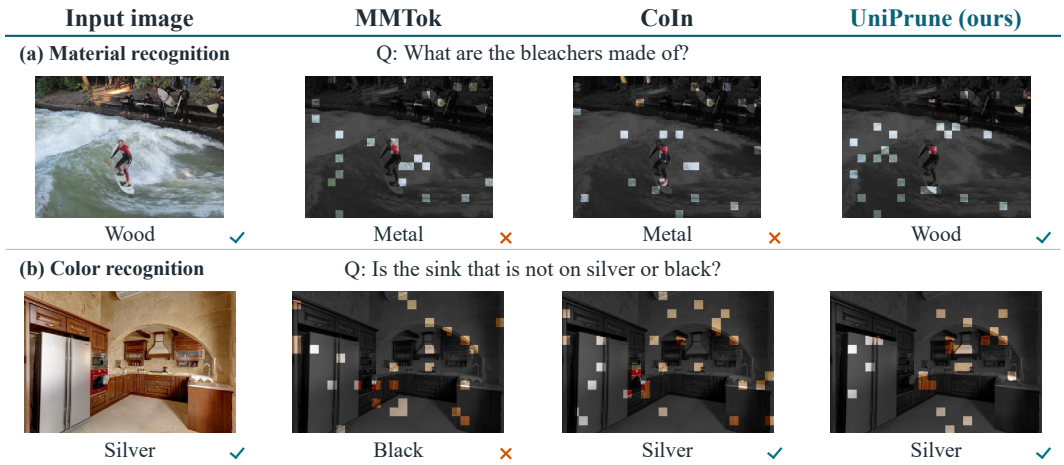


Figure 5: **Case studies at 32 retained visual tokens.** Comparison of regions retained by MMTok, CoIn, and UniPrune. Checkmarks and crosses indicate correct and incorrect answers, respectively.

4.3 Efficiency and Acceleration Analysis

Inference efficiency. To evaluate inference efficiency, we measure inference resource usage on LLaVA-NeXT-7B, as shown in Table 6. UniPrune achieves a 1.76× end-to-end (E2E) speedup and a 2.45× effective prefill speedup. Compared with set optimization methods MMTok and CoIn, UniPrune achieves lower E2E latency and higher average retention. Although UniPrune lacks a speed advantage over heuristic methods such as FastV, it improves average performance by 10.3 percentage points. Thus, its selection cost yields substantial gains over heuristic pruning and a better efficiency-performance tradeoff than set optimization baselines.

Table 6: **Inference efficiency, pruning overhead, memory, and performance.** Blue, green, and orange headers denote time in milliseconds, memory in GiB, and performance metrics, respectively. Avg. is average performance at 95% pruning over Tables 1–3.

Method	E2E	Prefill	Pre	Post	Prun	Mem.	POPE	Avg.
Vanilla	218.4	138.5	53.9	26.0	0.0	14.9	86.4	100.0
FastV	106.5	37.3	50.8	17.1	1.3	13.7	78.6	78.2
VisionDrop	116.4	33.7	50.7	16.5	15.5	14.6	85.1	83.8
CoIn	198.2	26.0	51.0	16.7	104.5	13.4	85.9	84.9
MMTok	130.3	26.3	51.3	16.7	36.0	13.5	85.4	85.3
UniPrune	124.1	33.0	50.7	16.8	23.6	13.6	85.6	**88.5**

4.4 Visualization Analysis

To visualize the actual token choices and QA outcomes of different pruning methods, Figure 5 presents their selections on the same samples. MMTok, CoIn, and UniPrune retain 32 visual tokens, enabling direct comparison of both selected regions and final answers. In both examples, UniPrune preserves complementary evidence for material and color recognition, whereas the competing methods focus on incomplete or redundant regions and therefore produce an incorrect answer in at least one case. Additional examples appear in Appendix Figures D4 and D5.

5 Conclusion

This paper presented **UniPrune**, a training-free visual token pruning framework that unifies relevance, semantic coverage, and redundancy suppression through a single cross-modal Log-Sum-Exp set objective. Its submodular formulation supports efficient greedy selection, while single-stage pruning requires minimal architectural changes and remains compatible with different MLLMs and acceleration techniques. Experiments show that UniPrune consistently preserves strong image and video understanding performance under aggressive token compression. Future work will investigate UniPrune on increasingly capable native MLLMs, where higher visual token counts and longer contexts make efficient token pruning increasingly important.

REFERENCES

Shuai Bai et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations (ICLR)*, 2023.

Qingqing Cao, Bhargavi Paranjape, and Hannaneh Hajishirzi. Pumer: Pruning and merging tokens for efficient vision language models. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023. doi: 10.18653/v1/2023.acl-long.721.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. *Lecture Notes in Computer Science*, 2024. doi: 10.1007/978-3-031-73004-7_2.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations (ICLR)*, 2024.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems 35*, 2022. doi: 10.52202/068431-1189.

Sixun Dong, Juhua Hu, Mian Zhang, Ming Yin, Yanjie Fu, and Qi Qian. Mmtok: Multimodal coverage maximization for efficient inference of vlms. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2508.18264.

Chenxi Du, Yongheng Deng, Jiani Liu, Yujia Zhang, Xi Chen, and Ju Ren. Coin: Coverage and informativeness-guided token reduction for efficient large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Zhengyao Fang, Pengyuan Lyu, Chengquan Zhang, Guangming Lu, Jun Yu, and Wenjie Pei. Prune redundancy, preserve essence: Vision token compression in vlms via synergistic importance-diversity. In *International Conference on Learning Representations (ICLR)*, 2026.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. Mme: A comprehensive evaluation benchmark for multimodal large language models. *Advances in Neural Information Processing Systems 38*, 2025a. doi: 10.52202/085713-4899.

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, Ran He, and Xing Sun. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025b. doi: 10.1109/cvpr52734.2025.02245.

Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Xiang Yue, Bo Li, Yuanhan Zhang, and Ziwei Liu. Video-mmmu: Evaluating knowledge acquisition from multidisciplinary professional videos. *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2026. doi: 10.18653/v1/2026.acl-long.1281.

Yuhang Huang, Fan Ma, Yun Shao, Jun Guo, Zhen Yu, Lin Cui, and Qi Tian. Nüwa: Mending the spatial integrity torn by VLM token pruning. In *International Conference on Learning Representations (ICLR)*, 2026.

Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. doi: 10.1109/cvpr.2019.00686.

Young Kim, Yifan Zhang, Haoran Liu, Ayoung Jung, Seong Lee, and Seung Hong. Zoo-prune: Training-free token pruning via zeroth-order gradient estimation in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Bo Li, Peiyuan Zhang, Kaichen Zhang, Fanyi Pu, Xinrun Du, Yuhao Dong, Haotian Liu, Yuanhan Zhang, Ge Zhang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Accelerating the development of large multimodal models, March 2024a. URL https://github.com/EvolvingLMMs-Lab/lmms-eval.

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. Mvbench: A comprehensive multi-modal video understanding benchmark. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b. doi: 10.1109/cvpr52733.2024.02095.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. doi: 10.18653/v1/2023.emnlp-main.20.

Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations (ICLR)*, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems 36*, 2023. doi: 10.52202/075280-1516.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge. https://llava-vl.github.io/blog/2024-01-30-llava-next/, 2024b. Blog post.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player? *Lecture Notes in Computer Science*, 2024c. doi: 10.1007/978-3-031-72658-3_13.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. Tempcompass: Do video llms really understand videos? *Findings of the Association for Computational Linguistics ACL 2024*, 2024d. doi: 10.18653/v1/2024.findings-acl.517.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems 35*, 2022. doi: 10.52202/068431-0182.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 1978. doi: 10.1007/bf01588971.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. doi: 10.1109/iccv51701.2025.02122.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. doi: 10.1109/cvpr.2019.00851.

Zichen Wen, Yifeng Gao, Shaobo Wang, Junyuan Zhang, Qintong Zhang, Weijia Li, Conghui He, and Linfeng Zhang. Stop looking for "important tokens" in multimodal language models: Duplication matters more. *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025. doi: 10.18653/v1/2025.emnlp-main.505.

Zhipeng Wu, Xin Ma, Zhichao Ni, Dan Zhang, Hai Shu, Xin Jiang, and Xiang Chen. Vlm-pruner: Buffering for spatial sparsity in an efficient vlm centrifugal token pruning paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

xAI. RealWorldQA. https://huggingface.co/datasets/xai-org/RealworldQA, 2024.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. doi: 10.1109/cvpr46437.2021.00965.

Rui Xu, Yunke Wang, Yong Luo, and Bo Du. Rethinking visual token reduction in lvlms under cross-modal misalignment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. doi: 10.1609/aaai.v40i32.39949.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. doi: 10.1109/cvpr52734.2025.01843.

Hao Yu, Wenhao Li, Xiao Qu, Shuang Wang, Jing Chen, and Jiayan Zhu. Visiontrim: Unified vision token compression for training-free mllm acceleration. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2601.22674.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. doi: 10.1109/cvpr52733.2024.00913.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models. *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025a. doi: 10.18653/v1/2025.findings-naacl.51.

Y. Zhang et al. Worldqa: Multimodal world knowledge in videos through long-chain reasoning. *arXiv preprint arXiv:2405.03272*, 2024.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. Sparsevlm: Visual token sparsification for efficient vision-language model inference. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2025b.

Jinguo Zhu et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

Xin Zou, Di Lu, Yizhou Wang, Yibo Yan, Yuanhuiyi Lyu, Xu Zheng, Linfeng Zhang, and Xuming Hu. Don't just chase "highlighted tokens" in mllms: Revisiting visual holistic context retention. *Advances in Neural Information Processing Systems 38*, 2025. doi: 10.52202/085713-1327.

A Detailed Experimental Setup

This appendix provides detailed descriptions of the datasets, models, baselines, and evaluation configurations. All models are frozen, and pruning does not require any additional training or task-specific fine-tuning. All baseline evaluations across all benchmarks were conducted using the official lmms-eval (Zhang et al., 2025a; Li et al., 2024a) evaluation toolkit, and the results were calculated according to its official evaluation protocols and metrics.

A.1 Benchmarks

A.1.1 Image Understanding

The main experiments are conducted on nine complementary image understanding benchmarks, covering various types of question answering, hallucination evaluation, OCR, real-world scene understanding, and related capabilities. Detailed descriptions of the benchmarks are provided below.

GQA (Hudson & Manning, 2019) constructs compositional questions based on scene graphs, primarily testing objects, attributes, spatial relations, and multi-step visual reasoning. We evaluate on the balanced test-dev split and report the percentage of answers that match the reference answers after ignoring case and punctuation.

MMBench-EN/CN (Liu et al., 2024c) uses English and Chinese multiple-choice questions to evaluate the overall capabilities of models across the two major categories of perception and reasoning, as well as more fine-grained capability dimensions. We report development-set accuracy (%), following the evaluator's circular protocol: all option permutations corresponding to the same original question must be answered correctly for that question to be counted as correct.

MME (Fu et al., 2025a) contains 14 subtasks across perception and cognition categories and uses paired yes/no questions to reduce the uncertainty of free-form evaluation. We report the sum of the perception and cognition scores. Each subtask contributes question-level accuracy plus paired-question accuracy (Acc+), both measured on a 0–100 scale; Acc+ requires both questions associated with an image to be answered correctly. The maximum total score is 2,800.

MMMU (Yue et al., 2024) covers university-level and professional multidisciplinary questions involving charts, diagrams, maps, music scores, and scientific images, and is designed to evaluate expert-level knowledge and visual reasoning. We report validation-set accuracy (%) using the benchmark's multiple-choice and open-ended answer-scoring rules.

POPE (Li et al., 2023) measures object hallucination by asking whether an object is present in an image. We report the F1 score (%), defined as the harmonic mean of precision and recall, with object presence (a "yes" answer) treated as the positive class.

RealWorldQA (RWQA) (xAI, 2024) focuses on understanding spatial relations, scale, distance, and physical layouts in real-world scenes. We report test-set accuracy (%), matching extracted options or normalized short answers against the reference answers.

ScienceQA-IMG (SQA-IMG) (Lu et al., 2022) selects science multiple-choice questions with image context, testing multi-step reasoning across natural sciences, social sciences, and language sciences. We report the percentage of multiple-choice questions answered correctly in the image-containing test subset.

TextVQA (Singh et al., 2019) requires reading and using text in natural images to answer questions, and is therefore sensitive to whether fine-grained text regions are pruned. We report validation-set VQA soft accuracy (%) following the standard TextVQA evaluation protocol.

A.1.2 Video Understanding

Video experiments are conducted on six benchmarks covering temporal perception, causal reasoning, world knowledge, and long-video understanding. Detailed descriptions of the benchmarks are provided below.

NExT-QA (Xiao et al., 2021) extends video question answering beyond scene description to causal action reasoning, temporal action reasoning, and commonsense scene understanding. We evaluate multiple-choice test questions and report the percentage of predictions matching the correct answers.

WorldQA (Zhang et al., 2024) contains 303 videos and 1,007 question-answer pairs, emphasizing the joint use of audio-visual information, world knowledge, and long-chain reasoning. We evaluate

its generation and multiple-choice test tasks. The former uses the evaluator's average graded answer score, while the latter uses the percentage of multiple-choice questions answered correctly. We report the average score of the generation and multiple-choice tasks for WorldQA.

Video-MMMU (Hu et al., 2026) consists of 300 expert-level videos and 900 human-annotated questions spanning six disciplines. It evaluates the acquisition of video knowledge across three stages: perception, understanding, and transfer application. We report the average accuracy across perception, understanding, and adaptation.

TempCompass (Liu et al., 2024d) diagnoses the temporal perception of video LLMs through different temporal attributes, such as speed, direction, and order, and multiple task formats. It uses videos with similar static content but conflicting dynamic attributes to reduce single-frame bias. We report the average accuracy across four task formats: caption matching, caption generation, multiple choice, and yes/no.

MVBench (Li et al., 2024b) contains 20 types of video tasks that require dynamic information and systematically evaluates capabilities ranging from temporal perception to cognitive reasoning in a multiple-choice format. We calculate the percentage of correct answers within each subtask and report the average score across all 20 subtasks.

Video-MME (Fu et al., 2025b) covers six major video domains and 30 subdomains, containing 900 videos and 2,700 questions. Video lengths range from 11 seconds to 1 hour, and the benchmark supports evaluation with or without subtitles and audio. We evaluate without subtitles and report the percentage of test questions answered correctly, aggregated across questions from all video durations.

The video input conditions used for all six benchmarks are specified in Appendix A.5.

A.2 Base Models

LLaVA-1.5-7B (Liu et al., 2024a) uses a CLIP ViT-L/14@336px visual encoder (Radford et al., 2021) and produces 576 visual tokens per image, with Vicuna-7B as the language model. It is used for a unified image evaluation against a large number of training-free pruning methods.

Qwen2.5-VL-7B (Bai et al., 2025) uses a native dynamic-resolution visual encoder with windowed visual processing; its visual token count varies with the input resolution, which tests the generalization of UniPrune to different visual architectures and token scales.

InternVL3-8B (Zhu et al., 2025) is natively multimodal-pretrained and supports extended multimodal contexts. Here, it is used with multi-frame video input to test the applicability of set selection to long visual sequences.

LLaVA-NeXT-7B (Liu et al., 2024b) adopts a higher-resolution AnyRes input strategy and is therefore suitable for comparing the end-to-end latency, pruning overhead, and memory footprint of different methods over longer visual sequences. It is also used as the common base model for measuring inference speed across all methods.

A.3 Baselines

We selected representative training-free methods spanning four design paradigms: single-stage pruning, multi-stage pruning, multi-module frameworks, and set-optimization methods. Some methods may belong to more than one category. These methods cover attention ranking, cross-modal relevance, visual redundancy, spatial integrity, cross-modal coverage, and explicit recovery or fusion mechanisms.

- **FastV** (Chen et al., 2024) is a representative single-stage pruning method that uses the attention scores from early LLM layers to rank visual tokens and removes the tokens with low attention in a single pruning step.
- **DART** (Wen et al., 2025) is a single-stage redundancy-based method. Rather than identifying important tokens solely from their individual scores, it selects a small set of pivot tokens and removes candidate tokens that are redundant with respect to the pivots.
- **SparseVLM** (Zhang et al., 2025b) is a multi-stage pruning framework that estimates token importance from cross-modal attention, progressively applies an adaptive sparsification ratio, and recycles discarded tokens to preserve useful visual information across stages.

- **VisionDrop** (Xu et al., 2026) performs progressive multi-stage pruning across the visual encoder and the LLM. It addresses cross-modal misalignment through intra-visual attention-based selection and combines pruning with lightweight context merging.
- **PruneSID** (Fang et al., 2026) is a multi-stage method that jointly models token importance and diversity during pruning, progressively retaining complementary visual evidence while removing redundant tokens.
- **VisionTrim** (Yu et al., 2026) is a multi-module framework that combines global and local dominant-token selection, text-guided visual supplementation, and token merging to perform visual selection and context recovery.
- **Nüwa** (Huang et al., 2026) introduces a dedicated spatial reconstruction module that repairs the spatial integrity damaged by token pruning and improves the continuity of the retained visual representation.
- **VLM-Pruner** (Wu et al., 2026) combines multiple modules within the LLM, including a buffering mechanism and spatial sparsity modeling, to reduce visual tokens while preserving spatially structured information.
- **HoloV** (Zou et al., 2025) uses holistic visual-context modeling to alleviate the local bias caused by selecting only high-response regions, and further incorporates an explicit token recovery mechanism.
- **Zoo-Prune** (Kim et al., 2026) estimates the influence of visual tokens on the model output through zeroth-order gradients. Its multi-module design supports both token ranking and token recovery without requiring backpropagation.
- **CoIn** (Du et al., 2026) formulates pruning as a set-selection problem that jointly considers token informativeness and coverage, making it a representative set-optimization method and one of the closest baselines to our set-based modeling view.
- **MMTok** (Dong et al., 2026) formulates visual token selection as a maximum-coverage problem over text tokens and the original visual tokens, explicitly optimizing the selected token set for cross-modal coverage.

All compared baselines follow the settings in their original papers or open-source repositories.

A.4 Summary Metric

For each benchmark b, let q_b^{pruned} and q_b^{full} denote the scores obtained by the pruned model and the corresponding unpruned full model, respectively. The average performance retention over B benchmarks is

$$\text{Avg.} = \frac{100}{B} \sum_{b=1}^{B} \frac{q_b^{\text{pruned}}}{q_b^{\text{full}}}.$$

This gives equal weight to benchmarks with different score scales; 100% denotes full-model performance on average. The image and video suites contain $B = 9$ and $B = 6$ benchmarks, respectively.

A.5 Implementation Details

UniPrune performs a single pruning after the second LLM layer. The support matrix and candidate marginal gains are computed in a vectorized manner on the GPU. The default relevance operator is cosine similarity, followed by row-wise z-score normalization over the text-token dimension.

Specifically, the default operator is defined as

$$r_{ij} = \frac{c_{ij} - \mu_i}{\sigma_i + 10^{-6}}, \qquad c_{ij} = \cos(v_i, t_j), \quad \mu_i = \text{Mean}_j(c_{ij}), \quad \sigma_i = \text{Std}_j(c_{ij}).$$

The main configuration determines the retained-token budget according to the target compression ratio. The temperature is computed independently for each sample from the relevance distribution and token budget, as described in Appendix B.

Video Input and Pruning. For InternVL3-8B, we sample eight frames from uniformly spaced temporal segments and process each frame as an individual 448×448 image tile. The input consists of the video frames and the task prompt, without audio or additional subtitles. For each pruning method, $\lceil \rho N_f \rceil$ tokens are retained from each frame, where N_f is the number of visual tokens in that frame and $\rho \in \{0.10, 0.05\}$. The retained tokens preserve their original sequence order.

Objective and Initialization. We implement the objective F defined in Equation 7, using the baseline support parameter $\epsilon = 10^{-8}$ in all experiments. Greedy selection starts from $S_0 = \varnothing$ with $Z_j(S_0) = 0$, and the same algorithm is used to compute marginal gains at every step.

Evaluation and Reproducibility. All experiments use the open-source multimodal evaluation framework lmms-eval (Zhang et al., 2025a; Li et al., 2024a), which provides unified model inference, data loading, question templates, answer parsing, and metric computation. Each method is evaluated under the same protocol for every backbone model and benchmark to ensure fair comparison. All experiments are conducted on NVIDIA H100 GPUs with a batch size of 1. Random seeds follow the fixed settings provided by lmms-eval. For benchmarks requiring LLM-as-a-judge evaluation, we use OpenAI's GPT-4o-mini following the configuration guidelines of lmms-eval. Image and video benchmarks use the corresponding official scoring protocols implemented in the framework. The average performance retention is computed according to the definition in Appendix A.4.

B Dynamic Temperature

B.1 Fixed-Temperature Sweep

This subsection reports the complete fixed-temperature sweep corresponding to Figure 4 and Table B1. Each run uses one fixed temperature for all samples. LLaVA-1.5-7B retains 64 or 32 of its 576 visual tokens, and Qwen2.5-VL-7B retains 10% or 5% of its dynamic number of visual tokens.

Avg. uses the nine-task retention metric in Appendix A.4, with the unpruned references in Tables 1 and 2. In Tables B2–B5, the shaded row marks the common reference $\tau = 1$, blue Avg. cells mark the optimal temperature ranges (within one percentage point of the best fixed result), and bold marks the best fixed temperature.

Effective temperature ranges across models. Tables B2–B5 show that good performance is not confined to one temperature. Using a gap of at most one percentage point from the best fixed result, all measured temperatures in $[0.7, 100]$ and $[0.2, 100]$ qualify for LLaVA-1.5-7B at 64 and 32 tokens, respectively. For Qwen2.5-VL-7B at 10% retention, qualifying measured temperatures are $0.07, 0.1, 0.2, 0.3, 0.5, 0.7, 1.5, 2$; at 5%, they are $0.1, 0.2, 0.3, 0.5$. Based on our analysis and observations, we attribute the differences in the optimal temperature distributions to differences in r_{ij} distributions produced by different base models. The role of the temperature mechanism is precisely to modulate the effective scaling of these differences.

Necessity of the temperature mechanism. By Equation 21, finite temperatures allow other visual tokens to contribute. The sweep demonstrates the value of this accumulation: for LLaVA-1.5-7B at 32 tokens, $\tau = 1$ improves average retention from 82.26% at $\tau = 0.01$ to 90.80%, with POPE increasing from 56.34 to 78.77 and TextVQA from 27.75 to 34.11. This is consistent with the need for sufficient visual evidence in object judgment and scene-text recognition. Therefore, the temperature mechanism effectively adjusts the preference between emphasizing key matches and accumulating complementary support by changing the matching strength.

B.2 Computing the Per-Sample Dynamic Temperature

We configure temperature dynamically from each sample's relevance distribution and token budget. For sample x, let $U_x^{\text{raw}} = \{c_{ij}\}$ denote the original cosine-similarity matrix before normalization, and measure score dispersion by

$$s_x = \text{Std}(U_x^{\text{raw}}). \tag{13}$$

The normalized relevance scores r_{ij} are used for support computation and greedy selection, whereas s_x is computed from the raw cosine-similarity matrix. When s_x is small, candidates have similar relevance, so a lower temperature amplifies subtle differences. When s_x is large, candidates are already well distinguished, so a higher temperature can be used. We combine this motivation with candidate scale and token budget. Let N_x and K_x be the original visual-token count and retained-token budget, and construct the effective candidate scale

$$C_x = N_x^{1-s_x^2}. \tag{14}$$

This heuristic decreases as s_x increases, reflecting more competitive candidates when relevance scores are similar. Taking s_x as a typical relevance difference, a token with score $r - s_x$ provides the following exponential support relative to a representative token with score r:

$$\frac{\exp((r - s_x)/\tau_x)}{\exp(r/\tau_x)} = \exp\left(-\frac{s_x}{\tau_x}\right).$$

To connect this support difference to the budget, we match its relative attenuation to the effective retention ratio K_x/C_x:

$$\exp\left(-\frac{s_x}{\tau_x}\right) \approx \frac{K_x}{C_x} \quad \Longleftrightarrow \quad \exp\left(\frac{s_x}{\tau_x}\right) \approx \frac{C_x}{K_x}. \tag{15}$$

For $s_x > 0$ and $C_x/K_x > 1$, this gives

$$\tau_x = \frac{s_x}{\log(C_x/K_x)}. \tag{16}$$

To avoid a zero denominator or negative temperature, the implementation enforces a denominator lower bound of $\log(1 + 1/K_x) > 0$ and a temperature lower bound of 10^{-6}, ensuring a positive, finite temperature even when $C_x \leq K_x$.

The rule adapts to both relevance differences and budget constraints. For a fixed original token count and budget, more similar relevance scores lead to a lower temperature, thereby amplifying their differences. For the same sample, a tighter budget also results in a lower temperature, thereby strengthening the preference for highly relevant matches.

B.3 Fixed versus Dynamic Temperature

Table B1: **Fixed versus dynamic temperature.** Indicators are average performance retention (%); parentheses give the τ corresponding to the best fixed result.

Model	Budget	$\tau = 1.0$	Best fixed (τ)	Dynamic
LLaVA-1.5-7B	64 tokens	94.8	**95.3** (50.0)	95.2
LLaVA-1.5-7B	32 tokens	90.8	90.9 (1.5)	**91.0**
Qwen2.5-VL-7B	10%	88.2	**89.3** (0.2)	88.6
Qwen2.5-VL-7B	5%	82.4	83.8 (0.2)	**84.0**

Automatic configuration across models. In Table B1, dynamic temperature improves over fixed $\tau = 1$ by 0.45, 0.19, 0.36, and 1.60 retention percentage points across the four configurations, with the largest gain for Qwen2.5-VL-7B at 5% retention. Appendix B.1 shows that smaller finite temperatures suit this Qwen budget. The dynamic rule adjusts support contrast to each sample's distribution and budget, improving on a common fixed value.

Comparison with the best swept configurations. Dynamic temperature achieves 95.24% and 90.99% at the two LLaVA budgets, close to the best fixed results of 95.30% and 90.90%. For Qwen at 5%, it achieves 84.03%, above the best fixed result of 83.84%; at 10%, the best fixed result is 89.34% versus 88.58% dynamically. Thus, the dynamic rule achieves overall performance close to the sweep optimum without sweeping fixed temperatures for each configuration. Its main value is simplifying temperature configuration across models and budgets, rather than providing a performance advantage.

Table B2: **LLaVA-1.5-7B, full fixed-temperature ablation keeping 64 visual tokens.**

τ	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
100	58.1	50.6	62.5	1821.7	35.8	80.9	53.2	69.0	40.4	95.1%
50	58.3	50.7	62.6	1819.0	36.1	81.1	53.3	69.1	40.4	**95.3%**
10	58.2	50.5	62.8	1809.7	35.4	81.1	52.9	69.3	40.5	95.0%
5	58.2	50.7	62.5	1814.4	35.1	81.2	53.2	69.1	40.5	94.9%
2	58.0	50.5	61.9	1785.1	35.6	81.8	53.6	69.3	40.5	94.9%
1.5	58.2	50.7	62.1	1811.2	34.9	81.9	53.2	69.3	40.3	94.9%
1	58.0	50.3	62.1	1813.7	35.0	81.8	53.2	69.3	40.3	94.8%
0.7	58.3	51.0	62.3	1814.0	35.1	81.6	52.8	69.2	39.8	94.8%
0.5	58.3	51.1	61.9	1775.0	35.2	81.0	51.6	69.0	39.2	94.1%
0.3	58.5	51.0	61.8	1818.8	35.2	80.8	51.5	68.9	38.1	94.0%
0.2	58.7	51.4	61.9	1795.7	35.0	80.8	51.0	69.3	37.6	93.8%
0.1	58.8	50.6	61.5	1819.3	35.1	81.5	51.2	68.4	37.4	93.7%
0.07	58.7	51.7	61.6	1743.7	35.8	79.4	51.0	69.0	37.8	93.5%
0.05	57.6	51.8	61.7	1695.8	35.7	74.1	51.5	69.5	37.8	92.5%
0.03	56.2	51.3	60.5	1743.7	35.6	70.5	51.5	68.2	35.0	90.9%
0.01	56.0	51.0	59.6	1635.7	34.2	69.6	50.2	68.3	34.1	89.0%

Table B3: **LLaVA-1.5-7B, full fixed-temperature ablation keeping 32 visual tokens.**

τ	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
100	54.4	49.0	59.9	1689.3	34.2	75.1	51.6	68.5	35.5	90.1%
50	54.5	48.6	60.0	1702.1	34.8	76.0	52.4	68.2	35.1	90.4%
10	54.8	48.5	59.5	1712.1	34.3	76.6	52.8	68.3	35.1	90.5%
5	54.8	48.5	58.5	1702.3	34.0	76.9	53.6	68.2	35.3	90.3%
2	55.6	48.6	60.0	1695.3	34.3	78.3	53.5	68.3	34.4	90.8%
1.5	55.7	48.8	59.9	1705.2	34.3	78.8	53.2	69.1	34.0	**90.9%**
1	55.9	49.1	59.7	1701.2	34.2	78.8	52.9	68.6	34.1	90.8%
0.7	56.1	48.9	59.9	1696.0	34.0	78.6	52.4	69.0	34.1	90.7%
0.5	56.3	48.9	60.0	1698.4	35.4	78.1	51.1	68.6	33.9	90.8%
0.3	56.5	48.9	60.3	1666.3	35.0	78.8	52.0	69.1	33.1	90.7%
0.2	56.6	48.5	60.4	1655.4	34.6	78.2	52.3	69.1	33.1	90.4%
0.1	56.6	47.3	59.3	1595.3	34.9	77.0	50.9	68.2	33.8	89.3%
0.07	55.8	48.9	59.1	1604.3	34.7	69.8	51.4	69.0	33.0	88.5%
0.05	55.0	48.5	57.6	1567.0	34.1	63.0	48.8	68.3	31.9	85.8%
0.03	53.5	48.3	57.5	1461.5	35.0	58.8	47.3	68.3	29.9	83.8%
0.01	52.7	46.9	57.2	1498.8	33.2	56.3	47.6	68.2	27.8	82.3%

Table B4: **Qwen2.5-VL-7B, full fixed-temperature ablation keeping 10% of the visual tokens.**

τ	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
100	56.2	72.9	76.7	2117.6	45.3	81.5	62.2	80.7	52.3	88.3%
50	56.4	73.4	76.9	2080.0	45.3	81.6	62.4	80.8	52.3	88.3%
10	56.1	73.9	76.9	2103.0	44.6	81.8	61.4	80.8	51.9	88.0%
5	56.4	73.5	76.7	2094.9	45.3	81.8	62.1	80.8	52.1	88.3%
2	56.1	73.4	77.3	2097.6	45.4	82.2	62.2	81.5	51.8	88.5%
1.5	56.0	73.4	77.2	2097.3	45.4	82.2	61.2	81.8	52.5	88.4%
1	55.6	73.3	77.2	2092.2	45.2	82.2	60.3	81.8	53.5	88.2%
0.7	55.5	73.5	76.7	2086.7	46.2	81.9	60.4	81.8	54.5	88.5%
0.5	55.3	73.5	77.2	2096.4	46.1	81.8	59.9	81.3	55.9	88.5%
0.3	55.0	73.7	77.1	2148.4	47.2	82.0	60.9	81.8	56.4	89.3%
0.2	55.5	74.5	77.8	2115.0	46.4	81.9	60.8	82.6	56.2	**89.3%**
0.1	56.0	72.5	76.6	2113.3	47.3	82.5	59.7	81.2	55.3	88.8%
0.07	55.8	73.9	75.9	2097.8	46.7	80.9	63.3	80.7	56.7	89.1%
0.05	54.4	72.8	76.2	2025.4	46.0	74.9	62.8	79.9	56.7	87.3%
0.03	53.4	72.0	74.6	1978.1	45.9	72.7	59.7	81.0	50.4	85.0%
0.01	53.3	72.8	73.8	1998.0	45.6	69.6	57.1	81.3	53.4	84.7%

Table B5: **Qwen2.5-VL-7B, full fixed-temperature ablation keeping 5% of the visual tokens.**

τ	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
100	52.2	67.3	72.4	1986.6	43.4	77.0	58.3	77.5	39.4	81.8%
50	52.4	67.5	71.2	1965.8	42.7	77.0	57.5	78.1	39.4	81.4%
10	52.2	67.3	72.4	1986.6	43.4	77.0	58.3	77.5	39.4	81.8%
5	52.3	66.6	71.6	1980.1	43.1	77.1	59.0	78.8	39.3	81.8%
2	52.5	66.7	72.0	2003.3	43.8	77.2	58.6	77.9	39.3	82.0%
1.5	52.3	68.2	72.7	2015.3	44.0	77.0	57.4	79.5	40.4	82.5%
1	52.4	68.0	72.2	1994.3	43.6	78.0	57.1	79.3	41.5	82.4%
0.7	52.3	68.2	72.7	2015.3	44.0	77.0	57.4	79.5	40.4	82.5%
0.5	52.5	68.3	72.6	2035.4	44.8	78.1	56.2	79.7	44.0	83.3%
0.3	52.7	69.1	73.5	2019.1	44.1	77.8	55.8	80.3	45.6	83.5%
0.2	53.1	68.6	73.5	2043.6	44.3	77.7	55.8	81.0	46.3	**83.8%**
0.1	53.6	68.6	71.1	1977.6	45.7	77.8	56.3	79.7	46.1	83.5%
0.07	51.8	67.3	70.2	1914.0	45.7	70.5	57.1	78.5	46.8	81.7%
0.05	50.1	68.0	70.5	1812.2	46.1	64.9	58.7	78.9	44.5	80.4%
0.03	49.8	68.9	70.1	1832.4	44.9	64.9	54.6	78.5	41.8	79.2%
0.01	49.7	67.2	69.0	1807.6	46.7	62.9	53.2	78.9	42.3	78.7%

C Supplementary Theoretical Analysis

C.1 Greedy Approximation Guarantee and Objective Interpretation

Greedy approximation guarantee. Let S_k be the set selected after k greedy steps, with $S_0 = \varnothing$, and let S^* be an optimal subset satisfying $|S^*| \le K$. For $k = 0, \dots, K-1$, enumerate the elements of $S^* \setminus S_k$ as

$$S^* \setminus S_k = \{x_1, \dots, x_m\}, \qquad m = |S^* \setminus S_k| \le K.$$

By monotonicity, $F(S^*) \le F(S_k \cup S^*)$. Expanding the gain from adding the elements of $S^* \setminus S_k$ one by one and applying submodularity gives

$$\begin{aligned} F(S^*) - F(S_k) &\le F(S_k \cup S^*) - F(S_k) \\ &= \sum_{i=1}^{m} \Delta(x_i \mid S_k \cup \{x_1, \dots, x_{i-1}\}) \\ &\le \sum_{i=1}^{m} \Delta(x_i \mid S_k) \\ &\le K \max_{x \in V \setminus S_k} \Delta(x \mid S_k). \end{aligned} \tag{17}$$

The second inequality follows from diminishing returns, since $S_k \subseteq S_k \cup \{x_1, \dots, x_{i-1}\}$, while the last inequality uses $m \le K$.

By the greedy rule,

$$\Delta(x_{k+1} \mid S_k) = \max_{x \in V \setminus S_k} \Delta(x \mid S_k) = F(S_{k+1}) - F(S_k).$$

Substituting this identity into Equation 17 and rearranging yields

$$F(S^*) - F(S_{k+1}) \le \left(1 - \frac{1}{K}\right) \big[F(S^*) - F(S_k)\big]. \tag{18}$$

Because $F(S_0) = F(\varnothing) = 0$, iterating Equation 18 for $k = 0, \dots, K-1$ gives

$$\begin{aligned} F(S_{\text{greedy}}) = F(S_K) &\ge \left[1 - \left(1 - \frac{1}{K}\right)^K\right] F(S^*) \\ &\ge (1 - 1/e) F(S^*), \end{aligned} \tag{19}$$

which is the standard cardinality-constrained greedy guarantee (Nemhauser et al., 1978). This guarantee applies to the regularized set objective F under the specified token budget.

LSE interpretation. For a fixed text token t_j, define the exponential support of each visual token by

$$q_i = \exp(r_{ij}/\tau), \qquad v_i \in S,$$

and let

$$q_{\max} = \exp\left(\frac{r_j^{\max}(S)}{\tau}\right), \qquad r_j^{\max}(S) = \max_{v_i \in S} r_{ij}.$$

The regularized support score can then be written as

$$G_S(t_j) = \tau \log\left(\epsilon + \sum_{v_i \in S} q_i\right) - \tau \log \epsilon. \tag{20}$$

Since $q_{\max}$ is the largest term in the support sum,

$$q_{\max} \le \sum_{v_i \in S} q_i \le |S| q_{\max}.$$

Therefore,

$$\epsilon + q_{\max} \le \epsilon + \sum_{v_i \in S} q_i \le |S|(\epsilon + q_{\max}),$$

where the last inequality uses $|S| \geq 1$ and hence $\epsilon \leq |S|\epsilon$.

Because the logarithm is monotone increasing and $\tau > 0$, applying $\tau \log(\cdot) - \tau \log \epsilon$ to the preceding inequality gives

$$m_j(S) \leq G_S(t_j) \leq m_j(S) + \tau \log |S|, \tag{21}$$

where

$$m_j(S) = \tau \log\left(\epsilon + \exp\left(\frac{r_j^{\max}(S)}{\tau}\right)\right) - \tau \log \epsilon.$$

Thus, as $\tau \to 0^+$, each per-token aggregate is dominated by the strongest match within the selected set. The gap between the aggregate and the corresponding strongest-match score is bounded by $\tau \log |S|$. Consequently, a smaller selected set yields a tighter approximation to the strongest match, while other visual tokens continue to provide additional support at finite temperature.

Set-dependent selection. The marginal term $a_{xj}/(\epsilon + Z_j(S))$ measures candidate support relative to both the baseline and the support already accumulated by the selected set. A candidate that provides stronger support for undercovered text semantics can therefore obtain a larger marginal gain than another candidate with similar total relevance. This set-dependent update jointly captures relevance and semantic coverage while reducing redundant evidence, which explains why marginal gains are recomputed after every greedy selection step.

C.2 Computational Complexity

We analyze language-model prefill computation, pruning savings, and UniPrune's selection overhead. Let the language model have T Transformer layers, hidden dimension d, and FFN intermediate dimension m. With N visual and M text tokens, the full sequence length is $n = N + M$; retaining K visual tokens reduces it to $\hat{n} = K + M$.

Full-model computation. Following the estimate in DART (Wen et al., 2025), each layer mainly computes attention projections, self-attention matrix operations, and the FFN. Counting one multiply-add as one computation unit, the Q, K, V and output projections cost $4nd^2$, attention scores and value aggregation cost $2n^2d$, and an FFN with two linear maps costs $2ndm$. Full-model prefill computation is approximately

$$\mathrm{FLOPs}_{\mathrm{full}} = T\left(4nd^2 + 2n^2d + 2ndm\right). \tag{22}$$

Projection and FFN costs grow linearly with sequence length, while self-attention matrix operations grow quadratically. For a gated FFN with three linear maps, replace $2ndm$ by $3ndm$. Counting multiplication and addition separately doubles the matrix-operation terms without changing relative savings under the same convention.

Computation after single-stage pruning. UniPrune prunes once after layer L. The first L layers process the full sequence; the remaining $T - L$ layers process only retained visual tokens and all text tokens. Excluding selection, language-model prefill computation is

$$\begin{aligned}\mathrm{FLOPs}_{\mathrm{pruned}} = {} & L\left(4nd^2 + 2n^2d + 2ndm\right) \\ & + (T - L)\left(4\hat{n}d^2 + 2\hat{n}^2d + 2\hat{n}dm\right).\end{aligned} \tag{23}$$

Let $f(n) = 4nd^2 + 2n^2d + 2ndm$. The theoretical saving fraction is

$$\eta_{\mathrm{LLM}} = 1 - \frac{\mathrm{FLOPs}_{\mathrm{pruned}}}{\mathrm{FLOPs}_{\mathrm{full}}} = \frac{T - L}{T}\left(1 - \frac{f(\hat{n})}{f(n)}\right). \tag{24}$$

Expanding gives

$$\Delta\,\mathrm{FLOPs}_{\mathrm{LLM}} = (T - L)(N - K)\left[4d^2 + 2dm + 2d(N + K + 2M)\right]. \tag{25}$$

Removing the same number of tokens saves more attention computation when the original visual sequence is longer, making early pruning particularly suitable for high-resolution images and multi-frame videos.

UniPrune selection overhead. Constructing the $N \times M$ visual–text relevance matrix takes $O(NMd)$ time. Preprocessing, dynamic temperature statistics, and exponential support computation take $O(NM)$. Evaluating the first-round marginal gain $\Delta(x \mid \varnothing) = F(\{x\})$ for all N visual-token candidates requires $O(NM)$ time. Each subsequent greedy round sums M marginal-gain terms for

at most N candidates, costing $O(NM)$; updating support after selection takes only $O(M)$. At most K rounds yield

$$\mathcal{C}_{\text{select}} = O(NMd + KNM). \tag{26}$$

Selection occurs at one position, and subsequent layers directly use the shorter sequence. Denoting actual selection operations by C_{select}, the saving fraction including this cost is $1 - (\text{FLOPs}_{\text{pruned}} + C_{\text{select}})/\,\text{FLOPs}_{\text{full}}$. This connects the one-time selection cost to gains shared across subsequent layers; Appendix D.4 reports pruning and end-to-end times.

Storage complexity. Relevance and support matrices each require $O(NM)$ storage. Cumulative support, candidate scores, and selected indices require $O(M)$, $O(N)$, and $O(K)$, respectively, so additional storage is $O(NM)$. Typically $M \ll N$, and UniPrune constructs neither an $N \times N$ visual–visual similarity matrix nor explicit attention maps. This computation and storage organization remains compatible with efficient attention such as FlashAttention, combining set selection with subsequent inference acceleration.

D Supplementary Experiments and Visualizations

D.1 Pruning Position

Table D1 compares selection after the projector and after early LLM layers on LLaVA-1.5-7B, keeping 64 tokens. All positions share the relevance operator and fixed-temperature protocol of Appendix B.1, with $\tau = 1$.

Table D1: **Ablation over the pruning position.** LLaVA-1.5-7B retains 64 tokens with fixed $\tau = 1$; only the pruning position changes. Avg. is nine-task retention relative to the full model, and bold marks the best pruned result per column.

Pruning position	GQA	MMB^{CN}	MMB^{EN}	MME	MMMU	POPE	RWQA	SQA^{IMG}	VQA^{Text}	Avg.
Vanilla	62.0	55.8	64.1	1867.3	35.9	85.8	56.0	69.5	46.0	100.0%
After projector	54.3	50.3	59.0	1610.5	34.7	75.2	48.6	68.4	37.4	89.7%
After 1st LLM layer	55.8	49.0	60.2	1700.7	**35.9**	79.6	52.9	67.7	38.3	92.3%
After 2nd LLM layer	58.0	50.3	62.1	**1813.7**	35.0	81.8	**53.2**	**69.3**	**40.3**	**94.8%**
After 3rd LLM layer	**59.0**	**51.5**	**62.5**	1784.5	35.0	**82.8**	52.4	68.8	39.3	94.8%

Early cross-modal interaction improves selection. Moving selection from the projector output to after the first and second LLM layers increases average retention from 89.65% to 92.34% and 94.79%, respectively. The second layer improves on the projector by 5.14 percentage points, with POPE, RWQA, and TextVQA rising from 75.20, 48.63, and 37.37 to 81.75, 53.20, and 40.33. This supports the motivation for contextualized cross-modal representations: after initial visual–text interaction, relevance better guides selection of evidence needed for the current question.

The second layer balances representation quality and early compression. Moving to the third layer gives 94.78% average retention, essentially the same as the second layer. The second layer achieves the highest average retention while shortening the sequence earlier, making it a suitable default.

D.2 Performance Across Visual Token Budgets

We extend LLaVA-1.5-7B evaluation to five visual token budgets, from retaining half the tokens to retaining only 32, to examine whether unified set selection adapts to different compression levels. Table D2 uses the default configuration and reports average performance retention over nine benchmarks. Figure D1 visualizes the same results.

Table D2: **Average performance retention of LLaVA-1.5-7B under different visual-token budgets.** Values report averages over nine benchmarks. Parenthesized values denote retention ratios relative to the full 576 visual tokens input.

Method	288 (50%)	192 (33.3%)	128 (22.2%)	64 (11.1%)	32 (5.6%)
FastV	98.5	96.9	94.6	89.4	81.0
CoIn	98.0	97.2	96.1	93.5	89.3
MMTok	98.3	97.5	96.6	94.1	90.3
VisionDrop	98.5	98.0	97.1	93.5	89.2
UniPrune	**99.2**	**98.6**	**97.7**	**95.2**	**91.0**

Consistent gains from moderate to severe compression. UniPrune achieves the highest retention across five budgets in Table D2, outperforming the strongest competitor by 0.68, 0.53, 0.62, 1.10, and 0.70 percentage points, respectively. At a 50% pruning ratio, performance is unchanged. With 288 tokens, it retains 99.21% of full-model performance while halving tokens. At 128 tokens, it retains 97.70%. Thus, selection reduces unnecessary visual input across compression levels.

Allocation under tight budgets. When the budget decreases from 128 to 32 tokens, FastV's average retention falls by 13.59 percentage points, versus 6.70 for UniPrune. At 32 tokens, UniPrune leads FastV, CoIn, and MMTok by 10.0, 1.7, and 0.7 percentage points, respectively. As fewer slots remain, repeatedly retaining similar visual support consumes a larger budget share. UniPrune updates text support after each selection, allowing subsequent slots to supplement semantics insufficiently supported by the current set, consistent with its performance under tight budgets.

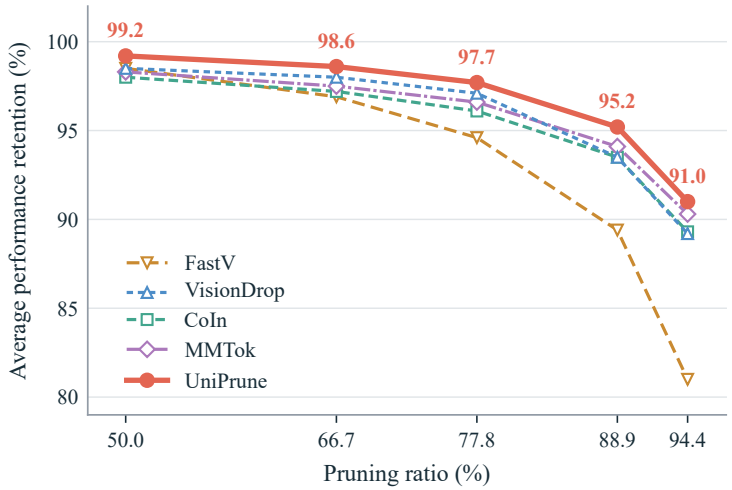


Figure D1: **Performance across visual-token pruning ratios.** Average performance retention of LLaVA-1.5-7B over nine image benchmarks, computed from the values in Table D2. The pruning ratio is defined as $(1 - K/576) \times 100\%$, where K is the number of retained visual tokens.

D.3 Fine-Grained Ability Grouping

To examine the abilities underlying the overall performance gains, we report fine-grained MMBench-EN and MVBench results at an approximately 5% visual-token retention rate. Table D3 summarizes the six official MMBench-EN L2 abilities together with their L3 subcategories, while Table D4 reports the 20 official MVBench subtasks. Figures D2 and D3 visualize the corresponding comparisons.

MMBench-EN. UniPrune is particularly strong on coarse perception, fine-grained cross-instance perception, and relation reasoning, where it achieves 73.65, 53.85, and 46.09, respectively. At the L3 level, it reaches 93.27 on image-scene understanding, 52.27 on attribute comparison, and 60.47 on social relations. Its overall accuracy reaches 60.05, exceeding the strongest competitor by 0.94 points. Other pruning strategies exhibit different strengths: HoloV performs best on fine-grained instance perception, MMTok leads attribute and logic reasoning, while FastV, PruneSID, and CoIn are strong on specific L3 categories. These results indicate that UniPrune effectively preserves complementary visual evidence for global and relational semantics.

MVBench. UniPrune is particularly effective on action and object reasoning, achieving the best results on action antonym, action count, episodic reasoning, fine-grained pose, and object existence, while tying for the best fine-grained action performance. It reaches 56.50 on action count, 79.50 on object existence, and 81.50 on action antonym, exceeding the strongest competitor by 22.50, 15.00, and 4.50 points, respectively. UniPrune also obtains the second-highest average MVBench accuracy under an approximately 5% visual-token budget. Other methods show different strengths: CoIn leads localization and several motion-centric tasks, FastV performs best on character order and navigation, and MMTok and VisionDrop lead selected event and object-centric tasks. These results indicate that UniPrune effectively preserves complementary evidence for action and object reasoning.

Taken together, these results suggest that UniPrune is most effective when the task requires integrating complementary visual evidence rather than relying on a single highly relevant token.

Table D3: **Fine-grained MMBench-EN results at 5% visual-token retention on LLaVA-1.5-7B.** Pruned methods retain 32 of 576 tokens. Scores are accuracies (%) on the 4,329-question development set. Bold marks the best pruned result; ties are all bolded. Overall pools all questions; bold group rows are official L-2 abilities and indented rows are L-3 categories. VDrop and VLM-P abbreviate VisionDrop and VLM-Pruner.

Ability or category	Full	FastV	DART	HoloV	VDrop	VLM-P	CoIn	MMTok	Nüwa	PruneSID	UniPrune
Overall	64.09	54.30	58.16	58.59	59.11	57.73	57.65	58.85	58.51	56.79	**60.05**
Coarse perception	77.36	66.55	68.24	67.23	69.93	70.95	68.24	69.93	67.57	67.57	**73.65**
Image scene	96.15	83.65	87.50	89.42	89.42	90.38	86.54	90.38	88.46	86.54	**93.27**
Image style	77.36	**66.04**	56.60	56.60	60.38	56.60	64.15	**66.04**	62.26	**66.04**	62.26
Image emotion	78.00	62.00	64.00	70.00	76.00	78.00	80.00	78.00	72.00	70.00	76.00
Image quality	35.85	32.08	**39.62**	22.64	30.19	37.74	20.75	20.75	18.87	26.42	**39.62**
Image topic	83.33	75.00	77.78	**80.56**	77.78	75.00	75.00	77.78	**80.56**	72.22	**80.56**
Fine-grained perception (instance)	65.87	53.58	60.75	**64.16**	62.12	59.39	59.39	60.07	62.80	58.02	61.77
Celebrity recognition	79.80	64.65	75.76	**80.81**	**80.81**	72.73	68.69	72.73	76.77	72.73	76.77
Object localization	39.51	**38.27**	34.57	32.10	29.63	32.10	32.10	28.40	35.80	30.86	33.33
Attribute recognition	79.73	56.76	70.27	**78.38**	72.97	71.62	77.03	77.03	75.68	66.22	74.32
OCR	58.97	51.28	58.97	**61.54**	**61.54**	58.97	58.97	**61.54**	58.97	**61.54**	58.97
Fine-grained perception (cross-instance)	55.24	48.25	51.75	49.65	48.25	51.75	50.35	49.65	51.75	50.35	**53.85**
Action recognition	90.74	75.93	**85.19**	81.48	79.63	**85.19**	83.33	81.48	**85.19**	79.63	**85.19**
Spatial relationship	17.78	15.56	17.78	20.00	22.22	17.78	22.22	20.00	15.56	**24.44**	17.78
Attribute comparison	50.00	47.73	45.45	40.91	36.36	45.45	38.64	40.91	47.73	40.91	**52.27**
Attribute reasoning	70.35	62.81	69.35	70.85	70.85	67.84	70.85	**72.36**	70.35	69.35	68.84
Function reasoning	75.95	64.56	70.89	77.22	78.48	70.89	77.22	**79.75**	72.15	73.42	70.89
Physical-property reasoning	50.67	49.33	**52.00**	49.33	46.67	48.00	50.67	**52.00**	**52.00**	50.67	49.33
Identity reasoning	93.33	82.22	95.56	95.56	**97.78**	95.56	93.33	93.33	**97.78**	93.33	**97.78**
Relation reasoning	57.39	42.61	44.35	40.00	44.35	41.74	41.74	41.74	40.87	39.13	**46.09**
Nature relation	41.67	**41.67**	37.50	35.42	35.42	39.58	35.42	35.42	33.33	37.50	35.42
Social relation	88.37	46.51	53.49	46.51	53.49	48.84	51.16	53.49	51.16	46.51	**60.47**
Physical relation	33.33	37.50	41.67	37.50	**45.83**	33.33	37.50	33.33	37.50	29.17	41.67
Logic reasoning	33.05	29.66	28.81	31.36	32.20	26.27	28.81	**33.05**	30.51	30.51	27.97
Structured image–text understanding	26.92	**29.49**	26.92	26.92	28.21	24.36	25.64	26.92	28.21	26.92	25.64
Future prediction	45.00	30.00	32.50	40.00	40.00	30.00	35.00	**45.00**	35.00	37.50	32.50

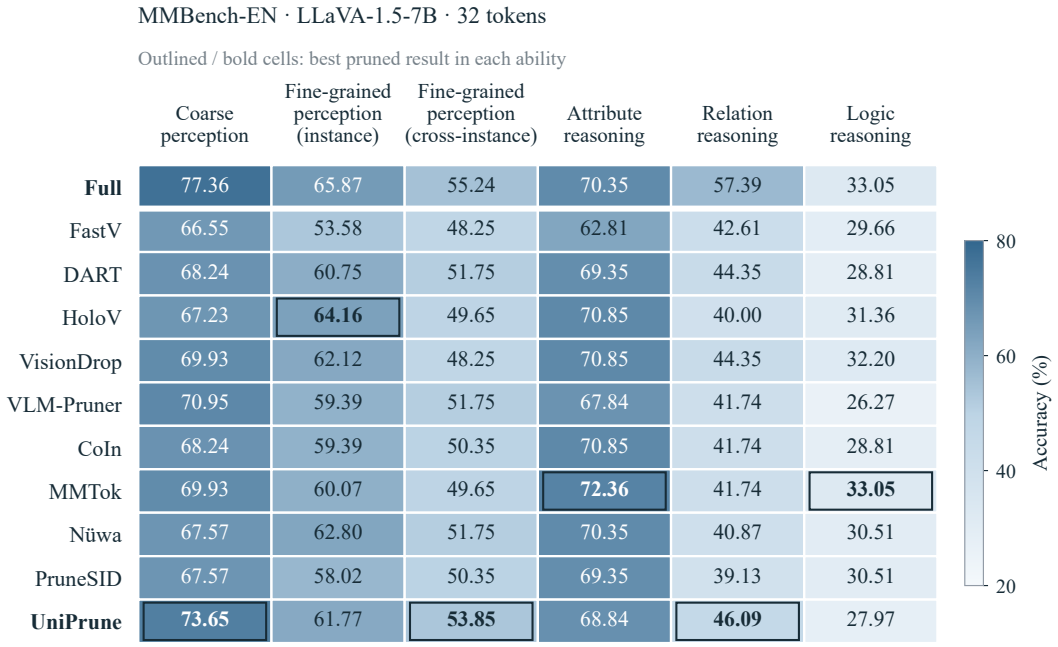


Figure D2: **MMBench-EN ability profiles.** The six L2 abilities from Table D3 are evaluated on LLaVA-1.5-7B. Pruned methods retain 32 of 576 visual tokens. Rows identify methods and columns identify abilities, and each cell reports accuracy (%) on a shared color scale. Outlined bold cells mark the best pruned result in each ability, including ties. The first row gives the full-model reference.

Table D4: **Fine-grained MVBench results at 5% visual-token retention on InternVL3-8B.** Entries are accuracy percentages. MVBench average is the unweighted mean of the 20 official subtasks. Bold marks the best pruned result; ties are all bolded.

Task	Full	FastV	VisionDrop	CoIn	MMTok	UniPrune
MVBench average	71.42	53.12	44.42	**60.12**	54.65	59.42
Action antonym	90.50	67.50	48.50	77.00	74.00	**81.50**
Action count	56.00	33.50	33.00	34.00	32.50	**56.50**
Action localization	58.00	41.50	39.00	**46.00**	45.00	41.50
Action prediction	62.50	52.50	31.00	**54.00**	51.50	52.00
Action sequence	83.00	**69.50**	41.00	**69.50**	61.00	68.50
Character order	75.50	**69.00**	51.50	59.00	63.00	66.00
Counterfactual inference	80.50	36.00	35.50	**61.50**	43.50	54.00
Egocentric navigation	35.50	**39.00**	35.50	35.00	38.00	31.00
Episodic reasoning	59.50	58.00	55.50	56.50	59.00	**61.50**
Fine-grained action	44.00	37.50	36.00	**40.50**	**40.50**	**40.50**
Fine-grained pose	80.50	71.00	39.00	68.00	66.50	**72.50**
Moving attribute	92.50	46.00	37.00	**82.50**	62.50	73.00
Moving count	63.50	37.00	33.00	**57.00**	44.00	45.50
Moving direction	87.50	28.00	29.50	**65.50**	40.00	47.00
Object existence	96.00	44.50	41.00	64.50	49.50	**79.50**
Object interaction	84.50	69.00	44.50	**71.00**	60.50	63.00
Object shuffle	36.00	39.00	**43.00**	38.50	38.00	37.50
Scene transition	93.00	86.00	89.00	86.50	**90.50**	87.00
State change	72.50	**65.00**	55.00	62.00	56.00	57.00
Unexpected action	77.50	73.00	71.00	74.00	**77.50**	73.50

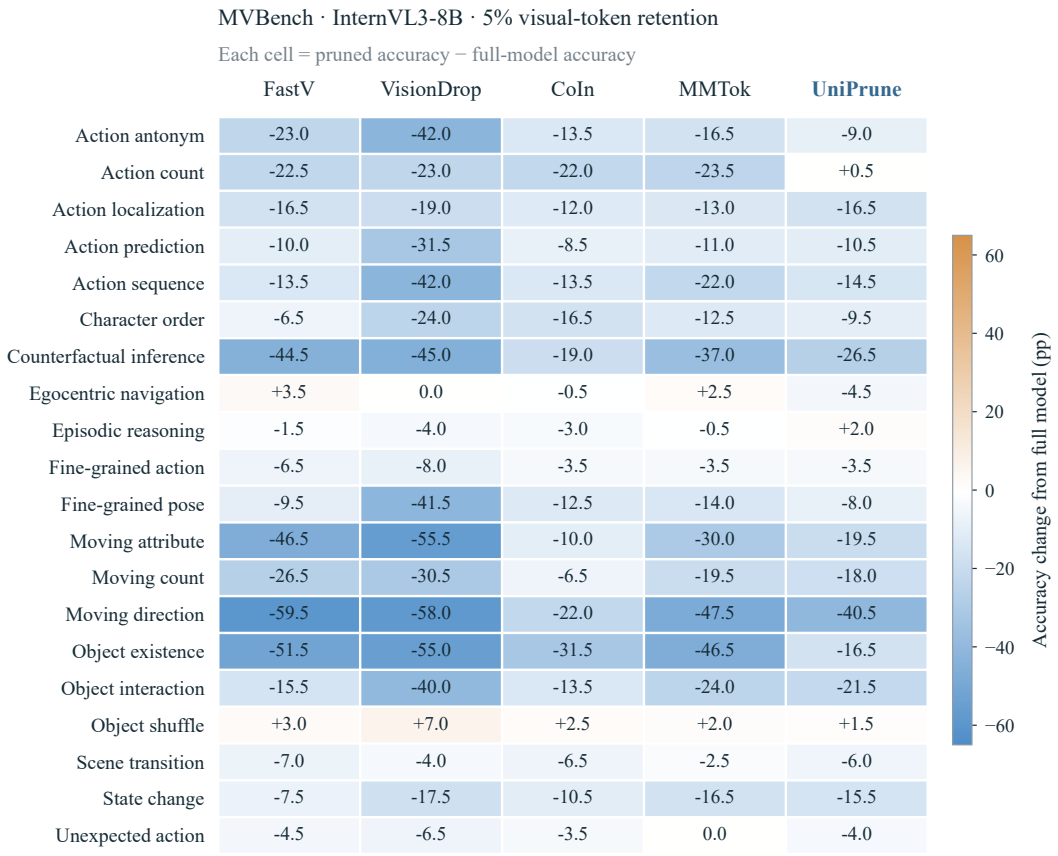


Figure D3: **Changes in MVBench subtask accuracy.** Each cell subtracts the full-model accuracy from the pruned accuracy in Table D4, in percentage points, on InternVL3-8B at 5% visual-token retention. The shared color scale is centered at zero: blue denotes a decrease and orange an increase.

D.4 Complete Efficiency Comparison

Table D5 supplements Table 6 with complete efficiency results, examining how savings from a shorter sequence translate into end-to-end acceleration after accounting for token-selection overhead and shared processing. For set-selection methods, both the pruned model's computation time and the additional time required to obtain the subset must be considered.

Table D5: **Complete inference-efficiency comparison on LLaVA-NeXT-7B (POPE).** Times are in milliseconds; GPU memory is in GiB.

Method	E2E	Prefill net	Other (pre)	Other (post)	Effective pruning	GPU memory (GiB)
Vanilla (unpruned)	218.4	138.5	53.9	26.0	0.0	14.947
FastV	106.5	37.3	50.8	17.1	1.3	13.665
SparseVLM	110.6	40.0	50.8	17.2	2.6	13.623
DART	105.1	33.5	50.2	16.3	5.1	13.744
HoloV	99.2	26.5	50.9	16.2	5.6	14.595
VisionDrop	116.4	33.7	50.7	16.5	15.5	14.597
VLM-Pruner	127.2	33.2	50.8	16.8	26.4	13.900
CoIn	198.2	26.0	51.0	16.7	104.5	13.416
Zoo-Prune	161.9	31.3	51.4	16.4	62.8	18.244
VisionTrim	103.7	26.5	51.1	16.3	9.8	14.595
MMTok	130.3	26.3	51.3	16.7	36.0	13.459
Nüwa	111.6	35.5	52.7	17.3	6.1	14.598
PruneSID	102.2	26.7	50.9	16.8	7.8	14.595
UniPrune	**124.1**	**33.0**	**50.7**	**16.8**	**23.6**	**13.578**

Set-selection overhead and practical acceleration. UniPrune reduces prefill time (net prefill plus pruning overhead) from 138.505 ms to 56.556 ms and end-to-end time from 218.436 ms to 124.115 ms, yielding $2.45\times$ and $1.76\times$ speedups, respectively. Compared with CoIn and MMTok, its effective pruning time falls by approximately 77.4% and 34.2%, and end-to-end time by approximately 37.4% and 4.8%. Although CoIn and MMTok have slightly shorter net prefill times, their additional selection overhead increases total latency. The cost of the set objective is therefore an important part of the performance–efficiency tradeoff. Incremental selection under a unified objective allows UniPrune's set-dependent decisions to translate into practical inference gains.

Shared costs and memory savings. Preprocessing times for pruned methods are concentrated around 50–53 ms. Visual sequence compression mainly accelerates subsequent language-model computation, while shared preprocessing remains a relatively stable cost. These stages jointly explain why end-to-end speedup is lower than net prefill speedup. UniPrune reduces GPU memory from 14.947 GiB to 13.578 GiB, approximately 9.2%, reducing storage requirements alongside inference time. Together with the performance results, this shows that UniPrune preserves task evidence at a controlled selection cost and balances accuracy, end-to-end latency, and GPU memory.

D.5 Additional Qualitative Comparisons

Figures D4 and D5 extend Figure 5 through cross-task comparisons and changes in the question condition. Figure D4 compares methods on four question types and two budgets to examine the correspondence between retained regions and task evidence. Figure D5 holds the image and budget fixed and changes only the question to examine responses to text.

Understanding evidence needs through region distributions. Color recognition and object-state judgment focus on local attributes of the target; material/object relations require the target and related context; scene recognition needs global semantics from multiple regions. The key comparison is whether selected regions jointly support the prompt, rather than whether they are spatially uniform or concentrated. Placing 64- and 128-token results side by side helps reveal whether additional budget supplements task evidence. UniPrune's gains are measured relative to existing support, directly explaining budget allocation from key to complementary regions.

Conditional selection for different questions about the same image. With image and budget fixed, a different question changes the relevance matrix and each text token's support requirements. The same visual token can therefore have different marginal value for different questions. Figure D5 pairs retained regions with model answers, allowing direct comparison of required evidence, selection, and answers. This qualitatively examines the central motivation for text-conditioned set selection.

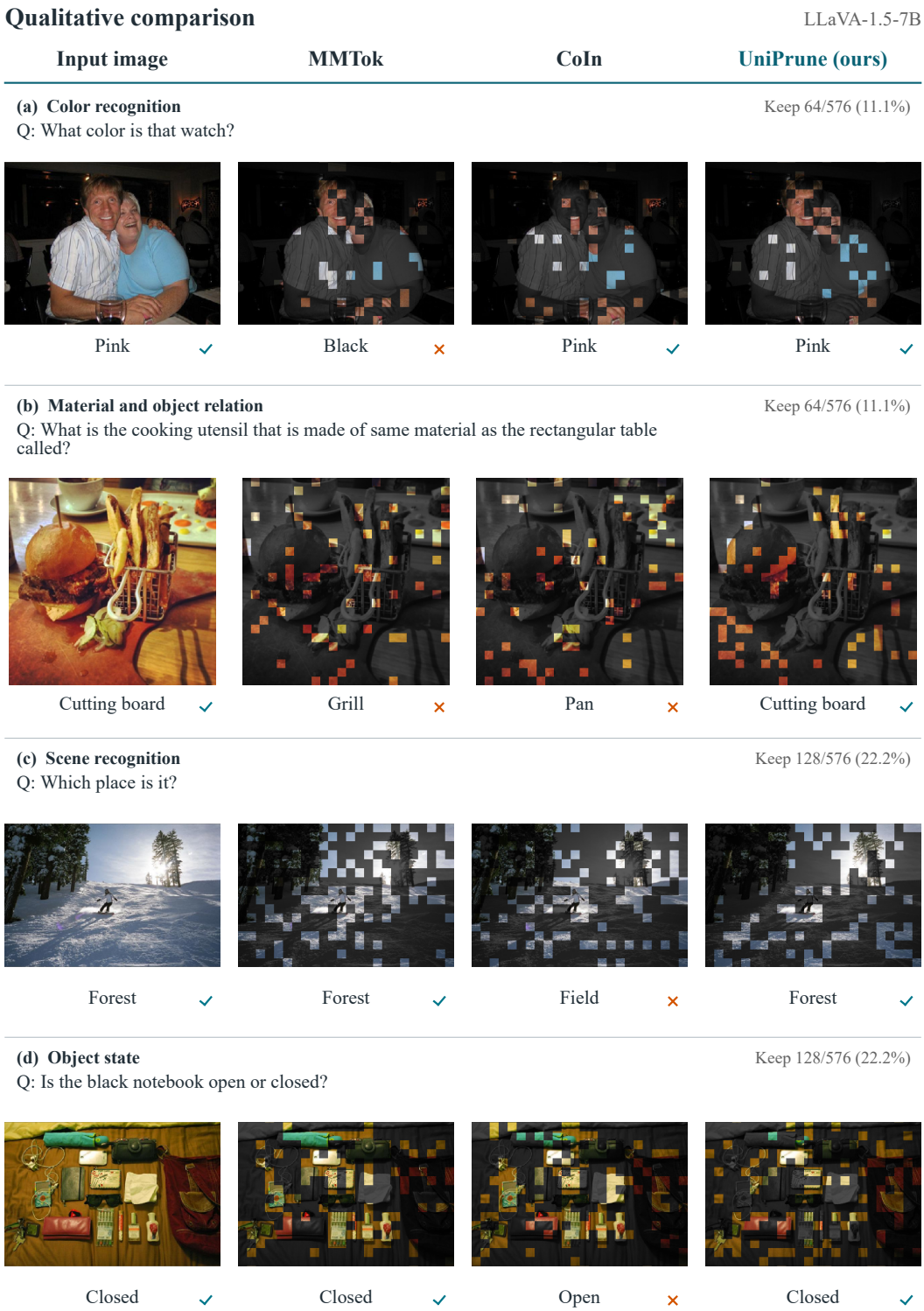


Figure D4: **Additional qualitative comparisons across images and tasks.** LLaVA-1.5-7B examples compare MMTok, CoIn, and UniPrune while retaining 64 or 128 of the original 576 visual tokens. The examples cover color recognition, material/object relations, scene recognition, and object state. Checkmarks and crosses denote correct and incorrect answers.

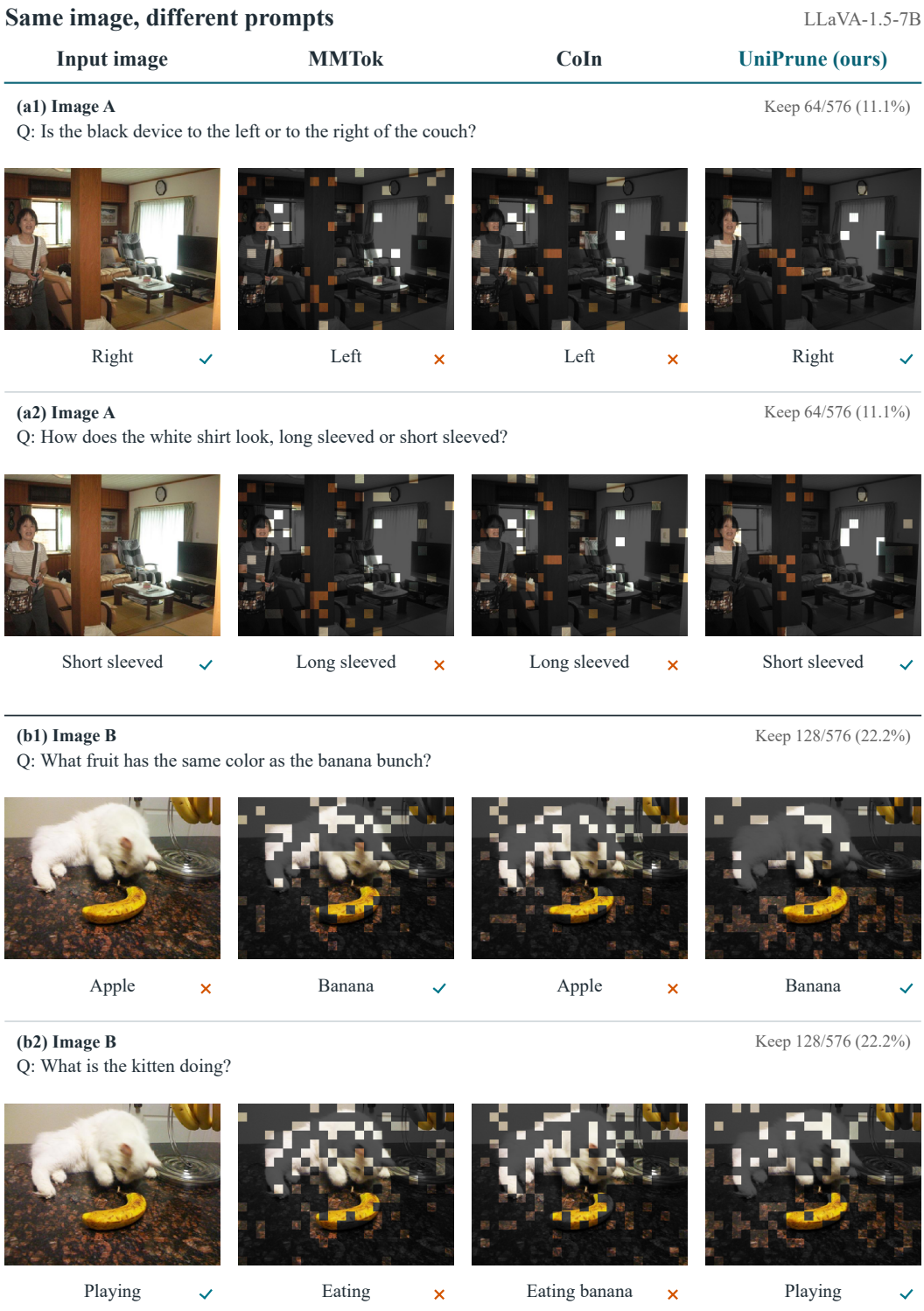


Figure D5: **Same image under different questions.** For each of two LLaVA-1.5-7B images, the question changes while the image and retention budget remain fixed. The retained regions and answers from MMTok, CoIn, and UniPrune show how selection responds to prompt-specific evidence. Checkmarks and crosses denote correct and incorrect answers.

D.6 Further Comparison of Relevance Operators

Table D6 supplements Table 5 by comparing alternative relevance-score representations and composite variants. The first group changes distances, kernels, or score transformations, whereas the second incorporates coverage, redundancy, or spatial information. All variants use LLaVA-1.5-7B, fixed $\tau = 1$, and the same nine-task protocol; preprocessing is shared unless explicitly stated otherwise. Figure D6 reports each result relative to the default cosine configuration under the same token budget. For the spatial-barycenter variant, $b_j = \sum_i \mathrm{softmax}_i(s_{ij})\mathrm{coord}_i$ and $\delta_{ij} = \|\mathrm{coord}_i - b_j\|_2$, where coord_i denotes the spatial coordinate of visual token i. The scale σ_j^2 is the mean of δ_{ij}^2 over visual tokens. Numerical-stability terms are omitted from the formulas.

Table D6 shows that the unified set objective remains effective across multiple distances and kernels. Although several composite variants outperform default cosine, their gains are limited to 0.06–0.25 retention percentage points. We therefore retain cosine as the default relevance operator for its simplicity. Together with the structural ablation, these results support combining a simple relevance operator with set-dependent support accumulation and marginal-gain updates.

Table D6: **Supplementary comparison of relevance operators.** Values are nine-task average retention (%). $c_{ij} = \cos(v_i, t_j)$, $d_{ij} = \|v_i - t_j\|_2$; $\bar{c}_{i\cdot}, \bar{c}_{\cdot j}, \bar{c}$ are row, column, and overall means; μ_v, μ_t denote mean visual and text token features. Except the first two variants, scores are row-normalized across text tokens. Cauchy σ^2 and Laplacian σ use median squared L2 and L1 distances.

Operator	Core formula	64 tokens	32 tokens
Basic operators and preprocessing variants			
Row-centered cosine	$c_{ij} - \bar{c}_{i\cdot}$	93.16	88.06
Double-centered cosine	$c_{ij} - \bar{c}_{i\cdot} - \bar{c}_{\cdot j} + \bar{c}$	93.34	88.16
Negative L1 distance	$-\|v_i - t_j\|_1$	92.15	88.51
Cauchy kernel	$(1 + d_{ij}^2/\sigma^2)^{-1}$	94.28	89.16
Laplacian kernel	$\exp(-\|v_i - t_j\|_1/\sigma)$	91.86	88.14
Pearson correlation	$\cos(v_i - \mu_v, t_j - \mu_t)$	91.36	86.07
Cosine (default)	c_{ij}	94.79	90.80
Composite variants			
Text coverage	$c_{ij}\ \mathrm{softmax}_j(c_{ij})$	95.02	90.76
Energy-normalized distance	$s_{ij} = -d_{ij}^2/(\|v_i\|_2^2 + \|t_j\|_2^2)$	94.86	90.95
Spatial barycenter	$s_{ij}/(1 + \delta_{ij}^2/\sigma_j^2)$	94.85	91.05
Negative redundancy	$c_{ij} - \max_k \cos(v_i, v_k)$	95.02	90.87

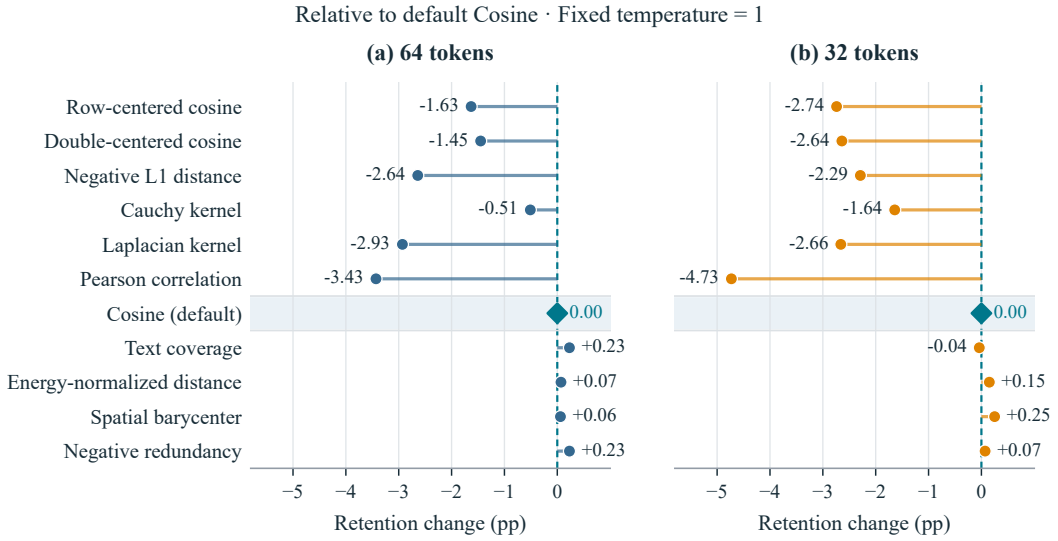


Figure D6: **Sensitivity to the relevance operator.** Nine-task average retention relative to the default Cosine operator, computed from Table D6, in percentage points. The two panels share an axis scale; zero denotes default Cosine at the corresponding budget (94.79% for 64 tokens and 90.80% for 32 tokens). All configurations use LLaVA-1.5-7B with fixed $\tau = 1$.

E STATEMENT ON THE USE OF LARGE LANGUAGE MODELS

LLMs were used in this work only for language polishing and writing assistance. Their role was limited to improving grammar, readability, and fluency. All research ideas, experimental designs, data analyses, and technical contributions were conceived and completed by the human authors. LLMs did not contribute to generating original ideas, interpreting results, or producing scientific claims. The authors take full responsibility for the accuracy and integrity of all content in this paper.