From Saliency Ranking to Evidence Preservation: Visual Token Condensation for Efficient Multimodal Reasoning

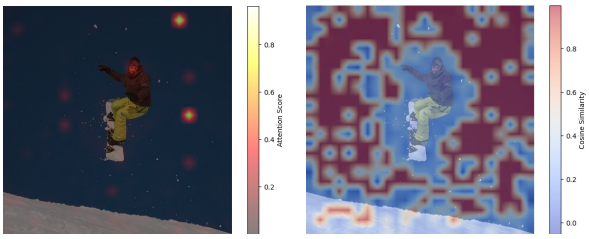
Anonymous ACL submission

Abstract

Visual tokens dominate the inference cost of multimodal large language models (MLLMs), but aggressive token reduction often degrades reasoning. We argue that the fragility comes from a *saliency-ranking view*, *i.e.*, existing training-free methods typically rank tokens by a scalar proxy, such as cross-attention, [CLS] attention, or feature similarity, and then prune or merge tokens independently. However, visual evidence is set-valued and structured: salient patches can be redundant, low-saliency tokens can be answer-critical, and feature-similar patches can be unsafe to merge when they are spatially incompatible. We propose CondenseVLM, a simple framework that formulates visual token compression as a budgeted *evidence-preservation* question. The formulation jointly optimizes salient evidence coverage, contextual coverage, and spatially constrained merge fidelity. It yields a monotone submodular coverage term, a marginal-gain interpretation of diversity-aware anchor selection, and a local distortion bound for spatial-semantic merging. Comprehensive experiments across six MLLM families and diverse general-purpose, grounding, text-dense, and video benchmarks demonstrate that CondenseVLM improves the accuracy-efficiency trade-off, and achieves the SOTA performance.

1 Introduction

Multimodal large language models (MLLMs) rely on long visual-token prefixes to connect image or video encoders with language decoders. This design enables general visual question answering, document understanding, grounding, and multimodal reasoning (Liu et al., 2024c,a,b; Li et al., 2024a; Bai et al., 2025), but it also makes inference cost highly sensitive to the number of visual tokens. A 336-resolution LLaVA-1.5 image produces 576 visual tokens, while high-resolution and video variants can expose the decoder to thousands of tokens. Because these visual tokens participate in decoder attention, reducing them directly lowers memory usage, prefill cost, and generation latency (Dao et al., 2022; Dao, 2024; Tan et al., 2026).



(a) [CLS] saliency (b) feature similarity

Figure 1: Why token ranking is insufficient. Saliency concentrates on a few regions, while feature similarity can connect distant patches. Compression must preserve an evidence set, not only high-scoring individual tokens.

Most training-free accelerators therefore start from a natural primitive: assign each visual token an importance score and keep the top-ranked tokens, or merge tokens using feature similarity. Attention-based methods use text–visual or layer-wise attention (Chen et al., 2024a; Zhang et al., 2024b); vision-centric methods use encoder-side saliency or clustering (Zhang et al., 2024a; Yang et al., 2024a; Zou et al., 2025b); and merging methods fuse visually similar tokens (Bolya et al., 2022; Shang et al., 2024). These approaches are attractive because they require no retraining. However, the primitive itself is incomplete: the frozen decoder does not consume a ranked list of independent patches; it consumes a *set* of visual evidence that must remain sufficient after compression.

The difference becomes critical under aggressive budgets. A good compressed sequence must satisfy three requirements. First, it must preserve *non-redundant salient evidence*: several high-attention patches around the same object may carry nearly identical information, so retaining all of them wastes budget. Second, it must preserve *contextual evidence*: background layout, object relations, and text surroundings can disambiguate an answer even

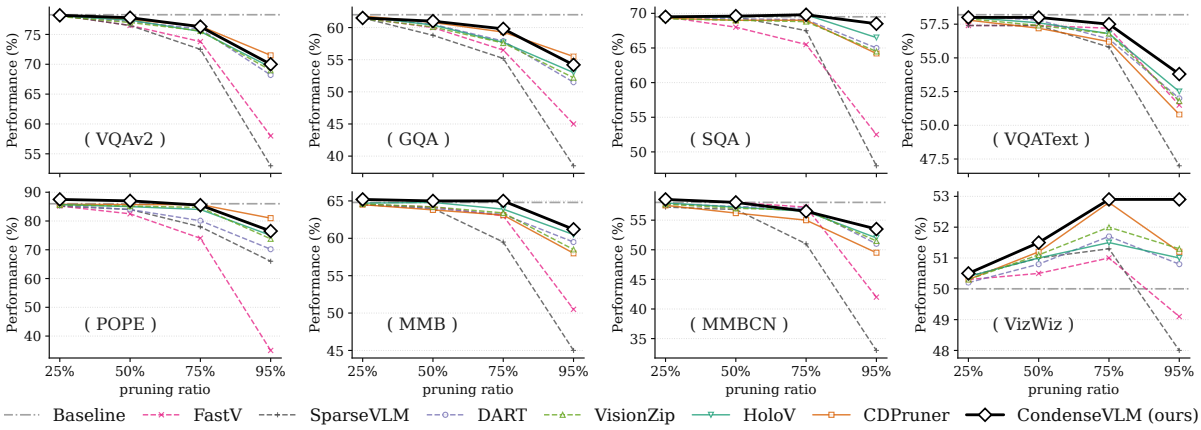


Figure 2: Performance trends under aggressive visual-token pruning. As the pruning ratio increases, CondenseVLM maintains a more stable accuracy curve than prior pruning and merging baselines, supporting our claim that visual-token compression should preserve structured evidence rather than rely on single-proxy token ranking or merging.

when no single patch receives high attention. Third, it must preserve *reconstructability*: if discarded tokens are merged into anchors, feature similarity alone may fuse spatially distant regions and corrupt local structure. Figure 1 illustrates the two observable symptoms behind these failures, and Figure 2 shows their empirical consequence: single-proxy methods degrade faster as the budget tightens.

We propose CondenseVLM, a visual token condensation framework derived from these three requirements. Instead of asking which individual tokens are most important, CondenseVLM constructs a compact visual evidence coreset. It selects faithful anchors according to marginal evidence coverage, reserves a small set of globally representative context anchors, and merges residual tokens with a spatial-semantic distortion criterion. The method is plug-and-play: it can be inserted before the LLM decoder of frozen MLLMs without changing model weights, prompts, or decoding hyperparameters.

The main contribution of this paper is to turn a practical compression recipe into a unified evidence-preservation principle. We first diagnose why token-level importance is not equivalent to evidence preservation. We then formulate condensation as a budgeted coreset objective whose terms correspond to salient coverage, contextual coverage, and local reconstruction fidelity. This objective provides a theoretical bridge to the algorithm: the coverage term is monotone submodular, diversity-aware anchor selection approximates marginal gains, and spatial-semantic merging minimizes a local perturbation bound. Finally, we validate the results across diverse models and benchmarks, larger backbones, showing robust accuracy-efficiency trade-offs under different pruning ratios.

2 Why Token Importance Is Not Evidence Preservation

The central limitation of scalar token ranking is that it treats compression as an independent decision for each patch. Evidence preservation is different: the utility of a token depends on what has already been retained and on whether discarded tokens can be safely represented by the retained set. We organize this mismatch into three desiderata that will later map directly to the objective in Eq. (2).

D1: marginal rather than absolute utility. A token should be retained when it contributes evidence that is not already covered by the selected anchors. High-attention visual tokens often cluster around a single object or textual region, so their absolute scores can be large while their incremental value is small. This is a diminishing-returns effect: once a region is represented, another near-duplicate patch should receive a lower marginal gain. Eq. (4) makes this set-dependent gain explicit.

D2: context as a separate evidence channel. Low-saliency does not imply irrelevance, as layout, scene type, position, and surrounding text can all determine how the decoder interprets foreground evidence. If context is forced to compete directly with foreground saliency, it is systematically removed under small budgets. A principled compressor should therefore reserve a lightweight channel for representative low-saliency context rather than treating all background tokens as discardable noise.

D3: merging as local reconstruction. When residual tokens are absorbed into retained anchors, the assignment must preserve the local geometry of the visual feature map. Feature similarity alone can

connect repeated textures or similar object parts across distant regions; such merging can blur spatial relations that grounding and text-rich reasoning depend on. A safe assignment should be both semantically close and spatially compatible.

These desiderata explain why aggressive pruning exposes the weakness of single-proxy methods: when K is large, redundant or misplaced tokens may be tolerated; when K is small, every retained token must either add new evidence, represent context, or reconstruct nearby residual information. Motivated by this, we turn D1-D3 into a compact coreset objective and derive a new objective.

3 Condensation as Budgeted Evidence Preservation

Given visual tokens $Z = \{z_i\}_{i=1}^n \subset \mathbb{R}^d$ and a retained budget K, CondenseVLM outputs a compressed sequence $\tilde{Z}_T = \{\tilde{z}_j : j \in T\}$ with $|T| = K$. The retained set is decomposed into faithful anchors T_a for task-relevant evidence and context anchors T_b for globally representative low-saliency context. Each discarded token $i \notin T$ is assigned to an anchor $\pi(i) \in T$ and reconstructed through local aggregation. Thus, visual compression is treated as coreset construction: select a small support set and use it to preserve the evidence carried by the full visual sequence.

3.1 A Coreset Objective for Visual Evidence

Let $V = \{1, \ldots, n\}$, let $\kappa_{ij} = (1 + \cos(z_i, z_j))/2 \in [0, 1]$ be normalized semantic similarity, and let $\bar{d}_{ij} \in [0, 1]$ be normalized spatial distance on the visual-token grid. Let $a_i \geq 0$ denote task saliency and $g_i \geq 0$ denote context weight, where g_i is concentrated on low-saliency but globally representative candidates. For any nonnegative weights w, define representative coverage as

$$F_w(T) = \sum_{i \in V} w_i \max_{j \in T} \kappa_{ij}. \tag{1}$$

This facility-location objective (Krause and Golovin, 2014) measures whether each weighted token is represented by at least one retained anchor. The evidence-preservation objective is

$$\max_{T_a, T_b, \pi} \quad F_a(T_a) + \eta F_g(T_b) - \rho \sum_{i \notin T_a \cup T_b} D(i, \pi(i)), \tag{2}$$

subject to $T = T_a \cup T_b$, $|T_a| + |T_b| = K$, and $\pi(i) \in T$, with

$$D(i, j) = 1 - \kappa_{ij} + \beta \bar{d}_{ij}. \tag{3}$$

The three terms correspond exactly to D1–D3: F_a rewards non-redundant salient evidence coverage, F_g reserves contextual coverage, and D penalizes semantically or spatially unsafe merging.

The key property of Eq. (1) is diminishing returns. For a selected set S and a candidate x, the exact marginal gain is

$$\begin{aligned} \Delta_w(x \mid S) &= F_w(S \cup \{x\}) - F_w(S) \\ &= \sum_{i \in V} w_i \big[\kappa_{ix} - \max_{j \in S} \kappa_{ij}\big]_+. \end{aligned} \tag{4}$$

Top-K saliency replaces this set-dependent quantity by a fixed score. It therefore cannot distinguish a genuinely new evidence carrier from a redundant patch around an already covered region.

Assumption 1 (Local semantic smoothness)**.** Let $\Delta(i, j)$ be the downstream perturbation caused by absorbing token i into anchor j in a frozen MLLM. There exist constants $L_z, L_c \geq 0$ such that

$$\Delta(i, j) \leq L_z(1 - \kappa_{ij}) + L_c \bar{d}_{ij}. \tag{5}$$

This assumption is deliberately local. It does not claim that the full nonlinear MLLM is globally Lipschitz-equivalent to cosine similarity and spatial distance. It only formalizes the weak condition needed for safe merging: near the original visual features, replacing a token is less harmful when the anchor is semantically similar and spatially nearby.

Proposition 1 (Evidence-preserving condensation)**.** *For nonnegative weights $w_i \geq 0$, F_w is normalized, monotone, and submodular; exact greedy maximization under a cardinality budget gives the standard $(1 - 1/e)$ coverage approximation. Under Assumption 1 and $L_z > 0$, assigning a discarded token by*

$$\pi(i) = \arg\max_{j \in T} \{\kappa_{ij} - \beta \bar{d}_{ij}\} \tag{6}$$

minimizes the per-token perturbation upper bound when $\beta = L_c/L_z$. Finally, the aggregation in Eq. (9) is the weighted least-squares reconstruction of each visual anchor cluster.

Appendix A gives the detailed proof. Proposition 1 does not assert that a training-free compressor globally solves the full MLLM computation. Instead, it establishes the logic, *i.e.*, coverage gives a principled diversity objective, context anchors instantiate a separate coverage channel, and spatial-semantic assignment controls local merge distortion.

3.2 Deriving the Training-Free Algorithm

Faithful anchors from marginal coverage. Exact greedy maximization of Eq. (1) requires recomputing Eq. (4) for every candidate at every step. We use the following MMR-style surrogate (Carbonell and Goldstein, 1998):

$$s_a(x \mid T_a) = a_x - \lambda \max_{j \in T_a} \kappa_{xj}. \tag{7}$$

The saliency term estimates direct task evidence, while the similarity penalty approximates the amount of evidence already represented by selected anchors. This turns selection from "keep the largest scores" into "add uncovered evidence."

Context anchors as a second coverage channel. From low-saliency candidates, CondenseVLM selects a small number of context anchors using

$$s_g(x) = \frac{1}{n-1} \sum_{i \neq x} \kappa_{xi}. \tag{8}$$

This one-step score approximates contextual coverage: a high-scoring context token is representative of many other low-saliency tokens. Keeping this budget separate is important because foreground saliency otherwise dominates the selection process.

Spatial-semantic merging from distortion control. For each discarded token, CondenseVLM assigns it by Eq. (6). This is the optimizer of the upper bound in Eq. (5) after normalizing $\beta \approx L_c/L_z$. The spatial penalty is therefore not an arbitrary add-on; it is the term that prevents feature-similar but spatially incompatible regions from being fused.

Aggregation as reconstruction. Given an anchor j and its assigned residual cluster $C_j = \{i : \pi(i) = j\}$, CondenseVLM outputs

$$\tilde{z}_j = \frac{z_j + \sum_{i \in C_j} \omega_i z_i}{1 + \sum_{i \in C_j} \omega_i}, \quad \omega_i = \sigma(a_i). \tag{9}$$

Equivalently, $\tilde{z}_j$ minimizes $\|u - z_j\|_2^2 + \sum_{i \in C_j} \omega_i \|u - z_i\|_2^2$. Merging is thus a lightweight reconstruction of residual information, not a hard deletion of all non-anchor visual tokens.

Algorithm and complexity. For each input, CondenseVLM computes saliency and feature similarities, selects $|T_a|$ faithful anchors by Eq. (7), selects $|T_b|$ context anchors by Eq. (8), assigns residual tokens by Eq. (6), and aggregates by Eq. (9). The dominant preprocessing cost is one $O(n^2 d)$ similarity matrix, paid once before generation; the decoder then attends to only K visual tokens. This trade-off is favorable when visual prefixes are long, generation has multiple decoding steps, or memory is the bottleneck. More details are in Appendix F.

4 Experiments

Experimental setup. The section is organized around five questions: (i) whether evidence preservation remains reliable under aggressive budgets, (ii) whether the same design transfers across model families, (iii) whether it preserves fine-grained spatial and text evidence, (iv) whether each objective component is necessary, and (v) whether the accuracy gains translate into practical latency and memory savings. The evaluation covers six representative backbones: LLaVA-1.5, LLaVA-NeXT, Qwen2.5-VL, InternVL2, LLaVA-OneVision, and VideoLLaVA. The benchmark suite covers general visual question answering and reasoning (GQA, MMBench, MMBench-CN, MME, POPE, ScienceQA, VQAv2, TextVQA, VizWiz, and MM-Vet), grounding-oriented RefCOCO variants, text-dense OCR/chart tasks, and video question answering. Unless otherwise stated, we report performance preservation relative to the corresponding uncompressed model, whose score is normalized, which allows comparisons across heterogeneous metrics. We compare against attention-based pruning, saliency-based pruning, token merging, and recent diversity- or instruction-aware pruners under the same retained-token budget for each backbone.

4.1 Main Results

The main question is whether the compressed visual sequence can preserve enough evidence when the token budget is severely limited. Table 1 reports the full LLaVA-1.5-7B results across nine benchmarks and three retained-token budgets. CondenseVLM achieves the best average preservation at all budgets: 99.5% with 192 tokens, 98.8% with 128 tokens, and 97.0% with only 64 tokens. Compared with the strongest prior result in each budget, the gains are +0.3, +0.7, and +0.6 points, respectively. These gains are most important at 128 and 64 tokens, where scalar-ranking methods begin to discard non-dominant but still useful evidence.

The per-benchmark results in Table 1 show that CondenseVLM is not simply improving one metric. At 192 tokens, it preserves or improves performance on POPE, VQAv2, TextVQA, and VizWiz while keeping strong GQA and MMBench scores.

Table 1: Performance comparison of various methods across general benchmarks. Results are shown for different pruning ratios, with accuracy and average performance highlighted. CondenseVLM consistently achieves SOTA.

Methods	GQA	MMB	MMB_{CN}	MME	POPE	SQA	VQA_{V2}	VQA_{Text}	VizWiz	Avg.
Upper Bound, 576 Tokens	61.9	64.7	58.1	1862	85.9	69.5	78.4	58.2	50.0	100%
LLaVA-1.5-7B					*Retain 192 Tokens* (↓ **66.7%**)					
FastV (ECCV25)	52.7 -14.9%	61.2 -5.4%	57.0 -1.9%	1612 -13.4%	64.8 -24.6%	67.3 -3.2%	67.1 -14.4%	52.5 -9.8%	50.8 +1.6%	90.5%
LLaVA-PruMerge (ICCV25)	54.3 -12.3%	59.6 -7.9%	52.9 -9.0%	1632 -12.4%	71.3 -17.0%	67.9 -2.3%	70.6 -9.9%	54.3 -6.7%	50.1 +0.2%	91.4%
PDrop (CVPR25)	57.1 -7.8%	63.2 -2.3%	56.8 -2.2%	1766 -5.2%	82.3 -4.2%	68.8 -1.0%	75.1 -4.2%	56.1 -3.6%	51.1 +2.2%	96.7%
FiCoCo-V (2025.03)	58.5 -5.5%	62.3 -3.7%	55.3 -4.8%	1732 -7.0%	82.5 -4.0%	67.8 -2.4%	74.4 -5.1%	55.7 -4.3%	51.0 +2.0%	96.1%
MustDrop (2024.11)	58.2 -6.0%	62.3 -3.7%	55.8 -4.0%	1787 -4.0%	82.6 -3.8%	69.2 -0.4%	76.0 -3.1%	56.5 -2.9%	51.4 +2.8%	97.2%
HiRED (AAAI25)	58.7 -5.2%	62.8 -2.9%	54.7 -5.9%	1737 -6.7%	82.8 -3.6%	68.4 -1.6%	74.9 -4.5%	47.4 -18.6%	50.1 +0.2%	94.6%
SparseVLM (ICML25)	57.6 -6.9%	62.5 -3.4%	53.7 -7.6%	1721 -7.6%	83.6 -2.7%	69.1 -0.6%	75.6 -3.6%	56.1 -3.6%	50.5 +1.0%	96.1%
DART (EMNLP25)	58.9 -4.8%	63.6 -1.7%	57.0 -1.9%	**1856** -0.3%	82.8 -3.6%	**69.8** +0.4%	76.7 -2.2%	57.4 -1.4%	51.1 +2.2%	98.5%
VisionZip (CVPR25)	59.3 -4.2%	64.5 -0.3%	57.3 -1.4%	1767 -5.1%	86.4 +0.6%	68.9 -0.9%	76.8 -2.0%	57.3 -1.5%	51.6 +3.2%	98.1%
VisPruner (ICCV25)	58.0 -6.3%	63.2 -2.3%	56.6 -2.6%	1765 -5.2%	85.0 -1.0%	69.3 -0.3%	76.5 -2.4%	57.2 -1.7%	51.9 +3.8%	98.0%
HoloV (NeurIPS25)	59.0 -4.7%	65.4 +1.1%	**58.0** -0.2%	1820 -2.3%	85.6 -0.3%	**69.8** +0.4%	76.7 -2.2%	57.4 -1.4%	50.9 +1.8%	99.2%
CDPruner (NeurIPS25)	59.7 -3.6%	63.8 -1.4%	56.4 -2.9%	1815 -2.5%	85.8 -0.1%	69.3 -0.4%	77.1 -1.7%	57.1 -1.9%	52.0 +4.0%	98.9%
CondenseVLM (Ours)	**59.8** -3.4%	**65.3** +0.9%	57.3 -1.4%	1794 -3.7%	**87.1** +1.4%	69.6 +0.1%	**77.7** -0.9%	**57.5** -1.2%	**51.7** +3.4%	**99.5%**
LLaVA-1.5-7B					*Retain 128 Tokens* (↓ **77.8%**)					
FastV (ECCV25)	49.6 -19.9%	56.1 -13.3%	56.4 -2.9%	1490 -20.0%	59.6 -30.6%	60.2 -13.4%	61.8 -21.2%	50.6 -13.1%	51.3 +2.6%	85.4%
LLaVA-PruMerge (ICCV25)	53.3 -13.9%	58.1 -10.2%	51.7 -11.0%	1554 -16.5%	67.2 -21.8%	67.1 -3.5%	68.8 -12.2%	54.3 -6.7%	50.3 +0.6%	89.4%
PDrop (CVPR25)	56.0 -9.5%	61.1 -5.6%	56.6 -2.6%	1644 -11.7%	82.3 -4.2%	68.3 -1.7%	72.9 -7.0%	55.1 -5.3%	51.0 +2.0%	94.9%
FiCoCo-V (2025.03)	57.6 -6.9%	61.1 -5.6%	54.3 -6.5%	1711 -8.1%	82.2 -4.3%	68.3 -1.7%	73.1 -6.8%	55.6 -4.5%	49.4 -1.2%	94.9%
MustDrop (2024.11)	56.9 -8.1%	61.1 -5.6%	55.2 -5.0%	1745 -6.3%	78.7 -8.4%	68.5 -1.4%	74.6 -4.8%	56.3 -3.3%	52.1 +4.2%	95.7%
HiRED (AAAI25)	57.2 -7.6%	61.5 -4.9%	53.6 -7.7%	1710 -8.2%	79.8 -7.1%	68.1 -2.0%	73.4 -6.4%	46.1 -20.8%	51.3 +2.6%	93.1%
SparseVLM (ICML25)	56.0 -9.5%	60.0 -7.3%	51.1 -12.0%	1696 -8.9%	80.5 -6.3%	67.1 -3.5%	73.8 -5.9%	54.9 -5.7%	51.4 +2.8%	93.8%
DART (EMNLP25)	57.9 -6.5%	63.2 -2.3%	**57.0** -1.9%	**1845** -0.9%	80.1 -6.8%	69.1 -0.6%	75.9 -3.2%	56.4 -3.1%	51.7 +3.4%	97.5%
VisionZip (CVPR25)	57.6 -6.9%	63.4 -2.0%	56.7 -2.4%	1768 -5.0%	84.7 -1.4%	68.8 -1.0%	75.6 -3.6%	56.8 -2.4%	52.0 +4.0%	97.2%
VisPruner (ICCV25)	57.0 -7.9%	62.6 -3.2%	56.0 -3.6%	1745 -6.3%	84.6 -1.5%	69.1 -0.6%	75.8 -3.3%	57.0 -2.1%	52.7 +5.4%	97.4%
HoloV (NeurIPS25)	57.7 -6.8%	63.9 -1.2%	56.5 -2.8%	1802 -3.2%	84.0 -2.2%	**69.8** +0.4%	75.5 -3.7%	56.8 -2.4%	51.5 +3.0%	98.0%
CDPruner (NeurIPS25)	**59.3** -4.2%	63.1 -2.5%	55.0 -5.3%	1792 -3.8%	85.7 -0.2%	69.0 -0.7%	**76.4** -2.6%	56.2 -3.4%	52.8 +5.6%	98.1%
CondenseVLM (Ours)	58.9 -4.8%	**64.5** -0.3%	56.3 -3.1%	1747 -6.2%	**86.2** +0.3%	69.7 +0.3%	76.3 -2.7%	**57.1** -1.9%	**52.9** +5.8%	**98.8%**
LLaVA-1.5-7B					*Retain 64 Tokens* (↓ **88.9%**)					
FastV (ECCV25)	46.1 -25.5%	48.0 -25.8%	52.7 -9.3%	1256 -32.5%	48.0 -44.1%	51.1 -26.5%	55.0 -29.8%	47.8 -17.9%	50.8 +1.6%	76.7%
LLaVA-PruMerge (ICCV25)	51.9 -16.2%	55.3 -14.5%	49.1 -15.5%	1549 -16.8%	65.3 -24.0%	68.1 -2.0%	67.4 -14.0%	54.0 -7.2%	50.1 +0.2%	87.7%
PDrop (CVPR25)	41.9 -32.3%	33.3 -48.5%	50.5 -13.1%	1092 -41.4%	55.9 -34.9%	68.6 -1.3%	69.2 -11.7%	45.9 -21.1%	50.7 +1.4%	77.5%
FiCoCo-V (2025.03)	52.4 -15.3%	60.3 -6.8%	53.0 -8.8%	1591 -14.6%	76.0 -11.5%	68.1 -2.0%	71.3 -9.1%	53.6 -7.9%	49.8 -0.4%	91.5%
MustDrop (2024.11)	53.1 -14.2%	60.0 -7.3%	53.1 -8.6%	1612 -13.4%	68.0 -20.8%	63.4 -8.8%	69.3 -11.6%	54.2 -6.9%	51.2 +2.4%	90.1%
HiRED (AAAI25)	54.6 -11.8%	60.2 -7.0%	51.4 -11.5%	1599 -14.1%	73.6 -14.3%	68.2 -1.9%	69.7 -11.1%	44.2 -24.1%	50.2 +0.4%	89.4%
SparseVLM (ICML25)	52.7 -14.9%	56.2 -13.1%	46.1 -20.7%	1505 -19.2%	75.1 -12.6%	62.2 -10.5%	68.2 -13.0%	51.8 -11.0%	50.1 +0.2%	87.3%
DART (EMNLP25)	55.9 -9.7%	60.6 -6.3%	53.2 -8.4%	**1765** -5.2%	73.9 -14.0%	69.8 +0.4%	72.4 -7.7%	54.4 -6.5%	51.6 +3.2%	93.9%
VisionZip (CVPR25)	55.1 -11.0%	60.1 -7.1%	**55.4** -4.6%	1690 -9.2%	77.0 -10.4%	69.0 -0.7%	72.4 -7.7%	55.5 -4.6%	52.9 +5.8%	94.5%
VisPruner (ICCV25)	54.4 -12.1%	61.5 -4.9%	54.2 -6.7%	1672 -10.2%	80.4 -6.4%	69.3 -0.3%	72.7 -7.3%	55.3 -5.0%	53.3 +6.6%	94.9%
HoloV (NeurIPS25)	55.3 -10.7%	**63.3** -2.2%	55.1 -5.2%	1715 -7.9%	80.3 -6.5%	69.5 +0.0%	72.8 -7.1%	55.4 -4.8%	52.8 +5.6%	95.8%
CDPruner (NeurIPS25)	**57.8** -6.6%	62.3 -3.7%	53.2 -8.4%	1703 -8.5%	**84.7** -1.4%	68.5 -1.4%	**75.3** -4.0%	55.2 -5.2%	53.4 +6.8%	96.4%
CondenseVLM (Ours)	56.5 -8.7%	63.1 -2.5%	55.1 -5.2%	1695 -9.0%	82.8 -3.6%	**70.1** +0.9%	74.2 -5.4%	**55.6** -4.5%	**54.1** +8.2%	**97.0%**

At 128 and 64 tokens, it remains competitive on both reasoning-oriented benchmarks and text-rich VQA, which supports the evidence-preservation view: the retained set should cover non-redundant salient regions, preserve sparse contextual cues, and reconstruct discarded tokens rather than only keep top-ranked patches. This conclusion is consistent with Figure 2, where the gap between our proposed CondenseVLM and single-proxy baselines becomes clearer as pruning becomes more aggressive. Rows for recent diversity- and instruction-aware pruners such as CDPruner and VisPruner in Table 1 further verify the same trend.

4.2 Generalization across Backbones

Tables 2 and 3 evaluate whether the same design transfers beyond LLaVA-1.5. On LLaVA-NeXT-7B, which uses a much longer high-resolution visual stream, CondenseVLM preserves 96.1% average performance with only 320 retained tokens under 88.9% pruning, outperforming the best prior average by 0.5 points. This setting is particularly challenging because high-resolution inputs introduce more visually similar local patterns; preserving non-redundant anchors and enforcing spatially compatible merging thus becomes more valuable.

On Qwen2.5-VL-7B, CondenseVLM also obtains the best average preservation across 66.7%, 77.8%, and 88.9% pruning, reaching 95.1%, 92.4%, and 88.5%, respectively. The improvements are consistent across POPE, ScienceQA, VQAv2, and VizWiz, suggesting that the method is not tied to a specific visual encoder or LLaVA-style architecture. Instead, the advantage comes from performing evidence-preserving condensation before the frozen decoder consumes the visual sequence.

Table 4 extends the comparison to InternVL2 and LLaVA-OneVision. On InternVL2-2B and InternVL2-8B with 50% token pruning, CondenseVLM preserves 98.9% and 98.6% average performance, respectively, outperforming FastV and VisionZip. On LLaVA-OneVision-8B, Conden-

Table 2: Performance comparison on LLaVA-NeXT-7B. Evaluation demonstrates that CondenseVLM significantly outperforms existing SOTA pruning methods in both individual task accuracy and overall average performance.

Methods	GQA	MMB	MMB_{CN}	MME	POPE	SQA	VQA_{V2}	VQA_{Text}	VizWiz	Avg.
Upper Bound, 2880 Tokens	64.2	67.4	60.6	1851	86.5	70.1	81.8	64.9	57.6	100%
LLaVA-NeXT-7B					*Retain 320 Tokens* (↓ **88.9%**)					
FastV (ECCV24)	55.9 -12.9%	61.6 -8.6%	51.9 -14.4%	1661 -10.3%	71.7 -17.1%	62.8 -10.4%	71.9 -12.1%	55.7 -14.2%	53.1 -7.8%	88.0%
LLaVA-PruMerge (ICCV25)	53.6 -16.5%	61.3 -9.1%	55.3 -8.7%	1534 -17.1%	60.8 -29.7%	66.4 -5.3%	69.7 -14.8%	50.6 -22.0%	54.0 -6.2%	85.6%
PDrop (CVPR25)	56.4 -12.1%	63.4 -5.9%	56.2 -7.3%	1663 -10.2%	77.6 -10.3%	67.5 -3.7%	73.5 -10.1%	54.4 -16.2%	54.1 -6.1%	90.9%
MustDrop (2024.11)	57.3 -10.7%	62.8 -6.8%	55.1 -9.1%	1641 -11.3%	82.1 -5.1%	68.0 -3.0%	73.7 -9.9%	59.9 -7.7%	54.0 -6.2%	92.2%
FasterVLM (ICCV25)	56.9 -11.4%	61.6 -8.6%	53.5 -11.7%	1701 -8.1%	83.6 -3.4%	66.5 -5.1%	74.0 -9.5%	56.5 -12.9%	52.6 -8.7%	91.1%
HiRED (AAAI25)	59.3 -7.6%	64.2 -4.7%	55.9 -7.8%	1690 -8.7%	83.3 -3.7%	66.7 -4.8%	75.7 -7.5%	58.8 -9.4%	54.2 -5.9%	93.3%
SparseVLM (ICML25)	56.1 -12.6%	60.6 -10.1%	54.5 -10.1%	1533 -17.2%	82.4 -4.7%	66.1 -5.7%	71.5 -12.6%	58.4 -10.0%	52.0 -9.7%	89.7%
GlobalCom^2 (2025.2)	57.1 -11.1%	61.8 -8.3%	53.4 -11.9%	1698 -8.3%	83.8 -3.1%	67.4 -3.9%	76.7 -6.2%	57.2 -11.9%	54.6 -5.2%	92.2%
DART (EMNLP25)	61.7 -3.9%	**65.3** -3.1%	**58.2** -4.0%	1710 -7.6%	84.1 -2.8%	68.4 -2.4%	79.1 -3.3%	58.7 -9.6%	56.1 -2.6%	93.9%
VisionZip (CVPR25)	59.3 -7.6%	63.1 -6.4%	55.3 -8.7%	1702 -8.0%	82.1 -5.1%	67.3 -4.0%	78.8 -3.7%	58.9 -9.2%	55.9 -3.0%	93.7%
VisPruner (ICCV25)	59.2 -7.8%	63.4 -5.9%	56.2 -7.3%	1705 -7.9%	81.5 -5.8%	68.2 -2.7%	77.2 -5.6%	59.4 -8.5%	55.4 -3.8%	93.9%
HoloV (NeurIPS25)	61.7 -3.9%	**65.3** -3.1%	57.5 -5.1%	1738 -6.1%	83.9 -3.0%	**68.9** -1.7%	79.5 -2.8%	58.7 -9.6%	55.3 -4.0%	95.6%
CDPruner (NeurIPS25)	**62.2** -3.1%	64.2 -4.7%	55.5 -8.4%	1730 -6.5%	**85.2** -1.5%	67.8 -3.3%	**80.0** -2.2%	59.2 -8.8%	55.6 -3.5%	95.3%
CondenseVLM (Ours)	61.8 -3.7%	65.1 -3.4%	57.6 -5.0%	**1745** -5.7%	84.2 -2.7%	68.8 -1.9%	79.6 -2.7%	**60.1** -7.4%	**56.3** -2.3%	**96.1%**

Table 3: Performance comparison on Qwen2.5-VL-7B across pruning ratios. CondenseVLM achieves SOTA.

Methods	GQA	MMB	MME	POPE	SQA	VQA_{V2}	VQA_{Text}	VizWiz	Avg.
Upper Bound, Unpruned	70.0	87.1	2220	84.8	80.0	91.7	85.0	68.1	100%
Qwen2.5-VL-7B				*Token Pruning Rate* = 66.7%					
FastV (ECCV24)	61.0 -12.9%	75.7 -13.1%	2072 -6.7%	82.2 -3.1%	78.5 -1.9%	86.5 -5.7%	77.9 -8.4%	64.1 -5.9%	92.8%
PDrop (CVPR25)	60.7 -13.3%	75.5 -13.3%	2043 -8.0%	81.8 -3.5%	78.0 -2.5%	86.6 -5.6%	77.2 -9.2%	63.7 -6.5%	92.3%
DART (EMNLP25)	63.2 -9.70%	77.5 -11.0%	2106 -5.1%	83.1 -2.0%	77.6 -3.0%	85.9 -6.3%	78.6 -7.5%	**65.4** -4.0%	93.9%
VisionZip (CVPR25)	62.5 -10.7%	76.0 -12.7%	2097 -5.5%	82.9 -2.2%	78.8 -1.5%	87.0 -5.1%	78.3 -7.9%	65.0 -4.6%	93.7%
VisPruner (ICCV25)	62.0 -11.4%	76.2 -12.5%	2085 -6.1%	82.5 -2.7%	78.1 -2.4%	86.8 -5.3%	77.5 -8.8%	64.2 -5.7%	93.1%
HoloV (NeurIPS25)	63.4 -9.40%	78.2 -10.2%	2110 -5.0%	83.0 -2.1%	78.9 -1.4%	87.1 -5.0%	78.5 -7.6%	65.2 -4.3%	94.4%
CDPruner (NeurIPS25)	63.5 -9.30%	77.0 -11.6%	2115 -4.7%	83.2 -1.9%	78.2 -2.2%	87.2 -4.9%	78.0 -8.2%	64.5 -5.3%	94.0%
CondenseVLM (Ours)	**63.8** -8.90%	**80.0** -8.20%	**2133** -3.9%	**83.3** -1.8%	**79.9** -0.1%	**87.5** -4.6%	**78.9** -7.2%	65.0 -4.6%	**95.1%**
Qwen2.5-VL-7B				*Token Pruning Rate* = 77.8%					
FastV (ECCV24)	60.5 -13.6%	74.9 -14.0%	2036 -8.3%	80.7 -4.8%	78.0 -2.5%	82.3 -10.3%	69.5 -18.2%	63.5 -6.8%	90.2%
PDrop (CVPR25)	60.2 -14.0%	75.0 -13.9%	2017 -9.1%	80.4 -5.2%	77.5 -3.1%	82.1 -10.5%	69.2 -18.6%	64.0 -6.0%	89.6%
DART (EMNLP25)	62.0 -11.4%	76.1 -12.6%	**2125** -4.3%	81.9 -3.4%	78.1 -2.4%	83.2 -9.3%	**71.2** -16.2%	63.5 -6.8%	91.7%
VisionZip (CVPR25)	61.8 -11.7%	75.7 -13.1%	2109 -5.0%	81.2 -4.2%	78.2 -2.2%	83.0 -9.5%	70.7 -16.8%	**65.0** -4.6%	91.4%
VisPruner (ICCV25)	61.2 -12.6%	75.2 -13.7%	2060 -7.2%	81.0 -4.5%	77.8 -2.8%	82.8 -9.7%	69.8 -17.9%	63.6 -6.6%	90.6%
HoloV (NeurIPS25)	62.2 -11.1%	76.5 -12.2%	2118 -4.6%	82.0 -3.3%	78.5 -1.9%	84.0 -8.4%	71.0 -16.5%	64.5 -5.3%	92.1%
CDPruner (NeurIPS25)	**62.8** -10.3%	75.8 -13.0%	2112 -4.9%	82.5 -2.7%	77.9 -2.6%	84.5 -7.9%	70.2 -17.4%	63.8 -6.3%	91.9%
CondenseVLM (Ours)	62.5 -10.7%	**77.0** -11.6%	2099 -5.5%	**82.7** -2.5%	**78.9** -1.4%	**85.6** -6.7%	70.7 -16.8%	64.2 -5.7%	**92.4%**
Qwen2.5-VL-7B				*Token Pruning Rate* = 88.9%					
FastV (ECCV24)	57.2 -18.3%	71.2 -18.3%	1949 -12.2%	78.6 -7.3%	77.4 -3.2%	81.0 -11.7%	60.3 -29.1%	60.5 -11.%	86.1%
PDrop (CVPR25)	56.3 -19.6%	71.4 -18.0%	1920 -13.5%	77.0 -9.2%	76.9 -3.9%	81.5 -11.1%	60.5 -28.8%	60.3 -11.%	85.5%
DART (EMNLP25)	58.5 -16.4%	71.9 -17.5%	**2042** -8.0%	77.9 -8.1%	76.9 -3.9%	81.3 -11.3%	61.7 -27.4%	61.2 -10.%	87.1%
VisionZip (CVPR25)	58.0 -17.1%	72.7 -16.5%	2006 -9.6%	77.5 -8.6%	77.8 -2.8%	81.7 -10.9%	**61.9** -27.2%	61.5 -9.7%	87.2%
VisPruner (ICCV25)	57.8 -17.4%	71.5 -17.9%	1970 -11.3%	77.3 -8.8%	77.2 -3.5%	81.4 -11.4%	60.6 -28.7%	61.0 -10.%	86.3%
HoloV (NeurIPS25)	59.2 -15.4%	72.9 -16.3%	2025 -8.8%	78.4 -7.5%	78.0 -2.5%	82.0 -10.6%	61.5 -27.6%	62.2 -8.7%	87.8%
CDPruner (NeurIPS25)	**60.8** -13.1%	72.0 -17.3%	2030 -8.6%	79.5 -6.2%	77.6 -3.0%	82.8 -9.70%	60.8 -28.5%	62.0 -9.0%	88.1%
CondenseVLM (Ours)	60.4 -13.7%	**73.2** -16.0%	2014 -9.3%	**79.7** -6.0%	**78.5** -1.9%	**83.2** -9.30%	61.3 -27.9%	**62.7** -7.9%	**88.5%**

seVLM preserves 97.3% average performance at 70% pruning and 90.0% at 90% pruning, improving over VisionZip by 1.6 and 6.4 points. The larger margin under 90% pruning again indicates that evidence-preserving condensation is most useful when the visual budget is too small for single-proxy token selection to remain reliable. The additional 13B-backbone results in the Appendix Table 9 show the same tendency on the larger LLaVA-1.5 and LLaVA-NeXT models.

4.3 Text-Dense, Grounding, and Video Tasks

We next test whether the compressed visual sequence preserves fine-grained and spatially localized evidence. Figure 3 summarizes OCRBench, ChartQA, and ChineseOCRBench under different pruning ratios. These tasks are difficult for visual token pruning because the answer often depends on small text regions, chart elements, or peripheral patches of the image rather than a single salient object. CondenseVLM maintains strong preservation across pruning ratios and remains robust even under extreme pruning, indicating that context anchors and residual aggregation help protect small but answer-critical visual evidence.

Grounding and video QA provide complementary stress tests. Table 6 shows that on RefCOCO, RefCOCO+, and RefCOCOg, CondenseVLM preserves 96.5% average grounding performance on Qwen2.5-VL-7B with a 50% retained-token ratio, outperforming VisionZip by 5.4 points. This result directly supports the spatial part of our design: when localization matters, merging distant but feature-similar patches is risky, and spatially con-

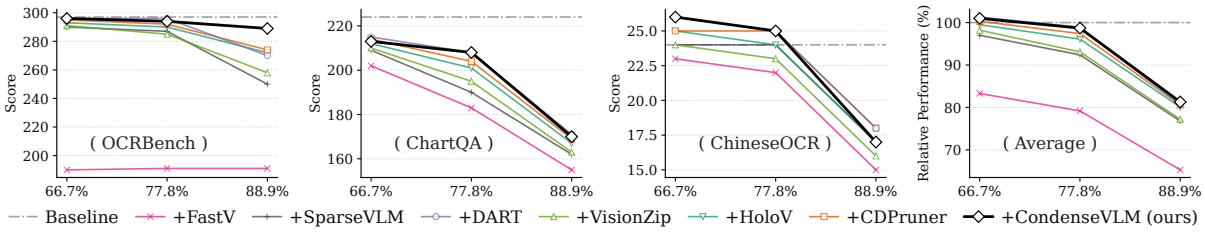


Figure 3: Performance comparison on text-dense benchmarks under different pruning ratios. CondenseVLM remains robust when small textual regions are critical, especially under 77.8% and 88.9% pruning.

Table 4: Performance comparison on InternVL2 and LLaVA-OneVision models with different token pruning ratios.

Methods	VizWiz	GQA	TextVQA	MME	MMB	MM-Vet	POPE	Avg.
InternVL2-2B (50%)	29.4	57.3	72.0	1821	72.5	39.6	85.4	100.%
FastV (ECCV24)	27.1 -7.8%	55.8 -2.6%	70.7 -1.8%	1774 -2.6%	71.8 -1.0%	34.1 -13.9%	**84.8** -0.7%	95.7%
VisionZip (CVPR25)	27.3 -7.1%	56.2 -1.9%	71.0 -1.4%	1813 -0.5%	72.1 -0.6%	34.6 -12.6%	82.9 -2.9%	96.1%
CondenseVLM (Ours)	**29.0** -1.4%	**56.8** -0.9%	**71.8** -1.8%	**1829** +0.4%	**72.3** -0.3%	**38.6** -2.5%	84.5 -1.1%	**98.9%**
InternVL2-8B (50%)	32.9	62.7	76.6	2205	81.8	60.0	86.7	100.%
FastV (ECCV24)	30.3 -7.9%	62.0 -1.1%	75.6 -1.3%	**2214** +0.4%	81.2 -0.7%	56.6 -5.7%	86.5 -0.2%	97.6%
VisionZip (CVPR25)	31.2 -5.2%	62.1 -1.0%	75.9 -0.9%	2185 -0.9%	81.3 -0.6%	56.8 -5.3%	**86.8** +0.1%	98.0%
CondenseVLM (Ours)	**32.1** -2.4%	**62.3** -0.6%	**76.2** -0.5%	2200 -0.2%	**81.5** -0.4%	**56.9** -5.2%	86.2 -0.6%	**98.6%**
LLaVA-OneVision-1.5-8B (70%)	66.0	69.2	79.5	2271	85.3	57.8	88.5	100.%
FastV (ECCV24)	64.1 -2.9%	65.2 -5.8%	72.3 -9.1%	2020 -11.1%	79.6 -6.7%	52.4 -9.3%	70.4 -20.5%	90.7%
VisionZip (CVPR25)	64.7 -2.0%	65.9 -4.8%	73.2 -7.9%	2105 -7.3%	83.3 -2.3%	55.3 -4.3%	**87.1** -1.6%	95.7%
CondenseVLM (Ours)	**65.0** -1.5%	**66.7** -3.6%	**77.1** -3.0%	**2210** -2.7%	**83.4** -2.2%	**56.2** -2.7%	85.7 -3.2%	**97.3%**
LLaVA-OneVision-1.5-8B (90%)	66.0	69.2	79.5	2271	85.3	57.8	88.5	100.%
FastV (ECCV24)	60.9 -7.7%	61.3 -11.4%	56.5 -28.9%	1800 -20.8%	71.1 -16.6%	46.8 -19.1%	62.9 -28.9%	80.9%
VisionZip (CVPR25)	59.8 -9.4%	60.7 -12.3%	48.2 -39.4%	1980 -12.8%	73.3 -14.1%	48.3 -16.4%	79.3 -10.4%	83.6%
CondenseVLM (Ours)	**62.6** -5.2%	**63.1** -8.8%	**63.4** -20.3%	**2002** -11.8%	**77.6** -9.0%	**52.0** -10.0%	**83.9** -5.2%	**90.0%**

Table 5: The results on diverse VideoQA tasks with VideoLLaVA-7B. CondenseVLM obtains better results.

Method	Token	TGIF-QA		MSVD-QA		MSRVTT		Average
		Acc	Score	Acc	Score	Acc	Score	Acc/Score
VideoLLaVA	2048	18.1	1.27	64.1	3.41	56.1	2.97	46.1/2.55
FastV (ECCV24)	456	16.4	1.17	60.3	3.24	53.0	2.84	43.2/2.42
VisPruner (CVPR24)	456	16.2	1.16	60.6	3.31	54.2	2.88	43.7/2.45
CondenseVLM	456	**17.5**	**1.25**	**63.6**	**3.36**	**55.3**	**2.93**	**45.5/2.51**
FastV (ECCV24)	228	12.4	1.00	55.1	2.93	50.4	2.70	39.3/2.21
VisPruner (CVPR25)	228	14.5	1.10	58.9	3.07	51.6	2.79	41.7/2.32
CondenseVLM	228	**16.3**	**1.17**	**61.5**	**3.30**	**53.8**	**2.87**	**43.9/2.45**

strained assignments better preserve region-level evidence. Table 5 further evaluates VideoLLaVA-7B on TGIF-QA, MSVD-QA, and MSRVTT-QA. Compared with VisPruner, CondenseVLM improves the average Acc./Score from 43.7/2.45 to 45.5/2.51 at 456 retained tokens and from 41.7/2.32 to 43.9/2.45 at 228 retained tokens. Although CondenseVLM is designed for image-token condensation, the video results suggest that diversity, context coverage, and locality are also useful when evidence is distributed across frames.

4.4 Ablation Study

Figure 4 isolates the contribution of each component under 90% pruning. Starting from a saliency-only anchor selection baseline, adding diversity-aware selection improves average preservation from 92.5% to 94.8%, validating the marginal-coverage view in Eq. (7). Adding token merging recovers information from discarded regions, and adding spatially constrained merging further raises average preservation to 96.5%. The same trend appears across MME, MM-Vet, MMBench, and POPE, showing that the improvement is not caused by a single benchmark artifact. Empirically, the ablation follows the objective in Eq. (2): faithful anchors preserve salient evidence, context anchors preserve global coverage, and spatial-semantic merging controls reconstruction distortion.

Figure 5 studies the context-anchor budget. Removing background anchors forces low-saliency context to be absorbed by foreground anchors, while allocating too many background anchors reduces the budget available for faithful evidence. A small non-zero context budget gives the best trade-off, supporting our choice to model contextual coverage as a separate channel rather than letting context compete directly with foreground saliency.

4.5 Efficiency–Accuracy Trade-off

Finally, Table 7 measures wall-clock efficiency on POPE using LLaVA-1.5-7B. At 66.7% pruning, CondenseVLM reduces the total evaluation time from 49:41 to 31:02 and lowers latency from 0.334s to 0.208s while preserving 101.0% POPE accuracy. At 90% pruning, it reduces latency to 0.191s and memory to 14.5G while preserving 95.5% accuracy. FasterVLM is slightly faster at the same 90% prun-

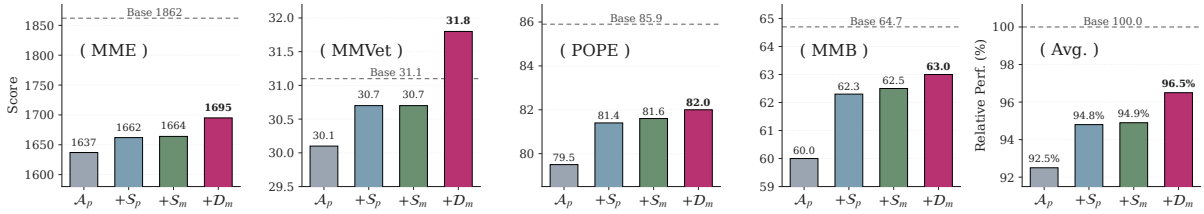


Figure 4: Ablation studies. Each component of CondenseVLM contributes to evidence preservation: diversity-aware selection improves coverage, merging recovers discarded evidence, and spatial constraints reduce harmful fusion.

Table 6: Grounding-oriented evaluation on RefCOCO, RefCOCO+, and RefCOCOg datasets with Qwen2.5-VL-7B.

Methods	RefCOCO testA	RefCOCO testB	RefCOCO val	RefCOCO+ testA	RefCOCO+ testB	RefCOCO+ val	RefCOCOg test	RefCOCOg val	Avg.
Qwen2.5-VL-7B (50%)	92.2	84.7	89.6	88.0	74.3	82.8	86.9	86.8	100%
FastV (ECCV24)	74.4 -19.3%	76.5 -9.7%	75.4 -15.8%	68.9 -21.7%	66.8 -10.1%	67.7 -18.2%	75.3 -13.3%	74.8 -13.8%	84.7%
VisionZip (CVPR25)	82.9 -10.1%	79.8 -5.8%	81.8 -8.7%	77.4 -12.0%	69.0 -7.1%	74.0 -10.6%	79.2 -8.9%	79.5 -8.4%	91.1%
CondenseVLM (Ours)	**90.2** -2.2%	**82.2** -3.0%	**86.7** -3.2%	**84.6** -3.9%	**70.6** -5.0%	**79.0** -4.6%	**83.6** -3.8%	**83.9** -3.3%	**96.5%**

Table 7: Efficiency and accuracy comparison on POPE. All methods use LLaVA-1.5-7B with 66.7% and 90% token pruning ratios. Our proposed CondenseVLM achieves superior trade-offs between inference speed, memory, acc.

Methods	Time	Prefill	Latency	Mem.	Acc.	Time	Prefill	Latency	Mem.	Acc.
Upper Bound, 576 Tokens	49:41	0.5ms	0.334s	19.0G	100.%	49:41	0.5ms	0.334s	19.0G	100.%
LLaVA-1.5-7B	*Retain 192 Tokens* (↓ 66.7%)					*Retain 58 Tokens* (↓ 90%)				
FastV (ECCV24)	35:34	0.5ms	0.239s	16.0G	75.4%	30:41	0.5ms	0.206s	15.6G	66.8%
MustDrop (2024.11)	32:30	0.5ms	0.273s	15.6G	96.2%	29:40	0.6ms	0.199s	14.5G	87.1%
FasterVLM (ICCV25)	30:09	0.5ms	**0.202s**	15.6G	100.%	25:08	0.5ms	**0.168s**	14.5G	92.5%
HiRED (AAAI25)	30:08	0.6ms	0.210s	15.7G	96.4%	25:03	0.6ms	0.168s	14.5G	92.7%
SparseVLM (ICML25)	40:51	0.6ms	0.251s	15.8G	97.3%	31:28	0.6ms	0.212s	14.6G	92.3%
CondenseVLM (Ours)	**31:02**	0.5ms	0.208s	**15.6G**	**101.%**	**28:28**	0.5ms	0.191s	**14.5G**	**95.5%**

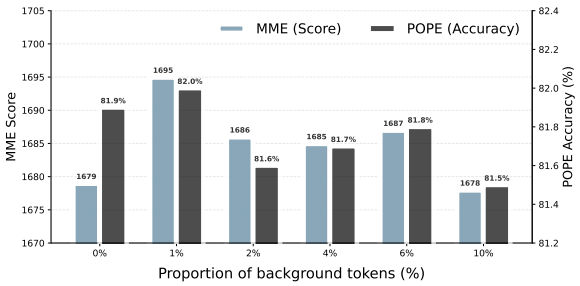


Figure 5: Background-anchor ratio with 90% pruning ratio. A small non-zero context budget performs best.

ing ratio, but its preservation drops to 92.5%. Thus, the one-time similarity computation in CondenseVLM strategy does not erase the decoding-time benefit, and it buys a stronger accuracy–efficiency trade-off in the aggressive-pruning regime.

5 Related Work

Efficient MLLM inference. MLLMs connect visual encoders to language models for multimodal instruction following and reasoning (Liu et al., 2024c,a,b; Li et al., 2024a; Bai et al., 2025). Their increasing input resolution and video support make visual-token efficiency a central deployment issue (Xu et al., 2024b; Li et al., 2024c; Lin et al., 2023).

Visual token pruning and merging. FastV and SparseVLM use attention-based pruning (Chen et al., 2024a; Zhang et al., 2024b); FasterVLM and VisionZip exploit visual saliency or clustering (Zhang et al., 2024a; Yang et al., 2024a); ToMe and LLaVA-PruMerge merge similar tokens (Bolya et al., 2022; Shang et al., 2024); and DART, CDPruner, IDPruner, VisPruner, and VLM-Pruner add duplication-, diversity-, instruction-, or sparsity-aware criteria (Wen et al., 2025; Zhang et al., 2025a; Tan et al., 2026; Zhang et al., 2025b; Wu et al., 2025). CondenseVLM is closest to this training-free family, but differs by deriving selection and merging from a single evidence-preservation view: anchors should cover non-redundant evidence, context should be explicitly reserved, and merging should respect locality.

6 Conclusion

We present CondenseVLM that rethinks visual token compression as evidence-preserving condensation. The central message is that efficient MLLM inference should not be reduced to ranking individual tokens: the retained sequence must cover non-redundant salient evidence, preserve sparse context, and reconstruct discarded information without breaking spatial structure. Experiments across model families, resolutions, grounding, text-dense tasks, video QA, and larger backbones show that this view yields robust accuracy-efficiency trade-offs under aggressive visual-token reduction.

Limitations

CondenseVLM adds a similarity-matrix computation before decoding, so the relative speedup can be smaller for extremely short generations or very small models. Integrating the same evidence-preservation objective into training or instruction tuning remains future work. The context-anchor selection is a lightweight approximation to global evidence coverage, not a substitute for task-specific perception modules in safety-critical settings.

References

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

Changwoo Baek, Jouwon Song, Sohyeon Kim, and Kyeongbo Kong. 2026. Agilepruner: An empirical study of attention and diversity for adaptive visual token pruning in large vision-language models. In *International Conference on Learning Representations (ICLR)*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

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

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336.

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

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2024b. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

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

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Ling Fu, Biao Yang, Zhebin Kuang, Jiajun Song, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, Mingxin Huang, Zhang Li, Guozhi Tang, Bin Shan, Chunhui Lin, Qi Liu, Binghong Wu, Hao Feng, Hao Liu, Can Huang, and 5 others. 2025. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning.

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

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617.

Yuhang Han, Xuyang Liu, Pengxiang Ding, Donglin Wang, Honggang Chen, Qingsen Yan, and Siteng Huang. 2024. Rethinking token reduction in mllms: Towards a unified paradigm for training-free acceleration. *arXiv preprint arXiv:2411.17686*.

Yefei He, Feng Chen, Jing Liu, Wenqi Shao, Hong Zhou, Kaipeng Zhang, and Bohan Zhuang. 2024. Zipvl: Efficient large vision-language models with dynamic token sparsification. *arXiv preprint arXiv:2410.08584*.

Kai Huang, Hao Zou, Ye Xi, BoChen Wang, Zhen Xie, and Liang Yu. 2024a. Ivtp: Instruction-guided visual

token pruning for large vision-language models. In *European Conference on Computer Vision*, pages 214–230. Springer.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. 2024b. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13418–13427.

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

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. 2017. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2758–2766.

Lei Jiang, Weizhe Huang, Tongxuan Liu, Yuting Zeng, Jing Li, Lechao Cheng, and Xiaohua Xu. 2024a. Fopru: Focal pruning for efficient large vision-language models. *arXiv preprint arXiv:2411.14164*.

Yifan Jiang, Kexuan Sun, Zhivar Sourati, Kian Ahrabian, Kaixin Ma, Filip Ilievski, Jay Pujara, and 1 others. 2024b. Marvel: Multidimensional abstraction and reasoning through visual evaluation and learning. *Advances in Neural Information Processing Systems*, 37:46567–46592.

Yutao Jiang, Qiong Wu, Wenhao Lin, Wei Yu, and Yiyi Zhou. 2025. What kind of visual tokens do we need? training-free visual token pruning for multi-modal large language models from the perspective of graph. *arXiv preprint arXiv:2502.17403*.

Andreas Krause and Daniel Golovin. 2014. Submodular function maximization. In *Tractability: Practical Approaches to Hard Problems*. Cambridge University Press.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and 1 others. 2024a. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Yanwei Li, Chengyao Wang, and Jiaya Jia. 2024b. LLaMA-VID: An image is worth 2 tokens in large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. 2024c. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814*.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2023. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

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

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024c. Visual instruction tuning. *Advances in neural information processing systems*, 36.

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

Xuyang Liu, Ziming Wang, Yuhang Han, Yingyao Wang, Jiale Yuan, Jun Song, Bo Zheng, Linfeng Zhang, Siteng Huang, and Honggang Chen. 2025a. Compression with global guidance: Towards training-free high-resolution mllms acceleration. *arXiv preprint arXiv:2501.05179*.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, and 1 others. 2025b. Mmbench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pages 216–233. Springer.

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

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xucheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. 2023. OCRBench: On the hidden mystery of OCR in large multimodal models. *arXiv preprint arXiv:2305.07895*.

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

Gen Luo, Yiyi Zhou, Yuxin Zhang, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. 2024. Feast your eyes: Mixture-of-resolution adaptation for multimodal large language models. *arXiv preprint arXiv:2403.03003*.

Yulin Luo, Ruichuan An, Bocheng Zou, Yiming Tang, Jiaming Liu, and Shanghang Zhang. 2025. Llm as dataset analyst: Subpopulation structure discovery with large language model. In *European Conference on Computer Vision*, pages 235–252. Springer.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L. Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11–20.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Michael Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. 2021. Tokenlearner: Adaptive space-time tokenization for videos. *Advances in neural information processing systems*, 34:12786–12797.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. 2024. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326.

Dingjie Song, Wenjun Wang, Shunian Chen, Xidong Wang, Michael Guan, and Benyou Wang. 2024. Less is more: A simple yet effective token reduction method for efficient multi-modal llms. *arXiv preprint arXiv:2409.10994*.

Zhichao Sun, Yidong Ma, Gang Liu, Yibo Chen, Xu Tang, Yao Hu, and Yongchao Xu. 2026. Ivc-prune: Revealing the implicit visual coordinates in lvlms for vision token pruning. In *International Conference on Learning Representations (ICLR)*.

Yifan Tan, Yifu Sun, Shirui Huang, Hong Liu, Guanghua Yu, Jianchen Zhu, and Yangdong Deng. 2026. IDPruner: Harmonizing importance and diversity in visual token pruning for MLLMs. *arXiv preprint arXiv:2602.13315*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, and 1 others. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Dezhan Tu, Danylo Vashchilenko, Yuzhe Lu, and Panpan Xu. 2024. Vl-cache: Sparsity and modality-aware kv cache compression for vision-language model inference acceleration. *arXiv preprint arXiv:2410.23317*.

Ao Wang, Fengyuan Sun, Hui Chen, Zijia Lin, Jungong Han, and Guiguang Ding. 2024. [cls] token tells everything needed for training-free efficient mllms. *arXiv preprint arXiv:2412.05819*.

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

Zhenkai Wu, Xiaowen Ma, Zhenliang Ni, Dengming Zhang, Han Shu, Xin Jiang, and Xinghao Chen. 2025. VLM-Pruner: Buffering for spatial sparsity in an efficient VLM centrifugal token pruning paradigm. *arXiv preprint arXiv:2512.02700*.

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

Bingxin Xu, Yuzhang Shang, Yunhao Ge, Qian Lou, and Yan Yan. 2024a. freepruner: A training-free approach for large multimodal model acceleration. *arXiv preprint arXiv:2411.15446*.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. 2017. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653.

Ruyi Xu, Yuan Yao, Zonghao Guo, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, Maosong Sun, and Gao Huang. 2024b. Llava-uhd: an lmm perceiving any aspect ratio and high-resolution images. *arXiv preprint arXiv:2403.11703*.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. 2024a. Visionzip: Longer is better but not necessary in vision language models. *arXiv preprint arXiv:2412.04467*.

Te Yang, Jian Jia, Xiangyu Zhu, Weisong Zhao, Bo Wang, Yanhua Cheng, Yan Li, Shengyuan Liu, Quan Chen, Peng Jiang, and 1 others. 2024b. Enhancing instruction-following capability of visual-language models by reducing image redundancy. *arXiv preprint arXiv:2411.15453*.

Linli Yao, Lei Li, Shuhuai Ren, Lean Wang, Yuanxin Liu, Xu Sun, and Lu Hou. 2024. DeCo: Decoupling token compression from semantic abstraction in multimodal large language models. *arXiv:2405.20985*.

Weihao Ye, Qiong Wu, Wenhao Lin, and Yiyi Zhou. 2025. Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 22128–22136.

Hanxun Yu, Wentong Li, Xuan Qu, Song Wang, Junbo Chen, and Jianke Zhu. 2026. Visiontrim: Unified vision token compression for training-free mllm acceleration. In *International Conference on Learning Representations (ICLR)*.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. 2016. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2023. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*.

Qizhe Zhang, Aosong Cheng, Ming Lu, Zhiyong Zhuo, Minqi Wang, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. 2024a. [cls] attention is all you need for training-free visual token pruning: Make vlm inference faster. *arXiv preprint arXiv:2412.01818*.

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

Qizhe Zhang and 1 others. 2025b. Beyond text-visual attention: Exploiting visual cues for effective token pruning in VLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, and 1 others. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

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

Yuke Zhu, Chi Xie, Shuang Liang, Bo Zheng, and Sheng Guo. 2024. Focusllava: A coarse-to-fine approach for efficient and effective visual token compression. *arXiv preprint arXiv:2411.14228*.

Xin Zou, Di Lu, Yizhou Wang, Yibo Yan, Yuanhuiyi Lyu, Xu Zheng, Linfeng Zhang, and Xuming Hu. 2025a. Don't just chase "highlighted tokens" in MLLMs: Revisiting visual holistic context retention. *arXiv preprint arXiv:2510.02912*.

Xin Zou, Di Lu, Yizhou Wang, Yibo Yan, Yuanhuiyi Lyu, Xu Zheng, Linfeng Zhang, and Xuming Hu. 2025b. Don't just chase "highlighted tokens" in mllms: Revisiting visual holistic context retention. *Advances in Neural Information Processing Systems*, 38:39800–39832.

Appendix Overview

This appendix is organized to make the main paper self-contained while keeping the 8-page narrative concise. Appendix A gives a detailed theoretical justification of CondenseVLM and clarifies the exact claims supported by the analysis. Appendix B specifies datasets, models, baselines, and reproducibility details. The remaining sections provide additional related work, complete experimental tables, complexity analysis, visualizations, fine-grain results, limitations, future directions, ethics and impact statements, and LLM-use disclosure.

A Detailed Theoretical Justification

This section expands Proposition 1. The purpose of the theory is not to prove that a compressed visual sequence is globally equivalent to the full nonlinear MLLM computation. Such a statement would be unrealistic for a training-free method applied to arbitrary frozen decoders. Instead, we prove a narrower and more relevant claim: each component of CondenseVLM is the solution, approximation, or computational surrogate of a well-defined part of the evidence-preservation objective in Eq. (2). This turns the method from a collection of ad-hoc heuristics into a structured coreset approximation.

A.1 Notation and Objective Recap

Let $V = \{1, \dots, n\}$ denote the original visual tokens, $z_i \in \mathbb{R}^d$ the visual feature of token i, $a_i \geq 0$ the task-saliency weight, $g_i \geq 0$ the context weight, and $\kappa_{ij} = (1 + \cos(z_i, z_j))/2 \in [0, 1]$ the normalized feature similarity. Let $\bar{d}_{ij} \in [0, 1]$ be the normalized spatial distance on the visual grid. For a nonnegative weight vector w, define the representative coverage function

$$F_w(T) = \sum_{i \in V} w_i c_i(T), \tag{10}$$

where

$$c_i(T) = \begin{cases} \max_{j \in T} \kappa_{ij}, & T \neq \emptyset, \\ 0, & T = \emptyset. \end{cases} \tag{11}$$

The main objective can be written as

$$\max_{T_a, T_b, \pi} \; F_a(T_a) + \eta F_g(T_b) - \rho \sum_{i \notin T_a \cup T_b} D(i, \pi(i)), \tag{12}$$

where

$$D(i, j) = 1 - \kappa_{ij} + \beta \bar{d}_{ij}. \tag{13}$$

The constraints are $T = T_a \cup T_b$, $|T_a| + |T_b| = K$, and $\pi(i) \in T$. The three terms respectively model salient evidence coverage, contextual coverage, and local reconstruction distortion.

A.2 Coverage is a Submodular Evidence Surrogate

Lemma 1 (Facility-location coverage)**.** *For any $w_i \geq 0$ and $\kappa_{ij} \in [0, 1]$, the function F_w in Eq. (10) is normalized, monotone, and submodular.*

Proof. Normalization follows from $c_i(\emptyset) = 0$, which gives $F_w(\emptyset) = 0$. For monotonicity, let $A \subseteq B \subseteq V$. For each i, the maximum over B is taken over a superset of the candidates used by A, hence

$$c_i(A) = \max_{j \in A} \kappa_{ij} \leq \max_{j \in B} \kappa_{ij} = c_i(B), \tag{14}$$

with the convention $c_i(\emptyset) = 0$. Multiplying by $w_i \geq 0$ and summing over i gives $F_w(A) \leq F_w(B)$.

For submodularity, take $A \subseteq B \subseteq V$ and $x \notin B$. The marginal gain of adding x to a set S is

$$\begin{aligned} \Delta_w(x \mid S) &= F_w(S \cup \{x\}) - F_w(S) \\ &= \sum_{i \in V} w_i \left[\kappa_{ix} - c_i(S)\right]_+ . \end{aligned} \tag{15}$$

Because $A \subseteq B$, monotonicity of c_i gives $c_i(A) \leq c_i(B)$ for every i. Therefore,

$$\left[\kappa_{ix} - c_i(A)\right]_+ \geq \left[\kappa_{ix} - c_i(B)\right]_+ . \tag{16}$$

After multiplying by $w_i \geq 0$ and summing, we obtain $\Delta_w(x \mid A) \geq \Delta_w(x \mid B)$, which is the diminishing-returns definition of submodularity. □

Lemma 2 (Greedy coverage approximation)**.** *Let O be an optimal set of size at most K for maximizing F_w. The exact greedy algorithm that repeatedly adds the token with the largest marginal gain obtains*

$$\begin{aligned} F_w(S_K) &\geq \left(1 - \left(1 - \frac{1}{K}\right)^K\right) F_w(O) \\ &\geq (1 - 1/e) F_w(O). \end{aligned} \tag{17}$$

Proof. Let S_t be the greedy set after t selections and let $\delta_t = F_w(O) - F_w(S_t)$ be the remaining optimality gap. By monotonicity and submodularity,

$$F_w(O) - F_w(S_t) \le F_w(O \cup S_t) - F_w(S_t) \le \sum_{x \in O \setminus S_t} \Delta_w(x \mid S_t). \tag{18}$$

Since $|O| \le K$, at least one element in $O \setminus S_t$ has marginal gain at least δ_t/K. Greedy chooses the largest available marginal gain, so

$$F_w(S_{t+1}) - F_w(S_t) \ge \frac{1}{K}\delta_t. \tag{19}$$

Thus $\delta_{t+1} \le (1 - 1/K)\delta_t$. Unrolling for K steps gives

$$\delta_K \le \left(1 - \frac{1}{K}\right)^K F_w(O), \tag{20}$$

which implies the stated bound. This is the classical result of Nemhauser et al. (1978) specialized to the facility-location coverage function. □

Why this matters for visual-token compression. Lemma 1 formalizes the central difference between token ranking and evidence preservation. A scalar-ranking method assigns each token a fixed score independent of the retained set. In contrast, the marginal gain in Eq. (15) decreases once a similar token has already been retained. This is exactly the redundancy pattern observed in attention and [CLS]-saliency maps: multiple individually salient tokens can represent the same visual evidence, while a lower-scoring token may add new coverage.

A.3 Relation Between Exact Marginal Gain and the Implemented Score

Exact greedy maximization of Eq. (10) would require recomputing Eq. (15) for all remaining candidates at every selection step. This is principled but inefficient for a plug-and-play inference-time compressor. CondenseVLM therefore uses the MMR-style surrogate

$$s_a(x \mid T_a) = a_x - \lambda \max_{j \in T_a} \kappa_{xj}, \tag{21}$$

which follows the same marginal-coverage logic: the first term rewards direct task evidence and the second term penalizes evidence already represented by selected anchors. This surrogate should not be interpreted as carrying the exact $(1 - 1/e)$ approximation guarantee, because it replaces the full weighted marginal sum over all tokens by a candidate-local redundancy penalty. The correct interpretation is more limited and more accurate: Eq. (21) is a computationally cheap proxy for the diminishing-returns behavior proven above. It is used because it preserves the sign of the desired design principle–prefer salient but non-redundant anchors–while avoiding repeated full-set scans during inference.

The context-anchor score applies the same idea to a separate evidence channel. Selecting context tokens by

$$s_g(x) = \frac{1}{n-1} \sum_{i \ne x} \kappa_{xi} \tag{22}$$

chooses low-saliency candidates that are representative of many other low-saliency tokens. Separating the context budget from the faithful-anchor budget is important: if the two channels compete under one saliency-dominated ranking, globally useful layout or background tokens are likely to be removed. Formally, both F_a and F_g are submodular by Lemma 1, and any nonnegative weighted sum of such functions is also submodular. Operationally, CondenseVLM uses a fixed budget split because it is simple, deterministic, and prevents the context channel from being absorbed by foreground saliency.

A.4 Spatial-Semantic Assignment Controls Local Distortion

Lemma 3 (Decomposable optimal assignment)**.** *For a fixed retained set T, the assignment that minimizes*

$$\sum_{i \notin T} D(i, \pi(i)), \qquad \pi(i) \in T, \tag{23}$$

is obtained independently for each discarded token by

$$\pi(i) = \arg\min_{j \in T} D(i, j). \tag{24}$$

Proof. The objective is a sum of terms, and each term depends only on the individual choice $\pi(i)$ for that discarded token. Therefore, changing $\pi(i)$ cannot affect any other token's assignment cost. The global minimizer over assignments is obtained by minimizing each independent term separately. □

Lemma 4 (Spatial-semantic assignment)**.** *Under Assumption 1, if $L_z > 0$ and $\beta = L_c/L_z$, the rule*

$$\pi(i) = \arg\max_{j \in T} \{\kappa_{ij} - \beta \bar{d}_{ij}\} \tag{25}$$

minimizes the per-token perturbation upper bound $L_z(1-\kappa_{ij}) + L_c\bar{d}_{ij}$ *over anchors* $j \in T$.

Proof. For a fixed discarded token i, define the local perturbation upper bound

$$B(i,j) = L_z(1-\kappa_{ij}) + L_c\bar{d}_{ij}. \tag{26}$$

When $L_z > 0$, dividing by L_z and dropping the additive constant 1 gives

$$\begin{aligned}\arg\min_{j\in T} B(i,j) &= \arg\min_{j\in T}\left\{1-\kappa_{ij}+\frac{L_c}{L_z}\bar{d}_{ij}\right\}\\ &= \arg\max_{j\in T}\left\{\kappa_{ij}-\frac{L_c}{L_z}\bar{d}_{ij}\right\}.\end{aligned} \tag{27}$$

Setting $\beta = L_c/L_z$ yields Eq. (25). If $L_z = 0$ and $L_c > 0$, the bound becomes purely spatial; the safest assignment degenerates to nearest-neighbor spatial merging. In practice, β is a normalization hyperparameter that balances semantic and spatial scales. □

Combining Lemmas 3 and 4, for a fixed anchor set T we obtain

$$\sum_{i\notin T}\Delta(i,\pi(i)) \le L_z\sum_{i\notin T}\left(1-\kappa_{i\pi(i)}+\frac{L_c}{L_z}\bar{d}_{i\pi(i)}\right), \tag{28}$$

so the implemented assignment minimizes the additive upper bound on local replacement error. This explains why the spatial term is necessary: feature similarity alone can merge repeated textures or repeated object parts across distant regions, while the distance penalty discourages merges that are semantically plausible but geometrically unsafe.

A.5 Aggregation is Weighted Least-Squares Reconstruction

Lemma 5 (Weighted least-squares reconstruction)**.** *For an anchor* j *and its assigned cluster* $C_j = \{i : \pi(i) = j\}$*, with weights* $\omega_i \ge 0$*, the aggregated token*

$$\tilde{z}_j = \frac{z_j + \sum_{i\in C_j}\omega_i z_i}{1+\sum_{i\in C_j}\omega_i} \tag{29}$$

is the unique minimizer of

$$\mathcal{L}_j(u) = \|u-z_j\|_2^2 + \sum_{i\in C_j}\omega_i\|u-z_i\|_2^2 \tag{30}$$

whenever $1+\sum_{i\in C_j}\omega_i > 0$.

Proof. The objective is a strictly convex quadratic in u because the coefficient of $\|u\|_2^2$ is $1 + \sum_{i\in C_j}\omega_i > 0$. Its gradient is

$$\nabla_u\mathcal{L}_j(u) = 2(u-z_j) + 2\sum_{i\in C_j}\omega_i(u-z_i). \tag{31}$$

Setting the gradient to zero gives

$$\left(1+\sum_{i\in C_j}\omega_i\right)u = z_j + \sum_{i\in C_j}\omega_i z_i. \tag{32}$$

Solving for u yields Eq. (29). Therefore, the aggregation step is the exact weighted least-squares reconstruction of each local anchor cluster. □

The weights $\omega_i = \sigma(a_i)$ allow discarded tokens with higher task evidence to contribute more to the reconstructed anchor feature. When all residual weights are zero, aggregation reduces to keeping the original anchor unchanged; when residual weights are positive, the anchor becomes a local representative of its cluster rather than a hard deletion of all assigned tokens.

A.6 Unified Interpretation and Scope of the Claim

The above results justify the structure of CondenseVLM at three levels. First, representative coverage is a monotone submodular objective, so selecting anchors for coverage is a principled coreset problem rather than independent top-K ranking. Second, the implemented faithful-anchor and context-anchor scores are efficient surrogates that preserve the intended marginal-coverage behavior while enforcing a useful separation between foreground evidence and global context. Third, for a fixed retained set, spatial-semantic assignment optimizes a local perturbation bound, and aggregation exactly solves the corresponding local reconstruction problem.

The proof also clarifies the limitations of the theory. The MMR-style score is a surrogate and does not inherit the exact greedy approximation ratio. The local smoothness assumption only concerns small feature replacements around the original visual sequence and does not imply global equivalence of decoder outputs. The objective is therefore best understood as a principled design lens: it explains why the components are aligned with evidence preservation, and the experiments test whether the resulting approximation is effective across models and tasks.

B Detailed Experiment Settings

B.1 Datasets

Our evaluation suite follows the scope stated in the main text and covers four complementary settings: general image understanding, grounding-oriented localization, text-dense perception, and video question answering. This broader coverage is important for token condensation because different tasks stress different failure modes: general VQA tests semantic preservation, grounding tests spatial fidelity, OCR/chart tasks test small-region evidence, and video QA tests whether evidence distributed across frames can still be retained under a compressed visual budget.

General image understanding and reasoning. For image understanding, we evaluate on GQA (Hudson and Manning, 2019), MMBench and its Chinese split MMBench-CN (Liu et al., 2025b), MME (Fu et al., 2023), POPE (Li et al., 2023), ScienceQA (Lu et al., 2022), VQA-v2 (Goyal et al., 2017), TextVQA (Singh et al., 2019), VizWiz (Gurari et al., 2018), and MM-Vet (Yu et al., 2023). These benchmarks jointly cover compositional scene reasoning, multiple-choice multimodal reasoning, perceptual and cognition-oriented evaluation, object-hallucination assessment, science reasoning, open-ended VQA, text-centric VQA, real-world accessibility-oriented VQA, and integrated multimodal capability evaluation.

GQA. GQA (Hudson and Manning, 2019) is built around scene graphs and compositional questions. It stresses relational reasoning, attribute recognition, spatial understanding, and multi-step visual reasoning, making it useful for evaluating whether compressed visual tokens still preserve structured scene evidence.

MMBench and MMBench-CN. MMBench (Liu et al., 2025b) provides a hierarchical evaluation protocol for MLLMs, including perception and reasoning categories with more fine-grained sub-abilities. We use both the English MMBench split and MMBench-CN to test whether token condensation remains stable across language settings and ability categories.

MME. MME (Fu et al., 2023) evaluates both perceptual and cognition-oriented capabilities through manually designed instruction-answer pairs. Since MME reports a score rather than a standard accuracy, we normalize it by the corresponding uncompressed model when computing preservation ratios.

POPE. POPE (Li et al., 2023) evaluates object hallucination by asking binary questions about object presence. We include it because aggressive token pruning can remove low-saliency but answer-critical objects, which directly affects hallucination-sensitive evaluation.

ScienceQA. ScienceQA (Lu et al., 2022) contains multimodal science questions spanning natural science, social science, and language science. It is used to evaluate whether compressed visual representations preserve evidence needed for reasoning beyond direct object recognition.

VQA-v2. VQA-v2 (Goyal et al., 2017) is a standard open-ended VQA benchmark with human-annotated answers. It evaluates broad visual recognition and question-answering capability under natural-image inputs.

TextVQA. TextVQA (Singh et al., 2019) focuses on answering questions that require reading and reasoning over text appearing in images. It is particularly sensitive to small textual regions, so it serves as a stress test for evidence loss under aggressive pruning.

VizWiz. VizWiz (Gurari et al., 2018) contains real images and questions from blind or low-vision users. The images are often noisy, poorly framed, or visually ambiguous, which tests whether a compression method can preserve robust evidence under non-curated visual conditions.

MM-Vet. MM-Vet (Yu et al., 2023) evaluates integrated multimodal abilities, including recognition, OCR, knowledge, spatial awareness, language generation, and reasoning. We include it to test whether compression preserves the combination of capabilities required by open-ended multimodal tasks.

Grounding-oriented evaluation. To evaluate spatial evidence preservation, we use RefCOCO and RefCOCO+ (Yu et al., 2016) and RefCOCOg (Mao et al., 2016). These datasets require localizing or selecting the target region described by a referring expression. They are important for our method because spatially unconstrained merging can fuse feature-similar but geographically incompatible regions, which is especially harmful for grounding.

Text-dense and chart understanding. For text-heavy evaluation, we additionally report OCRBench (Liu et al., 2023), ChartQA (Masry et al.,

2022), and ChineseOCRBench (Liu et al., 2024e; Fu et al., 2025). OCRBench evaluates text recognition, document-oriented VQA, key-information extraction, and related OCR-centric abilities; ChartQA evaluates visual and logical reasoning over charts; ChineseOCRBench isolates Chinese OCR cases from OCRBench-style evaluation. These datasets test whether token condensation can preserve sparse fine-grained evidence, such as characters, numbers, table cells, and chart labels.

Video question answering. For video experiments, we use TGIF-QA (Jang et al., 2017), MSVD-QA, and MSRVTT-QA (Xu et al., 2017). TGIF-QA emphasizes temporal and spatio-temporal reasoning, while MSVD-QA and MSRVTT-QA evaluate question answering over short open-domain videos. These benchmarks test whether the same evidence-preserving principle transfers from image-token compression to frame-token compression.

B.2 Models

We evaluate CondenseVLM on six representative open-source MLLM families used in the main experiments: LLaVA-1.5, LLaVA-NeXT, Qwen2.5-VL, InternVL2, LLaVA-OneVision, and VideoLLaVA. This choice covers standard fixed-resolution image MLLMs, high-resolution image MLLMs, dynamic-resolution models, larger and stronger recent backbones, unified image/video models, and video-specific MLLMs.

LLaVA-1.5. LLaVA-1.5-7B and LLaVA-1.5-13B (Liu et al., 2024a) are used as the primary image-understanding backbones. They encode a standard image into a fixed visual-token sequence, making them a common testbed for studying visual-token reduction.

LLaVA-NeXT. LLaVA-NeXT-7B and LLaVA-NeXT-13B (Liu et al., 2024b) are used to test high-resolution settings. Compared with LLaVA-1.5, LLaVA-NeXT exposes the language model to more visual tokens, so it provides a stronger stress test for token condensation.

Qwen2.5-VL. Qwen2.5-VL-7B (Bai et al., 2025) is used for the Qwen-family experiments, background-anchor ablation, grounding evaluation, and text-dense evaluation. Its dynamic-resolution processing and strong OCR/localization ability make it suitable for testing whether CondenseVLM generalizes beyond the LLaVA architecture.

InternVL2. InternVL2-2B and InternVL2-8B are used to evaluate transferability to the InternVL family (Chen et al., 2024b). We include both a smaller and a larger variant to check whether the compression behavior is stable across model scale.

LLaVA-OneVision. LLaVA-OneVision-8B (Li et al., 2024a) is used as a stronger unified visual-language backbone with a substantially longer visual prefix. It is useful for evaluating aggressive compression when the original visual-token budget is large.

VideoLLaVA. VideoLLaVA-7B (Lin et al., 2023) is used for video QA experiments. We apply the same selection-and-merging principle to visual tokens extracted from video frames, which tests whether evidence-preserving condensation is not restricted to single-image inputs.

B.3 Baselines

We compare with representative training-free or plug-and-play visual-token reduction methods. Unless otherwise stated, all methods are evaluated under the same retained-token budget for the same backbone, and the uncompressed model is reported as the *Upper Bound* or *Baseline*. The baselines cover attention-based pruning, saliency-based pruning, token merging, spatial or hierarchical allocation, duplication/diversity-aware pruning, and recent instruction- or sparsity-aware methods.

FastV. FastV (Chen et al., 2024a) prunes visual tokens according to attention maps in early language-model layers. It is a representative text-visual-attention pruning baseline.

FasterVLM. FasterVLM (Zhang et al., 2024a) estimates token importance from `[CLS]`-visual attention in the vision encoder and prunes tokens before the visual sequence enters the language model.

SparseVLM. SparseVLM (Zhang et al., 2024b) uses cross-modal attention to rank visual-token importance and introduces adaptive sparsity with token recycling.

ToMe. ToMe (Bolya et al., 2022) is a general token-merging method that merges similar tokens through lightweight matching. We include it as a representative similarity-based merging baseline.

LLaVA-PruMerge. LLaVA-PruMerge (Shang et al., 2024) combines sparse attention-based pruning with key-similarity-based merging, representing hybrid prune-then-merge compression.

PDrop. PDrop (Xing et al., 2024) progressively drops visual tokens across stages, forming a pyramid-like token schedule.

MustDrop. MustDrop (Liu et al., 2024d) combines text-guided pruning, spatial merging, and output-aware cache policies to reduce tokens across multiple stages.

GlobalCom^2. GlobalCom^2 (Liu et al., 2025a) uses global thumbnail information to allocate compression ratios for high-resolution local crops, preserving local details under a global-local representation.

VisionZip. VisionZip (Yang et al., 2024a) compresses visual tokens by preserving a small set of important tokens and aggregating redundant tokens through similarity-based operations.

DART. DART (Wen et al., 2025) is a duplication-aware token reduction method. It keeps less-duplicated visual tokens by comparing candidate tokens with selected pivots, making it a strong recent diversity-oriented baseline.

HoloV. HoloV (Zou et al., 2025a) rethinks visual-token retention from a holistic perspective and allocates token budgets across spatial crops to avoid preserving only highlighted local regions.

CDPruner. CDPruner (Zhang et al., 2025a) formulates token pruning with conditional diversity, encouraging the retained set to cover complementary evidence rather than only individually salient tokens.

VisPruner. VisPruner (Zhang et al., 2025b) exploits visual cues beyond text-visual attention for effective token pruning in vision-language models.

IDPruner and VLM-Pruner. IDPruner (Tan et al., 2026) balances importance and diversity for visual-token pruning, while VLM-Pruner (Wu et al., 2025) introduces spatial sparsity buffering for efficient VLM token pruning. They are included in the related-method discussion to position CondenseVLM among recent diversity- and sparsity-aware pruning strategies.

B.4 Reproducibility Statement

All experiments are conducted on NVIDIA A800-80G GPUs. The implementation uses Python 3.10, PyTorch 2.1.2, and CUDA 11.8. Unless otherwise noted, we follow the official evaluation protocol and hyperparameter settings of each baseline. For CondenseVLM, the default cosine-similarity threshold is $\gamma = 0.75$, and pruning ratios are converted to the corresponding retained-token budgets for each backbone. SparseVLM results are produced from the public repository cloned on 2025-02-10. We note that the SparseVLM arXiv results changed across versions, while the repository was not updated accordingly; we therefore report our reproduced repository results for a fair code-based comparison. We set the background token proportion to 0.01/1. All results come from two runs, and we report the mean values.

C Additional Related Work

C.1 MLLMs and Their Challenges

The recent remarkable success of LLMs (Ouyang et al., 2022; Zhang et al., 2022; Touvron et al., 2023a; Dubey et al., 2024; Luo et al., 2025) has spurred the trend of applying their strong capabilities to multimodal comprehension tasks, fostering the development of MLLMs (Achiam et al., 2023; Team et al., 2023). Leveraging open-source LLMs such as LLaMA families (Touvron et al., 2023a,b; Dubey et al., 2024), MLLMs (Bai et al., 2023; Liu et al., 2024a,b) have demonstrated enhanced adaptability across a range of visual understanding tasks, leading to a more profound ability to interpret the world. While this empowers LLMs with the capability of visual perception, the incorporation of lengthy visual tokens significantly escalates the computational burdens. Moreover, studies have shown that existing MLLMs still suffer from certain visual deficiencies (Tong et al., 2024; Jiang et al., 2024b) and some hallucinations (Huang et al., 2024b, 2025). Some work mitigates these issues by increasing the resolution of input images or videos (Luo et al., 2024; Xu et al., 2024b), but this further exacerbates the computational overhead. For example, LLaVA-1.5 (Liu et al., 2024c) encodes a 336-resolution image into 576 visual tokens, while LLaVA-NeXT (Liu et al., 2024b) doubles the resolution and generates 2,880 tokens. LLaVA-OneVision (Li et al., 2024a) represents an image using 7,290 visual tokens, and Video-LLaVA (Lin et al., 2023) faces even higher costs, as it must process numerous visual tokens from multiple frames during inference. These visual tokens occupy a large portion of the context window of their LLMs. In this work, we conducted experiments on these representative models to verify CondenseVLM's applicability.

C.2 Visual Redundancy Identification

In MLLMs, visual redundancy identification facilitates the distillation of visual tokens with high informativeness for faster inference. There are two main research directions: a) Vision-centric strategies analyze the image's structure and feature

distribution to discard less relevant visual tokens (Chen et al., 2024a; Wang et al., 2024). Existing approaches include spatial-similarity clustering (*e.g.*, TokenLearner (Ryoo et al., 2021)), dynamic pruning based on attention scores (Han et al., 2024; Yang et al., 2024b; Xu et al., 2024a), and using information bottleneck or entropy metrics during the prefilling stage to estimate background redundancy. b) Instruction-centric strategies typically use cross-modal attention analysis or gradient accumulation to identify redundant tokens (Liu et al., 2024d; Zhu et al., 2024; Song et al., 2024). Tokens with low attention or negligible gradient impact are deemed redundant (He et al., 2024). Building on this, some studies explore learned importance scoring, training a lightweight end-to-end model to predict each patch's "instruction relevance," enabling even finer-grained pruning (Jiang et al., 2024a; Tu et al., 2024; Ye et al., 2025). As the existence of language bias in LLM may cause hallucinations, we use a vision-centric scheme.

C.3 Token Compression for MLLMs

The inclusion of visual information in MLLMs introduces long token sequences, leading to high computation and memory costs. For example, mini-Gemini-HD (Li et al., 2024c) generates 2880 tokens from high-definition images, creating inference bottlenecks. To address this, research has focused on token compression and pruning techniques in Vision Transformers (Bolya et al., 2022) and MLLMs (Huang et al., 2024a). Methods like LLaMA-VID (Li et al., 2024b) and DeCo (Yao et al., 2024) address this by modifying models and adding training, which increases computational costs. ToMe (Bolya et al., 2022) reduces tokens without training but disrupts early cross-modal interactions (Xing et al., 2024). LLaVA-PruMerge (Shang et al., 2024) selectively retains key tokens while merging less critical ones based on key similarity. FasterVLM (Zhang et al., 2024a) utilizes [CLS] attention scores from the visual encoder to re-rank and retain top visual tokens. FastV (Chen et al., 2024a) and SparseVLM (Zhang et al., 2024b) focus on token selection using attention scores or cross-modal guidance, but overlook the role of token duplication and lack Flash-Attention (Dao et al., 2022; Dao, 2024). Our proposed CondenseVLM maintains hard acceleration compatibility (*e.g.*, Flash-Attention), and effectively retains visual context during aggressive pruning.

D Feature Comparison with Related Token-Reduction Methods

Table 8 summarizes the design differences between CondenseVLM and representative token-reduction methods discussed in the introduction and related work. We mark a property as ✓when it is an explicit design objective, △when it is only partially covered or emerges indirectly, and ✗when it is not a main component. The comparison is qualitative because different methods operate at different layers and under different backbones, but it clarifies the central distinction: CondenseVLM is designed as evidence-preserving condensation rather than as independent saliency ranking or unconstrained similarity merging.

IVC-Prune focuses on spatial reasoning by identifying RoPE-induced implicit visual coordinates; AgilePruner adaptively combines attention and diversity cues; and VisionTrim integrates global-local token selection with text-guided complement. CondenseVLM is complementary in motivation but different in formulation: it treats compression as a budgeted evidence-preservation problem and ties its three modules to coverage, context, and local reconstruction terms.

E Additional Experimental Results

This section reports supplementary experiments that extend the main-text evaluation. The goal is to verify that the behavior of CondenseVLM is not tied to a single backbone or benchmark family. We therefore include larger 13B backbones, grounding-oriented tasks, text-dense tasks, background-anchor allocation, and video question answering.

Table 9 evaluates CondenseVLM on LLaVA-1.5-13B and LLaVA-NeXT-13B. The results show that the advantage of evidence-preserving condensation is maintained on larger models, with CondenseVLM consistently improving average preservation over VisionZip under comparable token budgets. This supports the main-text claim that the method is not merely tuned to the 7B setting.

F Computational Complexity

We follow FastV (Chen et al., 2024a) and estimate the effect of visual-token reduction through the transformer FLOPs induced by the sequence length. For a transformer with T layers, sequence length N, hidden dimension d, and FFN intermediate di-

Table 8: Feature-level comparison with related training-free visual-token reduction methods. Unlike methods that mainly rely on a single saliency, attention, similarity, diversity, or coordinate proxy, CondenseVLM unifies marginal evidence coverage, explicit context preservation, and spatial-semantic residual aggregation.

Method	Training-free	Set-level coverage	Explicit context	Spatial constraint	Residual aggregation	Theory-guided objective	**Main pruning / condensation signal**
ToMe (Bolya et al., 2022)	✓	✗	✗	✗	✓	✗	Feature-similarity token merging
LLaVA-PruMerge (Shang et al., 2024)	✓	△	✗	✗	✓	✗	Key-token retention with similarity merging
FastV (Chen et al., 2024a)	✓	✗	✗	✗	✗	✗	LLM-layer attention score
SparseVLM (Zhang et al., 2024b)	✓	✗	△	✗	△	✗	Cross-modal attention and token recycling
FasterVLM (Zhang et al., 2024a)	✓	✗	✗	✗	✗	✗	Visual-encoder `[CLS]` saliency
VisionZip (Yang et al., 2024a)	✓	△	✗	✗	✓	✗	Important-token selection plus similarity compression
FitPrune (Ye et al., 2025)	✓	✗	✗	✗	✗	△	Attention-distribution fitting
G-Prune (Jiang et al., 2025)	✓	△	△	✗	✗	✗	Graph propagation over token similarity
DART (Wen et al., 2025)	✓	✓	✗	✗	✗	△	Duplication-aware token selection
HoloV (Zou et al., 2025a)	✓	△	✓	△	✗	✗	Holistic context retention across crops
CDPruner (Zhang et al., 2025a)	✓	✓	✗	✗	✗	△	Conditional diversity maximization
IDPruner (Tan et al., 2026)	✓	✓	✗	✗	✗	△	Importance-diversity balancing
VLM-Pruner (Wu et al., 2025)	✓	△	✗	✓	✗	△	Spatial sparsity buffering
IVC-Prune (Sun et al., 2026)	✓	△	△	✓	✗	✓	Implicit visual coordinates and foreground tokens
AgilePruner (Baek et al., 2026)	✓	✓	✗	✗	✗	△	Adaptive attention-diversity pruning
VisionTrim (Yu et al., 2026)	✓	✓	✓	△	✓	△	Global-local selection and text-guided complement
CondenseVLM (ours)	✓	✓	✓	✓	✓	✓	Marginal evidence coverage plus spatial-semantic reconstruction

mension m, the approximate FLOPs are

$$\mathrm{FLOPs}_{\mathrm{full}} = T\left(4Nd^2 + 2N^2d + 2Ndm\right). \tag{33}$$

The linear terms correspond to projections and FFN computation, while the quadratic term comes from self-attention. Since visual tokens often dominate N in MLLMs, reducing the visual prefix can reduce both attention cost and memory pressure.

If a method prunes or condenses tokens after layer L, and the resulting sequence length is $\hat{N}$, the approximate post-compression FLOPs are

$$\widehat{\mathrm{FLOPs}} = L\left(4Nd^2 + 2N^2d + 2Ndm\right) + (T-L)\left(4\hat{N}d^2 + 2\hat{N}^2d + 2\hat{N}dm\right). \tag{34}$$

The corresponding theoretical reduction ratio is

$$1 - \frac{\widehat{\mathrm{FLOPs}}}{\mathrm{FLOPs}_{\mathrm{full}}}. \tag{35}$$

CondenseVLM adds a one-time preprocessing cost for computing visual-token similarities and assignments. This cost is paid before generation and is amortized when the decoder generates multiple tokens. The benefit is most pronounced for long visual prefixes, high-resolution inputs, video inputs, or memory-limited deployment, where reducing $\hat{N}$ lowers repeated decoder attention cost.

G Additional Visualizations of CondenseVLM

This section provides qualitative evidence for the design choices analyzed in Appendix A. The visualizations are intended to show where single-proxy pruning fails and why diversity-aware selection plus spatial-semantic merging is useful.

G.1 Token-retention visualization

Figure 6 compares the retained tokens of FastV, FasterVLM, and CondenseVLM at different pruning ratios. FastV tends to preserve redundant regions because language-layer attention is affected by positional and instruction-side biases. FasterVLM is less redundant but can still concentrate heavily around locally salient regions. In contrast, CondenseVLM retains a more spatially and semantically diverse evidence set, which is consistent with the marginal-coverage objective in Eq. (1).

Table 9: Additional results on larger 13B backbones. CondenseVLM remains competitive under aggressive pruning.

Model / Method	Tokens	GQA	MMB	MME	POPE	SQA	VQA_{V2}	VQA_{Text}	VizWiz	Avg.
LLaVA-1.5-13B	576	63.2	67.7	1818	85.9	72.8	80.0	61.3	56.6	100.0%
VisionZip	192	59.6	65.9	1770	86.4	72.8	78.0	58.6	54.9	97.6%
CondenseVLM	192	59.6	65.9	1770	86.4	72.8	78.0	58.6	**56.0**	**97.8%**
VisionZip	128	57.9	66.7	1743	85.2	74.0	76.8	58.7	55.0	97.1%
CondenseVLM	128	58.9	65.5	1811	85.9	73.1	76.7	57.5	56.8	**97.7%**
VisionZip	64	56.2	64.9	1676	76.0	74.4	73.7	57.4	55.9	93.8%
CondenseVLM	64	57.8	63.8	1711	82.0	71.8	75.2	56.3	57.3	**95.0%**
LLaVA-NeXT-13B	2880	65.4	70.0	1858	86.2	73.5	81.8	64.3	64.0	100.0%
VisionZip	640	62.1	66.8	1795	85.0	70.1	78.8	59.5	58.9	96.2%
CondenseVLM	640	62.4	67.0	1817	85.6	70.1	79.1	60.2	60.2	**96.7%**
VisionZip	320	60.8	64.9	1762	81.5	69.8	75.9	57.8	57.2	94.4%
CondenseVLM	320	61.5	65.4	1810	82.7	70.3	76.9	58.5	58.5	**95.2%**
VisionZip	160	58.2	64.8	1680	75.2	68.5	72.1	55.4	54.8	91.0%
CondenseVLM	160	59.5	65.5	1715	77.8	69.1	73.6	56.7	56.7	**92.2%**

Table 10: Fine-grained comparison on MMBench (Liu et al., 2025b) between FasterVLM and CondenseVLM at high pruning ratios.

Category (dev)	Vanilla (576)	FasterVLM (↓88.9%)	Condense-VLM (↓88.9%)	FasterVLM (↓75%)	Condense-VLM (↓75%)
Action Recognition	90.7	83.3	90.9	85.2	92.0
Attribute Comparison	50.0	56.8	52.6	56.8	51.3
Attribute Recognition	79.7	78.4	74.7	78.4	78.8
Celebrity Recognition	79.8	82.8	75.4	81.2	73.7
Function Reasoning	75.9	77.2	81.1	75.9	82.1
Future Prediction	45.0	37.5	55.6	37.5	58.3
Identity Reasoning	93.3	93.3	96.3	95.6	97.6
Image Emotion	78.0	78.0	68.7	80.0	72.3
Image Quality	35.8	30.2	38.8	39.6	40.0
Image Scene	96.2	93.3	92.2	96.2	90.7
Image Style	77.4	66.0	62.4	69.8	68.2
Image Topic	83.3	86.1	94.1	83.3	95.3
Nature Relation	41.7	33.3	55.6	35.4	54.3
Object Localization	39.5	34.6	26.0	35.8	28.8
OCR	59.0	53.8	81.8	53.8	80.5
Physical Property Reasoning	50.7	50.7	55.0	50.7	57.0
Physical Relation	33.3	20.8	28.8	20.8	26.9
Social Relation	88.4	67.4	67.9	67.4	78.2
Spatial Relationship	17.8	20.0	17.3	22.2	21.0
Structured Image-Text Understanding	26.9	26.9	21.8	26.9	21.8

Table 11: MMBench L2-category fine-grained comparison between FasterVLM and CondenseVLM at high pruning ratios.

Category (dev)	Vanilla	FasterVLM (↓88.9%)	Condense-VLM (↓88.9%)	FasterVLM (↓75%)	Condense-VLM (↓75%)
Attribute Reasoning	70.4	70.9	76.4	75.4	77.8
Coarse Perception	77.4	73.6	73.2	77.0	74.9
Finegrained Perception (Cross-Instance)	55.2	55.2	54.7	56.6	55.9
Finegrained Perception (Instance-Level)	65.9	64.5	63.6	64.5	64.6
Logic Reasoning	33.1	30.5	35.8	30.5	37.0
Relation Reasoning	57.4	43.5	53.6	44.3	56.4

G.2 Background-token similarity

Figure 8 visualizes cosine similarity between a selected background token and all other tokens. In several cases, semantically different but visually repetitive regions obtain high similarity, such as sky-like and beach-like patches. Pure similarity-based merging may therefore fuse tokens that are close in feature space but incompatible in image geometry. The spatial term in Eq. (3) addresses this failure mode by preferring anchors that are both semantically similar and spatially nearby.

H Fine-grained MMBench Results

Tables 10 and 11 provide fine-grained MMBench comparisons between FasterVLM (Zhang et al., 2024a) and CondenseVLM. At both 88.9% and 75% pruning ratios, CondenseVLM improves several categories that require retaining distributed or relational visual evidence, including action recognition, function reasoning, future prediction, identity reasoning, image quality, OCR, physical-property reasoning, physical relation, and social relation. The L2-category results further show gains in attribute reasoning and relation reasoning. These results align with the proposed objective: preserving a structured evidence set is especially useful when the answer depends on multiple complementary regions rather than a single salient patch.

I Limitations

CondenseVLM is a training-free compression method and therefore depends on the quality of the frozen visual features and similarity signal. Extremely fine-grained OCR, dense counting, or small-object localization may still require larger token budgets. The current implementation also uses a fixed budget split between faithful anchors and context anchors; an adaptive split could be better for inputs whose evidence distribution differs substantially from the average case. For video inputs, the current single-shot condensation does not explicitly model temporal feedback or long-range event structure, so temporal-aware condensation remains an important direction.

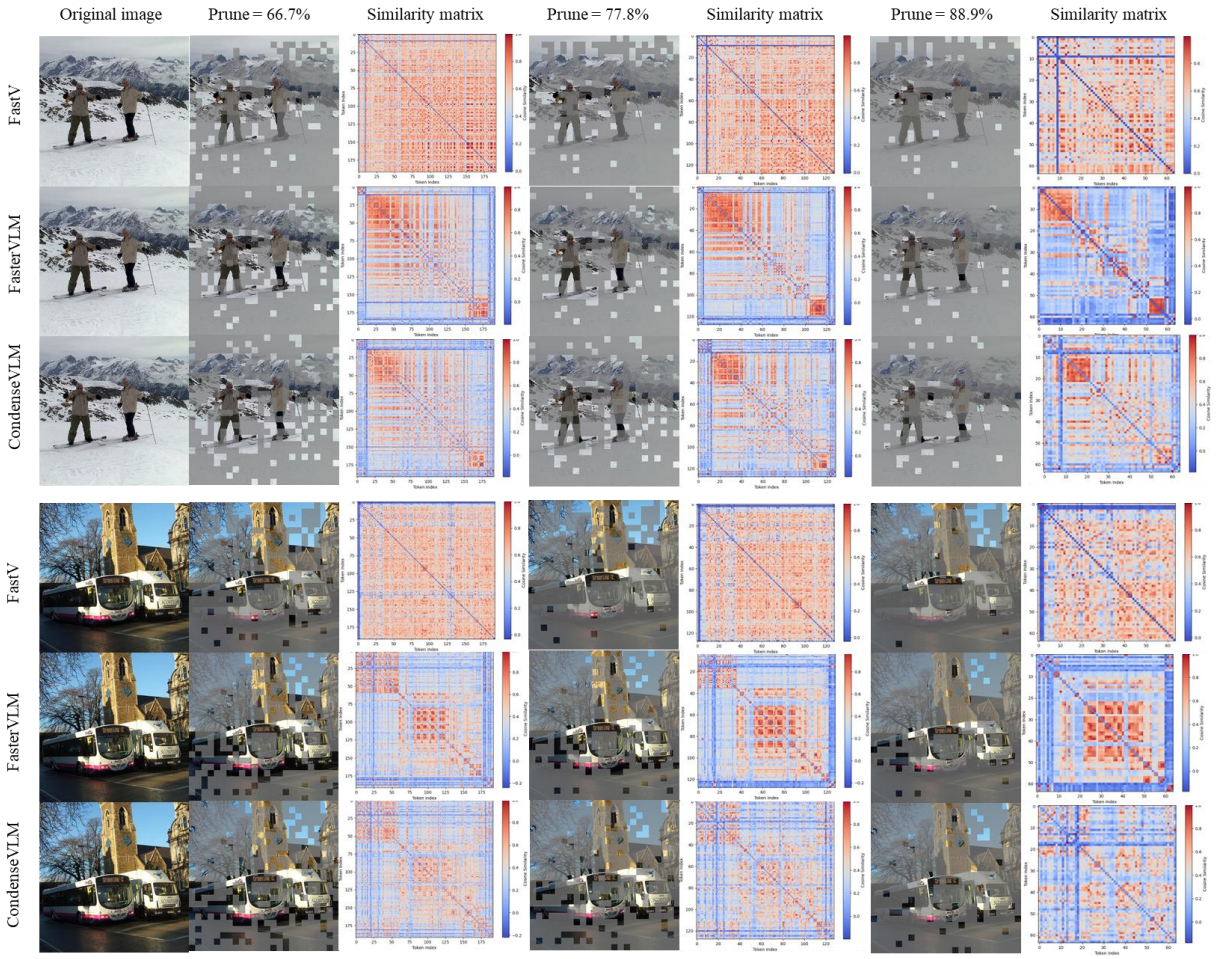


Figure 6: Qualitative comparison of retained visual tokens for FastV, FasterVLM, and CondenseVLM under pruning rates of 66.7%, 77.8%, and 88.9%.

J Future Work

The results suggest several extensions. First, adaptive token budgets could allocate more visual tokens to difficult samples and fewer tokens to simple samples. Second, temporal-aware condensation could preserve frame-level evidence and event transitions more explicitly for video MLLMs. Third, token condensation sometimes improves hallucination-sensitive performance on POPE, indicating that compression may remove distracting or redundant evidence. Studying how evidence-preserving condensation interacts with hallucination-aware decoding or post-training is a promising direction.

K Ethics Statement

This work reduces MLLM inference cost and may make multimodal systems more accessible on resource-constrained devices. It does not introduce new data collection, user profiling, or human-subject annotation. Because CondenseVLM operates on frozen MLLMs, it can inherit the biases, hallucinations, and safety limitations of the underlying model and its training data. Compression may also remove evidence needed for some safety-critical decisions, so deployment should use task-appropriate validation and conservative token budgets.

L Impact Statement

CondenseVLM aims to improve the efficiency of multimodal inference by reducing the number of visual tokens processed by the language decoder. This can lower memory usage and latency, making MLLMs easier to deploy in edge, mobile, and resource-limited environments. The method is training-free, so it can be applied to existing open-

Figure 7: Distribution of [CLS] attention across representative images. The attention mass often concentrates on a small number of regions, motivating diversity-aware evidence coverage.

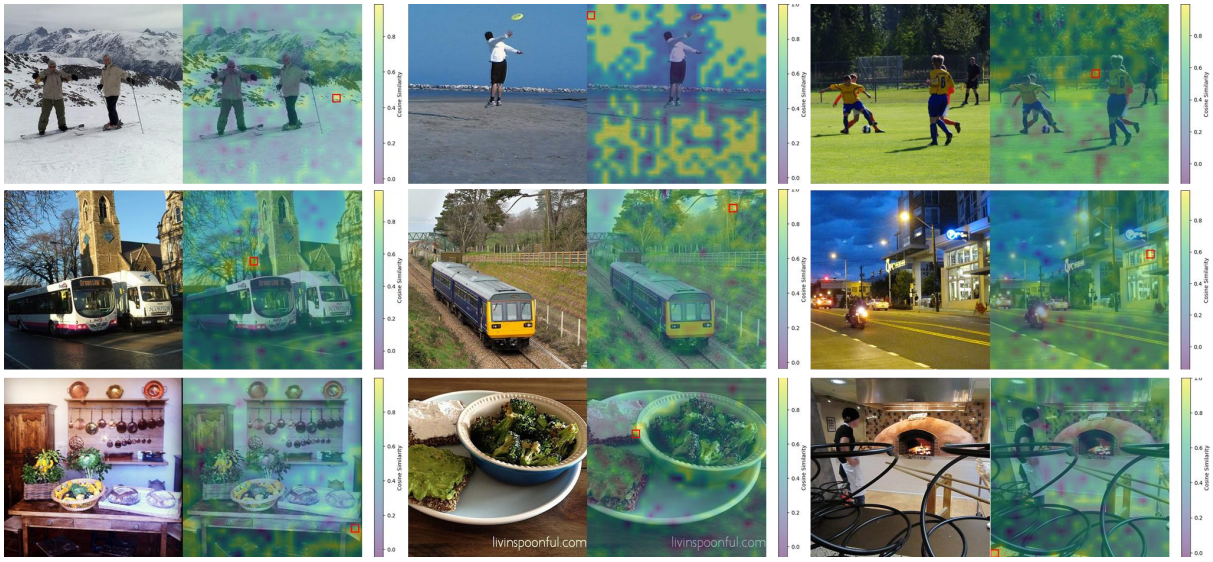


Figure 8: Similarity between selected background tokens and other visual tokens. High feature similarity can occur between spatially distant regions, motivating spatially constrained merging.

source MLLMs without additional large-scale training cost.

The main positive impact is practical accessibility: cheaper inference can broaden the use of multimodal systems in educational, assistive, and low-resource settings. The method can also contribute to more sustainable AI deployment by reducing repeated inference-time computation. At the same time, efficient deployment may increase the scale at which MLLMs are used, so standard concerns about hallucination, bias, privacy, and misuse remain relevant. Since CondenseVLM compresses visual evidence, users should evaluate it carefully for high-stakes applications where small visual details are critical.

M Artifact License and Usage

Our released artifacts are intended for research and reproducibility purposes. The implementation of CondenseVLM, including the token-condensation modules, evaluation scripts, and configuration files developed by the authors, will be released under a permissive open-source license such as MIT or Apache-2.0. The exact license file will be included in the public repository upon release.

This work does not redistribute the original model checkpoints or benchmark datasets. All evaluated MLLM backbones, datasets, and third-party baselines are used according to their respective licenses and terms of use. Users who reproduce our experiments should obtain these resources from their official sources and comply with the corresponding license requirements. In particular, our artifact only provides code and instructions for applying CondenseVLM to existing models; it does not modify the license of any upstream model, dataset, or baseline implementation.

For third-party methods compared in our experiments, we follow their publicly released implemen-

tations when available and cite the corresponding papers and repositories. Any derivative use of these baselines should follow their original licenses. The figures and tables reported in this paper are generated from our own experimental results and are included for academic dissemination.

N The Use of Large Language Models

In preparing this manuscript, we utilized GPT-5.5 and Doubao as a writing and editing assistant. Its role was limited to enhancing the clarity and fluency of the English in various sections. All scientific ideas, research methodology, experimental design, result analysis, and technical contributions are solely the product of the human authors.