ANSIGHT: ANSWER-SENSITIVE, SPATIALLY DIVERSE VISUAL TOKEN PRUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern vision-language models process hundreds to thousands of visual tokens for every query, making inference expensive in both computation and memory. Token pruning methods aim to reduce this cost, but the proxies they commonly use for token importance are not *explicitly* designed to capture the visual representations that actually matter for the model's answer. We show that the sensitivity of a VLM's answer to each visual representation can provide effective supervision for token selection. We identify Gradient $\times$ Activation as an effective label-free signal derived from the frozen model's own generated answer, and distill it into *Ansight*, a lightweight question-conditioned selector that operates before the language model. However, relevance alone can spend a small token budget on a single spatial region or on redundant evidence. We therefore combine *Ansight* with regional spatial coverage and global feature diversity, retaining visual evidence that is compact but complementary. Extensive experiments across several backbones, benchmarks, and retention budgets show that *Ansight* achieves the highest average relative performance in 13 of 15 backbone–budget settings, reducing the performance loss of the strongest prior method by 26%. On LLaVA-NeXT-7B, it retains 94.9% of full-model performance with only 5% of visual tokens, while cutting computation by 8.6$\times$, reducing KV-cache memory by 94%, and accelerating end-to-end inference by 2.7$\times$. We further show that *Ansight* transfers to long-video question answering without video-specific retraining and extends to multi-turn dialog over a fixed image.

1 INTRODUCTION

Vision-language models (VLMs) encode images/videos into hundreds or thousands of visual tokens before passing them to a language model. The cost of processing these tokens makes efficient inference challenging, particularly when computational resources are limited or visual context is long. At the same time, many questions require only a small fraction of the full visual context, suggesting that processing the entire sequence is often unnecessary (Chen et al., 2024). Visual-token pruning aims to reduce this cost by retaining a smaller subset of tokens. The central challenge is to determine which tokens can be removed without changing the frozen model's answer.

Existing visual-token pruning methods use proxy signals such as attention (Chen et al., 2024), text-guided relevance (Zhang et al., 2025c), saliency (Deng et al., 2025), or feature redundancy (Fang et al., 2026) to guide token retention. These signals can indicate visual importance, but they are not directly tied to the effect of pruning on the model's answer. Recent work has shown that explanation-derived relevance scores can also be distilled into lightweight input-stage token selectors (Lei et al., 2026). We therefore study which supervision signal best identifies the visual tokens needed to preserve the frozen model's answer.

We investigate answer attributions as a supervision signal for visual-token pruning. We assign each visual token a score based on the sensitivity of the frozen model's answer log-probability to its representation, measured by Gradient $\times$ Activation; this allows one teacher-forced backward pass to score every token at once. We distill this scoring into *Ansight* (**Ans**wer + hind**sight**), a small question-conditioned network that predicts relevance before the language model. Further, we show that the model's own full-image answers produce targets that closely match those from ground truth (GT) answers, removing the need for answer labels. At inference, Ansight requires neither gradients nor additional VLM evaluations.

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

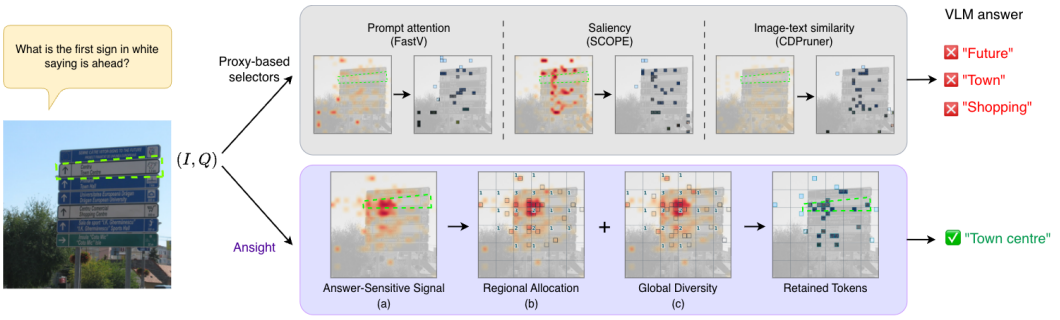


Figure 1: **Answer-sensitive visual-token selection.** Proxy-based selectors retain evidence that can miss the queried content. Ansight combines (a) answer-sensitive relevance with (b) regional allocation, with numbers indicating the tokens assigned to each spatial region, and (c) global feature diversity to retain compact, complementary evidence needed for the answer.

Predicted relevance (Fig. 1(a)) ranks the tokens but, by itself, is not a sufficient signal, especially for low-budget regimes. Retaining the highest-scoring tokens can concentrate a small budget on a single spatial region or redundant evidence. We therefore introduce two selection rules. *Regional Allocation* (RA) partitions the token grid into fixed cells and gives each cell a share of the budget proportional to the predicted relevance mass it holds (Fig. 1(b)), spreading the budget spatially without adding learned parameters. *Region-Constrained Global Diversity* (RCGD) keeps those per-cell counts and replaces independent ranking within each cell by one global greedy selection (Fig. 1(c)), admitting a token only when it is both relevant and far from the tokens already selected. The first rule decides where the budget is spent and the second what within it is redundant. An extensive set of ablations illustrates that relevance, region, and diversity are distinct and useful signals.

Across five backbones, six benchmarks, and three retention budgets, Ansight ranks first or second in 79 of 90 matched settings and achieves the highest per-backbone, per-budget average in 13 of 15 comparisons, improving over the strongest comparator by up to 5.4 pp. Ansight also transfers beyond single-image question answering. In multi-turn dialog, 25% of the retained token budget can be reused across questions while matching fully question-conditioned selection. Without retraining, Ansight also compresses long-video context by more than $20\times$ while preserving 81.8% of full-context answers. These reductions translate into substantial efficiency gains. At 5% retention, physical token compaction reduces counted per-query computation by 7.3–$8.6\times$ on the LLaVA backbones and 1.9–$2.8\times$ on the Qwen backbones, and KV-cache memory by 86–94%. End-to-end speedups reach $2.81\times$ on high-resolution inputs, and per-question speedups reach $4.34\times$ at 128 frames with cached video features. We make three main contributions:

- **Answer-attribution, label-free supervision.** We formulate visual-token selection based on the principle of preserving the frozen VLM's full-image answer and use Gradient $\times$ Activation of the answer log-probability as token-level supervision. Constructing targets from the model's own predictions removes the need for human answer labels, while distillation yields a lightweight question-conditioned feed-forward scorer used in inference.

- **Region-aware, diversity-constrained selection.** We introduce Regional Allocation to distribute a fixed token budget across spatial regions according to predicted answer-relevance mass, and region-constrained global diversity to select complementary evidence while preserving those regional budgets exactly. This separates where the token budget is allocated from which tokens are redundant within that allocation.

- **Transfer and reuse across visual contexts.** We show that our answer-sensitive selector can be transferred to broader applications such as long-video question answering without video-specific retraining, and, within a multi-turn dialog, can speed up each turn while preserving and reusing shared visual context across questions.

2 Related Work

Visual-token reduction. Visual-token reduction methods differ in both their selection signal and pruning stage. FastV (Chen et al., 2024) uses early language-model attention, while SparseVLM (Zhang et al., 2025c) combines text-guided sparsification with token recycling. Other methods reduce redundancy through token pruning, merging, or clustering (Shang et al., 2025; Yang et al., 2025b; Dhouib et al., 2025; Zhang et al., 2025a). TopV (Yang et al., 2025a) avoids materialized language-model attention maps, QuietPrune (Gao et al., 2026) introduces query guidance within the vision encoder, and ZOO-Prune (Kim et al., 2026) combines projection-stage zeroth-order sensitivity with feature diversity. Adaptive-budget methods additionally determine how many tokens to retain (Ye et al., 2025; Zeng et al., 2026; Li et al., 2026; He et al., 2026b); we instead use fixed budgets to isolate the quality of the selected subset.

Learned token selection. Learned selectors either distill token-relevance signals or optimize selection through a downstream objective. Lei et al. (2026) distill response-conditioned relevance derived from gradient-weighted attention propagation, while MAP (Sun et al., 2026) distills question-responsive middle-layer attention into a pre-language-model predictor. VisionSelector (**?**) and AutoSelect (He et al., 2026a) instead learn selection through differentiable mechanisms tied to the frozen VLM. Ansight differs by distilling Gradient $\times$ Activation of the full-answer log-probability and predicting relevance entirely from pre-language-model features, without language-model attention maps or gradients at inference. Using the model's own prediction to construct these targets is related to distillation and pseudo-labeling (Hinton et al., 2015; Xie et al., 2020).

Diversity and spatial coverage. Independent token ranking can retain redundant evidence under small budgets. DivPrune (Alvar et al., 2025) and CDPruner (Zhang et al., 2025b) explicitly promote diversity, while SCOPE (Deng et al., 2025), CoIn (Du et al., 2026), SCoRe (Xu et al., 2026b), TOPS (Wang et al., 2026a), and MMTok (Dong et al., 2026) combine relevance or informativeness with coverage. HoloV (Zou et al., 2025) distributes the pruning budget across spatial crops to preserve holistic visual context, while PruneSID (Fang et al., 2026) and RoRA (Lu et al., 2026) organize visual evidence into semantic groups or components before suppressing redundancy. Our approach instead combines answer-sensitive relevance with explicit spatial allocation and enforces the resulting regional budgets during global diversity selection.

Repeated queries and long visual contexts. Visual-token reduction is particularly valuable when visual context is reused across queries or grows over long videos. MetaCompress (Wang et al., 2026b) learns prompt-agnostic compression for future questions in multi-turn VQA, while SparseVILA (Khaki et al., 2025) uses query-aware visual sparsity to support multi-turn and long-context inference. For long video, FLoC (Cho et al., 2026) selects representative and diverse tokens through query-agnostic facility-location optimization, whereas KVTP (Liu et al., 2025) adapts pruning according to query-relevant frames. VLCache and Kamera (Qin et al., 2025; Ma et al., 2026) instead reuse visual computation or multimodal KV state across queries. We study whether an answer-sensitive selector trained on images can transfer directly to long video and whether part of the selected visual evidence can be reused across questions about the same image.

3 Method

We train a lightweight question-conditioned selector to preserve a frozen VLM's full-image answer. Offline, we construct answer-attribution targets from self-generated answers. At inference, Ansight predicts token relevance before the language model, regional allocation assigns regional budgets, and region-constrained global diversity selects complementary tokens subject to those budgets before token compaction. Figure 2 summarizes the offline and inference paths.

3.1 Problem Formulation

Given an image I and question q, a VLM with parameters θ produces visual representations $V = (v_1, \ldots, v_N)$, with $v_i \in \mathbb{R}^d$ in the language model's embedding space. Its full-image answer is $y^\star = \mathrm{Decode}_\theta(V, q)$. For a budget $k < N$, we seek a subset $S \subseteq \{1, \ldots, N\}$ with $|S| = k$ such that $\mathrm{Match}(\mathrm{Decode}_\theta(V_S, q), y^\star) = 1$. Here, V_S contains only the retained subset of visual tokens, and Match is the evaluation-specific answer-matching rule (Section 4.1). Preservation measures agreement with the frozen model, not correctness against a human answer. Direct subset optimization

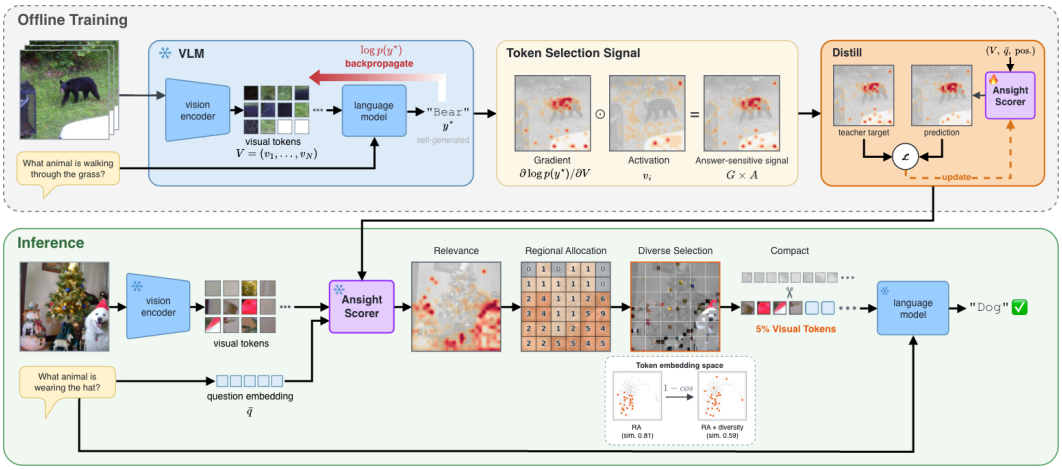


Figure 2: **Ansight training and inference.** Offline (top), the frozen VLM generates an answer, and a teacher-forced backward pass produces Gradient × Activation targets for distilling the lightweight scorer. At inference (bottom), Ansight predicts token relevance before the language model, RA allocates the token budget across spatial regions, and RCGD selects complementary tokens subject to those regional budgets. The retained tokens are then compacted before answer generation.

requires repeated VLM evaluations, so we instead distill per-token answer-attribution targets into a one-pass selector.

3.2 Answer-Sensitive Supervision and Distillation

Ansight is trained offline to imitate an attribution signal that measures how strongly the frozen model's own answer depends on each visual token. For an answer $y = (y_1, \ldots, y_T)$, let $\mathcal{J}(y; V, q) = \sum_{t=1}^{T} \log p_\theta(y_t \mid V, q, y_{<t})$ be its teacher-forced log-probability, summed over completion tokens only. We score each token by Gradient × Activation ($G \times A$):

$$a_i = \left\| v_i \odot \frac{\partial \mathcal{J}(y; V, q)}{\partial v_i} \right\|_1, \tag{1}$$

where $\odot$ is elementwise multiplication. Gradient × Activation weights the answer's sensitivity to each feature dimension by that dimension's value, while the ℓ_1 norm aggregates these contributions irrespective of sign.

Self-generated answers as targets. We decode the full-image answer $y^\star$ and teacher-force it to evaluate Eq. 1 with $y = y^\star$, keeping all parameters frozen (backbone-specific differentiation paths are in Appendix A.1). This aligns supervision with exactly the answer that pruning must preserve and requires no human answer labels. One backward pass gives relevance targets for all N tokens.

Scorer and distillation objective. We train a lightweight MLP, shared across visual tokens, to predict their answer relevance before the first language-model layer. For token i, the scorer produces $s_i = g_\phi(x_i)$ from the input

$$x_i = [v_i,\ \bar{q},\ \mathrm{LN}(v_i) \odot \mathrm{LN}(\bar{q}),\ \psi(p_i)],$$

where $\bar{q}$ is the mean input embedding of the question's non-special tokens, supplying task context; LN is feature-wise layer normalization, enabling the interaction term to relate the question to each visual token; and $\psi(p_i)$ is a fixed Fourier encoding that supplies spatial layout. Together with the visual token's image content, these features are available before the first language-model layer, without requiring hidden states or attention maps. Because raw attribution magnitudes vary across examples, we supervise relative importance within each example. With $u_i = \log(1 + a_i)$ and $\pi_i = u_i / (\sum_j u_j + \epsilon)$, the objective is

$$\mathcal{L} = \underbrace{-\sum_i \pi_i \log \frac{\exp(s_i)}{\sum_j \exp(s_j)}}_{\mathcal{L}_{\mathrm{rank}}} + \lambda\, \mathrm{SmoothL1}(z(s), z(u)), \tag{2}$$

where $z(\cdot)$ standardizes scores over the example's valid visual tokens, $\lambda = 0.25$, and $\epsilon = 10^{-6}$. The cross-entropy term prioritizes high-attribution tokens for global top-k selection, while the Smooth-L1 term preserves relative score magnitudes used by regional allocation (Section 3.3). Architecture, normalization, and loss safeguards are detailed in Appendix A.2, with ablations in Appendix E. At inference, Ansight scores all visual tokens in one pass and retains the k tokens with the highest predicted relevance.

3.3 Region-Aware Diverse Selection

Regional Allocation (RA). Global top-k can concentrate a small budget on redundant evidence in one region while omitting complementary evidence elsewhere. Regional Allocation uses aggregate relevance within spatial regions to distribute the token budget before selecting tokens. We partition the visual-token grid into fixed, non-overlapping 4×4 spatial cells and define

$$m_i = \frac{\exp(s_i/\tau)}{\sum_j \exp(s_j/\tau)}, \qquad M_c = \sum_{i \in c} m_i, \qquad \tilde{k}_c = kM_c,$$

with $\tau = 1$. We initialize each cell's allocation as $k_c = \min(|c|, \lfloor \tilde{k}_c \rfloor)$, where $|c|$ is its token count, then distribute the remaining slots in rounds to cells with available capacity and the largest allocation deficits $\tilde{k}_c - k_c$, until $\sum_c k_c = k$. Within each cell, the RA variant retains the k_c highest-scoring tokens. Exact redistribution is given in Appendix A.3.

Unlike uniform allocation, RA adapts the spatial budget to the image and question, and low-relevance cells may receive no tokens. It requires only predicted relevance scores and token coordinates, with no additional learned parameters, VLM evaluations, or pairwise feature comparisons.

Region-Constrained Global Diversity (RCGD). RA determines how many tokens each cell contributes, but independent within-cell ranking can still retain mutually redundant tokens. We therefore keep the regional budgets fixed while selecting tokens jointly across all cells. Let w_i denote the min–max-normalized Ansight relevance and z_i the corresponding visual representation in the language model's embedding space. We define the distance from token i to the current selected set S as

$$d(i, S) = \begin{cases} 1, & S = \emptyset, \\ \min\limits_{j \in S} \left(1 - \cos(z_i, z_j)\right), & \text{otherwise.} \end{cases}$$

Starting from $S = \emptyset$, we repeatedly select

$$i^\star = \underset{i \notin S,\ |S \cap c(i)| < k_{c(i)}}{\arg\max} \; w_i \, d(i, S). \tag{3}$$

The first step selects the most relevant token, while subsequent steps balance Ansight relevance with distance from the selected set. Once a cell reaches its allocated budget, its remaining tokens become ineligible. Selection is therefore global and set-aware while satisfying every regional count exactly. This capacity constraint distinguishes our method from unconstrained relevance–diversity selection, including ZOO-Prune (Kim et al., 2026). For all selection rules, we physically compact the retained visual tokens before the language model. Backbone-specific details are given in Appendix A.3.

4 Experiments

We compare Ansight with existing pruning methods at matched budgets (Section 4.2), test it on multi-turn dialog, long video, and high-resolution documents (Section 4.5), measure its speed and memory savings (Section 4.3), and ablate the supervision target and selection methods (Section 4.4).

4.1 Experimental Setup

Models and training. We evaluate Ansight on five frozen VLMs: LLaVA-1.5-7B (Liu et al., 2024a), Qwen3-VL-2B (Bai et al., 2025a), Qwen2.5-VL-7B (Bai et al., 2025b), LLaVA-NeXT-7B (Liu et al., 2024b), and LLaVA-OneVision-7B (Li et al., 2025). The Qwen models use native dynamic resolution, while LLaVA-NeXT and OneVision provide high-resolution inputs. We train one selector per backbone on DeFacto (Xu et al., 2026a) image–question pairs using attributions from the backbone's own full-image answers, without benchmark answer labels. Each selector is evaluated at 5%, 11%, and 22% visual-token retention using global top-k, RA, and RCGD.

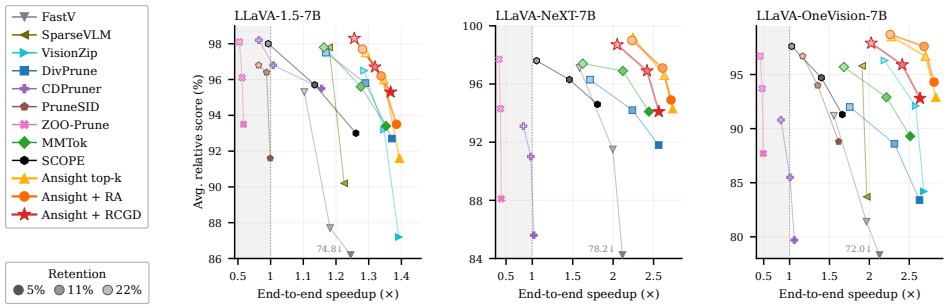


Figure 3: **Quality–efficiency trade-off on LLaVA-family backbones.** Each point shows the mean relative score across six benchmarks versus end-to-end speedup, including token-selection overhead. Connected points represent 5%, 11%, and 22% retention, from darkest to lightest. The vertical dotted line marks full-model latency (1× speedup).

Evaluation. Our main comparison covers six benchmarks: MMBench (Liu et al., 2024c), POPE (Li et al., 2023), GQA (Hudson & Manning, 2019), MME (Fu et al., 2025a), ScienceQA-IMG (Lu et al., 2022), and TextVQA (Singh et al., 2019). We compare against random selection and nine published pruning methods: FastV (Chen et al., 2024), SparseVLM (Zhang et al., 2025c), VisionZip (Yang et al., 2025b), PruneSID (Fang et al., 2026), DivPrune (Alvar et al., 2025), CDPruner (Zhang et al., 2025b), MMTok (Dong et al., 2026), SCOPE (Deng et al., 2025), and ZOO-Prune (Kim et al., 2026). We report benchmark scores along with answer preservation (the fraction of pruned answers matching the frozen model's full-token answer). Efficiency measurements include counted FLOPs, KV-cache memory, end-to-end speedup, and selector time. Additional evaluations cover Video-MME (Fu et al., 2025b), Visual Dialog (Das et al., 2017), and DocVQA (Mathew et al., 2021). Detailed protocols and paired 95% bootstrap procedures are given in Appendix B.

4.2 Main Results

Across five VLM backbones and six benchmarks, Ansight shows strong relative performance under aggressive visual-token compression. At 5% retention, Ansight + RCGD exceeds the strongest competing methods by 5.4 and 4.5 pp in average relative benchmark performance on Qwen3-VL and Qwen2.5-VL, respectively, while Ansight + RA outperforms SCOPE by 3.0 pp on LLaVA-OneVision. The margins narrow as retention budget increases. Answer-preservation results follow a similar trend. Table 1 provides the per-benchmark breakdown for LLaVA-OneVision, with corresponding results for the other four backbones in Appendix C (Tables 6–9) and the overall performance summary in Table 5.

Ansight combines strong performance with efficient inference on high-resolution backbones. On LLaVA-NeXT, it achieves both the highest average relative performance and the fastest end-to-end inference at all three retention budgets. On LLaVA-OneVision, it achieves the highest average relative performance while accelerating inference by up to 2.81× (Figure 3).

4.3 Inference Efficiency

Memory and computation savings. Physical token compaction shortens the sequence processed by the language model, reducing computation and KV-cache memory. At 5% retention, Ansight cuts counted per-query FLOPs by 86–88% on the three LLaVA backbones and 48–64% on the two Qwen backbones, which have shorter visual sequences (Appendix Tables 6–9). Across all five backbones, KV-cache memory falls by 86–94% and incremental peak memory by 51–82% (Appendix Table 12). The smaller memory footprint also improves batch capacity and throughput. At 11% retention, Ansight increases LLaVA-1.5's batch capacity by at least 2.17× on an H100 (118 to 256), and improves batch-32 throughput by 1.65× on H100 and 2.39× on L40S (Appendix Table 13).

End-to-end latency. Memory reduction alone does not guarantee faster inference, as vision encoding and decoding remain largely unchanged while selection adds overhead. On LLaVA-1.5 and the two Qwen backbones, prefill accounts for only 26.0–42.2% of full-query latency (Appendix Table 10), limiting the benefit of shorter visual sequences. At 5% retention on an L40S, LLaVA-1.5 achieves 1.37× speedup and Qwen2.5-VL 1.06×, while Qwen3-VL incurs a slight slow-

Table 1: **Quality and inference efficiency on LLaVA-OneVision-7B.** Benchmark cells report raw scores; Avg. rel. is the mean of the six benchmark scores normalized by their corresponding full-model scores. **Best** and **second-best** pruned quality scores are highlighted. Efficiency cells are shaded from **most** to **least** efficient within each retention block.

Method	MMB ↑	POPE F1 ↑	GQA ↑	MME ↑	SQA-IMG ↑	TextVQA ↑	Avg. rel. (%) ↑	E2E speed (×) ↑	Selector (ms) ↓
Full	**85.54**	**88.38**	**64.13**	**1965.9**	**95.93**	**73.71**	**100.0**	1.000	–
5% retention (median 135 of 2,709 visual tokens)									
Random	80.83	80.72	58.63	1626.3	83.99	50.61	86.0	–	–
FastV (ECCV'24)	72.72	57.67	48.88	1275.3	78.19	43.40	72.0	2.127	15.90
DivPrune (CVPR'25)	77.89	81.68	54.87	1511.1	81.66	50.97	83.4	2.629	10.53
CDPruner (NeurIPS'25)	77.27	75.56	53.37	1271.6	81.31	51.41	79.7	1.059	197.29
PruneSID (ICLR'26)	81.98	82.86	57.50	1718.3	86.76	55.83	88.8	1.615	36.06
ZOO-Prune (CVPR'26)	81.15	82.52	58.50	1631.4	85.32	55.31	87.7	0.492	533.15
MMTok (ICLR'26)	**82.61**	83.78	61.00	1653.5	84.98	56.28	89.3	2.512	16.34
SCOPE (NeurIPS'25)	82.54	84.09	**61.38**	1761.6	87.26	58.90	91.3	1.661	30.99
VisionZip (CVPR'25)	78.63	78.32	56.13	1601.5	83.54	50.52	84.2	2.683	8.84
Ansight top-k	82.56	86.02	59.75	1762.5	**88.65**	**64.94**	**92.9**	**2.834**	**2.57**
Ansight + RA	**83.92**	**86.51**	**62.88**	**1809.5**	**88.89**	**64.24**	**94.3**	2.811	3.49
Ansight + RCGD	82.44	**87.51**	60.25	**1778.8**	88.50	62.40	92.8	2.635	9.13
11% retention (median 298 of 2,709)									
Random	83.39	84.59	60.50	1748.7	87.36	58.42	91.1	–	–
FastV (ECCV'24)	78.45	71.41	52.63	1559.3	81.76	50.86	81.4	1.964	16.11
SparseVLM (ICML'25)	79.76	71.66	52.50	1653.0	83.49	55.36	83.7	1.968	30.52
DivPrune (CVPR'25)	81.68	85.11	58.50	1635.3	85.18	56.28	88.6	2.314	21.52
CDPruner (NeurIPS'25)	81.47	81.91	56.13	1443.4	83.39	56.77	85.5	1.006	209.33
PruneSID (ICLR'26)	83.76	86.79	61.38	1881.5	87.80	62.79	94.0	1.353	67.63
ZOO-Prune (CVPR'26)	83.92	**88.45**	62.13	1755.9	89.04	62.49	93.7	0.472	558.23
MMTok (ICLR'26)	**84.15**	87.22	62.00	1746.5	87.95	61.46	92.9	2.214	28.14
SCOPE (NeurIPS'25)	83.88	86.70	61.63	1866.1	88.84	65.38	94.7	1.399	60.51
VisionZip (CVPR'25)	82.74	84.44	61.25	1778.5	87.85	60.81	92.1	2.580	8.59
Ansight top-k	84.11	88.22	**63.00**	**1897.1**	90.43	**68.65**	**96.7**	**2.702**	**2.55**
Ansight + RA	**84.22**	**88.45**	**64.50**	**1928.8**	**91.52**	**68.60**	**97.6**	2.686	3.44
Ansight + RCGD	84.06	**89.16**	62.88	1827.7	**91.08**	66.41	95.9	2.412	11.48
22% retention (median 596 of 2,709)									
Random	84.66	86.90	62.38	1823.7	89.69	64.18	94.7	–	–
FastV (ECCV'24)	82.10	80.66	58.88	1845.8	86.96	61.65	91.2	1.556	16.14
SparseVLM (ICML'25)	84.78	85.16	59.88	1907.9	90.18	69.84	95.8	1.909	16.12
DivPrune (CVPR'25)	83.71	87.47	58.88	1746.4	87.85	61.34	92.0	1.754	42.42
CDPruner (NeurIPS'25)	82.47	86.19	58.50	1643.0	87.65	62.49	90.8	0.833	257.54
PruneSID (ICLR'26)	84.11	88.16	62.75	1929.1	91.03	67.22	96.7	1.164	87.27
ZOO-Prune (CVPR'26)	84.66	**89.37**	63.25	1864.5	92.22	66.66	96.7	0.432	599.00
MMTok (ICLR'26)	84.75	87.95	63.25	1873.3	89.69	65.23	95.7	1.681	51.34
SCOPE (NeurIPS'25)	84.64	88.79	**64.13**	1930.6	91.52	68.16	97.6	1.027	117.04
VisionZip (CVPR'25)	84.43	87.47	62.63	1931.1	90.23	66.62	96.3	2.189	8.65
Ansight top-k	**85.05**	89.14	64.00	**1933.3**	92.46	**71.00**	**98.5**	**2.278**	**2.57**
Ansight + RA	**85.17**	89.05	63.88	**1950.6**	**92.76**	**70.98**	**98.7**	2.261	3.48
Ansight + RCGD	84.66	**89.54**	**64.25**	1880.7	**92.96**	69.61	97.9	2.023	15.22

down (0.98×). For the longer visual sequences of LLaVA-NeXT and LLaVA-OneVision, Ansight achieves end-to-end speedups of 2.72× and 2.81×, respectively, including selection overhead.

Selection cost and context length. Table 2 compares retained quality and end-to-end efficiency at matched token budgets. Ansight + RA combines high retained quality with low selection cost on both backbones. On LLaVA-NeXT, it achieves 94.9% average relative performance and a 2.716× speedup with a 3.2 ms selector, compared with 94.6%, 1.806×, and 24.7 ms for SCOPE. Top-k offers marginally greater speedup at lower quality, whereas the more expensive selectors of CDPruner and ZOO-Prune make both methods slower than the full model.

Figure 4 shows how the advantage of compaction grows with visual context length. On LLaVA-NeXT, speedup increases from 2.38× to 2.90× as the median visual sequence grows from 1,512 to 2,340 tokens. The cached-video setting in panel (b) uses a narrower per-question timing boundary, illustrating the additional savings possible when video encoding is reused across questions. At high

Table 2: **Quality and inference efficiency at 5% retention.** Results on LLaVA-NeXT-7B and Qwen2.5-VL-7B. Avg. rel. is the six-benchmark mean relative to the full model, shaded in **green** by rank within each backbone, darker is better. Efficiency is measured on POPE using an L40S GPU, with cells shaded from **most** to **least** efficient within each backbone and metric. TTFT is time to first token; speedups are paired ratios and latencies are marginal medians.

Method	Avg. rel. (%)↑		Selector (ms)↓		E2E (ms)↓		TTFT (ms)↓		Speedup (×)↑		Peak (MB)↓	
	NeXT	Qwen2.5	NeXT	Qwen2.5	NeXT	Qwen2.5	NeXT	Qwen2.5	NeXT	Qwen2.5	NeXT	Qwen2.5
Full model	100.0	100.0	–	–	249.5	130.8	222.5	105.8	1.000	1.000	1,566	140
DivPrune	91.8	81.0	8.5	**1.5**	93.3	**124.4**	69.8	**99.6**	2.563	**1.070**	295	**68**
CDPruner	85.6	74.7	156.1	36.5	240.1	159.9	216.6	135.0	1.019	0.822	**268**	**68**
ZOO-Prune	88.1	81.0	513.4	124.3	602.3	250.0	578.6	225.1	0.411	0.546	1,821	366
MMTok	94.1	80.5	13.5	3.4	97.9	126.4	74.6	101.6	2.442	1.054	**268**	**68**
SCOPE	94.6	81.3	24.7	5.4	134.0	128.5	110.5	103.6	1.806	1.036	1,593	109
VisionZip	90.5	79.4	6.8	2.3	91.5	125.3	68.0	100.4	2.613	1.062	286	155
Ansight top-k	94.3	76.4	**2.4**	1.7	**87.1**	124.8	**63.6**	100.0	**2.737**	1.067	312	**68**
Ansight + RA	**94.9**	82.0	3.2	2.2	87.9	125.2	64.4	100.3	2.716	1.063	312	**68**
Ansight + RCGD	94.1	**85.8**	7.8	2.7	92.8	125.7	69.3	100.9	2.565	1.059	322	**68**

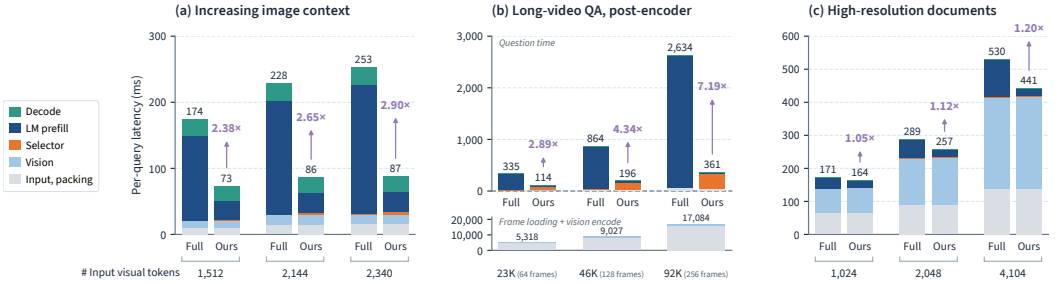


Figure 4: **Per-query latency as visual context grows.** Full-model and Ansight latency on **(a)** LLaVA-NeXT images (5% retention), **(b)** Qwen3-VL long videos (2,048 tokens retained), and **(c)** Qwen2.5-VL documents (512 tokens retained). Bar labels show median ms; purple annotations show speedups computed from medians in (a) and (c) and as paired per-question ratios in (b). In (b), frame loading and vision encoding are shown separately (bottom) and excluded from question time.

resolution, Ansight achieves a 70% larger speedup and nearly twice the KV-cache reduction of the low-resolution setting while retaining 89.5% of full-model quality (Appendix G.3).

4.4 Ablation Studies

Supervision signal. We compare relevance targets while holding the selector architecture, training pairs, and selection rule fixed (Table 3). On LLaVA-1.5 GQA, answer-gradient targets generally preserve more full-model answers than attention- and perturbation-based targets. At 5% retention, $G \times A$ achieves 74.92% answer preservation, compared with 71.71% for attention rollout and 66.25% for local occlusion. The consistently strong performance of gradient-based targets supports answer-sensitive supervision across attribution variants. Additional teacher comparisons and backbone-dependent outcomes are reported in Appendix E.

Answer source. We test whether supervision requires GT answers by training selectors with $G \times A$ targets derived from either the frozen model's own answers or GT answers. The $G \times A$ (self) and $G \times A$ (GT) rows in Table 3 show similar preservation across budgets (74.92% vs. 71.17% at 5% retention), with mean Spearman correlation 0.92 on their attribution maps (Appendix E). Ansight therefore can be trained in a self-supervised manner without human-provided answer labels.

Token selection rule. We fix the predicted relevance scores and vary only the selection rule to assess regional allocation and diversity (Table 4). At 5% retention, RA improves answer preservation over global top-k on all five backbones. Adding diversity within the regional quotas further improves preservation only on LLaVA-1.5 and Qwen2.5-VL. RA thus provides a consistent gain in this comparison, while RCGD can recover further gains when spatially distributed tokens remain redundant in feature space. Sensitivity to cell size and allocation temperature is examined in Appendix F.

Table 3: **Supervision-signal ablation.** Mean answer preservation / relative accuracy (%) on GQA over three seeds, using top-k selection without RA or diversity. Relative accuracy is normalized by the full model's accuracy. See Appendix E for definitions, full set of signals and results.

Kind	Supervision signal	5%	11%	22%
Answer gradient	$G \times A$ (self)	**74.92 / 89.30**	**78.46 / 93.43**	82.25 / 95.93
	$G \times A$ (GT)	71.17 / 85.84	78.33 / 92.75	**82.50 / 96.47**
Attention	Attention rollout (Lei et al.)	71.71 / 86.52	76.42 / 90.18	82.00 / 94.99
	Prompt attention (FastV, SparseVLM)	61.33 / 78.18	70.29 / 85.71	76.96 / 92.41
Perturbation	Local 2×2 occlusion (ZOO-Prune)	66.25 / 80.49	70.12 / 84.62	75.29 / 88.89
Feature magnitude	Activation norm	50.67 / 67.35	50.83 / 67.76	57.46 / 75.27
Similarity	CLIP question similarity (CDPruner)	71.67 / 87.93	76.42 / 93.02	80.25 / 95.33

Table 4: **Selection-rule ablation on MMBench.** Full-answer preservation (%) at 5% retention using identical predicted relevance scores. Full results are in Appendix Table 23.

Selection method	Region	Diversity	LLaVA-1.5	Qwen3-VL	Qwen2.5-VL	LLaVA-NeXT	LLaVA-OneVision
Global top-k	–	–	80.55	79.26	72.14	87.62	89.51
RA	Yes	–	83.90	**81.89**	75.79	**89.91**	**91.38**
Global Diversity	–	Yes	85.03	81.59	80.71	89.77	88.91
RCGD	Yes	Yes	**85.08**	81.03	**81.34**	89.56	90.37

4.5 Broader Applications

Repeated-image queries. When several questions refer to the same image, some selected tokens can be reused. On 200 held-out Visual Dialog dialogs (2,000 turns), selecting tokens for the current question preserves 6.75, 4.25, and 3.35 pp more answers than image-only selection at 5%, 11%, and 22% retention. We therefore reuse an image-dependent core for 25% of the token budget and select the remaining tokens for each question. This preserves 86.80%, 89.10%, and 91.05% of full-model answers, within the tested one-point non-inferiority margin of fully question-conditioned selection at every budget (Table 25). All settings use the same dialog history; only visual-token selection is reused, not the language model's KV cache.

Long-video question answering. We also apply the image-trained Qwen3-VL selector to long videos without retraining. On 900 Video-MME questions, it reduces an average of 42,565 visual tokens to 2,048 and preserves 81.78% of full-context answers. Accuracy is 48.78%, compared with 51.22% for the full model. Preservation is 6.00 pp higher than uniform temporal selection and 6.67–9.22 pp higher than the CDPruner, FastV, and ZOO-Prune ports; the difference from DivPrune is inconclusive (Table 26). Fixed-query selection is equally good, so the gain comes from the learned relevance rather than from conditioning on the current question.

Video inference cost. Longer videos also offer larger savings when their visual features are reused across questions. On a separate set of 64 videos, retaining 2,048 tokens cuts per-question latency from 864.1 to 196.2 ms at 128 frames ($4.34\times$), while reducing the KV cache from 5.00 to 0.30 GB. Speedup increases from $2.89\times$ at 64 frames to $7.19\times$ at 256 frames (Table 27)[1].

High-resolution documents. At higher resolution, a fixed budget buys more speed but drops more answers. Growing the Qwen2.5-VL DocVQA input from 1,024 to 4,104 visual tokens while keeping 512 raises the speedup of RA from $1.30\times$ to $4.22\times$, but lowers preservation from 78.9% to 58.6%. RCGD recovers 7.0–14.0 pp of preservation over RA, at a $3.25\times$ speedup (Appendix G.3).

5 Conclusion

We introduced Ansight, a visual-token selector distilled from $G \times A$ attributions of the frozen model's own answers, together with two selection rules that spread the token budget over the image (RA) and remove redundant evidence within it (RCGD). Across five backbones, six benchmarks, and three budgets, Ansight is the best or second-best pruning method in 79 of 90 settings, with the largest gains at 5% retention, and it transfers without retraining to long video and extends to multi-turn dialog.

[1]These measurements start from cached video features and generate one answer token; they are not end-to-end video-processing times.

AI USE STATEMENT

We used large language model (LLM) assistants to edit the manuscript for grammar, clarity, and concision; to support the literature search by locating and summarizing related work; and to assist with writing and debugging experiment, analysis, and plotting code. The authors checked every cited work against its original source, reviewed and tested all AI-assisted code, and verified all reported numbers against the experiment logs. The authors take full responsibility for the content of this paper, including any text, references, or code produced with AI assistance.

REPRODUCIBILITY STATEMENT

Section 3 defines the attribution target, selector inputs, distillation objective, and both selection rules. Appendix A gives the backbone-specific attribution, selector architecture and training, regional allocation, RCGD, token compaction, and the dialog selector. Appendix B describes the benchmarks, baseline implementations, timing protocol, and statistics. All backbones, training data, and benchmarks are public. We will release the code, trained selectors, and evaluation scripts upon acceptance.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4190–4197, 2020.

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. DivPrune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9392–9401, 2025.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. arXiv preprint arXiv:2511.21631, 2025a. URL https://arxiv.org/abs/2511.21631.

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, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. arXiv preprint arXiv:2502.13923, 2025b. URL https://arxiv.org/abs/2502.13923.

Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 397–406, 2021.

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

Janghoon Cho, Jungsoo Lee, Munawar Hayat, Kyuwoong Hwang, Fatih Porikli, and Sungha Choi. FLoC: Facility location-based efficient visual token compression for long video understanding. In *International Conference on Learning Representations (ICLR)*, 2026.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, Jose M. F. Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 326–335, 2017. doi: 10.1109/CVPR.2017.121.

Jinhong Deng, Wen Li, Joey Tianyi Zhou, and Yang He. SCOPE: Saliency-coverage oriented token pruning for efficient multimodel LLMs. In *Advances in Neural Information Processing Systems*, volume 38, pp. 179368–179393, 2025. doi: 10.52202/085713-5394. URL https://doi.org/10.52202/085713-5394.

Mohamed Dhouib, Davide Buscaldi, Sonia Vanier, and Aymen Shabou. PACT: Pruning and clustering-based token reduction for faster visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14582–14592, 2025.

Sixun Dong, Juhua Hu, Mian Zhang, Ming Yin, Yanjie Fu, and Qi Qian. MMTok: Multimodal coverage maximization for efficient inference of VLMs. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=GvPdSWZT31.

Chenxi Du, Yongheng Deng, Jiani Liu, Yujia Zhang, Xi Chen, and Ju Ren. CoIn: Coverage and informativeness-guided token reduction for efficient large multimodal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10492–10501, 2026.

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

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

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24108–24118, 2025b.

Tianxiao Gao, Shanwei Zhao, Shuo Fang, Shiai Zhu, and Chenguang Ma. QuietPrune: Query-guided early token pruning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3553–3562, 2026.

Landi He, Xiaoyu Yang, and Lijian Xu. The model knows which tokens matter: Automatic token selection via noise gating. arXiv preprint arXiv:2603.07135, 2026a. URL https://arxiv.org/abs/2603.07135.

Landi He, Shawn Young, and Lijian Xu. Stepwise token selection for efficient multimodal large language models. arXiv preprint arXiv:2606.16067, 2026b. URL https://arxiv.org/abs/2606.16067.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015. URL https://arxiv.org/abs/1503.02531.

Drew A. Hudson and Christopher D. Manning. 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 (CVPR)*, pp. 6700–6709, 2019.

Samir Khaki, Junxian Guo, Jiaming Tang, Shang Yang, Yukang Chen, Konstantinos N. Plataniotis, Yao Lu, Song Han, and Zhijian Liu. SparseVILA: Decoupling visual sparsity for efficient VLM inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23784–23794, 2025.

Youngeun Kim, Youjia Zhang, Huiling Liu, Aecheon Jung, Sunwoo Lee, and Sungeun Hong. ZOO-Prune: Training-free token pruning via zeroth-order gradient estimation in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 39572–39582, 2026.

Lei Lei, Jie Gu, Xiaokang Ma, Chu Tang, Jingmin Chen, and Tong Xu. Task-related token compression in multimodal large language models from an explainability perspective. In *International Conference on Learning Representations (ICLR)*, 2026.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025.

Geng Li, Guohao Chen, Ting Chen, Shilin Shan, Kuangji Zuo, Bofan Lyu, Tuo An, Gen Li, and Jianfei Yang. OccamToken: Efficient VLM inference with training-free and budget-adaptive token pruning. arXiv preprint arXiv:2605.29657, 2026. URL https://arxiv.org/abs/2605.29657.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 292–305. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.20.

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

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. https://llava-vl.github.io/blog/2024-01-30-llava-next/, January 2024b.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is your multi-modal model an all-around player? In *European Conference on Computer Vision*, pp. 216–233. Springer, 2024c. doi: 10.1007/978-3-031-72658-3_13.

Yudong Liu, Jingwei Sun, Yueqian Lin, Jianyi Zhang, Jingyang Zhang, Ming Yin, Qinsi Wang, Hai Li, and Yiran Chen. Keyframe-oriented vision token pruning: Enhancing efficiency of large vision language models on long-form video processing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20802–20811, 2025.

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

Qiyanhui Lu, Han Wu, Rongjian Xu, Tingzhang Luo, Cheng Fan, Xinghao Chen, Minjing Dong, Jufeng Yang, and Jianyuan Guo. RoRA: Role-oriented regional allocation for visual token pruning in MLLMs. arXiv preprint arXiv:2608.07088, 2026. URL https://arxiv.org/abs/2608.07088.

Bole Ma, Jan Eitzinger, Harald Koestler, and Gerhard Wellein. Kamera: Unified position-invariant multimodal KV cache for training-free reuse. arXiv preprint arXiv:2606.23581, 2026. URL https://arxiv.org/abs/2606.23581.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. DocVQA: A dataset for VQA on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2200–2209, 2021. doi: 10.1109/WACV48630.2021.00225.

Shengling Qin, Hao Yu, Chenxin Wu, Zheng Li, Yizhong Cao, Zhengyang Zhuge, Yuxin Zhou, Wentao Yao, Yi Zhang, Zhengheng Wang, Shuai Bai, Jianwei Zhang, and Junyang Lin. VLCache: Computing 2% vision tokens and reusing 98% for vision-language inference. arXiv preprint arXiv:2512.12977, 2025. URL https://arxiv.org/abs/2512.12977.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

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

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning (ICML)*, pp. 3145–3153, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *International Conference on Learning Representations (ICLR) Workshop*, 2014.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8317–8326, 2019. doi: 10.1109/CVPR.2019.00851.

Yuyao Sun, Tao Deng, Shuang Li, Deqing Wang, Hao Geng, and Minjun Yu. Learning to predict middle-layer attention in MLLMs for visual token pruning. arXiv preprint arXiv:2608.06411, 2026. URL https://arxiv.org/abs/2608.06411.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning (ICML)*, pp. 3319–3328, 2017.

Tinghao Wang, Yichen Guo, Rui Huang, Junchi Yao, Qizhe Zhang, Zheng Lu, Yuan Zhang, Jiajun Cao, Zhirong Shen, Yaosong Du, Lijie Hu, and Shanghang Zhang. TOPS: First-principles visual token pruning via constructing token optimal preservation sets for efficient MLLM inference. arXiv preprint arXiv:2606.27161, 2026a. URL https://arxiv.org/abs/2606.27161.

Yi Wang, Haofei Zhang, Qihan Huang, Anda Cao, Gongfan Fang, Wei Wang, Xuan Jin, Jie Song, Mingli Song, and Xinchao Wang. Rethinking token reduction for large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026b.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves ImageNet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10687–10698, 2020.

Tianrun Xu, Haoda Jing, Ye Li, Yuquan Wei, Jun Feng, Guanyu Chen, Haichuan Gao, Tianren Zhang, Jing Liu, and Feng Chen. DeFacto: Counterfactual thinking with images for enforcing evidence-grounded and faithful reasoning. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026a.

Tong Xu, Hailong Shi, and Xingyu Gao. SCoRe: Salience-coverage reduction for vision token pruning in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24686–24695, 2026b.

Cheng Yang, Yang Sui, Jinqi Xiao, Lingyi Huang, Yu Gong, Chendi Li, Jinghua Yan, Yu Bai, Ponnuswamy Sadayappan, Xia Hu, and Bo Yuan. TopV: Compatible token pruning with inference time optimization for fast and low-memory multimodal vision language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19803–19813, 2025a.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. VisionZip: Longer is better but not necessary in vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802, 2025b.

Xubing Ye, Yukang Gan, Yixiao Ge, Xiao-Ping Zhang, and Yansong Tang. ATP-LLaVA: Adaptive token pruning for large vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24972–24982, 2025.

Quan-Sheng Zeng, Yunheng Li, Qilong Wang, Peng-Tao Jiang, Zuxuan Wu, Ming-Ming Cheng, and Qibin Hou. A glimpse to compress: Dynamic visual token pruning for large vision-language models. *IEEE Transactions on Circuits and Systems for Video Technology*, 2026.

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

Qizhe Zhang, Mengzhen Liu, Lichen Li, Ming Lu, Yuan Zhang, Junwen Pan, Qi She, and Shanghang Zhang. Beyond attention or similarity: Maximizing conditional diversity for token pruning in MLLMs. In *Advances in Neural Information Processing Systems*, volume 38, pp. 28761–28791, 2025b. doi: 10.52202/085713-0855. URL https://doi.org/10.52202/085713-0855.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis A. Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. SparseVLM: Visual token sparsification for efficient vision-language model inference. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 74840–74857, 2025c. URL https://proceedings.mlr.press/v267/zhang25s.html.

Xin Zou, Di Lu, Yizhou Wang, Yibo Yan, Yuanhuiyi Lyu, Xu Zheng, Linfeng Zhang, and Xuming Hu. Don't just chase "highlighted tokens" in MLLMs: Revisiting visual holistic context retention. In *Advances in Neural Information Processing Systems*, volume 38, pp. 44496–44528, 2025. doi: 10.52202/085713-1327.

A Implementation Details

A.1 Attribution Targets

We differentiate the answer log-probability with respect to the visual tokens that enter the frozen language model, detached from the vision encoder. For LLaVA-1.5 these are the projected penultimate-layer features without the class token, and for the Qwen models they are the outputs of the visual merger. Qwen3-VL also injects intermediate visual features $D^{(\ell)}$ into early language-model layers. To include these paths, we differentiate with respect to a copy V' and substitute $V \mapsto V'$ and $D^{(\ell)} \mapsto D^{(\ell)} + (V' - V)$. The forward pass is unchanged at $V' = V$, but the gradient now reflects both the input embeddings and the injected features. The activation factor in Eq. 1 remains v_i.

A.2 Selector Architecture and Training

The scorer applies an affine LayerNorm to x_i, followed by an MLP with hidden widths 512 and 128, GELU activations, and a scalar output. The question feature $\bar{q}$ averages the frozen input embeddings of the question's non-special tokens, and the interaction term uses non-affine LayerNorm. The 64-dimensional position encoding $\psi(p_i)$ applies sine and cosine features at frequencies $2^r\pi$ ($r = 0, \ldots, 9$) to the coordinates (t, y, x), each normalized by its maximum, and pads the 60 resulting features with four zeros. In Eq. 2, Smooth-L1 uses $\beta = 1$, both terms are averaged per example and then over the minibatch, and standardization divides by the sample standard deviation plus ϵ. If an example has zero attribution mass, only the Smooth-L1 term is used.

We train on DeFacto (Xu et al., 2026a) image–question pairs with annotated supporting regions. The common pool of 855 pairs has one question per image and keeps only pairs for which LLaVA-1.5 answers correctly, changes its answer when the supporting region is masked, and keeps it when a control region is masked. The filter uses the GT answer only to choose pairs, and each backbone computes its targets from its own answers. Selectors trained without the filter perform as well (Appendix E). We split the pool 80/20 into training and validation, train for 40 epochs with AdamW (learning rate 10^{-4}, weight decay 0.01), and use the final checkpoint.

A.3 Regional Allocation, RCGD, and Compaction

Tokens are assigned to 4×4 cells by floor division of their spatial coordinates. Each cell first receives $k_c = \min(|c|, \lfloor kM_c \rfloor)$ tokens. The remaining slots go, one per cell per round, to cells that still have capacity and have the largest deficit $kM_c - k_c$, until exactly k tokens are allocated.

RCGD computes cosine distances and runs the greedy rule of Eq. 3 exactly. Its cost is reported in Appendix D.

Dropped visual tokens are removed before the language model, and kept tokens stay in their original order. LLaVA assigns consecutive position IDs to the shortened sequence. The Qwen models keep the original multi-axis rotary positions of all remaining tokens, and generation continues from the largest retained position. For Qwen3-VL, the injected intermediate features are compacted with the same indices.

A.4 High-Resolution Backbones

LLaVA-NeXT (`llava-v1.6-vicuna-7b`) and LLaVA-OneVision (`llava-onevision-qwen2-7b-ov`) pack a base grid, a higher-resolution detail grid, and one newline token per detail row into a single visual span, and OneVision also downsamples large detail grids. This gives 674–2,928 visual tokens per image on LLaVA-NeXT and 869–8,765 on OneVision. All packed tokens, including newline tokens, are candidates, and the budget is set per image as $k = \mathrm{round}(rN)$. We rebuild each token's position from the packing geometry, placing base tokens on plane $t = 0$ and detail and newline tokens on $t = 1$. Base coordinates are rescaled to the detail grid so that a 4×4 cell covers the same image region at both scales. Keeping all tokens through this path reproduces the native answers.

B Experimental Protocol

Benchmarks and metrics. MMBench uses 4,329 questions, POPE 9,000 questions over 500 images, and GQA a fixed subset of 800 questions over 201 images. Relative performance divides each pruned score by the full model's score on the same benchmark. Answer preservation is the fraction of pruned answers that match the full model's answer after the benchmark's answer normalization.

Baselines. On LLaVA-1.5, every baseline runs from its official implementation, which uses FP16 and its own preprocessing. We rescale each baseline score by the ratio of our full-model score to the full-model score of that implementation. As a check on this rescaling, we also ran our selector inside each official implementation. Against the same full-model reference, it is on par with or better than official CDPruner, MMTok, and SCOPE in GQA and MMBench preservation, and it exceeds official FastV on GQA by 15.3, 11.1, and 5.6 pp at 5%, 11%, and 22%. On Qwen2.5-VL we use the released MMTok and VisionZip code. All other baselines are ports that keep the published selection signal and change only the interface. Backbones without a CLIP (Radford et al., 2021) text encoder take CDPruner's question relevance from the language model's input embeddings. On OneVision, whose SigLIP encoder has no class token, SCOPE's class-to-patch saliency is replaced by the mean attention each patch receives. FastV and SparseVLM prune inside the language model. SparseVLM's budget follows a published per-layer schedule whose smallest configuration retains 64 tokens, so its 5% cells are left unevaluated rather than obtained by redesigning that schedule.

Timing. Main-paper timings use one L40S GPU, and the appendix adds an H100. Each query covers input preparation, vision encoding, selection, packing, language-model prefill, and one decode step, with batch size one and two forced output tokens. All methods for a backbone run in one process on the same 64 POPE images with five randomized, paired repeats. Speedup is the paired ratio of full-model to pruned-model time. Selector time covers feature construction, scoring, and selection, and includes the pre-pruning language-model layers for FastV and SparseVLM. FLOPs are counted over the same phases. KV memory is the stored key/value cache, and peak memory is measured above the loaded model.

Statistics. Intervals are paired 95% percentile bootstrap intervals. We resample images when an image has several questions, and dialogs or videos for the applications. Intervals are unadjusted unless a Holm correction is stated.

C Additional Benchmark Results

Table 5 summarizes quality and speed across backbones, and Tables 6–9 give the per-benchmark results for the four backbones not shown in Table 1. Table 5 shows the best Ansight variant in each setting. With one rule fixed for every backbone and budget, Ansight + RCGD has the highest average relative score in 11 of the 15 settings and ties in one, and Ansight + RA in 10. Summed over the 15 settings, Ansight's loss relative to the full model is 26% smaller than that of the best baseline in each setting.

Preservation counts agreement with the full model whether its answer is right or wrong. On Qwen3-VL with RCGD at 5% retention, averaged over the six benchmarks, 85.0% of the full model's correct answers stay correct and 19.9% of its wrong answers become correct (a TextVQA answer counts as correct at VQA accuracy of at least 0.5). Correct answers are preserved more often than wrong ones (83.3 vs. 62.1%). SQA-IMG depends least on the image, and on LLaVA-1.5 random selection even exceeds the full model. Averaging over the other five benchmarks, Ansight is best in 14 of the 15 settings, and Ansight + RCGD alone in 12.

Table 5: **Cross-backbone summary.** Avg. rel. is the mean relative score over six benchmarks, Pres. is the mean answer preservation, and E2E is the paired end-to-end speedup on one L40S. Ours is the best Ansight variant in each setting, shown with two strong baselines per backbone. Bold marks the best value in each setting.

		5% retention			11% retention			22% retention		
Backbone	**Method**	**Avg. rel.** (%) ↑	**Pres.** (%) ↑	**E2E** (×) ↑	**Avg. rel.** (%) ↑	**Pres.** (%) ↑	**E2E** (×) ↑	**Avg. rel.** (%) ↑	**Pres.** (%) ↑	**E2E** (×) ↑
LLaVA-1.5-7B	CDPruner	**95.5**	84.2	1.156	**96.8**	87.0	1.008	98.2	88.4	0.821
	ZOO-Prune	93.5	83.4	0.585	96.1	**87.5**	0.564	98.1	**90.1**	0.524
	Ours	95.3	**84.9**	**1.367**	96.7	87.4	**1.319**	**98.3**	89.2	**1.256**
Qwen3-VL-2B	DivPrune	81.1	74.5	**0.992**	90.1	83.6	**0.982**	95.5	89.6	0.965
	ZOO-Prune	82.7	78.2	0.680	91.5	86.3	0.654	96.2	90.9	0.638
	Ours	**88.1**	**81.8**	0.978	**94.2**	**88.0**	0.978	**97.9**	**92.3**	**0.978**
Qwen2.5-VL-7B	SCOPE	81.3	75.5	1.036	90.3	83.8	1.009	95.9	89.8	0.954
	MMTok	80.5	75.1	1.054	89.4	83.8	1.044	94.8	89.2	1.018
	Ours	**85.8**	**79.4**	**1.059**	**92.4**	**85.7**	**1.061**	**96.3**	**90.1**	**1.055**
LLaVA-NeXT-7B	MMTok	94.1	87.7	2.442	96.9	88.7	2.120	97.4	89.8	1.625
	SCOPE	94.6	87.2	1.806	96.3	88.5	1.459	97.6	89.7	1.053
	Ours	**94.9**	**88.0**	**2.716**	**97.1**	**89.3**	**2.612**	**99.2**	**91.5**	**2.246**
LLaVA-OneVision-7B	ZOO-Prune	87.7	83.6	0.492	93.7	89.9	0.472	96.7	92.4	0.432
	SCOPE	91.3	87.6	1.661	94.7	90.9	1.399	97.6	**92.8**	1.027
	Ours	**94.3**	**88.9**	**2.811**	**97.6**	**91.1**	**2.686**	**98.7**	92.6	**2.261**

At 5% retention, RA preserves significantly more answers than global top-k on MMBench and POPE for all five backbones, and on GQA for all but LLaVA-1.5. RCGD is significantly better than RA in 12 of 45 backbone–benchmark–budget cells and significantly worse in 4, three of them on OneVision. The largest gains are on Qwen2.5-VL at 5%, where RCGD adds 5.5, 3.7, and 4.3 pp of preservation on MMBench, POPE, and GQA.

Because DeFacto and the benchmarks draw on public image collections, we repeated the analysis after removing every benchmark image that resembles a DeFacto training image, which leaves 4,246 MMBench and 8,946 POPE questions. On LLaVA-1.5 and both Qwen models, the 5% gains of RA over top-k change by at most 0.1 pp.

Table 6: **LLaVA-1.5-7B results.** Raw scores at 5%, 11%, and 22% retention; Avg. rel. is the mean score relative to the full model. Baseline scores come from each method's official implementation, rescaled by the ratio of our full-model score to that of its own stack (Appendix B). **Green** and **amber** mark the best and second-best pruned scores. Speed and selector time are measured on one L40S and shaded from **best** to **worst** within each budget.

Method	MMB ↑	POPE F1 ↑	GQA ↑	MME ↑	SQA-IMG ↑	TextVQA ↑	Avg. rel. (%) ↑	FLOPs (T) ↓	E2E speed (×) ↑	Selector (ms) ↓
Full	**73.48**	**84.62**	**61.50**	**1771.9**	**65.74**	**57.87**	**100.0**	8.59	1.000	–
5% retention (29 of 576 visual tokens)										
Random	66.39	71.55	53.00	1390.3	**67.28**	46.81	87.1	1.17	–	–
FastV (ECCV'24)	58.43	30.82	46.25	1234.1	65.44	51.32	74.8	2.92	1.245	4.43
VisionZip (CVPR'25)	67.03	68.01	49.50	1428.1	65.12	52.85	87.2	1.18	1.392	1.43
DivPrune (CVPR'25)	68.73	81.02	55.88	1523.1	64.74	53.00	92.7	1.18	1.371	2.30
CDPruner (NeurIPS'25)	70.09	**86.43**	**57.73**	1531.9	**67.48**	53.45	**95.5**	1.19	1.156	15.84
PruneSID (ICLR'26)	68.93	79.26	54.25	1485.3	66.29	51.70	91.6	1.18	0.994	28.37
ZOO-Prune (CVPR'26)	69.60	82.31	55.88	1506.1	65.36	**54.08**	93.5	4.27	0.585	97.05
MMTok (ICLR'26)	69.68	82.91	54.73	1533.9	65.37	53.52	93.4	1.17	1.353	3.26
SCOPE (NeurIPS'25)	**70.45**	79.85	54.13	1539.2	65.46	**53.92**	93.0	1.17	1.261	6.18
Ansight top-k	67.29	80.60	55.75	1448.2	65.74	52.16	91.6	1.18	**1.394**	**1.19**
Ansight + RA	70.02	82.50	55.50	**1561.5**	65.94	51.73	93.5	1.18	1.385	1.73
Ansight + RCGD	**70.52**	**84.78**	**57.12**	**1568.1**	66.68	53.67	**95.3**	1.18	1.367	2.49
11% retention (63 of 576 visual tokens)										
Random	69.05	78.31	55.88	1518.0	**67.38**	49.18	91.8	1.63	–	–
FastV (ECCV'24)	68.78	56.80	52.12	1498.6	66.34	55.33	87.7	3.77	1.182	4.39
SparseVLM (ICML'25)	68.52	75.91	52.38	1495.9	64.12	52.81	90.2	1.93	1.225	6.97
VisionZip (CVPR'25)	70.38	76.13	54.87	1557.8	65.36	55.92	93.2	1.63	**1.347**	1.39
DivPrune (CVPR'25)	70.54	84.53	56.25	**1629.2**	65.36	55.56	95.8	1.63	1.290	4.65
CDPruner (NeurIPS'25)	70.90	**86.85**	**58.74**	1565.1	67.03	55.63	**96.8**	1.64	1.008	25.74
PruneSID (ICLR'26)	70.94	84.49	56.38	**1640.6**	**67.97**	54.51	96.4	1.63	0.938	33.27
ZOO-Prune (CVPR'26)	71.12	85.33	57.25	1589.1	65.27	**56.27**	96.1	4.72	0.564	101.13
MMTok (ICLR'26)	71.16	84.80	57.16	1577.1	65.09	55.31	95.6	1.63	1.276	5.48
SCOPE (NeurIPS'25)	**71.98**	83.56	56.19	1607.9	64.71	56.17	95.7	1.63	1.135	12.52
Ansight top-k	**71.19**	83.97	**58.00**	1587.4	66.09	55.39	96.0	1.63	**1.347**	**1.19**
Ansight + RA	71.17	84.89	56.88	1615.7	66.34	55.28	96.2	1.63	1.338	1.70
Ansight + RCGD	70.85	**86.68**	57.50	1584.8	66.48	**56.39**	**96.7**	1.64	1.319	2.79
22% retention (127 of 576 visual tokens)										
Random	71.17	81.95	58.13	1594.6	**68.07**	51.52	95.1	2.48	–	–
FastV (ECCV'24)	71.82	73.52	57.38	**1685.9**	66.19	56.81	95.3	5.37	1.103	4.39
SparseVLM (ICML'25)	**72.77**	83.68	59.13	**1697.3**	65.17	56.74	97.8	2.79	1.179	6.93
VisionZip (CVPR'25)	72.01	81.65	57.12	1660.3	65.41	56.94	96.5	2.49	1.286	1.41
DivPrune (CVPR'25)	71.59	85.69	59.13	1645.3	65.69	56.51	97.5	2.48	1.169	9.15
CDPruner (NeurIPS'25)	71.92	**86.96**	**59.86**	1620.6	66.58	56.88	**98.2**	2.50	0.821	44.72
PruneSID (ICLR'26)	72.03	84.32	58.50	1672.3	65.59	54.49	96.8	2.48	0.817	40.11
ZOO-Prune (CVPR'26)	72.24	86.00	**60.38**	1640.3	65.93	56.43	98.1	5.58	0.524	109.46
MMTok (ICLR'26)	72.58	85.51	59.20	1647.3	65.51	56.66	97.8	2.48	1.163	9.73
SCOPE (NeurIPS'25)	**72.84**	85.02	59.08	1672.1	64.94	**57.29**	98.0	2.48	0.964	24.67
Ansight top-k	72.03	85.22	57.63	1669.9	65.64	56.89	97.5	2.49	**1.291**	**1.20**
Ansight + RA	71.91	85.64	58.13	1678.3	65.89	56.67	97.7	2.49	1.281	1.71
Ansight + RCGD	71.84	**86.58**	59.00	1651.5	**67.13**	**57.15**	**98.3**	2.49	1.256	3.25

Table 7: **Qwen3-VL-2B results**, in the format of Table 6. All baselines are our ports, and SparseVLM has no 5% schedule.

Method	MMB ↑	POPE F1 ↑	GQA ↑	MME ↑	SQA-IMG ↑	TextVQA ↑	Avg. rel. (%) ↑	FLOPs (T) ↓	E2E speed (×) ↑	Selector (ms) ↓
Full	**85.15**	**89.13**	**64.00**	**1987.4**	**87.11**	**77.65**	**100.0**	1.79	1.000	–
5% retention (median 13 of 260 visual tokens)										
Random	67.50	63.76	45.75	1463.3	74.22	42.86	72.7	0.92	–	–
FastV (ECCV'24)	46.50	11.28	38.88	1110.5	69.91	37.71	52.1	1.16	0.984	4.45
SparseVLM (ICML'25)	–	–	–	–	–	–	–	–	–	–
DivPrune (CVPR'25)	71.89	70.32	49.62	1621.7	79.38	56.67	81.1	0.92	**0.992**	**1.34**
CDPruner (NeurIPS'25)	64.91	45.29	42.12	1402.1	74.96	45.43	68.0	0.95	0.743	35.52
ZOO-Prune (CVPR'26)	74.17	79.10	51.12	1620.0	79.92	52.10	82.7	2.60	0.680	50.26
MMTok (ICLR'26)	72.60	77.20	51.63	1587.0	78.58	52.76	81.8	0.92	0.977	3.02
SCOPE (NeurIPS'25)	69.85	64.25	46.75	1528.6	76.05	52.40	76.5	0.93	0.962	4.57
VisionZip (CVPR'25)	60.98	42.64	44.50	1387.3	72.73	41.00	65.8	0.93	0.976	3.13
Ansight top-k	73.62	75.62	52.00	1603.0	**80.42**	**62.65**	84.4	0.93	0.987	1.98
Ansight + RA	**76.09**	**80.68**	**54.62**	**1690.1**	79.77	59.89	**86.5**	0.93	0.983	2.50
Ansight + RCGD	**75.38**	**84.23**	**54.25**	**1760.7**	**80.17**	**62.15**	**88.1**	0.93	0.978	3.07
11% retention (median 29 of 260)										
Random	75.49	75.25	51.12	1584.9	77.39	50.75	81.2	0.98	–	–
FastV (ECCV'24)	60.78	47.59	42.88	1336.2	71.19	51.28	67.8	1.27	0.986	4.14
SparseVLM (ICML'25)	78.15	86.45	56.00	1687.8	81.31	59.30	88.5	1.01	0.922	7.82
DivPrune (CVPR'25)	78.79	84.18	53.87	1797.0	81.85	65.86	90.1	0.98	0.982	2.42
CDPruner (NeurIPS'25)	73.97	59.98	46.62	1600.8	78.04	54.03	77.8	1.01	0.738	35.13
ZOO-Prune (CVPR'26)	80.25	**87.13**	56.38	1807.4	**83.69**	63.54	91.5	2.66	0.654	54.13
MMTok (ICLR'26)	78.68	84.45	56.38	1808.8	80.07	64.39	90.2	0.98	0.967	4.12
SCOPE (NeurIPS'25)	78.10	80.18	54.87	1744.6	80.71	64.07	88.4	0.99	0.937	7.53
VisionZip (CVPR'25)	72.42	67.79	52.13	1527.4	76.65	46.82	77.9	0.99	0.978	2.82
Ansight top-k	80.02	86.58	56.62	1805.6	**82.94**	**70.79**	92.8	0.98	**0.990**	**1.69**
Ansight + RA	**80.90**	86.90	**58.13**	**1866.0**	**82.94**	69.44	**93.6**	0.98	0.986	2.16
Ansight + RCGD	**81.27**	**87.98**	**57.38**	**1883.9**	82.60	**71.09**	**94.2**	0.98	0.978	2.77
22% retention (median 57 of 260)										
Random	80.30	82.64	57.38	1746.9	80.96	59.95	89.1	1.08	–	–
FastV (ECCV'24)	72.77	73.16	51.12	1532.4	75.01	67.84	83.0	1.45	0.988	4.13
SparseVLM (ICML'25)	82.28	**89.12**	**61.50**	1897.8	83.44	70.56	95.8	1.11	0.920	7.94
DivPrune (CVPR'25)	**83.32**	88.10	59.00	1894.9	84.38	71.25	95.5	1.08	0.965	4.51
CDPruner (NeurIPS'25)	78.98	72.48	50.50	1727.2	81.51	62.93	85.7	1.11	0.729	37.14
ZOO-Prune (CVPR'26)	83.02	88.91	**61.88**	1864.3	**85.18**	71.29	96.2	2.75	0.638	57.92
MMTok (ICLR'26)	82.03	86.91	58.63	1866.2	84.23	70.20	94.4	1.08	0.947	6.43
SCOPE (NeurIPS'25)	81.94	86.39	60.50	1899.1	84.28	71.65	95.4	1.09	0.890	13.15
VisionZip (CVPR'25)	78.31	82.89	57.75	1735.5	81.76	60.52	89.1	1.09	0.980	2.82
Ansight top-k	82.93	88.90	60.38	1945.1	84.18	**75.30**	97.2	1.08	**0.990**	**1.69**
Ansight + RA	83.02	88.73	60.75	**1973.2**	**85.28**	75.01	**97.6**	1.08	0.986	2.16
Ansight + RCGD	**83.25**	**89.02**	**61.88**	**1958.7**	84.88	**75.58**	**97.9**	1.08	0.978	2.97

Table 8: **Qwen2.5-VL-7B results**, in the format of Table 6. MMTok and VisionZip use their released Qwen2.5-VL code, and the other baselines are our ports. SparseVLM has no 5% schedule.

Method	MMB ↑	POPE F1 ↑	GQA ↑	MME ↑	SQA-IMG ↑	TextVQA ↑	Avg. rel. (%) ↑	FLOPs (T) ↓	E2E speed (×) ↑	Selector (ms) ↓
Full	**87.53**	**86.43**	**61.50**	**2316.4**	**88.45**	**78.34**	**100.0**	7.28	1.000	–
5% retention (median 17 of 345 visual tokens)										
Random	75.47	67.09	50.75	1721.8	75.90	49.54	78.3	2.59	–	–
FastV (ECCV'24)	54.59	35.02	43.50	1357.3	74.27	60.21	65.5	3.98	1.011	4.29
SparseVLM (ICML'25)	–	–	–	–	–	–	–	–	–	–
DivPrune (CVPR'25)	74.96	70.11	50.50	**1765.6**	78.24	56.55	81.0	2.59	**1.070**	**1.46**
CDPruner (NeurIPS'25)	72.56	63.45	44.88	1619.1	76.20	49.05	74.7	2.70	0.822	36.54
ZOO-Prune (CVPR'26)	74.01	**75.08**	**55.38**	1727.5	78.88	47.40	81.0	6.53	0.546	124.33
MMTok (ICLR'26)	**75.58**	68.56	50.38	1693.3	78.04	58.24	80.5	2.59	1.054	3.38
SCOPE (NeurIPS'25)	73.71	72.67	52.25	1676.7	80.22	56.22	81.3	2.61	1.036	5.38
VisionZip (CVPR'25)	73.69	69.46	50.43	1623.7	78.33	55.68	79.4	2.61	1.062	2.34
Ansight top-k	69.21	57.37	48.00	1461.0	79.67	**64.27**	76.4	2.59	1.067	1.72
Ansight + RA	73.30	69.02	52.75	1690.2	**80.32**	61.55	**82.0**	2.59	1.063	2.19
Ansight + RCGD	**77.80**	**74.60**	**53.87**	**1845.7**	**81.16**	**63.33**	**85.8**	2.59	1.059	2.72
11% retention (median 38 of 345)										
Random	80.69	75.22	52.88	1926.0	80.52	57.25	85.4	2.89	–	–
FastV (ECCV'24)	67.06	70.02	50.62	1712.0	78.38	**72.54**	82.5	4.54	1.015	4.28
SparseVLM (ICML'25)	76.00	64.72	45.88	1810.3	79.42	69.02	82.1	3.10	1.014	7.39
DivPrune (CVPR'25)	**80.71**	79.16	55.50	1928.9	82.20	66.00	89.1	2.89	1.060	2.96
CDPruner (NeurIPS'25)	79.81	73.20	52.00	1841.5	78.63	57.84	83.8	3.00	0.817	38.04
ZOO-Prune (CVPR'26)	79.88	**82.03**	**58.25**	1832.3	82.60	53.96	87.0	6.82	0.540	128.42
MMTok (ICLR'26)	80.27	78.84	57.88	**1933.0**	80.96	66.11	89.4	2.89	1.044	4.99
SCOPE (NeurIPS'25)	79.70	**81.61**	57.00	1867.0	**83.74**	69.12	**90.3**	2.91	1.009	9.44
VisionZip (CVPR'25)	80.15	79.24	57.74	1769.6	**83.39**	66.11	88.7	2.91	1.065	2.33
Ansight top-k	77.80	76.80	54.75	1811.1	82.80	**73.07**	88.6	2.89	**1.072**	**1.67**
Ansight + RA	79.30	78.00	55.75	1881.6	82.55	70.29	89.3	2.89	1.067	2.14
Ansight + RCGD	**82.56**	80.94	**58.88**	**1977.0**	83.24	71.55	**92.4**	2.89	1.061	2.87
22% retention (median 76 of 345)										
Random	83.39	81.01	58.00	2060.6	82.99	66.67	91.9	3.43	–	–
FastV (ECCV'24)	81.22	80.16	57.38	2030.8	83.34	**76.60**	93.1	5.55	1.008	4.29
SparseVLM (ICML'25)	84.52	79.60	55.00	2099.0	82.25	75.26	93.0	3.66	1.006	7.47
DivPrune (CVPR'25)	84.01	83.55	60.25	2067.7	83.29	72.08	94.3	3.43	1.035	5.65
CDPruner (NeurIPS'25)	83.07	78.80	55.12	2082.7	81.31	65.00	90.1	3.54	0.804	40.56
ZOO-Prune (CVPR'26)	82.79	**84.84**	60.00	1932.2	84.38	60.80	91.1	7.36	0.531	133.05
MMTok (ICLR'26)	**84.57**	83.85	**60.50**	2096.3	83.14	72.08	94.8	3.43	1.018	7.98
SCOPE (NeurIPS'25)	83.85	**84.31**	**61.25**	**2104.7**	84.33	75.39	**95.9**	3.45	0.954	16.83
VisionZip (CVPR'25)	83.64	83.77	59.50	2026.8	83.74	72.98	94.1	3.45	1.061	2.33
Ansight top-k	83.23	82.79	57.38	2005.6	**85.77**	**76.35**	94.2	3.43	**1.067**	**1.67**
Ansight + RA	**84.57**	82.89	58.25	2017.5	85.18	75.43	94.5	3.43	1.063	2.14
Ansight + RCGD	**85.33**	83.89	59.75	**2143.4**	**85.72**	75.73	**96.3**	3.43	1.055	3.11

Table 9: **LLaVA-NeXT-7B results**, in the format of Table 6. All baselines are our ports.

Method	MMB ↑	POPE F1 ↑	GQA ↑	MME ↑	SQA-IMG ↑	TextVQA ↑	Avg. rel. (%) ↑	FLOPs (T) ↓	E2E speed (×) ↑	Selector (ms) ↓
Full	**75.74**	**85.77**	**65.00**	**1777.1**	**70.70**	**59.78**	**100.0**	33.22	1.000	–
5% retention (median 115 of 2,298 visual tokens)										
Random	70.27	71.31	56.38	1450.7	66.88	43.74	85.3	3.84	–	–
FastV (ECCV'24)	63.96	43.35	51.00	1326.4	67.18	51.62	78.2	9.53	2.115	13.02
DivPrune (CVPR'25)	71.75	81.56	57.63	1549.2	**68.42**	52.70	91.8	3.87	2.563	8.51
CDPruner (NeurIPS'25)	71.22	61.91	52.50	1493.0	67.33	52.31	85.6	3.87	1.019	156.13
ZOO-Prune (CVPR'26)	69.79	79.10	53.63	1486.0	67.77	49.29	88.1	19.31	0.411	513.35
MMTok (ICLR'26)	73.55	**84.71**	60.38	1609.5	67.72	53.34	94.1	3.84	2.442	13.46
SCOPE (NeurIPS'25)	73.67	80.73	**60.62**	**1684.5**	67.48	**55.52**	**94.6**	3.84	1.806	24.73
Ansight top-k	73.25	81.95	59.38	1640.6	**68.37**	**55.67**	94.3	3.86	**2.737**	**2.38**
Ansight + RA	**74.17**	82.33	**61.12**	**1650.6**	**68.37**	54.95	**94.9**	3.86	2.716	3.20
Ansight + RCGD	**73.94**	**82.56**	60.50	1611.5	68.07	54.26	94.1	3.90	2.565	7.82
11% retention (median 252 of 2,298)										
Random	72.88	78.98	61.12	1539.1	66.19	46.79	90.1	5.57	–	–
FastV (ECCV'24)	71.77	72.52	58.13	1571.2	68.27	56.85	91.5	12.78	1.996	12.92
DivPrune (CVPR'25)	73.16	84.47	59.75	1629.5	68.22	53.90	94.2	5.61	2.238	18.27
CDPruner (NeurIPS'25)	72.46	71.66	56.63	1646.6	67.67	54.54	91.0	5.61	0.972	157.26
ZOO-Prune (CVPR'26)	72.49	**86.85**	60.00	1605.5	67.67	54.26	94.3	21.04	0.394	531.61
MMTok (ICLR'26)	73.37	86.24	**62.38**	**1724.2**	**68.42**	56.41	**96.9**	5.57	2.120	24.71
SCOPE (NeurIPS'25)	74.29	84.77	60.12	**1712.2**	67.08	**58.01**	96.3	5.57	1.459	51.46
Ansight top-k	**74.66**	85.53	61.00	1671.6	67.82	**58.14**	96.6	5.60	**2.633**	**2.34**
Ansight + RA	74.52	86.05	**62.38**	1697.2	67.82	57.59	**97.1**	5.60	2.612	3.12
Ansight + RCGD	**74.68**	**87.40**	**61.63**	1676.5	**68.57**	56.48	**96.9**	5.63	2.419	9.03
22% retention (median 506 of 2,298)										
Random	74.29	82.83	62.75	1605.8	66.44	50.69	93.4	8.78	–	–
FastV (ECCV'24)	74.64	82.32	62.63	1700.2	**69.11**	58.90	97.1	18.81	1.584	12.95
DivPrune (CVPR'25)	74.41	86.33	62.00	1701.7	67.58	55.24	96.3	8.82	1.714	36.31
CDPruner (NeurIPS'25)	73.34	77.40	57.13	1681.2	67.33	55.94	93.1	8.82	0.831	188.62
ZOO-Prune (CVPR'26)	74.17	**87.57**	63.75	1756.2	67.23	56.44	97.7	24.25	0.366	566.04
MMTok (ICLR'26)	73.43	86.83	62.75	1726.1	67.77	57.69	97.4	8.78	1.625	44.43
SCOPE (NeurIPS'25)	74.45	85.83	62.13	1754.4	67.28	58.63	97.6	8.78	1.053	99.90
Ansight top-k	74.84	86.74	**64.25**	**1803.3**	67.87	**59.32**	**99.2**	8.81	**2.246**	**2.37**
Ansight + RA	**75.19**	86.93	63.38	**1787.0**	68.42	**59.05**	**99.0**	8.81	2.233	3.16
Ansight + RCGD	**74.96**	**87.65**	**64.13**	1746.9	**68.47**	58.04	98.7	8.85	2.050	11.89

D EFFICIENCY MEASUREMENTS

Where the time goes. Pruning visual tokens shortens only the language-model prefill. Vision encoding, input preparation, and decoding stay the same (Table 10). Prefill is 26–42% of a query on LLaVA-1.5 and the Qwen models but about 62–64% on LLaVA-NeXT and OneVision, which is why the end-to-end gains are much larger on the high-resolution backbones. We report the widest timing boundary. With narrower ones, the same 5% runs on LLaVA-1.5, Qwen3-VL, and Qwen2.5-VL give $1.110\times$, $0.955\times$, and $0.969\times$ in time to first token and $1.279\times$, $1.006\times$, and $1.060\times$ in prefill alone, compared with $1.075\times$, $0.970\times$, and $0.981\times$ end to end. Every baseline is charged its own selection cost in the same way, which is why methods with expensive selection fall below $1\times$.

All six benchmarks have short answers, so two output tokens are close to their natural length. Longer outputs dilute the gain because the added decoding steps shrink only slightly with a shorter visual prefix. On LLaVA-1.5 at 5% retention on the L40S, forcing 64 or 256 output tokens lowers the speedup of RA to $1.03\times$ and $1.02\times$. For long generations the main benefit is the smaller KV cache and the larger batch it allows.

Offline cost. Building the target and selector inputs for one training pair (answer generation, one backward pass, and feature extraction) takes about 1.5 GPU-seconds on an H100 for Qwen3-VL-2B, so its 855-pair pool takes about 20 GPU-minutes.

Table 10: **Where a query's time goes.** Median wall time per phase (ms) on an H100 for the full model and Ansight + RA at 5% retention (64 POPE images). Only the language-model prefill shrinks when tokens are removed. The prunable share is the prefill fraction of the full query, and the ceiling is the speedup if prefill took no time. Phase medians need not sum to the end-to-end median.

	Qwen3-VL-2B		Qwen2.5-VL-7B		LLaVA-1.5-7B		LLaVA-NeXT-7B		LLaVA-OV-7B	
Median visual tokens	260		345		576		2,298		2,709	
Phase	Full	Ours	Full	Ours	Full	Ours	Full	Ours	Full	Ours
Input preparation (CPU)	13.33	13.29	13.93	14.04	7.12	7.01	12.69	12.66	15.65	15.48
Vision encoder	12.71	12.64	23.39	23.28	7.28	7.30	8.16	8.07	13.03	12.96
Selector (scoring + selection)	0.02	1.68	0.02	1.95	0.03	1.48	0.02	2.49	0.02	2.75
Prompt packing	1.22	1.22	1.33	1.35	0.41	0.58	0.65	0.65	0.72	0.67
LM prefill	**18.37**	**18.24**	**20.03**	**18.96**	**24.89**	**19.33**	**77.34**	**20.53**	**83.38**	**19.42**
Decode step	17.67	17.39	18.46	18.14	19.13	19.03	20.57	20.27	19.15	18.77
End-to-end	63.70	65.40	77.17	78.95	58.99	54.73	120.03	65.10	134.61	70.32
Examples	64		64		64		64		64	
Prunable share	28.8%		26.0%		42.2%		64.4%		61.9%	
Ceiling if prefill $\to 0$	$1.41\times$		$1.35\times$		$1.73\times$		$2.81\times$		$2.63\times$	

Speed of every method. Table 11 reports the end-to-end speedup of all methods on both GPUs. Speedups are larger on the L40S for nearly every method, and the ordering of methods is similar on both GPUs. At matched budgets, the MMTok and SCOPE selectors take 0.3–4.5 ms and 1.4–11.2 ms longer than RA, while their language-model prefill time is the same.

Memory and batching. Table 12 gives single-query latency and memory at 5% retention. Counted FLOPs for the other backbones are in Tables 6–9; on LLaVA-OneVision they fall from 44.98 to 5.62 TFLOPs at 5% retention. The KV cache includes text and generated tokens, so it shrinks less than the visual-token count. With 11% retention, LLaVA-1.5 fits a batch of at least 256 on both GPUs, compared with 118 on the H100 and 63 on the L40S for the full model (Table 13).

Cost of RCGD. Table 14 gives the cost of the exact greedy rule. Allocation stays under 9 ms up to 4,104 tokens. The distance matrix grows quadratically and reaches 257 MB at 8,200 tokens, so the dense implementation suits sequences of up to a few thousand tokens.

Table 11: **End-to-end speedup of every method on H100 and L40S.** Paired full/method ratios on the 64 POPE timing images (batch size one, two output tokens). The first row is the full-model time in ms.

	LLaVA-1.5		Qwen3-VL		Qwen2.5-VL		LLaVA-NeXT		LLaVA-OV	
Method	H100	L40S	H100	L40S	H100	L40S	H100	L40S	H100	L40S
Full E2E (ms)	59.0	99.8	63.7	96.3	77.2	129.3	120.0	247.3	134.6	294.1
5% retention										
FastV	1.221	1.245	0.999	0.984	1.007	1.011	1.825	2.115	1.891	2.127
VisionZip	1.101	1.392	0.972	0.976	0.982	1.062	1.749	2.613	1.791	2.683
DivPrune	1.089	1.371	0.986	0.992	0.993	1.070	1.735	2.563	1.755	2.629
CDPruner	0.958	1.156	0.714	0.743	0.746	0.822	0.606	1.019	0.563	1.059
ZOO-Prune	0.471	0.585	0.600	0.680	0.456	0.546	0.327	0.411	0.357	0.492
MMTok	1.074	1.353	0.969	0.977	0.981	1.054	1.644	2.442	1.676	2.512
SCOPE	1.057	1.261	0.960	0.962	0.961	1.036	1.356	1.806	1.295	1.661
Ansight top-k	1.094	1.394	0.976	0.987	0.985	1.067	1.818	2.737	1.857	2.834
Ansight + RA	1.075	1.385	0.970	0.983	0.981	1.063	1.808	2.716	1.844	2.811
Ansight + RCGD	1.076	1.367	0.971	0.978	0.980	1.059	1.749	2.565	1.788	2.635
11% retention										
FastV	1.225	1.182	0.990	0.986	1.009	1.015	1.828	1.996	1.840	1.964
SparseVLM	1.081	1.225	0.919	0.922	0.934	1.014	1.592	1.985	1.615	1.968
VisionZip	1.100	1.347	0.968	0.978	0.981	1.065	1.740	2.513	1.765	2.580
DivPrune	1.070	1.290	0.974	0.982	0.976	1.060	1.606	2.238	1.589	2.314
CDPruner	0.886	1.008	0.708	0.738	0.740	0.817	0.586	0.972	0.536	1.006
ZOO-Prune	0.464	0.564	0.595	0.654	0.453	0.540	0.318	0.394	0.345	0.472
MMTok	1.048	1.276	0.965	0.967	0.959	1.044	1.517	2.120	1.516	2.214
SCOPE	0.995	1.135	0.941	0.937	0.933	1.009	1.156	1.459	1.097	1.399
Ansight top-k	1.098	1.347	0.979	0.990	0.982	1.072	1.819	2.633	1.836	2.702
Ansight + RA	1.084	1.338	0.972	0.986	0.975	1.067	1.796	2.612	1.812	2.686
Ansight + RCGD	1.064	1.319	0.967	0.978	0.975	1.061	1.716	2.419	1.719	2.412
22% retention										
FastV	1.214	1.103	0.992	0.988	1.008	1.008	1.663	1.584	1.596	1.556
SparseVLM	1.087	1.179	0.919	0.920	0.933	1.006	1.602	1.911	1.614	1.909
VisionZip	1.085	1.286	0.974	0.980	0.986	1.061	1.659	2.163	1.647	2.189
DivPrune	1.021	1.169	0.962	0.965	0.969	1.035	1.358	1.714	1.336	1.754
CDPruner	0.773	0.821	0.703	0.729	0.732	0.804	0.535	0.831	0.480	0.833
ZOO-Prune	0.448	0.524	0.582	0.638	0.445	0.531	0.301	0.366	0.322	0.432
MMTok	1.002	1.163	0.940	0.947	0.948	1.018	1.275	1.625	1.247	1.681
SCOPE	0.903	0.964	0.898	0.890	0.892	0.954	0.900	1.053	0.837	1.027
Ansight top-k	1.089	1.291	0.974	0.990	0.983	1.067	1.721	2.246	1.709	2.278
Ansight + RA	1.076	1.281	0.971	0.986	0.979	1.063	1.715	2.233	1.699	2.261
Ansight + RCGD	1.054	1.256	0.957	0.978	0.971	1.055	1.583	2.050	1.554	2.023

Table 12: **Single-query latency and memory on an H100** at 5% retention (Ansight + RA, 64 POPE images). Times are medians (ms), speed is the paired full/compact ratio with a 95% interval, and memory reductions are measured above the loaded model. Bold marks the lower latency.

Model	Full	Compact	Speed [95% CI]	KV reduction	Peak reduction
LLaVA	58.99	**54.73**	1.075 [1.066,1.084]	90.4%	81.6%
Qwen3-VL	**63.70**	65.40	0.970 [0.962,0.977]	86.4%	59.0%
Qwen2.5-VL	**77.17**	78.95	0.981 [0.975,0.988]	85.9%	51.4%
LLaVA-NeXT	120.03	**65.10**	1.808 [1.775,1.841]	93.8%	80.1%
LLaVA-OneVision	134.61	**70.32**	1.844 [1.800,1.886]	94.1%	67.8%

Table 13: **Batch capacity and throughput for LLaVA-1.5** at 11% retention (two output tokens). Capacity is the largest batch that fits, up to a search limit of 256, and throughput is compared at batch size 32 with 95% intervals.

GPU	Path	Max. batch	Capacity gain	Throughput [95% CI]
H100 80GB	Full	118	1.00×	1.000
	Ansight	$\geq$ **256**	$\geq$ 2.17×	1.648 [1.570,1.726]
L40S 48GB	Full	63	1.00×	1.000
	Ansight	$\geq$ **256**	$\geq$ 4.06×	2.392 [2.216,2.484]

Table 14: **RCGD cost on Qwen2.5-VL features.** Median milliseconds over 16 images. N is the number of visual tokens, B the budget, and N^2 MB the size of the distance matrix. Build computes the distances, select runs the greedy rule, and GPU total covers the whole allocation. Peak is added GPU memory (MB).

N	B	N^2 MB	Build	Select	GPU total [p95]	Peak
345	38	0.5	0.13	0.40	0.79 [0.87]	10
345	17	0.5	0.13	0.21	0.59 [0.63]	10
1,024	113	4.0	0.26	1.09	1.58 [1.78]	36
1,024	51	4.0	0.26	0.52	1.03 [1.22]	36
1,024	512	4.0	0.27	4.58	5.20 [5.78]	36
2,048	225	16.0	0.69	2.12	3.16 [3.36]	88
2,048	102	16.0	0.69	0.99	2.00 [2.21]	88
2,048	512	16.0	0.70	4.77	5.82 [6.06]	88
4,104	451	64.3	3.06	4.54	8.09 [8.31]	241
4,104	205	64.3	3.06	2.10	5.58 [5.77]	241
4,104	512	64.3	3.06	5.15	8.75 [8.95]	241
8,200	902	256.5	9.94	9.85	20.49 [20.67]	737
8,200	410	256.5	9.94	4.51	15.11 [15.31]	737
8,200	512	256.5	9.95	5.61	16.22 [16.39]	737

E SUPERVISION ABLATIONS

Signal definitions. All signals score each visual token v_i of the frozen model. With $g_i = \partial \mathcal{J} / \partial v_i$ the gradient of the answer log-probability (Eq. 1), $G \times A$ scores $\|g_i \odot v_i\|_1$, the Gradient×Input attribution of Shrikumar et al. (2017). $G{\times}\mathrm{LN}(A)$ replaces v_i by its layer-normalized value. Gradient saliency scores $\|g_i\|_1$ (Simonyan et al., 2014), the size of the gradient alone, and activation norm scores $\|v_i\|_1$, which uses no answer or question.

Integrated gradients (Sundararajan et al., 2017) accumulate gradients along a straight path from a baseline input V^0 to V instead of taking the gradient at V only. A token scores $\|(v_i - v_i^0) \odot \bar{g}_i\|_1$, where $\bar{g}_i$ averages the gradient at 16 evenly spaced points on the path. With the usual all-zero baseline the ranking changed with the number of steps, because all-zero tokens lie far from the inputs the language model sees. We therefore use the tokens of an all-black image as V^0, for which 16 and 128 steps give the same ranking.

Attention rollout multiplies gradient-weighted attention maps through all layers (Abnar & Zuidema, 2020; Chefer et al., 2021) and averages the result over answer tokens, following Lei et al. (2026). Prompt attention averages the attention that prompt tokens pay to each visual token, the signal used by FastV and SparseVLM. Answer attention does the same from the positions of the GT answer. CLIP question similarity is CDPruner's instruction relevance, the negated cosine between each CLIP patch embedding and the CLIP embedding of the question. Local occlusion scores the drop in answer log-probability when the 2×2 window around a token is removed. Rows marked (GT) compute the target from the GT answer instead of the model's own answer. Method names in parentheses refer to methods that prune with a related signal. We reproduce the signal, not the full method.

Teacher comparison. Table 15 lists every signal on GQA, including the rows omitted from Table 3, and Table 16 repeats the comparison on MMBench. The ordering of signal families is the same on both benchmarks, although gradient saliency is slightly ahead of $G \times A$ at 5% and 11%. Table 17 checks whether this holds on other backbones. After Holm correction, gradient saliency is significantly worse than $G \times A$ on all three Qwen3-VL benchmarks and on Qwen2.5-VL POPE, and never significantly better, so we keep $G \times A$. On MMBench at 5%, $G \times A$ preserves 3.6 pp more answers than attention rollout after Holm correction, and it wins in each of five training seeds by 2.3–5.5 pp (Table 18).

The closest learned selector distills attention rollout into a small convolutional network that reads first-layer attention (Lei et al., 2026). Trained on our pairs, it preserves 70.84% of MMBench answers at 5%, against 81.27% for Ansight with global top-k (Table 16). Both parts contribute. With the rollout teacher, our MLP reaches 77.65%, and switching the teacher to $G \times A$ adds another 3.6 pp.

Table 15: **All supervision signals on LLaVA-1.5 GQA** (800 questions, global top-k). Entries are preservation / relative accuracy (%), averaged over three seeds; this is the complete version of Table 3. Bold marks the best value in each column.

Kind	Supervision signal	5%	11%	22%
Answer gradient	$G \times A$ (self)	**74.92 / 89.30**	78.46 / **93.43**	82.25 / 95.93
	Gradient saliency	72.46 / 87.27	78.21 / 92.34	**82.92** / 96.00
	Integrated gradients	73.04 / 87.40	**78.96** / 92.62	81.83 / 95.25
	$G \times A$ (GT)	71.17 / 85.84	78.33 / 92.75	82.50 / **96.47**
Attention	Attention rollout	71.71 / 86.52	76.42 / 90.18	82.00 / 94.99
	Answer attention (GT)	63.71 / 79.67	70.96 / 85.77	77.33 / 90.59
	Prompt attention	61.33 / 78.18	70.29 / 85.71	76.96 / 92.41
Perturbation	Local 2×2 occlusion	66.25 / 80.49	70.12 / 84.62	75.29 / 88.89
Feature magnitude	Activation norm	50.67 / 67.35	50.83 / 67.76	57.46 / 75.27
Similarity	CLIP question similarity	71.67 / 87.93	76.42 / 93.02	80.25 / 95.33

Table 16: **Supervision targets on LLaVA-1.5 MMBench** (4,329 questions, global top-k). Entries are preservation / accuracy (%), averaged over three seeds. The last row is the selector of Lei et al. (2026), a convolutional network over first-layer attention trained on attention rollout. Bold marks the best value in each column.

Teacher	5%	11%	22%
$G \times A$ (ours)	81.27 / 68.06	86.02 / 70.76	89.40 / 72.56
Attention rollout	77.65 / 64.94	85.44 / 70.28	88.90 / 71.53
Gradient saliency	**82.11** / 68.61	**86.49 / 71.30**	89.32 / **72.60**
$G \times \mathrm{LN}(A)$	81.77 / 68.62	86.31 / 71.29	89.37 / 72.56
Integrated Gradients	82.10 / 68.79	86.25 / 71.07	**89.53** / 71.86
Activation norm	45.07 / 37.68	44.48 / 36.94	52.82 / 44.11
Local 2×2 occlusion	72.81 / 61.51	77.93 / 64.97	81.48 / 67.10
GT-answer $G \times A$	79.65 / 66.38	86.09 / 71.15	89.17 / 72.19
Answer-token attention, GT answer	70.16 / 58.64	81.17 / 67.11	87.33 / 70.89
Prompt attention	66.25 / 56.05	79.95 / 66.88	87.10 / 70.86
CLIP question similarity	80.33 / **69.01**	84.80 / 70.78	88.23 / 71.70
Lei et al.	70.84 / 60.31	78.77 / 65.23	85.56 / 68.97

Answer source and training-data robustness. Attribution maps derived from self-generated and ground-truth answers have a mean Spearman correlation of 0.92, and the two answers agree on 97.6% of examples. The resulting selectors show no detectable difference in preservation, supporting supervision from the frozen model's own answers without human answer labels.

Training data. Our default training pool is small and filtered using LLaVA-1.5 (Appendix A.2). We examine whether the main image-QA selector depends on this filtering or the original DeFacto training distribution by retraining Qwen3-VL on three alternative pools (Table 19): all 6,717 unfiltered DeFacto pairs, a mixed pool of 19,129 pairs combining DeFacto with seven additional VQA datasets, and a size-matched sample of the mixed pool. All targets come from the model's own answers, training images matching evaluation images are excluded, and all selectors are decoded in one process.

Both the unfiltered DeFacto and full mixed pools match or exceed the default in every reported setting, while the size-matched mixed pool remains within 0.4 pp. The gains from the full mixed pool extend to MMBench, POPE, GQA, and MME, whose training splits are absent from all pools. On LLaVA-OneVision, removing the DeFacto filter significantly improves preservation in 11 of 54 benchmark–rule–budget settings and reduces it in only one.

We also perform an additional training without DeFacto. The dialog selector in Section 4.5 is trained exclusively on Visual Dialog, using self-generated answer-attribution targets and no DeFacto examples. On 200 held-out dialogs, current-question selection preserves 86.50/88.60/91.00% of full-model answers at 5/11/22% retention, compared with 77.90/82.55/87.15% for random selec-

Table 17: $G \times A$ **versus gradient saliency** as the teacher of Ansight + RA. Entries are preservation / correctness (%), where correctness is accuracy on MMBench and GQA and F1 on POPE. Bold marks the better teacher in each column.

Benchmark	Teacher	5%	11%	22%
LLaVA-1.5-7B				
MMBench	$G \times A$	83.90 / 70.13	86.88 / 70.92	**89.56 / 72.21**
	Gradient saliency	**84.25 / 70.57**	**87.13 / 71.66**	89.05 / 72.12
POPE	$G \times A$	91.56 / 82.50	93.72 / 84.89	94.87 / 85.64
	Gradient saliency	**92.19 / 83.01**	**93.96 / 85.48**	**94.92 / 86.04**
GQA	$G \times A$	**76.50** / 55.50	**80.12** / 56.88	82.75 / 58.13
	Gradient saliency	76.38 / **56.38**	**80.12 / 57.50**	**83.00 / 59.25**
Qwen3-VL-2B				
MMBench	$G \times A$	**81.89 / 76.09**	**87.92 / 80.90**	**91.61 / 83.02**
	Gradient saliency	72.72 / 68.03	80.80 / 75.44	86.93 / 79.97
POPE	$G \times A$	**88.33 / 80.68**	**94.00 / 86.90**	**97.09 / 88.73**
	Gradient saliency	79.98 / 67.90	86.96 / 78.31	93.37 / 86.23
GQA	$G \times A$	**73.12 / 54.62**	**79.12 / 58.13**	**86.88 / 60.75**
	Gradient saliency	66.88 / 50.12	75.12 / 56.00	83.25 / 59.00
Qwen2.5-VL-7B				
MMBench	$G \times A$	75.79 / 73.30	83.34 / 79.30	**90.30 / 84.57**
	Gradient saliency	**77.99 / 74.54**	**84.11 / 80.04**	89.19 / 83.92
POPE	$G \times A$	**84.23 / 69.02**	**90.87 / 78.00**	**94.40 / 82.89**
	Gradient saliency	81.03 / 63.71	87.66 / 73.41	92.44 / 80.05
GQA	$G \times A$	**64.75 / 52.75**	**74.38 / 55.75**	**81.00 / 58.25**
	Gradient saliency	62.75 / 50.88	70.75 / 53.62	78.62 / 57.88

Table 18: $G \times A$ **minus attention rollout for each training seed** on LLaVA-1.5 MMBench (4,329 questions, global top-k). Entries are differences in answer preservation (pp) with 95% image-clustered bootstrap intervals. Seeds 13, 17, and 43 are averaged in Table 16.

Seed	5%	11%	22%
13	+3.12 [+1.86, +4.41]	+0.21 [−0.75, +1.17]	+0.32 [−0.40, +1.05]
17	+5.45 [+4.12, +6.81]	+1.13 [+0.09, +2.20]	+0.92 [+0.16, +1.70]
43	+2.26 [+1.13, +3.42]	+0.39 [−0.58, +1.35]	+0.25 [−0.53, +1.04]
67	+3.60 [+2.31, +4.92]	+0.97 [+0.00, +1.94]	+0.28 [−0.45, +1.00]
101	+5.34 [+3.87, +6.85]	+1.55 [+0.44, +2.67]	+0.99 [+0.16, +1.84]

tion (Table 25). Together, these results show that answer-attribution supervision remains effective beyond specific training data.

Table 19: **Training pools for the Qwen3-VL selector.** Mean answer preservation (%) over the six benchmarks. The last column averages MMBench, POPE, GQA, and MME, whose training splits are in no pool. All targets come from the model's own answers, and all selectors are decoded in one process. Bold marks the best pool in each column.

		5%			RCGD		RCGD 5%
Pool	Pairs	Top-k	RA	RCGD	11%	22%	four bench.
Default (filtered DeFacto)	684	76.14	78.06	79.43	86.16	90.77	82.24
All DeFacto	6,717	76.73	78.09	79.60	86.50	91.11	82.56
Mixed, size-matched	6,718	75.87	77.71	80.02	86.64	**91.15**	82.46
Mixed	19,129	**77.41**	**78.57**	**80.23**	**87.04**	91.04	**83.38**

Selector inputs and capacity. Table 20 removes one input group at a time. On single-question MMBench, visual and position features carry most of the signal, and question features matter mainly when one image receives several questions (Appendix G.1). Table 21 varies the scorer. Wider or deeper MLPs, a low-rank bilinear scorer, and pooled cross-attention do not improve 5% preservation

by a point or more with a positive lower bound, so we keep the default MLP. Table 22 replaces the pre-LM inputs with final-layer hidden states. They help LLaVA-1.5 on POPE (+4.2 F1 at 5%), hurt Qwen3-VL (-7.1 F1), and make no significant difference on Qwen2.5-VL, and they require a full language-model pass before selection.

Token-wise scores. $G \times A$ is a first-order score computed with every token present, while pruning removes many tokens at once, so the effect of dropping a token depends on which others remain. Local occlusion measures removal directly, one 2×2 region at a time, and trains a weaker selector than $G \times A$ (66.25 vs. 74.92% preservation on GQA and 72.81 vs. 81.27% on MMBench at 5%). When the same evidence is spread over several tokens, removing any one of them changes little, so single-removal scores underrate all of them.

Table 20: **Selector-input ablation** on LLaVA-1.5 MMBench. Entries are preservation / accuracy (%), averaged over three seeds that are trained separately from the main checkpoint. Bold marks the best variant in each column.

Selector input	5%	11%	22%
Global top-k			
Complete input	81.42 / 67.98	86.47 / 71.21	89.27 / 72.10
No position	81.39 / 68.15	86.09 / 70.96	89.04 / 72.04
No interaction	80.77 / 67.54	86.42 / 71.16	**89.28 / 72.55**
Interaction only, no position	80.18 / 67.54	85.86 / 70.73	88.91 / 71.89
Visual + question only	81.00 / 67.60	86.05 / 70.95	89.10 / 72.35
Visual + position only	**81.66 / 68.48**	**86.66 / 71.39**	89.14 / 72.46
Regional Allocation			
Complete input	84.19 / 69.79	86.84 / 71.24	89.23 / 72.11
No position	83.98 / 69.69	86.71 / 71.28	89.04 / 71.84
No interaction	83.85 / 69.71	87.06 / 71.26	**89.31** / 72.20
Interaction only, no position	83.38 / 69.62	86.26 / 71.01	88.83 / 71.81
Visual + question only	83.60 / 70.09	87.09 / 71.48	89.10 / **72.34**
Visual + position only	**84.42 / 70.42**	**87.36 / 71.79**	89.12 / 72.29

Table 21: **Scorer architecture** on LLaVA-1.5, evaluated on the 171 held-out DeFacto validation pairs. Val. Spearman is the correlation with the teacher on validation, head time is the scorer forward pass plus top-k (ms), and the last three columns give preservation (%). The default is MLP $[512, 128]$. Bold marks the best value in each column.

Scorer	Val. Spearman	Head ms	5%	11%	22%
MLP $[256, 64]$	.8148	.332	80.1	**88.3**	**90.1**
MLP $[512, 128]$	.8164	.456	83.0	87.7	**90.1**
MLP $[1024, 512, 128]$	.8201	.676	81.9	85.4	87.7
MLP $[512, 512, 256, 128]$	.8193	.478	75.4	86.0	87.7
Low-rank bilinear	**.8246**	**.315**	**83.6**	86.0	87.7
Pooled cross-attention	.8215	.538	77.2	86.0	88.9

Table 22: **Pre-LM inputs versus final-layer hidden states** on POPE (9,000 questions, global top-k). Entries are F1 / preservation (%); the full model reaches 84.62, 89.13, and 86.43 F1 on LLaVA-1.5, Qwen3-VL, and Qwen2.5-VL. Bold marks the better input in each pair.

Model	Features	5%	11%	22%
LLaVA-1.5	Pre-LM	80.60 / 89.94	83.97 / 93.33	85.22 / 94.89
	Hidden	**84.83 / 92.34**	**86.19 / 93.96**	**86.15 / 95.21**
Qwen3-VL	Pre-LM	**75.66 / 84.50**	**86.32 / 93.52**	**88.68 / 96.27**
	Hidden	68.55 / 79.80	84.58 / 92.12	87.98 / 95.58
Qwen2.5-VL	Pre-LM	57.37 / 78.26	76.80 / 89.28	82.79 / 94.17
	Hidden	**59.00 / 78.86**	**77.46 / 89.56**	**82.94 / 94.68**

F SELECTION-RULE ABLATIONS

Relevance, region, and diversity. Table 23 extends Table 4 to all budgets and three benchmarks. Global Diversity accounts for most of the diversity gain, and RCGD adds the regional constraint. Neither diversity rule wins everywhere. Global Diversity is higher in some Qwen POPE cells, and RCGD is higher on Qwen2.5-VL MMBench.

When diversity helps. The value of diversity depends on the backbone. Across the 45 backbone, benchmark, and budget cells of Table 23, whether RCGD beats RA agrees with whether region-free Global Diversity beats RA in 38, and both gains are largest on Qwen2.5-VL ($+3.0$ and $+3.2$ pp on average). One comparison of Global Diversity against RA on held-out data therefore decides the rule for a new backbone, and RCGD is the safer choice at high resolution, where RA falls below top-k (Appendix G.3).

Table 23: **Selection rules with identical Ansight scores.** Answer preservation (%) on MMBench (4,329 questions), POPE (9,000), and GQA (800). Global Diversity is the RCGD greedy rule without regional quotas.

	MMBench			POPE			GQA		
Allocation	5%	11%	22%	5%	11%	22%	5%	11%	22%
LLaVA									
Global top-k	80.55	86.67	89.19	89.94	93.33	94.89	74.75	79.88	82.88
Regional Allocation	83.90	86.88	89.56	91.56	93.72	94.87	76.50	80.12	82.75
Global Diversity	85.03	87.99	89.72	91.62	93.77	94.33	76.00	80.62	83.25
RCGD	85.08	87.73	89.97	92.74	94.01	94.38	76.88	80.50	83.25
Qwen3-VL									
Global top-k	79.26	86.72	91.25	84.71	93.87	96.99	68.12	77.88	87.12
Regional Allocation	81.89	87.92	91.61	88.33	94.00	97.09	73.12	79.12	86.88
Global Diversity	81.59	88.20	92.56	92.37	95.53	97.13	73.62	81.12	86.75
RCGD	81.03	88.08	91.98	91.37	95.39	97.21	73.00	80.62	87.62
Qwen2.5-VL									
Global top-k	72.14	82.47	89.26	78.26	89.28	94.17	58.38	71.75	79.75
Regional Allocation	75.79	83.34	90.30	84.23	90.87	94.40	64.75	74.38	81.00
Global Diversity	80.71	86.79	91.50	89.18	93.81	95.88	69.25	76.25	84.38
RCGD	81.34	87.48	91.61	87.98	92.63	95.23	69.00	77.00	83.38
LLaVA-NeXT									
Global top-k	87.62	89.74	91.91	92.22	94.27	96.00	75.50	80.25	86.63
Regional Allocation	89.91	90.46	91.96	93.71	95.22	96.27	80.25	82.25	87.38
Global Diversity	89.77	90.92	91.64	93.39	94.84	95.98	81.63	84.00	84.88
RCGD	89.56	90.46	92.35	93.39	95.27	95.70	81.25	84.13	87.75
LLaVA-OneVision									
Global top-k	89.51	92.47	95.06	93.43	96.04	97.34	75.75	81.63	85.75
Regional Allocation	91.38	93.44	95.10	94.76	96.20	97.42	80.50	83.75	85.25
Global Diversity	88.91	91.45	93.39	94.13	95.58	96.27	76.75	81.00	83.63
RCGD	90.37	92.33	94.06	94.76	95.93	96.64	79.13	81.88	85.25

Relevance-weighted versus uniform allocation. To separate spatial coverage from relevance weighting, we also tested uniform allocation, which gives every 4×4 cell the same share of the budget and ranks tokens within each cell by the same Ansight scores. At 5% retention, uniform allocation already improves on global top-k for every backbone we tested. The gain is large on LLaVA-1.5 (80.6 to 83.1% MMBench preservation) and Qwen2.5-VL (72.1 to 74.5% on MMBench and 78.3 to 85.6% on POPE), and small on Qwen3-VL (0.6 and 0.8 pp). Weighting cells by relevance helps most where coverage alone helps least. On Qwen3-VL, RA adds 2.1 pp on MMBench and 2.9 pp on POPE over uniform allocation. On Qwen2.5-VL it adds 1.3 pp on MMBench but loses 1.3 pp on POPE, and on LLaVA-1.5 the 0.8 pp difference is not significant. Most of RA's gain on LLaVA-1.5 and Qwen2.5-VL therefore comes from spreading the budget over the image, while on Qwen3-VL it comes from placing the budget where relevance is concentrated. POPE asks whether an object is present anywhere in the image, which favors even coverage and may explain why uniform allocation beats RA on Qwen2.5-VL POPE.

Other selection rules. On LLaVA-1.5 GQA at 5%, a determinantal point process (DPP) allocator that trades relevance against similarity on the same scores improves preservation by 1.0 pp (interval

$[-1.4, 3.5]$) at 4.7 times the selection cost. Reserving 20–30% of the budget for unconstrained top-k gives no gain over RA. Neither farthest-point sampling in feature space nor k-means cluster representatives (the token nearest each center) beat RA significantly. RA lowers the largest number of tokens in any cell from about 11 to 7 while keeping 87–88% of top-k's relevance mass.

Cell size and temperature. We sweep cell sizes of 2×2, 4×4, and 8×8 tokens and allocation temperatures $\tau \in \{0.5, 1, 2\}$ while holding relevance scores fixed (Table 24). None of the 80 alternative configurations significantly outperforms the default for either rule. RCGD is also less sensitive to these choices and only 3 configurations significantly reduce its performance, compared with 13 for RA, primarily at $\tau = 0.5$, where the budget becomes concentrated in fewer cells. This distinction is particularly pronounced on high-resolution Qwen2.5-VL DocVQA with 512 retained tokens. While RA alone can underperform global top-k at 2,048 and 4,104 input tokens, adding global diversity allows RCGD to match or exceed top-k in 26 of 27 configurations. These results highlight the importance of selecting complementary evidence within the allocated regional budgets, particularly under aggressive compression of high-resolution inputs.

Table 24: **Cell size and temperature on GQA** (800 questions). Entries are answer preservation (%) with fixed Ansight scores, and the default (4×4, τ=1) is shaded. $^{\downarrow}$ marks settings significantly below the default (paired bootstrap, Holm over eight settings). Spread is the range over the nine settings.

			2×2 cells			4×4 cells			8×8 cells			
Backbone	Ret.	Top-k	τ=0.5	τ=1	τ=2	τ=0.5	τ=1	τ=2	τ=0.5	τ=1	τ=2	Spread
Regional allocation												
LLaVA-1.5	5%	74.75	75.38	77.00	77.12	74.88	76.50	75.00	76.50	75.00	75.62	2.25
	11%	79.88	79.62	79.38	79.62	79.62	80.12	78.38	81.25	80.25	80.12	2.88
Qwen3-VL	5%	68.25	68.38$^{\downarrow}$	69.25$^{\downarrow}$	71.12	69.38$^{\downarrow}$	73.25	73.12	71.00	71.25	71.38	4.88
	11%	78.00	76.75	79.62	79.75	79.00	79.25	80.12	77.75	79.00	79.62	3.38
Qwen2.5-VL	5%	58.38	56.25$^{\downarrow}$	63.50	63.75	60.12$^{\downarrow}$	64.75	66.00	59.38$^{\downarrow}$	61.25	60.12$^{\downarrow}$	9.75
	11%	71.75	72.50	73.75	73.88	70.62$^{\downarrow}$	74.38	73.88	69.00$^{\downarrow}$	72.50	74.25	5.38
LLaVA-NeXT	5%	75.50	75.50$^{\downarrow}$	77.50	79.50	76.75$^{\downarrow}$	80.25	80.25	77.25	78.50	77.38	4.75
	11%	80.25	81.62	82.38	84.50	81.12	82.25	82.62	79.38$^{\downarrow}$	80.38	81.38	5.12
LLaVA-OV	5%	75.75	77.50	79.62	80.38	78.38	80.50	79.88	76.50$^{\downarrow}$	78.50	78.25	4.00
	11%	81.62	83.62	84.12	85.12	82.25	83.75	82.62	82.00	82.75	82.88	3.12
RCGD												
LLaVA-1.5	5%	74.75	76.88	77.38	76.38	77.88	76.88	78.38	78.38	77.00	75.75	2.62
	11%	79.88	79.62	81.12	80.50	80.25	80.50	79.88	81.50	80.00	79.88	1.88
Qwen3-VL	5%	68.25	69.88	70.50	70.62	70.88	72.88	73.38	72.62	72.88	72.75	3.50
	11%	78.00	77.25	80.75	81.62	79.75	80.50	79.50	80.12	80.62	80.88	4.38
Qwen2.5-VL	5%	58.38	59.38$^{\downarrow}$	67.25	68.12	62.88$^{\downarrow}$	69.12	70.00	65.50	71.62	72.12	12.75
	11%	71.75	74.88	77.25	77.88	74.75	77.00	77.75	75.25	77.75	79.25	4.50
LLaVA-NeXT	5%	75.50	78.25	80.75	81.00	79.75	81.25	79.88	79.50	81.12	80.38	3.00
	11%	80.25	82.38	84.25	84.25	83.12	84.12	83.38	82.62	82.88	82.50	1.88
LLaVA-OV	5%	75.75	78.88	79.38	79.38	78.00	79.12	77.00	77.88	76.38$^{\downarrow}$	76.50	3.00
	11%	81.62	82.00	81.75	83.25	81.38	81.88	82.38	81.00	82.12	82.12	2.25

G Broader Applications

G.1 Multi-Turn Dialog

Dialog selector. The dialog selector splits relevance into an image score b_i that is shared across turns and corrections for the current question and the dialog history. Each visual token and its position encoding are projected to 128 dimensions, and an MLP maps them to b_i. The current question and the history are mean-pooled from frozen input embeddings and projected to $\tilde{q}_t$ and $\tilde{h}_t$. Two residual MLPs score $[\tilde{v}_i, \tilde{q}_t, \tilde{v}_i \odot \tilde{q}_t]$ and $[\tilde{v}_i, \tilde{h}_t, \tilde{v}_i \odot \tilde{h}_t]$ to give corrections $d_i^{(t)}$ and $e_i^{(t)}$, each centered over tokens. All MLPs have hidden widths $[128, 64]$.

For a dialog of M turns with attribution targets $a^{(t)}$, let $\bar{a} = M^{-1} \sum_t a^{(t)}$, let $\mathcal{D}(s, a)$ be Eq. 2 with $\lambda = 0.25$, and let $s^{(t)} = b + d^{(t)}$. With H the Smooth-L1 loss, z_D per-example standardization,

$\hat{a}^{(t)} = z_D(\log(1 + a^{(t)}))$, and $\hat{a}_{\text{img}} = z_D(\log(1 + \bar{a}))$, we minimize

$$\mathcal{L}_{\text{dialog}} = \mathcal{D}(b, \bar{a}) + \frac{1}{M}\sum_t \mathcal{D}(s^{(t)}, a^{(t)}) + \frac{0.5}{M}\sum_t H\Big(z_D(d^{(t)}), \hat{a}^{(t)} - \hat{a}_{\text{img}}\Big)$$
$$+ \frac{0.25}{M-1}\sum_{t=1}^{M-1} H\Big(z_D(s^{(t+1)}) - z_D(s^{(t)}), \hat{a}^{(t+1)} - \hat{a}^{(t)}\Big)$$
$$+ \frac{0.25}{M}\sum_t \mathcal{D}(s^{(t)} + e^{(t)}, a^{(t)}) + \frac{0.01}{MN}\sum_{t,i}(e_i^{(t)})^2.$$

The image-only, current-question, and history-conditioned variants use b, $b + d^{(t)}$, and $b + d^{(t)} + e^{(t)}$ from the same model. A shared core of $\text{round}(\rho k)$ tokens is chosen once per image by b_i, and each turn fills the remaining budget by $s^{(t)}$ from outside the core.

Setup and results. We use 200 held-out Visual Dialog dialogs (2,000 turns) on LLaVA-1.5, with a 400-dialog training split that shares no images with them. The language model always sees the same caption and reference-answer history, and only the visual-token selection changes (Table 25). Adding history to the selector gives no benefit beyond the current question. The 25% core passes a one-point non-inferiority test at every budget, while a 50% core fails at 5% (lower bound -1.1 pp).

Table 25: **Multi-turn dialog on LLaVA-1.5.** Answer preservation (%) on 200 held-out Visual Dialog dialogs of ten turns. Core percentages refer to the retained budget.

Selection	5%	11%	22%
Random	77.90	82.55	87.15
Image-only	79.75	84.35	87.65
Current question	86.50	88.60	91.00
Current question + history	86.90	88.40	91.20
25% shared image core	86.80	89.10	91.05

Without the image, the full model still gives its original answer on 1,411 of the 2,000 turns. On the remaining 589, question-conditioned selection beats image-only selection by 20.6, 11.6, and 9.8 pp at 5%, 11%, and 22%, and the 25% core stays on par with full question conditioning. Maps for different questions about the same image are only moderately correlated (mean Spearman 0.55), and using another question's map for the same image costs 13.8, 10.4, and 6.2 pp. Shared image structure therefore does not make token subsets interchangeable between questions. With the dialog history re-prefilled each turn, question-conditioned selection speeds up each turn by 1.23–1.26×.

G.2 Long Video

We use the long-duration Video-MME subset (30–60 minute videos, no subtitles), with 300 videos and 900 questions, and apply the Qwen3-VL selector without retraining. Each video is encoded once per frame count, and all methods for a question are decoded in one process against the same full-token reference. Uniform temporal selection spreads the budget evenly over frames. DivPrune, CDPruner, FastV, and ZOO-Prune are ports to the Qwen3-VL video interface. Two reduced-input settings give the unpruned model a comparable budget directly, either with fewer frames (4–18) or with all 128 frames at lower resolution. All video selections use global top-k, since dense RCGD would need a distance matrix over more than 40,000 tokens (Table 14).

Conditioning on the current question makes no detectable difference. It is 0.1 pp above a fixed query and 0.8 pp above a different question about the same video, with both intervals containing zero. Even with a fixed query, Ansight preserves 81.67% of answers, against 75.78% for uniform temporal selection and at most 78.89% for the ported baselines, so the gain comes from the learned answer-sensitive relevance. Across a video's three questions, the selected sets overlap with a median Jaccard index of 0.48, but their distribution over frames barely changes (median histogram overlap 0.90). Selection changes which patches are kept within frames rather than which frames are kept. The 2.9 pp gain over DivPrune is not significant, and the gains over the reduced-input settings and the other ports are, after Holm correction (Table 26).

Table 26: **Long-video quality on Video-MME** (Qwen3-VL-2B, 128 frames, 900 questions). Entries are accuracy / preservation (%) at absolute token budgets; the full model uses 42,565 tokens on average and reaches 51.22% accuracy. The mean-of-three-questions row assumes all questions are known in advance. Reduced-input settings shrink the input instead of pruning it and are run only near 2,048 tokens. Bold marks the best value in each column.

Selection	2,048 tokens	1,024 tokens
Ansight (current question)	**48.78** / 81.78	47.44 / 77.22
Ansight (fixed query, question-independent)	47.89 / 81.67	47.67 / 77.33
Ansight (different question, control)	48.67 / 81.00	46.83 / 77.89
Ansight (mean of three questions, oracle)	48.56 / **82.00**	**48.11** / **79.00**
Uniform temporal	45.33 / 75.78	44.11 / 71.78
DivPrune	45.78 / 78.89	45.56 / 74.67
CDPruner	44.22 / 75.11	45.22 / 72.67
FastV	46.00 / 72.56	40.89 / 65.56
ZOO-Prune	44.44 / 74.56	42.89 / 72.11
Full model, fewer frames	40.44 / 63.00	–
Full model, lower resolution	45.89 / 75.44	–

For cost (Table 27), both methods reuse one decoded and encoded video and hold the same cached features, and each measurement runs from the tokenized question to the first answer token. Peak memory above the cached state falls from 7.88 to 1.09 GB at 128 frames. Scoring all cached tokens in one batch instead of 64-token chunks cuts per-question time to 49, 66, and 103 ms at 64, 128, and 256 frames, at the cost of more peak memory.

Table 27: **Per-question cost on long video** (64 videos, 192 questions). Both methods start from the same encoded video and cached features. Values are medians over questions, and speedups are paired per-question ratios (128-frame interval $[4.27, 4.36]$). Bold marks the lower value.

Frames	Method	Visual tokens	Latency (ms)	KV cache (GB)
64	Full	23,040	335.4	2.51
	Compact (Ansight)	2,048	**114.0** ($2.89\times\downarrow$)	**0.26** ($9.5\times\downarrow$)
128	Full	46,080	864.1	5.00
	Compact (Ansight)	2,048	**196.2** ($4.34\times$)	**0.30** ($16.6\times\downarrow$)
256	Full	92,160	2,633.7	10.00
	Compact (Ansight)	2,048	**361.3** ($7.19\times$)	**0.37** ($26.8\times\downarrow$)

G.3 High-Resolution Documents

We take native-resolution DocVQA validation images that support 1,024, 2,048, and 4,104 visual tokens on Qwen2.5-VL without upsampling, and keep 512 tokens at every size. Table 28 times the full model, RA, and RCGD on the first 128 images. Prefill speeds up by $1.45\times$, $2.62\times$, and $5.18\times$, and cached-vision latency by $1.30\times$ to $4.22\times$ for RA. From the raw image the speedups are 1.03–$1.20\times$, because the fixed shared vision encoder grows with resolution. RCGD costs 5–8 ms more than RA but recovers much of the lost preservation.

Table 29 adds the pre-LM baselines on 384 images, all decoded in one process with the full-context reference. RCGD preserves significantly more answers than RA, CDPruner, ZOO-Prune, and random selection at every resolution. It is also ahead of DivPrune at every resolution, by 2.1–4.4 pp.

Table 28: **Timing at high resolution** (Qwen2.5-VL, 128 DocVQA images, 512 tokens kept). Prefill is language-model time, cached is time to the first token from cached vision features (speedup in parentheses), and raw is end-to-end time from the image, all in ms. Allocator is the RA or RCGD selection time, KV is in MiB, and ANLS (average normalized Levenshtein similarity) measures agreement with the GT answer. The $B{=}1024$ rows are untimed quality references.

Full tokens	Selection	Prefill	Cached	Raw	Allocator	KV	ANLS	Preservation
1,024	Full	31.92	33.07	171.04	–	58.1	0.957	1.000
	RA	21.97	25.35 (1.30×)	163.56	0.37	30.1	0.840	0.789
	RCGD	21.96	30.18 (1.10×)	167.73	5.20	30.1	0.897	0.859
	RA, $B{=}1024$	–	–	–	–	–	0.957	1.000
2,048	Full	58.75	59.88	288.55	–	114.1	0.943	1.000
	RA	22.29	26.45 (2.27×)	256.71	0.52	30.1	0.760	0.633
	RCGD	22.27	32.05 (1.87×)	261.95	5.87	30.1	0.844	0.773
	RA, $B{=}1024$	–	–	–	–	–	0.898	0.867
4,104	Full	117.86	118.76	529.58	–	226.5	0.946	1.000
	RA	22.67	28.07 (4.22×)	441.07	0.97	30.1	0.717	0.586
	RCGD	22.68	36.49 (3.25×)	449.34	8.73	30.1	0.785	0.680
	RA, $B{=}1024$	–	–	–	–	–	0.826	0.750

Table 29: **Pre-LM baselines at high resolution** (Qwen2.5-VL, 384 DocVQA images, 512 tokens kept). Entries are preservation (%) and ANLS (×100). All methods are decoded in one process with the full-context reference. Bold marks the best pruned value in each column.

	1,024 full tokens		**2,048 full tokens**		**4,104 full tokens**	
Selection	Pres.	ANLS	Pres.	ANLS	Pres.	ANLS
Full context	100.0	94.4	100.0	95.0	100.0	95.4
Random	66.9	81.1	49.0	68.0	40.6	59.4
DivPrune	85.4	**91.2**	76.0	85.3	65.1	79.7
CDPruner	62.5	75.7	47.1	64.4	34.1	52.5
ZOO-Prune	71.4	83.3	55.5	71.8	31.8	50.1
Ansight + RA	79.9	86.0	63.8	75.4	55.7	70.9
Ansight + RCGD	**87.8**	90.6	**78.1**	**85.4**	**69.5**	**81.8**

H Limitations and Future Work

Ansight already outperforms prior pruning methods at matched budgets, and several directions could extend its gains. It scores tokens after the vision encoder, so the encoder still processes every patch. At 4,104 visual tokens this limits the raw-image speedup to $1.20\times$, against $4.22\times$ from cached features (Appendix G.3). Predicting relevance inside or before the encoder would remove much of this cost. Integration with serving systems, such as fused selection kernels and batched prefill, could also turn more of the saved computation into lower latency on short visual sequences. Finally, adaptive budgets and a single selector shared across backbones are natural next steps.