BIFOCAL: TRAINING-FREE MIXED-GRANULARITY VISUAL TOKEN PRUNING FOR MULTIMODAL LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Training-free visual token pruning allocates a fixed budget among candidate tokens, and existing methods keep refining the scoring function, hovering near the ceiling of the general benchmarks, yet most collapse on fine-grained tasks such as grounding. We instead ask what visual evidence the LLM needs at inference, and find two answers. First, the evidence can come from only two sources, the image and the question, and no single score serves every task. Second, keeping only a subset of native patches breaks the integrity of the visual sequence in both content and position. Building on both findings, we present **Bifocal**, a training-free, two-stage visual token pruning method built on mixed-granularity evidence. Before the LLM, it keeps two kinds of fine evidence and a coarse context. Query-focused evidence keeps the tokens the question refers to, image-intrinsic evidence maximises how well the token set represents and covers the image, and a pooled coarse context preserves the dual integrity of the retained tokens. Inside the LLM, it further keeps the top-K tokens by text-to-image attention. Evaluated across three models and several budgets, spanning general VQA, reasoning, OCR, hallucination and grounding, Bifocal achieves state-of-the-art results: it attains the best general average in every setting while, at the tightest budget, leading the strongest baseline on grounding by 9.9 to 33.1 percentage points.

1 INTRODUCTION

Visual tokens are the main source of inference cost in multimodal large language models, so dropping some of them during inference is the most direct way to make these models faster (Chen et al., 2024; Jin et al., 2025; Shao et al., 2026). A training-free, plug-and-play pruner has only two things to go on at inference time: the image itself (Yang et al., 2025; Zhang et al., 2025b; Alvar et al., 2025), and the sentence being asked (Zhang et al., 2025c; Huang et al., 2024; Li et al., 2026). Existing methods have been wavering between the two, one scoring function after another, trading fractions of a point under the ceiling of the general benchmarks, while on fine-grained tasks they collapse (Endo et al., 2025; Chien et al., 2025; Yang et al., 2026).

We go back to what causes this lopsidedness. Inside the LLM, causal self-attention already enables deep vision–language interaction, which pruning methods can naturally leverage (Chen et al., 2024; Zhang et al., 2025c). Pruning before the LLM asks a different question: what evidence does the model need to be given in order to think about this question at all? Approached this way, the lopsidedness turns out not to be a matter of scoring functions being insufficiently accurate. Evidence simply comes in two kinds, what the image itself makes salient and what the question points to, and different tasks need them in different proportions (Radford et al., 2021; Li et al., 2023a; Liu et al., 2023), so any single scoring function is bound to lean to one side, with the task deciding which side.

We therefore propose a training-free mixed-granularity pruning framework. Combining coarse and fine granularity is a long-standing idea, from the image pyramid (Burt & Adelson, 1987; Lin et al., 2017) to recent high-resolution multimodal models (Li et al., 2025; Huang et al., 2025). We bring the idea to pruning: a coarse context spends a small part of the budget on a low-resolution covering of the whole image and preserves the dual integrity of the retained tokens, while the fine evidence comes in two kinds, query-focused evidence for the content the question points to and image-intrinsic evidence for what the image itself makes salient (Yuan et al., 2026). The three have clear-cut goals: query-focused evidence keeps as much of the content the question is about as it can, image-intrinsic

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.

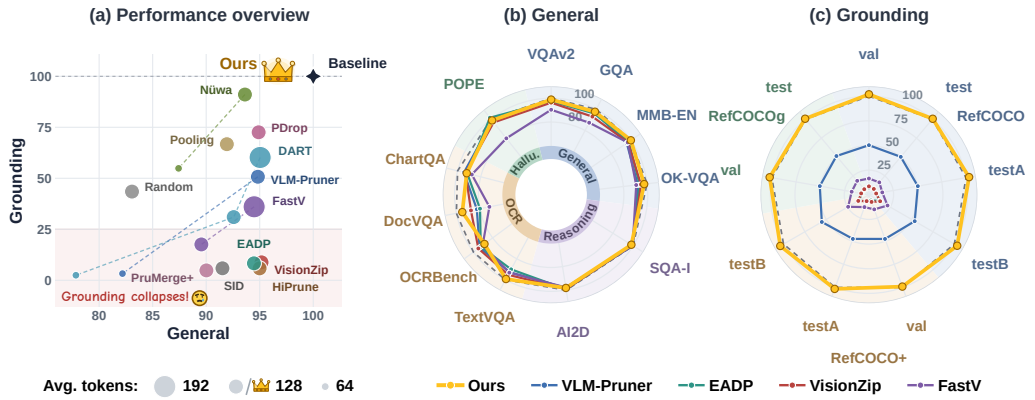


Figure 1: **Performance overview of Bifocal on LLaVA-1.5-7B. (a)** General versus grounding performance across pruning methods. **(b, c)** Radar charts over 20 evaluations in five ability groups, at an average of 128 visual tokens. Bifocal achieves state-of-the-art overall performance.

evidence covers what is diverse beyond the question, and the coarse context maintains the scattered fine tokens as a coherent whole.

The two kinds of fine evidence are where the care goes. For query-focused evidence, the words of a sentence do not contribute equally when one decides which token matters, so we compute a saliency map per word and let an image saliency map, defined as the residual of a regression, referee the word weights: a word earns a larger vote the more the places it attends to agree with the places the image itself finds salient. The word saliency maps are then conjoined by a geometric mean into the query-focused score. For image-intrinsic evidence, we measure how well the tokens already kept represent the image, and search the remaining patches for those that represent what has not yet been expressed. Finally we lay down an evenly pooled coarse context, and the mixed-granularity representation is complete.

Bifocal is evaluated on LLaVA-1.5-7B (Liu et al., 2024a), LLaVA-NeXT-7B (Liu et al., 2024b) and Qwen2.5-VL-7B (Bai et al., 2025) across several budgets, covering five kinds of ability: general VQA, reasoning, OCR and documents, hallucination, and grounding. The baselines include both the classic pruning methods and recent, strong open-source work, each reproduced by us. On the general tasks Bifocal attains the best average at every budget, ahead of several recent methods optimised chiefly for general ability. On the fine-grained side the margin is not subtle: on LLaVA-1.5-7B grounding still exceeds the unpruned model at an average of 128 tokens, where several other methods are down to single digits. We also measure end-to-end efficiency, and obtain speed-ups beyond most methods while keeping accuracy. Ablations and visualisations then show that the method does behave differently when the question changes.

Our contributions are threefold. **(1) A mixed-granularity framework for visual token pruning.** Across 20 benchmarks we analyse what visual evidence the LLM needs, and show that a low-resolution coarse context preserves the dual integrity of the retained tokens and, together with the fine evidence, keeps performance balanced across tasks. **(2) Bifocal, a training-free two-stage method.** Before the LLM, it instantiates the mixed-granularity framework with a dedicated selection rule for each kind of token; inside the LLM, it keeps the top-K tokens by text-to-image attention. **(3) Evaluation on three models, three budgets and five abilities**, where Bifocal achieves the best overall performance in all nine settings.

2 Mixed-Granularity Evidence Selection

2.1 Information: Two Sources

Existing training-free pruning methods often exhibit pronounced task-wise biases . We therefore turn the question around: ***what visual evidence does an LLM actually need to solve different tasks?***

At inference time, a training-free pruning method has access to only two sources of information: the image itself and the question. We therefore examine how token selections driven by each source

behave under the same budget. We fix the token budget at one quarter of the visual tokens and turn each source into a route with a common, simple scoring rule: the image route ranks patches by CLS attention, and the text route ranks them by their cosine similarity to the mean word embedding of the question.

Figure 2 (left) shows that the two routes retain noticeably different regions, and their selections overlap only slightly. Figure 2 (right) shows that this difference is systematic across tasks: OCR and document tasks favour the image route, REC tasks favour the text route, and the general benchmarks show much smaller gaps. Relying on either source alone therefore leads to a clear task-wise bias, which suggests that the two sources provide complementary rather than interchangeable evidence. We further include two references: random retention and an even split of the budget between the two routes. The even-split curve largely tracks the upper envelope of the two single-route curves; its advantage is not concentrated on any particular task but lies in robustness across tasks. **Appendix B analyses the effect of each single route.**

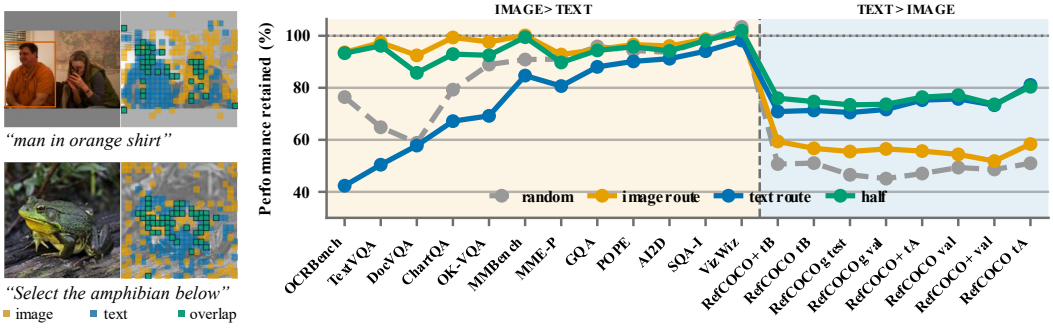


Figure 2: **The image and text routes under a fixed token budget.** ■ image route; ■ text route. **Left:** tokens selected by each route on two examples; ■ marks tokens selected by both. **Right:** performance relative to the baseline on 20 benchmarks at 144 of 576 tokens; ■ *half* splits the budget evenly between the two routes, and ■ *random* retains tokens at random.

2.2 Structure: Two Integrities

Even after combining the two routes, grounding remains the least recovered capability. Is the remaining gap simply a matter of inaccurate token selection? Figure 3 suggests otherwise. In ③, we push selection to the extreme and retain only the patches on the ground-truth target, yet localisation falls well below the two-route selection in ②. Conventional visual token pruning methods, including our two routes, share the same compression paradigm: retaining a subset of native patches from the full visual sequence. ***This subset-based compression breaks the integrity of the original visual sequence in two ways: positional integrity and content integrity.***

Positional integrity. Subset pruning leaves tokens at only a sparse and irregular set of positions, breaking the global positional organization of the original visual sequence. Starting from this extreme strategy in ③, we progressively restore coverage beyond the target—first around its boundary (④), then across the full image with random or lattice layouts (⑤, ⑥). Localisation improves substantially as the retained tokens recover a more complete image-wide distribution.

Content integrity. token pruning also incurs an unavoidable loss of content. To examine this effect, we keep the image-wide positional layout fixed and vary only what the distributed tokens contain. In ⑧, all distributed tokens carry the same image-wide average, yet localisation improves over the native lattice in ⑥. Replacing this global summary with region-specific summaries (⑨) improves performance further. Discarded content need not be preserved at its original resolution, but retaining a low-resolution account of it is important for content integrity.

A layer of pooled coarse tokens addresses both deficiencies at once: each coarse token summarizes content that would otherwise be discarded, while together they restore an even positional layout over the image. The strong performance of ⑨ confirms the effectiveness of this design. This closely mirrors the coarse–fine decomposition of classical image pyramids (Burt & Adelson, 1987), where

	Visual tokens	Center ↑	ACC@0.5 ↑
①	No pruning, 576 patches	91.07 / 85.16	64.47 / 47.48
②	144 patches, both routes	85.95 / 77.64	51.83 / 35.47
③	144 patches on the GT bbox	71.92 / 65.78	39.14 / 27.80
④	80 patches + 64 rim	81.37 / 79.25	42.50 / 29.62
⑤	80 patches + 64 random	88.19 / 84.20	54.52 / 40.82
⑥	80 patches + 64 lattice	87.64 / 84.79	54.98 / 40.88
⑦	144 summaries, no patches	85.66 / 77.74	44.62 / 29.11
⑧	80 patches + 64 blanked	<u>89.92</u> / **86.65**	<u>56.62</u> / <u>42.59</u>
⑨	80 patches + 64 pooled	**90.53** / <u>86.12</u>	**63.02** / **46.18**

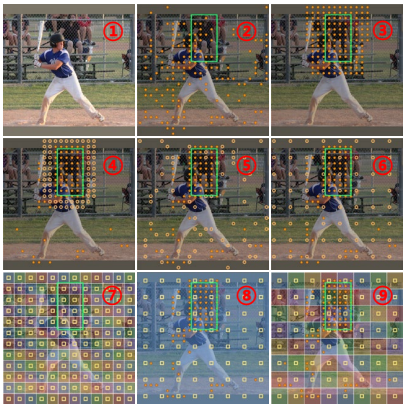


Figure 3: **Left:** experiments testing positional and content integrity. Every row except ① keeps 144 of the 576 visual tokens; each cell reports RefCOCO testA / testB, with the best pruned result in bold and the second best underlined. **Right:** the tokens each row retains on one example. ground-truth box; predicted box; retained patch; extra native patch; pooled token, averaging its tile.

local detail and global structure are represented at different resolutions. It also motivates our mixed-granularity design: fine tokens preserve task-relevant native detail, while coarse tokens recover, at a lower resolution, the content and positional integrity lost through subset selection. **Appendix C validates the framework by integrating existing pruning methods into it.**

3 METHODOLOGY

Figure 4 gives an overview of the design of Bifocal. We describe query-focused evidence in §3.1, image-intrinsic evidence in §3.2, coarse context in §3.3, and Stage 2 in §3.4.

3.1 QUERY-FOCUSED EVIDENCE

Query-focused evidence assigns each patch a question-dependent score I_j and retains the top-K patches as the evidence the question needs. Existing methods typically compress the question into a single vector and score each patch by its cosine similarity to it. Whether the words are pooled by a max or a mean, a strong match to one word can stand in for the absence of another, an OR-like rule under which a "red shirt" is no different from a red fire hydrant elsewhere in the image. Since no model reads the question as a sentence before the LLM, we treat it as a set of words and model two ingredients explicitly: word weighting and word conjunction. For weighting, we invite the image back to sit on the jury alongside the question: a word earns a louder voice when its saliency map agrees with what the image itself finds salient. For conjunction, we combine the words with a geometric mean, a soft AND that favours patches supported by all relevant words jointly.

Word Saliency Maps We construct a word saliency map for each word to locate the image patches it attends to. Let e_w be the embedding of word w and p_j the feature of patch j after the projector maps it into the LLM space. Cosine similarity measures how strongly word w attends to patch j:

$$s_w(j) = \cos(\hat{e}_w,\ \hat{p}_j), \qquad \hat{e}_w = e_w/\|e_w\|. \tag{1}$$

In fact, we observe that this commonly used operation of assigning each patch a score via cosine similarity leads to large score gaps between neighboring patches on the same object. Such discrepancies arise from estimation noise rather than genuine differences in content. Because patches are selected by ranking these scores, the noise can promote patches that happen to score high while displacing those that actually lie on the target. Fortunately, signal and noise occupy different spatial scales: an object spans several patches, so the correlation of interest is spatially coherent, whereas estimation noise fluctuates independently from patch to patch. We find that, on a word saliency map, a simple spatial smoothing $\mathcal{G}$ is enough: this low-pass filter averages away uncorrelated noise while leaving extended structure largely intact, yielding $\tilde{s}_w = \mathcal{G}\, s_w$. We then min-max normalise each smoothed map over the patches, $\hat{s}_w = \mathrm{mm}(\tilde{s}_w)$, so that the conjunction compares the spatial shape of the word maps rather than their overall level, which is typically high for content words and low for function words. **Appendix D analyses the smoothing further.**

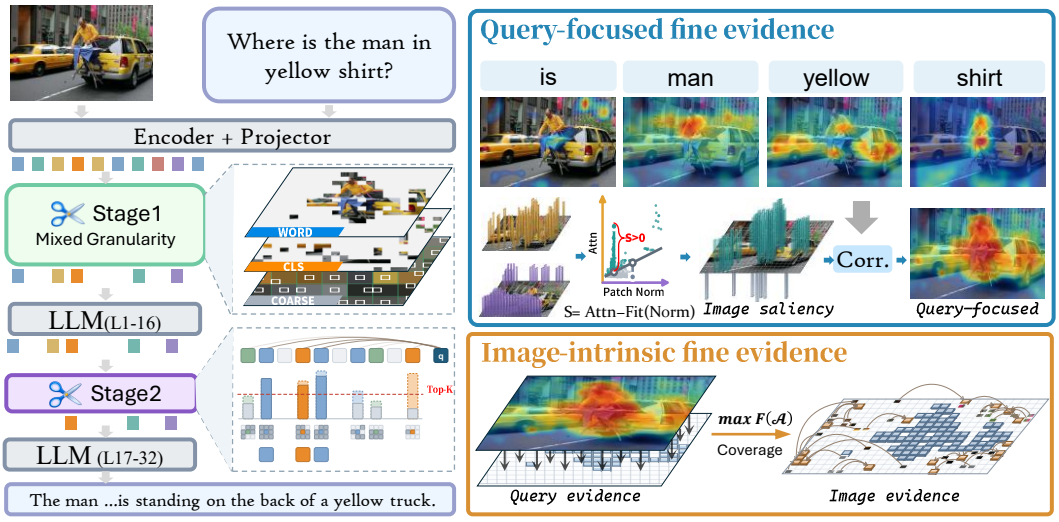


Figure 4: **Overview of Bifocal**, a training-free, two-stage visual token pruning method. **Stage 1** runs before the LLM and retains a mixed-granularity representation: fine evidence at native-patch resolution, of two kinds, query-focused evidence selected for the question and image-intrinsic evidence selected from the image itself, and coarse tokens obtained by average pooling. **Stage 2** keeps the top-K tokens by text-to-image attention inside the LLM.

Image Saliency Map Word weighting needs a referee for the word saliency maps, and **Appendix B.1** suggests a natural one: what the image itself attends to after the ViT. Following common practice, we start from the attention of the ViT's CLS token, which aggregates global information. It is not, however, a clean content map: a substantial part of the CLS score follows the norm of a patch rather than its content, most visibly in the attention sink (Darcet et al., 2024; Xiao et al., 2024), a few patches of abnormally large norm that mostly sit on low-information background. Since this map weights the words, one that points at the background hands the say to words unrelated to the query.

The reason, derived and verified in **Appendix E**, is that the CLS attention factorises exactly as $a_j = n_j\, c_j$, a gain n_j set by the norm times a content score c_j. To obtain the content saliency, we need to decouple the norm from the content. Neither factor is observed, so we separate the product directly in the attention, using only the attention scores and the norm. By the second-order Taylor expansion of xy about $(\bar{x}, \bar{y})$, which is exact, for any x, y and any reference values $\bar{x}, \bar{y}$,

$$xy = \bar{y}\, x + \bar{x}\, y - \bar{x}\bar{y} + (x - \bar{x})(y - \bar{y}). \tag{2}$$

Substituting $x = n_j$ and $y = c_j$, with the reference values taken to be their image-wide means $\bar{n}$ and $\bar{c}$:

$$a_j = \bar{c}\, n_j \;+\; \bar{n}\, c_j \;-\; \bar{n}\bar{c} \;+\; \big(n_j - \bar{n}\big)\big(c_j - \bar{c}\big). \tag{3}$$

With the means as reference values, the four terms separate cleanly. The first depends only on the gain and is linear in it; the second ranks patches exactly as c_j does; the third is a constant; only the cross term involves both, and it is small wherever the gain stays near its mean. Removing the first term therefore leaves, up to a constant and the cross term, a map that ranks patches by content. The first term is not observable, but the gain rises with the norm, so over one image we approximate it by a linear function of the smoothed norm map ν of the projected patch features: we regress the smoothed CLS attention $\tilde{a} = \mathcal{G}a$ on ν over the patches of the image by least squares and subtract the fit patch by patch. If content does not co-vary with the norm, the fit recovers the first term; whatever does co-vary with the norm is removed along with it. The residual is the image saliency map:

$$S_j = \tilde{a}_j - \big(\hat{\beta}\, \nu_j + \hat{b}\big). \tag{4}$$

Word Weighting and Word Conjunction With both maps in hand, each word is weighted by how well its saliency map agrees with the image saliency map:

$$\alpha_w \;\propto\; \mathrm{relu}\big(\mathrm{corr}(\hat{s}_w,\, S)\big), \qquad \sum_w \alpha_w = 1, \tag{5}$$

where corr is the Pearson correlation over patches and relu discards words whose map disagrees with the image. The words are then combined by a weighted geometric mean, a soft AND under

which a patch scores high only if every weighted word points to it:

$$R_j \;=\; \prod_w \hat{s}_w(j)^{\,\alpha_w}. \tag{6}$$

The norm map ν removed above also carries information of its own: high-norm patches mostly lie on background, so $B = -\nu$ serves as a query-independent background penalty. The query-focused score combines the three maps by an unweighted geometric mean of their min-max normalised versions:

$$I_j \;=\; \big[\mathrm{mm}(R_j)\cdot \mathrm{mm}(S_j)\cdot \mathrm{mm}(B_j)\big]^{1/3} \tag{7}$$

3.2 Image-Intrinsic Evidence

With part of the token budget spent on query-focused evidence, and following the analysis of Figure 2, we now select a second kind of evidence that comes from the image itself. Query-focused evidence concentrates on the object the question names; image-intrinsic evidence spends its budget elsewhere, on the N_i patches that best represent the rest of the image. This calls for a measure of how well the image is represented. Let $\mathcal{A}$ be the set of tokens currently in place. For each patch j we take its similarity to the closest token in $\mathcal{A}$, and sum these values weighted by the CLS attention a_j, we call the result the *representational coverage* of $\mathcal{A}$:

$$F(\mathcal{A}) \;=\; \sum_j a_j \cdot \max_{i\in\mathcal{A}} \cos(x_j,\, x_i) \tag{8}$$

Because of the $\max$, a patch needs only one representative, and a near-copy of a token already in place adds nothing. F is monotone and submodular, so greedy selection is near-optimal: at each step we add the patch with the largest marginal gain,

$$\Delta(t) \;=\; F(\mathcal{A}\cup\{t\}) - F(\mathcal{A}) \;=\; \sum_j a_j \,\mathrm{relu}\big(\cos(x_t, x_j) - \mathrm{cov}_j\big), \qquad \mathrm{cov}_j = \max_{i\in\mathcal{A}} \cos(x_j, x_i) \tag{9}$$

where cov_j is how well patch j is represented so far (Algorithm 1, Appendix F).

3.3 Coarse Context

Following the pooling analysis of Figure 3, we add a layer of pooled coarse tokens after the fine evidence. We make one adaptation here: the content of each pooling block averages only the patches inside the block that were *not* selected as fine evidence, and its position is taken to be the attention-weighted centroid of the block. Let $\mathcal{B}_b$ denote the members of block b and $\mathcal{Q}$ the patches selected as fine evidence, so that the residual set is $\mathcal{R}_b = \mathcal{B}_b \setminus \mathcal{Q}$. Then

$$\bar{x}_b = \sum_{j\in\mathcal{R}_b} \frac{a_j}{\sum_{k\in\mathcal{R}_b} a_k}\, x_j, \qquad \hat{r}_b = \Big\lfloor \sum_{j\in\mathcal{B}_b} \frac{a_j}{\sum_{k\in\mathcal{B}_b} a_k}\, r_j + \tfrac{1}{2} \Big\rfloor, \tag{10}$$

where r_j is the grid coordinate of patch j, the rounding puts the centroid back on the nearest patch inside the block, and $\hat{r}_b$ is the position this coarse token takes in the sequence. Under a fixed budget this reduces redundancy among the tokens: every patch is either present natively or enters exactly one coarse token.

3.4 Stage 2: Pruning inside the LLM

Stage 1 decides which visual evidence enters the LLM, yet the visual tokens remain highly redundant inside it (Chen et al., 2024). By then the causal attention has also let them absorb the question, so the text-to-image attention reliably reflects which visual information the answer still needs. Following FastV, at layer L we score each visual token by the attention t_j it receives from the last text token and keep the top-K.

We make one adjustment, with the integrity of the retained set in mind. After Stage 1 the grid is sparse, and this attention often peaks on isolated survivors whose neighbours have been pruned. We therefore weight each score by a *local-density prior*:

$$\tilde{t}_j = t_j \cdot d_j, \tag{11}$$

where d_j is the fraction of the neighbourhood of token j that survived Stage 1. Pruned patches no longer receive attention; they enter Stage 2 only by lowering the density of their neighbours.

Table 1: **General multimodal benchmarks on LLaVA-1.5-7B.** Methods are matched by layer-averaged visual-token budget. Within each budget, the best and second-best results are **bold** and underlined. Avg. averages benchmark-wise performance relative to the baseline.

Method	General				Reasoning		OCR				Hall.	Avg.
	VQA^{v2}	GQA	MMB	VQA^{OK}	SQA-I	AI2D	VQA^{T}	OCR-B	DocVQA	ChartQA	POPE	
Baseline	78.5	61.9	64.1	53.5	69.4	55.2	46.0	314	21.5	18.2	85.9	100.0%
Average Tokens = 192												
FastV	75.0	57.0	63.6	51.1	69.1	54.1	44.6	300	18.2	17.1	75.8	94.5%
PDrop	77.4	60.0	63.5	52.4	68.9	54.1	45.9	308	19.9	**18.2**	85.3	98.1%
PruMerge+	74.6	57.6	61.3	48.6	69.0	54.6	39.0	282	14.8	16.5	79.9	90.9%
VisionZip	76.8	59.3	63.8	52.8	68.7	55.5	44.5	312	19.4	17.4	85.3	97.5%
SID	76.4	58.9	63.3	51.9	68.9	54.9	40.9	289	15.4	17.0	85.8	93.9%
DART	76.7	59.5	62.9	51.3	68.9	54.7	43.6	300	19.6	17.5	81.8	96.2%
Nüwa	77.8	61.0	63.8	**53.5**	68.6	54.4	45.7	244	20.2	**18.2**	84.6	96.7%
EADP	77.2	60.3	62.9	51.3	68.8	54.5	43.3	301	19.3	17.2	**87.4**	96.7%
VLM-Pruner	77.9	61.2	62.0	51.3	**69.9**	55.1	44.1	299	19.5	17.6	85.6	97.2%
HiPrune	76.8	59.2	63.0	52.4	68.8	54.5	44.9	**316**	20.5	17.8	86.1	98.1%
Bifocal (ours)	**78.2**	**61.5**	**64.4**	53.4	68.4	**55.6**	**46.2**	284	**21.6**	17.0	85.7	**98.5%**
Average Tokens = 128												
FastV	71.0	54.1	62.0	48.6	69.1	53.9	42.9	284	15.4	16.0	68.2	89.5%
PDrop	74.9	57.4	61.4	50.4	**70.9**	52.9	44.5	**299**	18.2	17.5	81.8	94.9%
PruMerge+	74.0	57.6	60.3	47.2	69.7	54.3	39.2	279	14.2	15.9	81.1	90.0%
VisionZip	75.6	57.7	62.2	50.7	68.8	54.7	43.8	298	18.8	17.0	82.9	95.2%
SID	75.4	58.0	62.2	50.1	68.5	54.6	39.3	272	14.7	16.4	84.4	91.5%
DART	74.8	57.8	61.6	49.9	69.6	53.9	41.6	275	17.0	17.2	77.8	92.6%
Nüwa	77.1	60.3	63.1	51.9	68.2	54.2	43.5	202	19.0	**17.8**	83.6	93.6%
EADP	76.5	59.6	62.7	50.1	68.8	54.8	41.4	290	17.2	16.7	**87.2**	94.5%
VLM-Pruner	77.1	60.3	61.4	49.3	69.3	54.2	42.6	294	17.8	17.2	84.4	94.8%
HiPrune	74.9	57.3	61.9	50.5	68.4	**55.2**	43.6	296	19.3	16.8	82.8	95.0%
Bifocal (ours)	**77.6**	**60.5**	**64.1**	**52.1**	68.2	54.2	**45.6**	262	**20.4**	16.8	84.7	**96.3%**
Average Tokens = 64												
SID	74.2	57.3	59.5	48.4	68.3	53.6	38.0	**276**	13.8	16.2	84.3	89.8%
DART	65.4	51.1	54.8	42.5	68.8	51.7	29.3	214	11.7	14.2	59.0	77.8%
Nüwa	74.6	**58.6**	62.0	**49.4**	67.8	53.0	39.9	152	15.5	**16.3**	79.8	87.4%
EADP	**75.3**	58.1	60.7	47.9	69.1	**53.9**	37.6	268	14.4	15.6	**85.6**	90.0%
VLM-Pruner	69.9	54.4	57.7	43.0	**69.3**	52.5	32.8	240	11.9	14.3	69.1	82.2%
HiPrune	69.1	53.6	59.5	46.3	68.9	53.4	39.9	258	16.1	16.1	73.0	87.9%
Bifocal (ours)	**75.3**	58.5	**62.7**	46.8	67.6	52.5	**40.9**	254	**17.0**	15.4	81.7	**90.6%**

4 EXPERIMENTS

We evaluate Bifocal at three budgets on LLaVA-1.5-7B (Tables 1 and 2), Qwen2.5-VL-7B (Table 3) and LLaVA-NeXT-7B (**Appendix H**), followed by ablations (§4.3), an efficiency analysis (**Appendix I**) and visualisations (§4.4).

4.1 EXPERIMENTAL SETUP

Models and benchmarks. We report main results on LLaVA-1.5-7B and verify on LLaVA-NeXT-7B and Qwen2.5-VL-7B. The benchmarks are VQA^{v2} (Goyal et al., 2017), GQA (Hudson & Manning, 2019), MMBench (Liu et al., 2024d), OK-VQA (Marino et al., 2019), ScienceQA-IMG (Lu et al., 2022), AI2D (Kembhavi et al., 2016), TextVQA (Singh et al., 2019), OCRBench (Liu et al., 2024e), DocVQA (Mathew et al., 2021), ChartQA (Masry et al., 2022), POPE (Li et al., 2023b), RefCOCO, RefCOCO+ (Yu et al., 2016) and RefCOCOg (Mao et al., 2016).

Baselines. We select representative visual-token pruning methods as baselines, including FastV (Chen et al., 2024), PDrop (Xing et al., 2025), VisionZip (Yang et al., 2025), and PruMerge+ (Shang et al., 2025). We further reproduce recent publicly available representative works and conduct fair comparisons with open-source methods that provide favorable reproducibility, including DART (Wen et al., 2025), Nüwa (Huang et al., 2026), HiPrune (Liu et al., 2026a), EADP (Wang et al., 2026a), SID (Fang et al., 2026), and VLM-Pruner (Wu et al., 2026).

4.2 MAIN RESULTS

Results on LLaVA-1.5-7B. Existing methods are markedly lopsided across tasks. On the general benchmarks they are hard to tell apart: at the 192-token budget in Table 1, the top five are all crammed between 96.69% and 98.19%. Yet they diverge sharply the moment they leave the general

tasks: the OCRBench score of Nüwa drops from 314 to 244 at 192 tokens and to only 152 at 64, and on grounding (Table 2) HiPrune, EADP and SID retain merely 10.56%, 13.97% and 9.53% even at 192 tokens, no longer a mild degradation but a breakdown. Bifocal holds both ends. It attains the best general average at all three budgets, reaching 98.19%, 96.26% and 90.65%, while on grounding it reaches 111.02% and 101.81% at 192 and 128 tokens, and still retains 66.70% at 64, 11.83 points above the strongest baseline, Nüwa.

Table 2: **Visual grounding on LLaVA-1.5-7B.** All entries are ACC@0.5. Budget groups and highlighting follow Table 1.

Method	RefCOCO val	testA	testB	RefCOCO+ val	testA	testB	RefCOCOg val	test	Avg.
Baseline	56.2	64.4	47.5	50.1	59.4	38.9	48.8	48.4	100.0%
Average Tokens = 192									
DART	32.9	36.5	30.7	28.5	33.0	26.8	29.6	29.4	60.2%
Nüwa	59.0	67.3	49.0	50.5	60.8	40.2	51.7	50.1	103.6%
HiPrune	5.3	5.5	7.2	4.5	4.8	5.6	5.2	4.8	10.6%
EADP	7.4	7.7	9.2	6.4	7.0	7.1	6.1	5.9	14.0%
SID	5.2	5.2	6.4	4.1	4.2	5.2	4.3	4.1	9.5%
VLM-Pruner	46.0	54.6	39.7	39.8	48.2	32.5	38.9	39.1	81.9%
Bifocal (ours)	**62.4**	**71.4**	**52.9**	**54.7**	**66.2**	**43.1**	**55.1**	**54.2**	**111.0%**
Average Tokens = 128									
DART	17.0	17.5	17.1	14.8	15.7	15.1	15.4	14.6	31.0%
Nüwa	52.1	61.3	42.8	43.4	53.0	35.4	45.1	43.6	91.1%
HiPrune	3.2	3.0	4.3	2.3	2.4	3.4	2.6	2.4	5.8%
EADP	4.2	5.0	5.8	3.1	4.2	4.6	3.5	3.4	8.3%
SID	3.0	3.0	3.9	2.6	2.8	3.5	2.5	2.4	5.9%
VLM-Pruner	28.3	32.4	25.5	23.9	28.3	21.5	24.7	24.8	50.8%
Bifocal (ours)	**57.3**	**66.4**	**49.1**	**49.6**	**60.3**	**40.4**	**49.8**	**48.7**	**101.8%**
Average Tokens = 64									
DART	1.4	0.9	1.9	0.9	0.8	1.9	1.1	1.1	2.5%
Nüwa	32.7	39.7	24.4	26.2	33.9	20.5	25.6	24.3	54.9%
HiPrune	1.3	1.3	2.1	1.1	1.1	1.6	1.3	0.9	2.6%
EADP	3.2	3.7	4.2	2.6	3.2	3.2	2.6	2.3	6.2%
SID	3.0	2.3	3.8	2.5	2.6	3.1	2.2	2.2	5.4%
VLM-Pruner	1.7	1.3	2.5	1.4	1.2	2.2	1.5	1.3	3.3%
Bifocal (ours)	**38.6**	**49.0**	**30.7**	**31.9**	**41.9**	**24.3**	**31.0**	**30.4**	**66.7%**

Results on Qwen2.5-VL-7B. The same pattern persists on a stronger backbone. As shown in Table 3, Bifocal ranks first in both general and grounding averages at all three token budgets, and it leads on every grounding split (**Appendix G**).

Table 3: **Results on Qwen2.5-VL-7B.** Budgets are fractions of the visual tokens. Gen. and REC are the performance relative to the baseline on the general and grounding benchmarks.

Method	General		Reason.		OCR			Hall.	Grounding			Avg.	
	GQA	MMB	SQA	AI2D	VQA^{T}	OCR-B	DocVQA	POPE	RefCOCO	RefCOCO+	RefCOCOg	Gen.	REC
Baseline	60.9	84.0	88.2	82.3	82.8	842	94.7	86.4	88.7	82.6	84.1	100.0%	100.0%
Retain 50% Tokens													
VisionZip	57.4	78.7	79.6	73.5	74.8	734	89.5	86.5	77.8	74.9	77.5	91.5%	89.6%
DART	59.3	81.5	86.2	78.0	79.5	721	77.5	85.6	76.1	69.8	72.5	93.1%	85.5%
Nüwa	**60.1**	**83.7**	**87.6**	81.6	81.4	813	93.2	86.0	86.7	79.6	81.9	97.6%	97.2%
HiPrune	57.3	79.2	79.3	74.0	74.8	731	89.3	**86.6**	77.5	74.7	77.7	91.7%	89.4%
EADP	59.5	82.6	85.5	80.0	77.3	741	81.0	84.9	82.1	74.5	76.7	93.4%	91.5%
VLM-Pruner	59.9	83.0	87.3	**82.1**	81.0	**820**	91.3	85.8	83.7	77.2	79.2	97.4%	94.0%
Bifocal (ours)	**60.1**	82.6	87.3	82.0	**82.8**	814	**94.0**	86.3	**87.5**	**80.7**	**82.8**	**98.4%**	**98.3%**
Retain 25% Tokens													
VisionZip	55.1	77.0	78.3	70.0	68.6	601	77.5	84.0	61.2	58.5	59.4	84.8%	69.9%
DART	55.7	77.3	82.8	70.1	66.4	528	48.7	80.8	26.6	24.1	25.9	80.3%	29.9%
Nüwa	58.5	**82.4**	86.7	81.0	77.9	680	87.8	85.7	77.0	68.2	72.7	93.4%	85.4%
HiPrune	54.9	77.7	77.7	70.2	68.8	604	78.4	84.5	59.3	56.3	58.7	85.1%	67.9%
EADP	58.6	79.5	83.7	76.2	70.8	641	66.7	83.3	61.0	55.7	56.3	86.8%	67.9%
VLM-Pruner	58.9	80.7	85.7	78.5	76.3	678	76.8	84.9	44.1	40.5	40.1	90.3%	49.1%
Bifocal (ours)	**59.5**	82.0	**87.2**	**81.3**	**80.3**	**711**	**90.3**	**86.2**	**81.2**	**72.0**	**77.0**	**94.9%**	**90.2%**
Retain 15% Tokens													
VisionZip	53.3	75.1	76.7	66.1	60.8	463	60.3	82.1	38.9	36.3	36.6	77.0%	43.8%
DART	50.8	71.4	80.0	66.5	49.9	383	30.1	70.1	6.3	5.5	5.8	68.0%	6.9%
Nüwa	56.2	**80.5**	85.2	77.6	72.2	540	75.5	84.0	53.7	44.3	49.4	86.7%	58.0%
HiPrune	52.5	75.0	76.6	66.9	62.8	457	64.0	82.3	35.7	33.0	34.4	77.7%	40.3%
EADP	**57.7**	78.0	82.3	72.7	65.5	564	56.8	80.9	36.6	33.7	34.1	81.4%	40.9%
VLM-Pruner	55.2	76.0	83.5	71.1	66.5	487	53.2	78.5	8.5	8.0	7.8	77.8%	9.5%
Bifocal (ours)	**57.7**	79.5	**85.8**	**77.9**	**76.3**	**578**	**81.1**	**85.3**	**62.7**	**52.4**	**57.5**	**89.0%**	**67.9%**

4.3 ABLATION STUDY

Bifocal selects and allocates three kinds of tokens from different angles, so we verify that each of them is indeed useful. Taking LLaVA-1.5-7B with an average of 128 tokens per LLM layer as an

example, we vary how the 224 Stage-1 tokens are split among C, I and Q (Table 4). C clearly pays off on grounding: adding it lifts REC from 83.6% to 89.0% with I alone, and from 97.0% to 101.8% with Q alone. Like I, C is query-free, so the general average changes little. As expected, removing I lowers POPE from 84.7 to 83.2, and removing Q drops REC from 102.7% to 89.0%.

Table 4: **Evidence ablation**. C, I and Q are the numbers of coarse-context, image-intrinsic and query-focused tokens, all rows apply the same Stage-2 pruning.

Token allocation			General		Reasoning		OCR			Hall.	RefCOCO+		Avg.	
C	I	Q	GQA	MMB	SQA	AI2D	OCR-B	DocVQA	ChartQA	POPE	testA	testB	Gen.	REC
Baseline			61.9	64.1	69.4	55.2	314	21.5	18.2	85.9	59.4	38.9	100.0%	100.0%
–	224	–	**60.6**	63.8	68.2	**54.8**	200	20.9	17.3	84.3	49.7	32.5	93.6%	83.6%
–	112	112	59.8	63.9	**68.3**	53.1	213	**21.2**	**17.4**	84.2	55.0	38.3	93.9%	95.5%
–	–	224	59.6	63.7	68.0	51.9	242	20.8	16.8	83.7	56.5	38.5	93.9%	97.0%
64	160	–	60.3	63.7	68.2	54.6	209	19.4	17.3	83.7	53.1	34.4	92.9%	89.0%
64	**80**	**80**	60.5	**64.1**	68.2	54.2	**262**	20.4	16.8	**84.7**	**60.3**	40.4	**95.4%**	**102.7%**
64	–	160	60.0	63.1	67.7	53.5	226	19.3	16.3	83.2	58.6	**40.8**	92.2%	101.8%

4.4 VISUALISATION

We further analyse the effectiveness of Bifocal qualitatively through visualisations. Figure 5 shows which tokens Bifocal keeps and how this helps grounding. In the top row, the image-intrinsic evidence spreads over the whole image and does not change with the query, while the query-focused evidence moves with the query to the region it names. The two kinds of fine evidence thus split the work as intended: one covers the image, the other follows the question. In the bottom row, Bifocal localises the target as the unpruned model does, whereas the other methods misplace the box. More cases are shown in **Appendix K**.

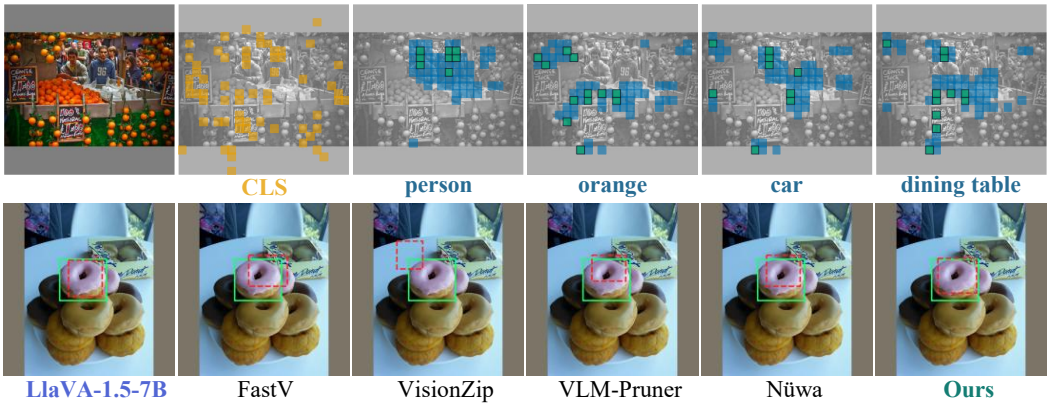


Figure 5: *Top:* the fine evidence Bifocal selects for one image under different queries, ● image-intrinsic evidence and ■ query-focused evidence. *Bottom:* grounding results of LLaVA-1.5-7B and pruning methods under the same budget, with ▭ the ground truth and ▭ the prediction.

5 CONCLUSION

This paper approaches visual token pruning from a novel perspective and proposes a mixed-granularity pruning framework before the LLM, composed of query-focused evidence, image-intrinsic evidence and a coarse context, which preserves the abilities of the model more comprehensively. Building on this framework, we optimise the selection of each kind of token. We combine the word and image saliency maps to better preserve query-focused evidence through word weighting and word conjunction, then select the optimal token subset by representational coverage, and finally add a coarse context pooled from the remaining patches. Together with Stage 2, which further removes redundant tokens inside the LLM, Bifocal achieves the best overall performance across multiple models and budgets.

AI USE STATEMENT

In this work, we used generative AI tools for (1) surveying progress in the field and translating the related literature, and (2) polishing the language and checking the grammar of the manuscript. We have not used generative AI tools to generate data or to produce any reported result. We have reviewed all AI-assisted work, and we take responsibility for the final content of this work.

REFERENCES

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

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*, 2025.

Peter J Burt and Edward H Adelson. The laplacian pyramid as a compact image code. In *Readings in computer vision*, pp. 671–679. Elsevier, 1987.

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*, pp. 19–35. Springer, 2024.

Tzu-Chun Chien, Chieh-Kai Lin, Shiang-Feng Tsai, Ruei-Chi Lai, Hung-Jen Chen, and Min Sun. Grounding-aware token pruning: Recovering from drastic performance drops in visual grounding caused by pruning. *arXiv preprint arXiv:2506.21873*, 2025.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International conference on learning representations*, volume 2024, pp. 2632–2652, 2024.

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*, pp. 10492–10501, 2026.

Mark Endo, Xiaohan Wang, and Serena Yeung-Levy. Feather the throttle: Revisiting visual token pruning for vision-language model acceleration. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22826–22835. IEEE, 2025.

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 *The Fourteenth International Conference on Learning Representations*, 2026.

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

Jiayi Han, Liang Du, Yiwen Wu, Guanming Liang, Xiangguo Zhou, Weibo Zheng, Donghong Han, and Zixun Sun. Adav: Adaptive text-visual redirection for vision-language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 4985–4997, 2025.

Kai Huang, Hao Zou, Ye Xi, BoChen Wang, Zhen Xie, and Liang Yu. Ivtp: Instruction-guided visual token pruning for large vision-language models. In *European conference on computer vision*, pp. 214–230. Springer, 2024.

Runhui Huang, Xinpeng Ding, Chunwei Wang, Jianhua Han, Yulong Liu, Hengshuang Zhao, Hang Xu, Lu Hou, Wei Zhang, and Xiaodan Liang. Hires-llava: Restoring fragmentation input in high-resolution large vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29814–29824. IEEE, 2025.

Yihong Huang, Fei Ma, Yihua Shao, Jingcai Guo, Zitong YU, Laizhong Cui, and Qi Tian. Nüwa: Mending the spatial integrity torn by vlm token pruning. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 100825–100856, 2026.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6693–6702. IEEE, 2019.

Yizhang Jin, Jian Li, Tianjun Gu, Yexin Liu, Bo Zhao, Jinxiang Lai, Zhenye Gan, Yabiao Wang, Chengjie Wang, Xin Tan, et al. Efficient multimodal large language models: A survey. *Visual Intelligence*, 3(1):27, 2025.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 787–798, 2014.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *European conference on computer vision*, pp. 235–251. Springer, 2016.

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

Xu Li, Yi Zheng, Mengyang Zhao, Yuxuan Liang, Zhe Liu, Rui Zhu, Xiaolei Chen, Wei Zhou, Baoquan Zhao, and Juncen Guo. Crisp: Pre-llm yet text-driven visual token pruning for efficient lvlm inference. *arXiv preprint arXiv:2607.16326*, 2026.

Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-gemini: Mining the potential of multi-modality vision language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, 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, 2023b.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 936–944. Ieee, 2017.

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

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26286–26296. IEEE, 2024a.

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

Jizhihui Liu, Guangdao Zhu, Feiyi Du, Niu Lian, Jun Li, Bin Chen, Weili Guan, and Yaowei Wang. Hiprune: Hierarchical attention for efficient token pruning in vision-language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 3274–3291, 2026a.

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

Yingen Liu, Fan Wu, Ruihui Li, Zhuo Tang, and Kenli Li. Par: Prompt-aware token reduction method for efficient large multimodal models. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 12897–12901. IEEE, 2026b.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *European conference on computer vision*, pp. 216–233. Springer, 2024d.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 67(12):220102, 2024e.

Ziniu Liu, Shuheng Zhou, Mingqing Liu, Hao Deng, and Huijia Zhu. Crisprune: Combining contextual relevance and intrinsic saliency for efficient visual token pruning in mllms. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13546–13564, 2026c.

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

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 11–20, 2016.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *2019 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 3190–3199. IEEE, 2019.

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

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 2199–2208. IEEE, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

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

Kele Shao, Keda Tao, Kejia Zhang, Sicheng Feng, Mu Cai, Yuzhang Shang, Haoxuan You, Can Qin, Yang Sui, and Huan Wang. A survey of token compression for efficient multimodal large language models. *Transactions on Machine Learning Research*, 2026.

Aditya Kumar Singh, Hitesh Kandala, Pratik Prabhanjan Brahma, Zicheng Liu, and Emad Barsoum. Duet-vlm: Dual stage unified efficient token reduction for vlm training and inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17651–17660, June 2026.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8309–8318. IEEE, 2019.

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

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. *arXiv preprint arXiv:2402.17762*, 2024.

Yizheng Sun, Yanze Xin, Hao Li, Jingyuan Sun, Chenghua Lin, and Riza Batista-Navarro. LVPruning: An effective yet simple language-guided vision token pruning approach for multi-modal large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 4299–4308, 2025.

Xuehui Wang, Xuankun Yang, and Wei Shen. Combating textual noise and redundancy: Entropy-aware dense visual token pruning. In *European Conference on Computer Vision*, pp. 559–580. Springer, 2026a.

Yahong Wang, Juncheng Wu, Zhangkai Ni, Chengmei Yang, Yihang Liu, Longzhen Yang, Yuyin Zhou, Ying Wen, and Lianghua He. Entropyprune: Matrix entropy guided visual token pruning for multimodal large language models. *arXiv preprint arXiv:2602.17196*, 2026b.

Zichen Wen, Yifeng Gao, Shaobo Wang, Junyuan Zhang, Qintong Zhang, Weijia Li, Conghui He, and Linfeng Zhang. Stop looking for "important tokens" in multimodal language models: Duplication matters more. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9972–9991, 2025.

Zhenkai Wu, Xiaowen Ma, Zhenliang Ni, Dengming Zhang, Han Shu, Xin Jiang, and Xinghao Chen. Vlm-pruner: Buffering for spatial sparsity in an efficient vlm centrifugal token pruning paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 31952–31961, 2026.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, 2024.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Conical visual concentration for efficient large vision-language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14593–14603. IEEE, 2025.

Cheng-Yu Yang, Shao-Yuan Lo, and Yu-Lun Liu. Reroute, don't remove: Recoverable visual token routing for vision-language models. *arXiv preprint arXiv:2606.12412*, 2026.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802. IEEE, 2025.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European conference on computer vision*, pp. 69–85. Springer, 2016.

Yunqian Yu, Biao Chen, Yunya Zhang, Tonglan Xie, Mengmeng Jing, and Lin Zuo. Instruction-guided cross-modal clustering for training-free visual token pruning in vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 12213–12221, 2026.

Hanzhang Yuan, Mengxuan Hu, Wenhao Zhang, Tianlong Wang, Zhongliang Zhou, Jiasen Lu, and Sheng Li. Reducing token redundancy in lvlms: A systematic review of token pruning methods. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7231–7251, 2026.

Pu Zhang, Yuwei Li, Xingyuan Xian, and Guoming Tang. Zspaprune: Zero-shot prompt-aware token pruning for vision-language models. *arXiv preprint arXiv:2510.17197*, 2025a.

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20857–20867. IEEE, 2025b.

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 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 74840–74857. PMLR, 2025c.

A Related Work

Text-Guided Visual Token Pruning. Within an LLM, textual hidden states progressively integrate linguistic context and visual information through multiple layers of causal self-attention. A simple and computationally efficient strategy is therefore to use the final text token as a proxy for the full instruction and estimate visual token importance from its attention to visual tokens (Chen et al., 2024; Xing et al., 2025; Endo et al., 2025). Building on this idea, subsequent methods select or aggregate multiple text tokens to extract task-relevant textual cues for visual token pruning (Zhang et al., 2025c; Singh et al., 2026; Liu et al., 2024c). Beyond directly reusing existing textual hidden states, some studies explicitly construct or learn text-conditioned representations for visual token pruning. IVTP (Huang et al., 2024) constructs a pseudo-CLS token that is semantically associated with the current instruction and uses it as a reference for visual token selection. LVPruning (Sun et al., 2025) introduces learnable cross-attention modules to explicitly fuse textual and visual representations and predict visual token retention decisions.

Before visual features enter the LLM, instruction embeddings have not undergone deep contextualization or extensive interaction with the current visual input, although pretrained vision encoders and modality projectors already provide a certain degree of cross-modal semantic alignment. Accordingly, methods such as TRIM (Song et al., 2025) exploit embedding-level text–vision relevance by measuring the similarity between CLIP text embeddings and visual patch embeddings, while subsequent approaches apply pooling or aggregation to derive text–vision relevance scores (Zhang et al., 2025a; Han et al., 2025; Liu et al., 2026b). Rather than treating the entire instruction as a single unit, CRISP (Li et al., 2026) extracts keywords and compares them with projected visual tokens, whereas ICCTP (Yu et al., 2026) uses instruction nouns as visual-semantic anchors for cross-modal clustering and visual token compression.

Image-Guided Visual Token Pruning. Unlike methods conditioned on the current textual instruction, image-guided approaches compress visual tokens solely according to intrinsic visual cues, such as saliency, informativeness, similarity, and redundancy. One line of work estimates visual token saliency from the attention maps produced by the vision encoder. A common strategy is to use the attention from the CLS token to image patches to identify visually important tokens (Shang et al., 2025). HiPrune (Liu et al., 2026a) further exploits distinct attention patterns across different vision-encoder layers to capture object-level and global visual information, while EntropyPrune (Wang et al., 2026b) measures visual token informativeness from an information-theoretic perspective using representation entropy. Another line of work focuses on similarity and redundancy among visual tokens, reducing repeated information by retaining representative or diverse visual features (Yang et al., 2025; Alvar et al., 2025). VisPruner (Zhang et al., 2025b) combines these two types of visual cues by first selecting salient tokens according to visual attention and then removing redundant tokens based on feature similarity.

Joint Text–Image-Guided Visual Token Pruning. Some approaches jointly exploit textual and visual cues for visual token compression. Multi-stage pruning strategies naturally fall into this category because they use complementary guidance signals at different compression stages. MustDrop (Liu et al., 2024c) first removes redundant tokens according to visual similarity during visual encoding and subsequently leverages textual semantics to further filter visual information. DUET-VLM (Singh et al., 2026) first performs redundancy-aware merging based on correlations among visual tokens and then applies progressive pruning within the LLM using text-to-vision attention.

Another line of work combines textual and visual signals within a unified selection process to estimate visual token importance. AdaV (Han et al., 2025) and CrisPrune (Liu et al., 2026c) both use ViT CLS–patch self-attention to characterize visual saliency and the cosine similarity between visual tokens and text embeddings to estimate textual relevance. AdaV aggregates textual relevance using the maximum similarity, whereas CrisPrune further combines the two signals into a token-quality score. CoIn (Du et al., 2026) measures intrinsic visual saliency using the ℓ_2 norm of visual token embeddings and evaluates textual relevance through the cosine similarity between visual tokens and the mean-pooled text embedding. The two normalized scores are then combined to obtain an overall informativeness score.

B THE CONTRIBUTION OF EACH EVIDENCE ROUTE

The main text (§2.1) concludes that the two evidence routes fail in different ways and cannot substitute for each other. In this appendix we discuss the contribution of each route in turn.

B.1 WHICH ROUTE MATCHES THE INTUITION OF THE DENSE MODEL?

We first look at which evidence a dense model relies on when it solves a task normally. For each sample, let A_j be the attention that the dense model pays to visual token j while generating its answer, and s_j^r the score that route r assigns to it. Since pruning only depends on the ranking of the scores, we measure their agreement by the Spearman correlation over the 576 visual tokens:

$$\rho^r = \mathrm{corr}\big(\mathrm{rank}(A),\ \mathrm{rank}(s^r)\big), \qquad r \in \{\text{text},\ \text{image}\}. \tag{12}$$

Figure 6 shows a clear trend. The image route is positively correlated with the attention of the dense model on all 18 tasks, and is higher than the text route on 92% to 100% of the samples in every task. The strength of this agreement, however, varies with the task: it is around 0.38 on grounding and reaches 0.74 on OCR and document tasks. The text route stays much weaker, slightly positive only on grounding and negative on the other tasks. The image thus lays the foundation of what the dense model reads, and different tasks attend to it to different degrees. Although the text route agrees less with the dense model, it is still necessary: as the next subsection shows, on grounding it decides whether the right tokens are kept at all.

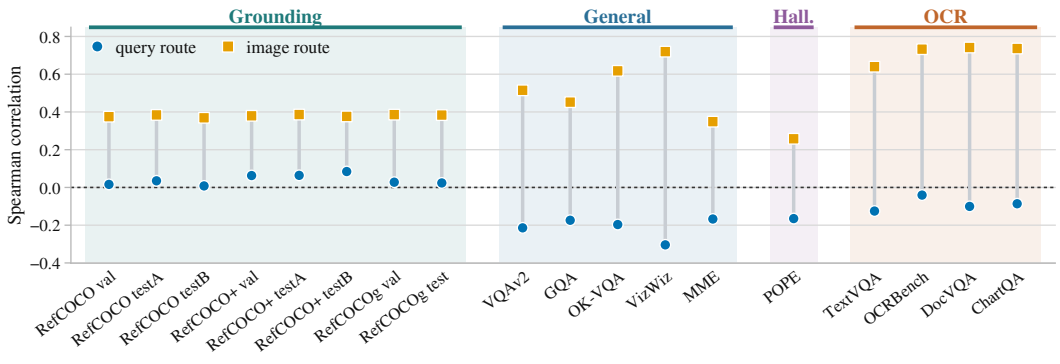


Figure 6: **Alignment of the image and text routes with the baseline's attention.** ● text route. ■ image route. Error bars are 95% confidence intervals.

B.2 THE TEXT ROUTE: A NECESSARY CONDITION FOR GROUNDING

For the text route, we use a deletion test to ask the most direct question: can we do without it? Starting from the unpruned sequence of 576 visual tokens, we remove equal numbers of tokens selected by the text route, the image route, or at random, while progressively increasing the number removed. Figure 7 shows a consistent trend across eight REC splits: at low removal rates, random deletion has only a limited effect, whereas removing text-route tokens causes ACC@0.5 to drop sharply, usually faster than removing image-route tokens. The text route may not carry the majority of the evidence, but it provides query-specific evidence that grounding cannot do without—if the model does not know what the question refers to, how can it draw the box accurately? As the removal rate approaches one half, the curves of the two evidence routes gradually converge. At this point, large-scale information loss begins to dominate, and which route the removed tokens come from is no longer the main issue. This further supports our conclusion in the main text: the two evidence routes cannot substitute for each other and should each be allocated their own share of the fixed budget.

B.3 THE IMAGE ROUTE: A WIDER VIEW

However well the text route is optimised, what it keeps is still word-level evidence: before the LLM there is no real understanding of the question, so the route can only keep the patches that look like

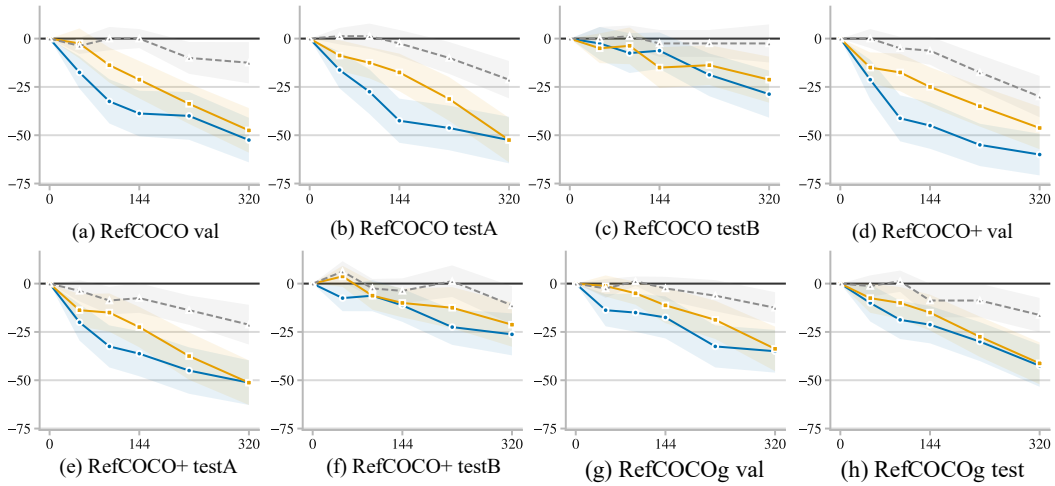


Figure 7: **Deletion test on the eight REC splits**. The horizontal axis is the number of tokens removed from the unpruned 576. The vertical axis is the change in ACC@0.5 relative to keeping all 576, in percentage points. Each panel shows three curves: text route, image route, and random.

the words it was given. This fails in two ways: the object the question refers to is not in the image at all, or the answer sits on something the question never names (or the question has no target to begin with). In both cases, selecting tokens purely by word similarity means that, under a reduced budget, the model is shown the wrong evidence, and hallucination follows. This is exactly what the image route is for: it does not read the question, and its wider view brings a more diverse sample of the scene into the budget.

Figure 8 shows three cases. In the first, the same image is asked two different questions: the tokens selected by the text route change with the question, while the image route does not read the query and its two rows are identical patch for patch. When asked about a bottle that is not in the image, only the text route answers "yes", while the image route and the half-and-half mix both answer correctly. When asked about the sink that is present, all four inputs answer correctly—so the text route does not say "yes" to everything, it says "yes" on fragments that merely look like the queried object. The remaining two cases show the other failure: what the question names is not where the answer lies, the text route again answers wrong, and the image route is needed to bring that wider view back.

C Plugging Existing Pruning Methods into the Framework

To verify whether the mixed-granularity framework is effective in itself, we plug existing pruning methods into it (Table 5). Each method replaces the selection rule of only one kind of fine evidence, while the budgets of all three kinds of tokens stay fixed, and is compared with the same method used on its own under the same total budget. For image-intrinsic evidence, HiPrune and VisionZip replace our image-evidence selection on LLaVA-1.5-7B. Grounding on LLaVA-1.5-7B hardly separates different query-focused methods, so for query-focused evidence EADP replaces our query-evidence selection on Qwen2.5-VL-7B. With the mixed-granularity framework, every method improves substantially, most clearly on grounding: the per-dataset gains range from 9.7 to 20.6 points, and Avg. rises by 6.8 to 9.7 points.

D Per-Cell Noise in the Cosine Map and What Smoothing Removes

The starting point is simple. Pruning keeps the top-m cells by score, and 576 cells compete for that small handful of seats at the upper tail. If the score of each cell has picked up the perturbation that



Figure 8: **Answers under single-route inputs.** Each row is one question, with the ground truth shown by the icon on the right. The four columns are four visual token inputs: image route (CLS top-144), text route (scoring-chain top-144), mix (half from each route), and baseline (all 576 tokens). Within each panel, kept tokens retain their colour and discarded ones are dimmed.

comes with computing a cosine, the quality of the top-k suffers. This section is therefore about three things: that the noise exists, that a simple averaging operation can filter it out, and that this operation is safe, since the low-pass filter does not remove the information that matters.

Which cells of a cosine map are content and which are jitter can be told apart by a property that requires no labels: the two spread over different ranges. Content is contiguous—an object always occupies several cells, so the values of neighbouring cells are correlated—whereas estimation jitter is independent from cell to cell and is already unrelated one cell away. The vague question of whether there is noise, and how much, thus becomes a measurable one: how much of the variance in the map lives only at the scale of a single cell.

Table 5: **Existing methods with and without our framework.** HiPrune and VisionZip select the image-intrinsic evidence within the framework on LLaVA-1.5-7B, and EADP selects the query-focused evidence on Qwen2.5-VL-7B. Each grounding column averages ACC@0.5 over the val and test splits of that dataset, and Avg. averages all metrics relative to the unpruned model.

Method	General		Reasoning		OCR		Hall.	Grounding			Avg.
	GQA	MMB	SQA-I	AI2D	VQA^T	OCR-B	POPE	RefCOCO	RefCOCO+	RefCOCOg	
LLaVA-1.5-7B, image-intrinsic evidence											
HiPrune	58.2	64.0	68.3	53.0	44.2	290	81.2	16.3	13.9	15.2	75.9%
+ framework	59.3	63.3	68.2	52.9	43.8	288	80.5	**36.9** +20.6	**30.7** +16.8	**28.6** +13.4	**85.6%** +9.7
VisionZip	58.4	63.4	68.2	52.3	44.6	301	81.1	26.4	22.6	22.8	81.3%
+ framework	59.2	63.6	68.1	52.7	44.5	297	80.4	**41.5** +15.1	**34.8** +12.2	**32.5** +9.7	**88.5%** +7.2
Qwen2.5-VL-7B, query-focused evidence											
EADP	58.6	79.5	83.7	76.2	70.8	641	83.3	61.0	55.7	56.3	84.0%
+ framework	58.8	81.7	85.8	80.2	78.2	703	84.7	**73.4** +12.4	**66.6** +10.9	**67.4** +11.1	**90.8%** +6.8

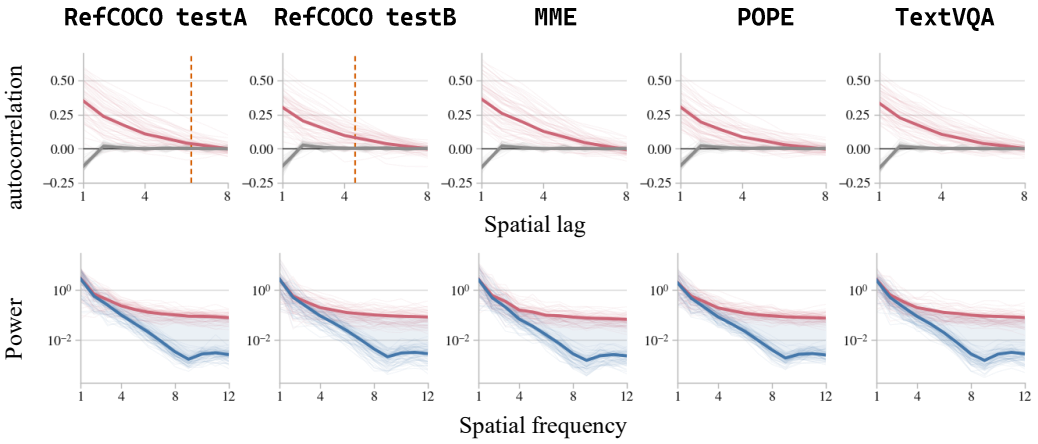


Figure 9: Top row: autocorrelation against spatial lag, in cells. Bottom row: radially averaged power spectrum against spatial frequency, on a logarithmic vertical axis. Curves are ■ the raw map s_w, ■ the map after the 3×3 mean filter $\tilde{s}_w$, and ■ the removed component $s_w - \tilde{s}_w$. Thin lines are individual samples and thick lines their mean. The dashed vertical line marks the median half-width of the ground-truth boxes.

We therefore look for evidence in two directions. One is spatial: how much relation a token still has with the cells around it. The other is spectral: among the light-and-dark fluctuations on the map, how many are large patches that vary slowly, and how many vary from one cell to the next? Objects are contiguous, so the fluctuations they produce can only be the former, that is, concentrated at low frequency.

Figure 9 shows both. The first is the spatial one, which plots autocorrelation against distance. The horizontal axis is the distance d between two cells (in cells), and the vertical axis is the correlation coefficient of the values of two cells that far apart, averaged over samples,

$$\rho(d) = \frac{\mathrm{Cov}\left(s(i),\, s(j)\right)}{\mathrm{Var}(s)}, \qquad \|i - j\| = d.$$

The red curve is $\rho(d)$ of the raw per-word map. The correlation of a cell with itself is always 1. It decreases with distance, and once the gap between two cells exceeds the scale of the content itself it falls to zero. What we actually read is this: the red curve has already dropped to about a half at $d = 1$, and only then decays gently, reaching zero at around six cells, which matches the median half-width of the ground-truth boxes (dashed line). The initial drop and the subsequent decay say two different things. Writing the per-word map as content plus noise, $s = c + \varepsilon$ with ε independent from cell to cell, ε contributes nothing to the covariance at $d \geq 1$, so

$$\rho(d) = (1 - \alpha)\, \rho_c(d) \quad (d \geq 1), \qquad \alpha = \frac{\mathrm{Var}(\varepsilon)}{\mathrm{Var}(s)}.$$

That is, the drop from 1 to $\rho(1)$ is exactly the share of variance α held by the cell-independent component, while the distance over which the gentle part extends is the correlation length of the content. The grey curve is the autocorrelation of the removed component: one cell out it already sits at zero, or even slightly below—it is unrelated even to its own neighbours and cannot carry content.

The second figure is another way of looking at the same thing. Autocorrelation splits the variance by distance, the power spectrum splits the same variance by scale, and the two are strict Fourier pairs (Wiener–Khinchin theorem). Concretely, a two-dimensional discrete Fourier transform of the 24×24 per-word map gives coefficients $F(u)$, and the energy at a given frequency is $|F(u)|^2$, which we then average radially over the modulus of the frequency:

$$P(f) = \left\langle\, |F(u)|^2 \,\right\rangle_{\|u\|=f}.$$

Summing the energy over all frequencies gives exactly the total variance of the map (Parseval's identity), so asking which frequencies the energy falls on is asking how coarse the fluctuations that make up this variance are: the horizontal axis is spatial frequency, where low frequency corresponds to large, slow variation and high frequency to jitter that changes cell by cell. The vertical axis is logarithmic because the magnitudes are too far apart— the energy of the content at low frequency is one to two orders above the noise plateau, and on a linear axis the plateau would be squashed against zero, hiding the fact that it is flat.

The red curve is markedly high at low frequency, which says the map does carry contiguous content, while at high frequency it flattens into a horizontal plateau. Flatness means equal energy at every frequency, which is the signature of cell-independent noise, and the height of the plateau is the power of that noise. The blue curve is the spectrum after smoothing: at low frequency the two curves almost coincide, while at high frequency it is an order of magnitude lower—what has been removed is exactly that plateau, and the low frequency where the content lives is barely touched.

The other two curves in the figure confirm this. The grey curve is the removed part, and one cell out it is unrelated to its neighbours, which says that what we dropped is not content. The blue curve is the spectrum after smoothing, pushed down only at high frequency and almost coinciding with the red curve at low frequency, which says the content at low frequency is still there as it was. The 3×3 filter is therefore not smearing detail away but removing the jitter that should not win a seat: the part it removes carries no content, and the step is safe.

E Norm Bias in the CLS Attention

The image saliency map S of Section 3.1 is the residual of the smoothed CLS attention after a linear regression on the patch norm (Eq. equation 4). This appendix explains why this correction is needed. We first derive why the CLS attention follows the norm and verify each step of the derivation empirically (Appendix E.1), and then show how removing the norm-explained component changes the patches ranked at the top (Appendix E.2).

E.1 Why the CLS Attention Follows the Norm

LayerNorm rescales every patch feature to the same length before the attention, so it is not obvious why the norm should affect the CLS attention at all. The CLS logit for patch j is the inner product between the CLS query q and the key, and the key is a linear transform of the normalised feature z_j. LayerNorm fixes the length of z_j but not its composition. A large norm comes from a few channels with massive activations (Sun et al., 2024; Darcet et al., 2024), which point along a shared direction v that is stable across images. We call it the *high-norm direction*. We therefore split z_j into its component along v and the rest:

$$z_j = \|z\|\big(\rho_j\, v + \sqrt{1-\rho_j^2}\, u_j\big), \tag{13}$$

where $\rho_j = \cos(z_j, v)$, which we call the *norm alignment* of patch j, and $u_j \perp v$ is the direction of the remainder. The larger the norm, the more these channels weigh after normalisation, and the higher the norm alignment: LayerNorm removes the norm, but the norm survives as alignment.

Since both the key projection and the inner product are linear, substituting this split separates the logit in each attention head into two terms:

$$\text{logit}_j = \frac{q^\top W_k z_j}{\sqrt{d}} = \rho_j \frac{\|z\|\, q^\top W_k v}{\sqrt{d}} + \sqrt{1-\rho_j^2}\, \frac{\|z\|\, q^\top W_k u_j}{\sqrt{d}} = G\,\rho_j + g_j. \tag{14}$$

The first term is where the norm enters. G is shared by all patches, so $G\rho_j$ is a bonus that each patch receives in proportion to its norm alignment, regardless of what it contains, and only g_j reflects the content. The softmax then turns this sum into a product:

$$a_j = \frac{e^{\text{logit}_j}}{\sum_k e^{\text{logit}_k}} = n_j\, c_j, \qquad n_j = e^{G\rho_j}, \quad c_j = \frac{e^{g_j}}{Z}, \tag{15}$$

where $Z = \sum_k e^{G\rho_k + g_k}$ is the softmax normaliser, n_j is a gain set by the norm alignment, and c_j is the content score. This is why the CLS attention follows the norm: a patch of large norm is multiplied by a large gain, whatever it contains. The derivation only needs G to exist, and its sign and size differ from head to head.

To verify this account, we feed all images of GQA, TextVQA, RefCOCO testA and RefCOCO testB to LLaVA-1.5-7B and measure, on the ViT layer used for scoring, three things: how strongly the CLS attention is biased toward high-norm patches, whether the bias arises through the mechanism derived above, and what the fit of Eq. equation 4 does to the map. Table 6 reports the medians over images. Throughout, $\mathcal{H}$ denotes the eight highest-norm patches of an image, 1.4% of the grid.

Bias. π is the share of the CLS attention received by $\mathcal{H}$, and κ is the ratio of its mean attention to that of an ordinary patch. r and $r_{\setminus\mathcal{H}}$ are the Pearson correlations between the attention map $\tilde{a}$ and the norm map ν, over all patches and with $\mathcal{H}$ excluded. The bias is large: $\mathcal{H}$ receives 17%–22% of the attention, 14 to 19 times an ordinary patch. It is also not confined to $\mathcal{H}$, since the correlation drops only slightly once $\mathcal{H}$ is excluded. We therefore remove the whole norm-explained component rather than a few patches.

Mechanism. γ is the mean cosine between the high-norm directions v of different images, and R^2_ρ and $R^2_{\|h\|}$ are the shares of the logit variance explained by a linear fit on the norm alignment ρ_j and on the patch norm $\|h_j\|$. With $\gamma = 0.96$ on every split, v is a property of the model rather than of the image. R^2_ρ is about four times $R^2_{\|h\|}$, so the logit follows the norm alignment rather than the norm itself, as the derivation predicts once LayerNorm has removed the length.

Correction. Taking the top 5% of patches by ν as high-norm, $\eta_{\tilde{a}}$ and η_S are their share of the top 10% of $\tilde{a}$ and of S, where chance is 10%. r_s is the Spearman correlation between $\tilde{a}$ and S. The fit of Eq. equation 4 lowers the high-norm share from 64%–75% to 46%–50%, while $r_s \geq 0.87$ shows that the ranking of the other patches stays largely intact. The share remains above chance because the correction is first-order: sinks that rise far above the fitted line stay on top.

Table 6: **Empirical check of the norm bias** on LLaVA-1.5-7B, median over all images of each split. Symbols are defined in the text.

	Bias				Mechanism			Correction		
Split	π	κ	r	$r_{\setminus\mathcal{H}}$	γ	R^2_ρ	$R^2_{\|h\|}$	$\eta_{\tilde{a}}$	η_S	r_s
GQA	22%	19×	0.53	0.49	0.96	0.67	0.17	75%	46%	0.87
TextVQA	17%	14×	0.30	0.25	0.96	0.59	0.16	64%	46%	0.92
RefCOCO testA	21%	18×	0.53	0.48	0.96	0.67	0.18	68%	46%	0.88
RefCOCO testB	22%	19×	0.54	0.49	0.96	0.63	0.14	75%	50%	0.87

E.2 Effect on Token Selection

The derivation predicts that raw CLS attention hands its top ranks to high-norm patches, which mostly lie on the background, and that removing the norm-explained component returns these ranks to the content. Figure 10 illustrates this step from left to right on three cases. The two panels on

the left are what one sees when raw CLS is borrowed as a content criterion: many of the bright cells fall on sea, grass, sky and other background, and most of the top-80 seats go to them. The middle panel shows why. The 576 points spread along a rising line — attention does follow norm — and that line is the baseline to be subtracted. The largest-norm cells are thrown far out to the right and almost all of them lie outside the box, so their score is bought by norm. The cells inside the box, in contrast, crowd into the low-norm side and float above the line, and it is the part above the baseline that counts as content. Subtracting the baseline deducts the more from a cell the larger its norm, so the cells at the right end fall the furthest and those inside the box rise relative to them. The two panels on the right are the result of that subtraction: sink cells have not disappeared, but the background is markedly weakened and the seats have returned to the target. Nor should they disappear — they were never pushed below the line, and a sink is not meaningless to begin with, it aggregates information from the whole image.

For token pruning a more direct reading is to ask where the high-norm tokens rank and where the tokens inside the ground-truth box rank. Figure 11 measures both on five grounding benchmarks. The top row plots the probability that the cell ranked r is one of the eight largest-norm cells: raw CLS hands its foremost seats to sink cells, about six in ten of the first eight being sink cells. The bottom row plots the probability that the cell ranked r falls inside the ground-truth box. An ideal ranking should be most accurate at $r = 1$ and decay from there, yet raw CLS starts *below* chance and climbs past it only around rank twenty, landing inside the box just 10% of the time over the first eight ranks. The norm control stays at the bottom throughout, which confirms the premise this section rests on: where the norm is high is background.

After debiasing both rows move in the intended direction. The peak in the top row roughly halves and shifts right, but it does not vanish: the median rank of a sink cell moves only from 5–6 to 13–15. That is the result we want rather than a defect — a sink still does its job of aggregating information over the whole image, and this step only weakens the say of the background, sink cells included. The in-box rate of the first eight ranks rises accordingly in the bottom row, which is the direct gain for token pruning: at the same budget, more of the selected tokens land on the target.

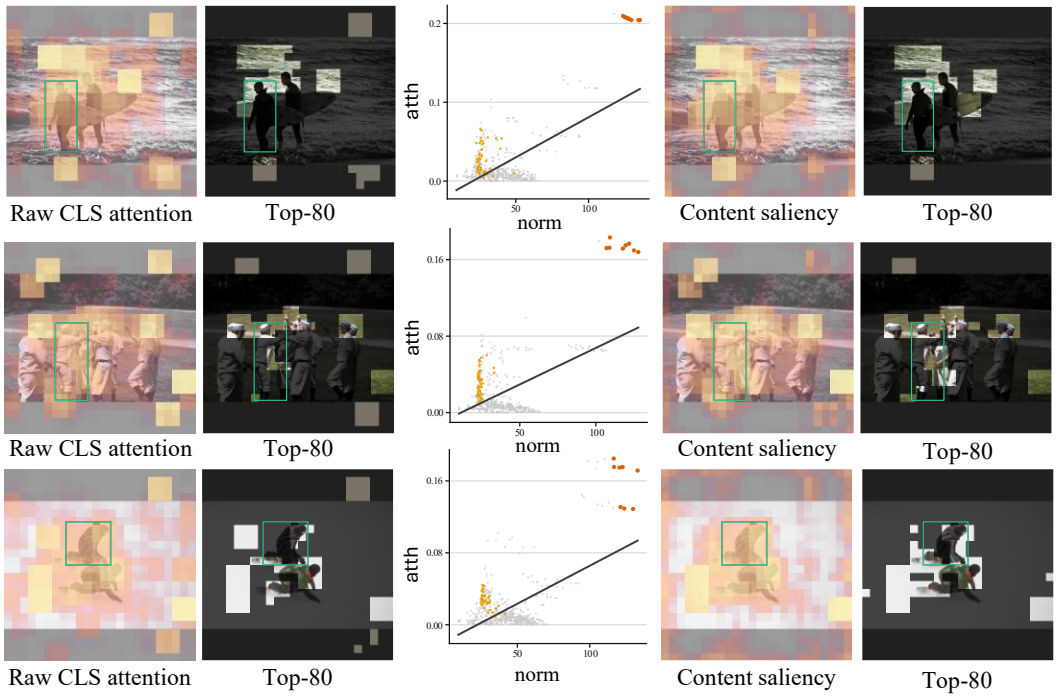


Figure 10: Three cases, one per row, five panels read left to right. (a) raw CLS attention $\tilde{a}$; (b) the top-80 cells kept by $\tilde{a}$; (c) the 576 cells as points, abscissa the patch norm ν, ordinate $\tilde{a}$, with the least-squares baseline in dark — points are ■ inside the box, ■ outside it, and ■ the eight cells of largest norm; (d) the residual S; (e) the top-80 cells kept by S. In (b) and (e) kept cells retain their colour and discarded ones are dimmed. The ■ box is the ground truth.

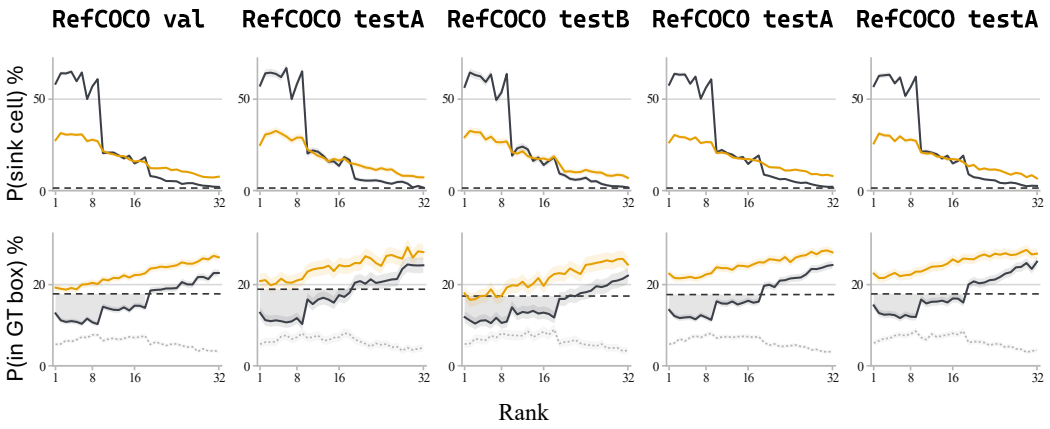


Figure 11: One column per split, $n = 600$ images each. The horizontal axis is the rank r under the score ($r = 1$ is the highest-scoring cell). Top row: the probability that the cell ranked r is one of the eight largest-norm cells. Bottom row: the probability that it falls inside the ground-truth box. Curves are ■ raw CLS $\tilde{a}$ and ■ the debiased S, with ■ the norm ν itself as a control in the bottom row. Bands are 95% confidence intervals of the mean and the dashed line is chance.

F Algorithm for Image-Intrinsic Evidence Selection

Algorithm 1 shows the greedy procedure that maximises the representational coverage in Eq. equation 8.

Algorithm 1 Image-intrinsic evidence by attention-weighted coverage

Require: patch features $\{x_j\}_{j=1}^{N}$, CLS attention $\{a_j\}$, tokens in place $\mathcal{A}_0$, budget N_i
Ensure: image-intrinsic evidence $\mathcal{Q}_i$ with $|\mathcal{Q}_i| = N_i$
$C_{jk} \leftarrow \cos(x_j, x_k)$ for all j, k
$\mathrm{cov}_j \leftarrow \max\big(0, \max_{i \in \mathcal{A}_0} C_{ij}\big)$ ▷ 0 if $\mathcal{A}_0 = \emptyset$
$\mathcal{A} \leftarrow \mathcal{A}_0$, $\mathcal{Q}_i \leftarrow \emptyset$
for $n = 1, \ldots, N_i$ **do**
  $t^\star \leftarrow \arg\max_{t \notin \mathcal{A}} \sum_j a_j \,\mathrm{relu}\big(C_{tj} - \mathrm{cov}_j\big)$
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{t^\star\}$, $\mathcal{Q}_i \leftarrow \mathcal{Q}_i \cup \{t^\star\}$
  $\mathrm{cov}_j \leftarrow \max\big(\mathrm{cov}_j, C_{t^\star j}\big)$ for all j
end for
return $\mathcal{Q}_i$

G Detailed Grounding Results on Qwen2.5-VL-7B

Table 7 lists the per-split grounding results behind the REC average in Table 3. Bifocal is best on every split at all three budgets.

H Results on LLaVA-NeXT-7B

The ANYRES backbone is a harder test: it splits each image into a variable number of crops, so the visual sequence is about four times longer than on LLaVA-1.5 and varies across images, and a fixed token budget no longer corresponds to a fixed fraction of the sequence. As shown in Tables 8 and 9, the gap between general and grounding performance widens further on this backbone. On the general benchmarks all methods fall within a narrow band, and even at the 160-token budget the baselines differ by at most 10.9 points in Avg., whereas on grounding, Avg. ranges from 3.1% to 52.1%. At this budget, VisionZip reaches a general Avg. of 78.2%, close to the 79.8% of Bifocal, yet retains only 3.1% on grounding, about 17× lower than ours. General VQA is therefore largely insensitive to which tokens are retained, while grounding exposes the difference. Moreover, the four baselines that prune once before the LLM (VisionZip, HiPrune, EADP and SID) perform similarly

Table 7: **Visual grounding on Qwen2.5-VL-7B.** All entries are ACC@0.5.

Method	RefCOCO val	test	testA	testB	RefCOCO+ val	testA	testB	RefCOCOg val	test	Avg.
Baseline	89.4	89.4	91.6	84.4	83.0	88.9	75.8	83.9	84.3	100.0%
Retain 50% Tokens										
VisionZip	77.9	78.4	81.9	72.9	75.2	81.1	68.4	77.5	77.4	89.6%
DART	75.7	76.7	80.8	71.2	70.0	76.7	62.8	72.2	72.8	85.5%
Nüwa	86.8	87.6	89.2	83.0	79.8	85.4	73.6	81.9	81.8	97.2%
HiPrune	77.5	78.0	81.2	73.2	75.1	80.6	68.3	77.5	77.8	89.4%
EADP	82.6	83.0	86.4	76.3	75.5	81.9	66.0	76.7	76.6	91.5%
VLM-Pruner	84.6	84.3	87.8	78.1	78.2	84.0	69.4	78.8	79.5	94.0%
Bifocal (ours)	**87.8**	**88.1**	**90.8**	**83.5**	**81.4**	**86.9**	**73.7**	**83.1**	**82.6**	**98.3%**
Retain 25% Tokens										
VisionZip	61.5	61.7	66.6	54.9	58.5	65.0	52.0	59.9	59.0	69.9%
DART	26.5	26.6	29.9	23.3	24.2	27.9	20.3	26.2	25.5	29.9%
Nüwa	77.0	77.6	80.3	73.2	68.2	73.9	62.5	72.8	72.7	85.4%
HiPrune	59.4	60.2	64.4	53.1	56.4	63.2	49.2	59.0	58.4	67.9%
EADP	60.9	62.1	68.6	52.4	56.8	64.9	45.4	56.4	56.1	67.9%
VLM-Pruner	43.9	44.4	51.3	36.7	41.1	48.2	32.3	40.4	39.8	49.1%
Bifocal (ours)	**81.0**	**81.8**	**84.2**	**77.8**	**72.7**	**77.2**	**66.1**	**77.1**	**77.0**	**90.2%**
Retain 15% Tokens										
VisionZip	38.6	38.9	44.2	34.1	36.7	41.6	30.6	36.6	36.5	43.8%
DART	6.0	6.1	6.3	6.6	5.2	5.8	5.4	6.2	5.5	6.9%
Nüwa	53.2	53.9	57.7	50.1	44.8	48.0	40.2	49.3	49.4	58.0%
HiPrune	35.6	35.9	39.9	31.6	32.9	37.3	28.9	34.6	34.1	40.3%
EADP	36.4	36.9	43.1	29.8	33.6	41.5	26.1	34.1	34.0	40.9%
VLM-Pruner	8.2	8.9	8.5	8.2	7.9	8.3	7.8	8.0	7.7	9.5%
Bifocal (ours)	**62.1**	**63.4**	**67.2**	**58.1**	**51.5**	**58.1**	**47.8**	**57.6**	**57.3**	**67.9%**

on grounding at every budget, whether or not they use the query text, suggesting that where the budget is spent in the pipeline matters more than text awareness.

Table 8: **General benchmarks on LLaVA-NeXT-7B.** Within each tier, the best and second-best results are **bold** and underlined. Avg. averages benchmark-wise performance relative to the baseline.

Method	General			Reasoning		OCR				Hall.	Avg.
	GQA	MMB	VQA^{OK}	SQA-I	AI2D	VQA^{T}	OCR-B	DocVQA	ChartQA	POPE	
Baseline	64.2	67.2	44.2	70.2	65.4	64.9	521	68.5	54.9	86.4	100.0%
Average Tokens ≈ 640											
VisionZip	61.2	65.0	**46.8**	68.0	63.5	61.2	474	55.3	42.0	86.2	93.4%
SID	62.5	65.1	40.5	68.3	63.8	58.1	415	50.0	39.5	86.1	89.5%
Nüwa	63.4	64.6	43.7	68.8	64.0	63.0	376	60.9	48.5	85.6	93.5%
EADP	63.2	65.5	39.2	67.0	64.4	56.8	405	47.6	38.3	**87.7**	88.4%
HiPrune	61.8	66.2	41.6	68.6	**65.0**	61.6	486	57.1	44.0	85.6	93.7%
Bifocal (ours)	**63.9**	**67.0**	46.6	**69.9**	64.6	**64.2**	**507**	**65.2**	**53.2**	86.3	**99.1%**
Average Tokens ≈ 320											
VisionZip	58.9	63.6	43.8	67.5	62.4	58.3	397	41.8	35.9	82.0	86.4%
SID	60.6	63.0	30.8	67.7	62.1	53.4	363	40.8	34.2	85.8	82.2%
Nüwa	61.9	63.1	39.8	**68.0**	**63.8**	58.5	312	49.5	34.7	83.3	85.7%
EADP	61.7	63.1	26.1	67.6	63.2	49.7	312	32.6	27.4	**86.1**	77.5%
HiPrune	57.6	64.2	30.2	67.7	62.7	56.9	402	40.5	35.8	79.9	82.8%
Bifocal (ours)	**63.3**	**64.8**	**45.9**	66.4	62.5	**61.0**	**415**	**54.4**	**46.8**	84.8	**92.6%**
Average Tokens ≈ 160											
VisionZip	55.5	59.6	**41.6**	68.1	62.0	52.6	**308**	27.5	29.6	74.7	78.2%
SID	57.7	58.9	18.1	67.4	61.4	46.6	283	28.0	24.7	82.7	71.6%
Nüwa	58.5	60.1	33.3	66.5	**62.2**	51.8	219	31.4	28.1	77.6	75.4%
EADP	59.2	59.8	16.3	67.1	61.1	39.2	253	22.6	22.3	82.3	68.5%
HiPrune	51.6	59.9	17.5	**68.7**	60.7	42.7	259	22.0	26.8	65.9	67.3%
Bifocal (ours)	**60.4**	**60.8**	38.6	59.7	58.8	**54.4**	298	**36.3**	**33.1**	**83.8**	**79.8%**

I EFFICIENCY ANALYSIS

Token pruning acts mainly on the prefill stage, so we report per-sample prefill time, time to first token (TTFT) and peak memory on TextVQA in Table 10, at the 128-token budget. While leading clearly on accuracy, Bifocal also achieves a 1.27× speedup in TTFT, faster than EADP, DART and VLM-Pruner and close to the fastest methods, with no extra memory cost.

Table 9: **Visual grounding on LLaVA-NeXT-7B (ANYRES).** All entries are ACC@0.5.

Method	RefCOCO val	testA	testB	RefCOCO+ val	testA	testB	RefCOCOg val	test	Avg.
Baseline	85.4	89.6	78.0	77.0	86.4	66.4	81.9	80.4	100.0%
Average Tokens ≈ 640									
VisionZip	8.5	8.4	8.2	7.1	7.9	7.0	9.0	9.2	10.1%
SID	35.0	38.9	30.6	30.8	34.8	26.0	33.1	33.5	40.7%
Nüwa	63.9	67.7	57.4	56.3	63.2	46.8	57.7	56.9	72.8%
EADP	35.9	39.0	30.9	31.4	35.1	25.1	33.8	34.2	41.1%
HiPrune	32.0	35.3	29.5	27.2	31.5	24.2	29.0	29.1	36.9%
Bifocal (ours)	**78.1**	**83.6**	**69.6**	**70.2**	**79.2**	**58.2**	**73.9**	**72.5**	**90.7%**
Average Tokens ≈ 320									
VisionZip	4.5	4.2	4.8	3.7	3.5	4.2	4.7	4.8	5.3%
SID	21.9	23.8	18.7	20.1	21.9	16.1	20.3	20.1	25.2%
Nüwa	33.7	35.4	32.8	26.1	29.6	22.5	28.6	28.4	36.8%
EADP	22.6	24.6	19.4	21.1	22.6	17.9	22.3	21.9	26.7%
HiPrune	13.2	11.2	13.0	11.7	10.6	11.5	11.6	12.7	14.8%
Bifocal (ours)	**64.7**	**70.7**	**56.7**	**56.1**	**65.3**	**45.1**	**59.9**	**58.7**	**74.0%**
Average Tokens ≈ 160									
VisionZip	2.6	2.1	3.3	2.0	1.6	2.7	2.7	2.8	3.1%
SID	12.4	12.3	12.2	11.1	11.5	10.6	10.6	11.5	14.3%
Nüwa	17.9	17.3	18.4	12.5	13.1	12.7	15.2	15.5	19.0%
EADP	16.0	17.4	14.5	15.1	15.9	13.0	14.6	14.7	18.8%
HiPrune	5.0	4.2	5.6	5.0	3.8	5.1	4.8	4.7	5.9%
Bifocal (ours)	**45.6**	**50.4**	**40.2**	**38.3**	**45.5**	**31.2**	**42.9**	**42.1**	**52.1%**

Table 10: **Efficiency on LLaVA-1.5-7B.**

Method	Avg. Tokens	FLOPs (T)	Peak Mem. (GiB)	Prefill (ms)	TTFT (ms)	Speedup
LLaVA-1.5-7B	576	8.450	13.5	50.36	89.32	1.00×
VLM-Pruner	128	2.459	13.8	45.54	85.57	1.04×
EADP	128	2.454	13.5	25.41	81.67	1.09×
DART	128	2.483	13.4	35.86	75.00	1.19×
VisionZip	128	2.454	13.5	24.37	65.42	1.37×
HiPrune	128	2.454	13.5	24.40	64.75	1.38×
Bifocal (ours)	128	2.458	13.4	28.61	70.61	1.27×

J DETAILS OF BENCHMARKS

VQAv2 (Goyal et al., 2017). evaluates open-ended visual question answering on balanced image–question pairs, with complementary images designed to reduce language priors and encourage stronger visual grounding.

GQA (Hudson & Manning, 2019). GQA evaluates compositional visual reasoning over natural images using questions derived from scene graphs, emphasizing objects, attributes, spatial relations, and structured reasoning.

MMBench (Liu et al., 2024d). MMBench is a comprehensive multimodal benchmark that evaluates a broad range of perception and reasoning abilities through carefully designed multiple-choice questions.

OK-VQA (Marino et al., 2019). OK-VQA evaluates knowledge-based visual question answering, where answering often requires external or commonsense knowledge beyond information directly observable in the image.

ScienceQA (Lu et al., 2022). ScienceQA evaluates multimodal scientific reasoning over questions from natural science, social science, and language domains, requiring models to combine visual understanding with contextual and domain knowledge.

AI2D (Kembhavi et al., 2016). AI2D evaluates diagram understanding and reasoning over scientific illustrations, requiring models to interpret visual components, textual labels, symbols, and their structural relationships.

TextVQA (Singh et al., 2019). TextVQA evaluates visual question answering over text-rich natural images, requiring models to recognize scene text and reason jointly over textual and surrounding visual information.

OCRBench (Liu et al., 2024e). OCRBench evaluates OCR-related capabilities of multimodal large language models across text recognition, scene-text understanding, document understanding, and information extraction.

DocVQA (Mathew et al., 2021). DocVQA evaluates question answering over document images, emphasizing text recognition, layout understanding, information localization, and reasoning over structured document content.

ChartQA (Masry et al., 2022). ChartQA evaluates question answering over chart images, requiring models to understand chart elements and perform visual, logical, and arithmetic reasoning over the underlying data.

POPE (Li et al., 2023b). POPE is a diagnostic benchmark for object hallucination, using binary questions to test whether multimodal models incorrectly report objects that are absent from the image.

RefCOCO (Kazemzadeh et al., 2014; Yu et al., 2016). RefCOCO evaluates referring expression comprehension by requiring models to localize objects described by natural-language expressions in complex visual scenes.

RefCOCO+ (Kazemzadeh et al., 2014; Yu et al., 2016). RefCOCO+ evaluates referring expression comprehension while restricting explicit location words, placing greater emphasis on object appearance, attributes, and relational cues.

RefCOCOg (Kazemzadeh et al., 2014; Yu et al., 2016). RefCOCOg contains longer and more descriptive referring expressions, requiring models to integrate fine-grained object attributes and contextual relationships for accurate visual grounding.

K VISUALISATIONS

We first show fine-grained tasks, grounding and OCR, on Qwen2.5-VL-7B and LLaVA-1.5-7B, and then visualise the two kinds of fine evidence Bifocal selects under different queries.

K.1 FINE-GRAINED TASKS ON QWEN2.5-VL-7B

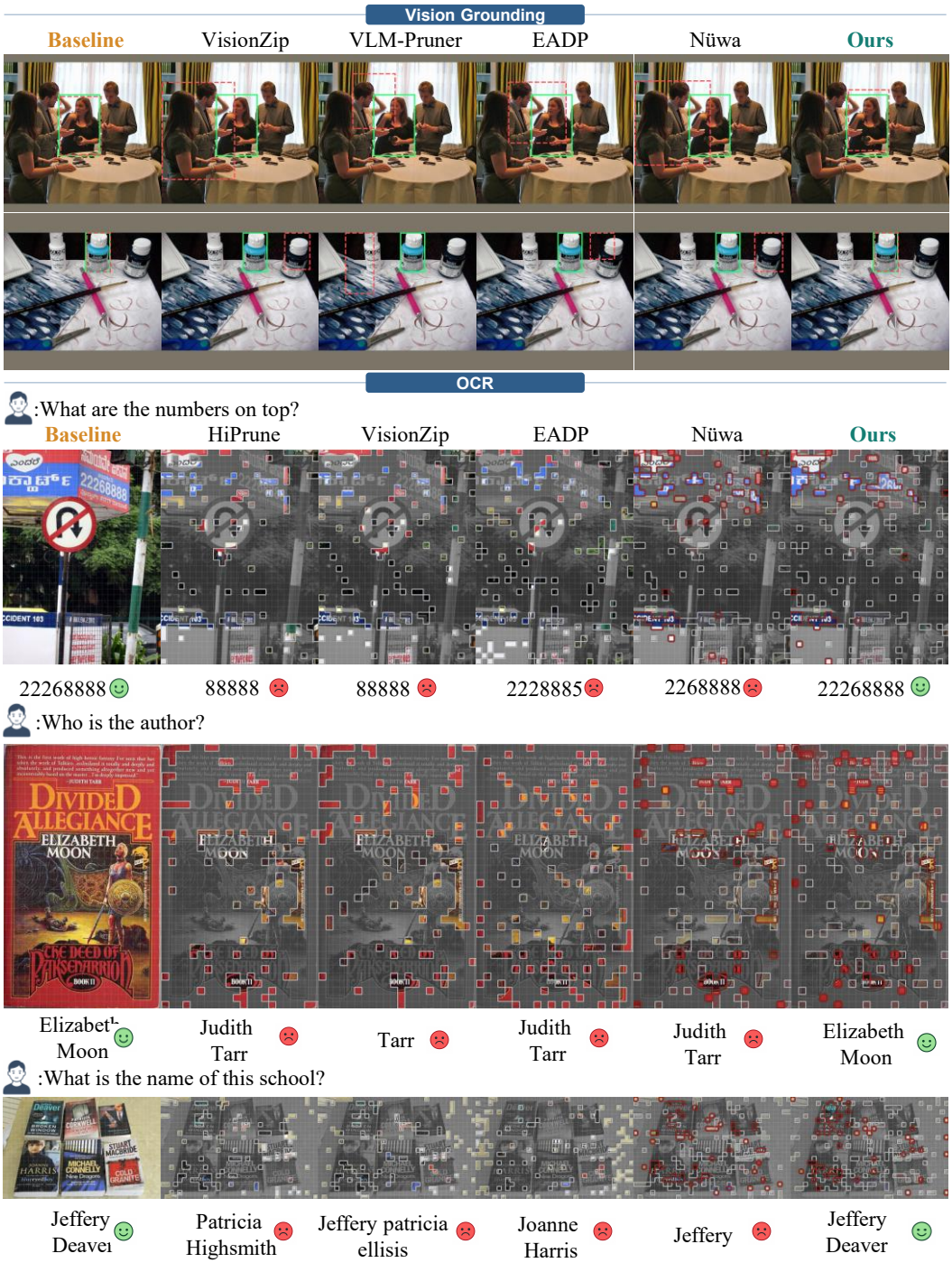


Figure 12: **Grounding and OCR cases on Qwen2.5-VL-7B.** In the grounding cases, ▭ is the ground truth and ▭ the prediction.

K.2 FINE-GRAINED TASKS ON LLAVA-1.5-7B



Figure 13: **Grounding and OCR cases on LLaVA-1.5-7B.** Outputs of the unpruned model and of pruning methods under the same budget. In the grounding cases, ▭ is the ground truth and ▭ the prediction.

K.3 Image-Intrinsic and Query-Focused Evidence

Figure 14 shows the fine evidence Bifocal selects for one image under several POPE queries. The image-intrinsic evidence, selected from the CLS attention, spreads over the whole image and barely changes across queries. The query-focused evidence instead changes with the query and gathers on the object it names. The two kinds of evidence thus play the roles they are designed for: one covers the image, the other follows the question.

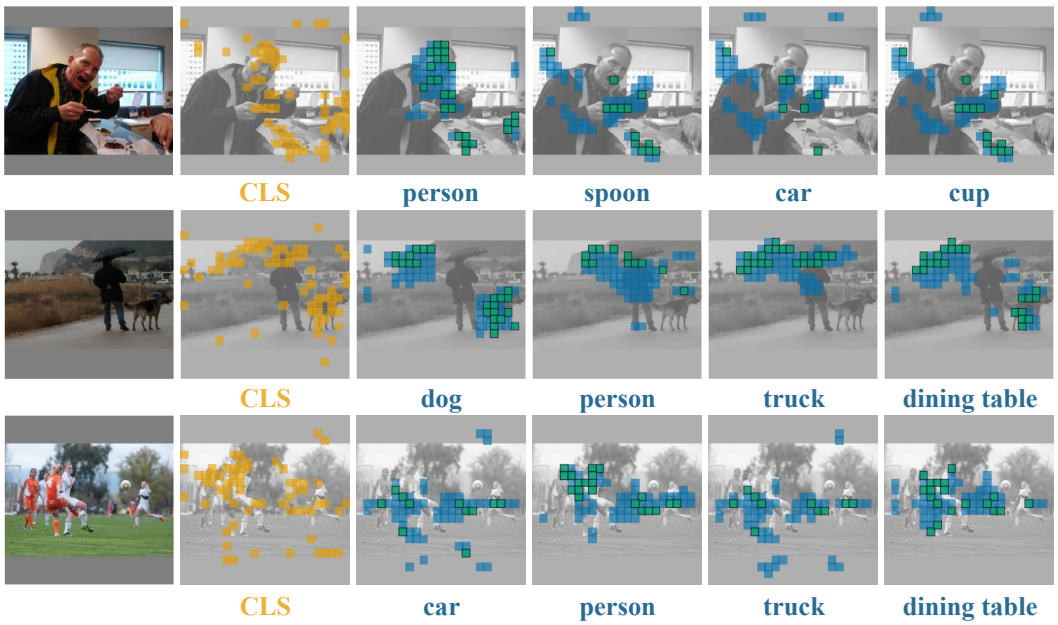


Figure 14: **The two kinds of fine evidence under different queries.** Each row is one POPE image. The first panel after the input shows ■ the image-intrinsic evidence, and the remaining panels show ■ the query-focused evidence for each query.