Glance, then Select: Evidence-Tempered Visual Token Selection for Extremely Compressed Multimodal LLMs

Yuxuan Wang[1*], Guangxiang Zhao[2*], Yuxuan Tian[1], Yahui Han[1], Xinyu Tan[1], Feiyu Wang[1], Zongwei Lv[1], Xiangzheng Zhang[3], Lin Sun[3], Tong Yang[1†]

[1]Peking University
[2]Qiyuan Tech
[3]Qihoo 360

[*]Equal contribution. [†]Corresponding author: Tong Yang.

Abstract

At a 5% visual-token budget a multimodal LLM sees about one token in twenty, and two costs remain: the vision encoder still processes every patch of a large image, and the few surviving tokens must still contain the answer. We present Glance-ETD, a two-stage pipeline that decides where to look *and* which tokens to keep. *A glance encoder encodes the whole image at a quarter of the patches, picks 25% of the coarse cells by max–min diversity and re-encodes only those at native resolution, producing a native-size token grid at a sixth of the encoder FLOPs. **E**vidence-**T**empered **D**iversity (ETD) then selects the final tokens by multiplying the max–min diversity gain of each token by its relative evidence $(Np_i)^{\beta t}$, where p and the text confidence t come from a 1.9M-parameter scorer trained on box annotations with the MLLM frozen. With the MLLM frozen, ETD raises retention at strict 5% from 75.8% (DivPrune) to 84.2%. On Qwen3.5-4B trained with on-policy self-distillation, the same training recipe with DivPrune retains 88.8%; ETD raises it to 91.7%, and the glance encoder then trades 1.4 points for a 2.9× shorter time-to-first-token on 4K images (483 to 65 prefill TFLOPs), giving 90.3±0.8%. The released LT-OPD checkpoint retains 85.4% at the same budget. On held-out document benchmarks ETD raises DocVQA from 52.5 to 74.5.*

1. Introduction and Scope

Multimodal LLMs spend most of their prefill on visual tokens. A 4K image produces tens of thousands of patches, and even a 1K image over a thousand merged tokens, for a question usually answered in a few words. Visual token reduction removes most of these tokens before the language model, and LT-OPD [9] shows that a student distilled on its own rollouts from a full-token copy can work with 5% of them, rather than the full visual sequence.

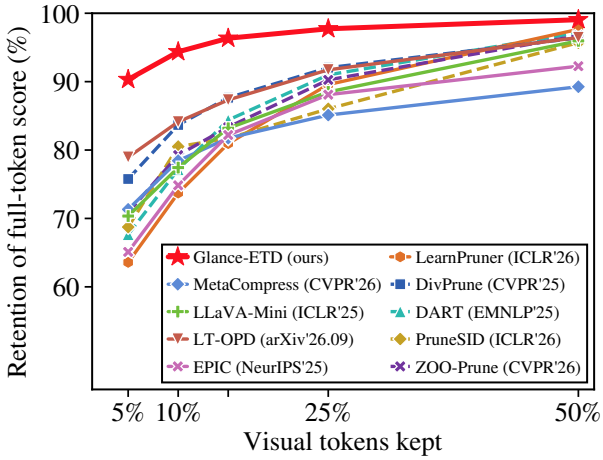


Figure 1. Retention across visual-token budgets on Qwen3.5-4B: the mean over nine benchmarks of each score divided by our non-thinking full-token score, with the same number of tokens per image. Solid lines are trained models; dashed lines are training-free and run on the released model, as in their papers. At 5% of the tokens, Glance-ETD keeps 90.3% (the mean of three training seeds; the other budgets use one seed), 14.5 points above DivPrune.

Two problems remain at this budget (Fig. 2). The first is cost. Pruning after the vision encoder shortens the LLM prefill, but the encoder still attends over every patch: on HR-Bench 4K it takes 2.6 s of a 3.9 s time to first token, so removing 95% of the tokens gives only a 1.20× speed-up (Fig. 2a). The second is selection: with one token in twenty, which tokens survive decides whether the answer is still in the input. Diversity selectors such as DivPrune [1] cover the scene but rarely keep the few tokens of a phone number or a street sign; on TextVQA, DivPrune keeps 10.4% of the

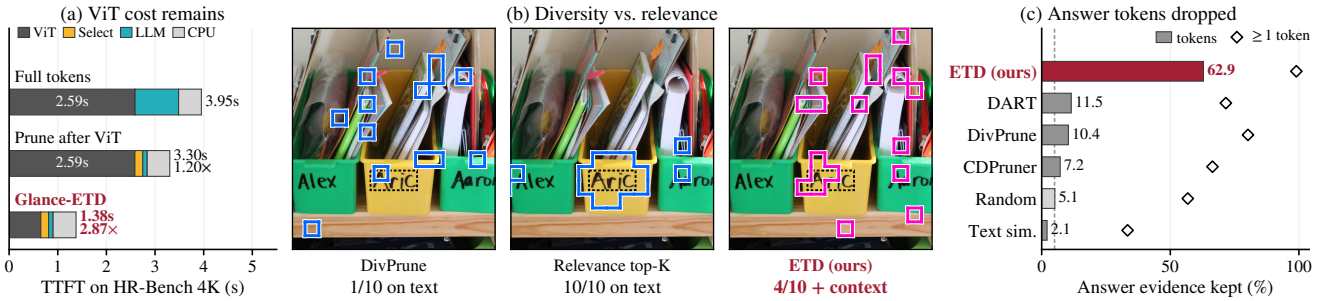


Figure 2. Two problems at a strict 5% visual-token budget on Qwen3.5-4B. (a) Time to first token on HR-Bench 4K for the full-token model, pruning after the ViT, and Glance-ETD (Tab. S14). Each bar is split into vision encoding, selection, LLM prefill and CPU preprocessing. Pruning after the ViT still spends 2.59 s in the encoder; the glance encoder cuts that stage to 0.65 s. (b) One TextVQA image at the same budget: outlines are kept tokens and the dotted box is the answer. DivPrune keeps 1 of 10 text tokens, relevance top-K keeps all 10 but little context, and ETD keeps 4 plus context. (c) On 4,123 TextVQA validation questions with the LT-OPD-recipe model, bars are the share of tokens overlapping the answer OCR box and diamonds the share of questions keeping at least one. The dashed line is the 5% budget.

answer tokens and loses the answer entirely for one question in five (Fig. 2b,c). Relevance alone does not solve this: question–token similarity keeps 2.1% of the answer tokens, and the top tokens of a learned evidence map find the text but lose the coverage that existence and counting questions need (GQA 54.4→51.2, Tab. 3, same budget).

Training the MLLM at the budget does not remove the selection problem. With the self-distillation recipe of LT-OPD and a 10× encoder learning rate, DivPrune retains 88.8% of the full-token score, and the remaining loss is concentrated on reading: TextVQA falls from 80.5 to 70.4 and OCRBench from 848 to 489 (Tab. 1). With the same recipe and the same K, changing only the selector raises them to 76.0 and 579, so on these questions the selector, not the training, sets the limit on these reading benchmarks.

We address both with **Glance-ETD**, a two-stage pipeline that first decides where to look and then which tokens to keep. The *glance encoder* sees the image at a quarter of the patches, picks a quarter of the coarse cells by diversity, and re-encodes only these at native resolution. Its output keeps the native token grid, so any selector applies afterwards. *Evidence-Tempered Diversity* (ETD) then keeps the DivPrune greedy loop and multiplies each candidate's diversity gain d_i by its evidence raised to a temperature,

$$i^\star = \arg\max_i \; d_i \, (Np_i)^{\beta t},$$

where p is the evidence map of a small query-conditioned scorer and $t \in [0, 1]$ is its confidence that the evidence is local text. A flat evidence map or a non-text question gives DivPrune; a confident reading question moves the selection toward the evidence. The scorer is trained on box annotations with the MLLM frozen, so ETD is a learned selector, and we compare it with selectors learned in the same way. ETD covers 62.9% of the answer tokens and halves the teacher KL relative to DivPrune.

On nine benchmarks with Qwen3.5-4B at a strict 5% budget, Glance-ETD retains 90.3±0.8% of the full-token score over three training seeds, 14.5 points above training-free DivPrune and 11.3 above LT-OPD, and reaches the first token on HR-Bench 4K 2.9× sooner than the full-token model (Fig. 1 and Tab. S14). The same pipeline retains 89.5±1.1% on Gemma 4 E4B-it, where LT-OPD retains 76.3% under that same strict 5% budget.

Our contributions are a glance encoder and ETD, together with a controlled study at a strict 5% budget.

- **A glance encoder** that spends full resolution on a quarter of the image and keeps the native token grid. It cuts time-to-first-token on 4K images by 2.9× at a cost of 1.4 retention points against a dense encoder trained with the same recipe and the same budget.
- **ETD**, a selection rule that keeps the diversity loop of DivPrune and tilts it by a supervised evidence map. With the MLLM frozen it improves strict-5% retention by 8.4 points over DivPrune on Qwen3.5-4B and 10.4 on Gemma 4, and after self-distillation by 2.9 points at the same recipe. Most of the gain is on reading.
- **A controlled study at a strict 5% budget**: three training regimes at the same K per image, the recipe separated from the components (a 10× encoder learning rate alone adds 3.5 points), three training seeds, and eight held-out benchmarks (DocVQA +22.0, InfoVQA +10.6 over DivPrune, held out of the experimental design).

2. Related Work and Context

2.1. Training-Free Visual Token Reduction.

Training-free methods reduce the visual sequence without updating the multimodal language model. They either prune tokens inside the language model using text-to-image attention (FastV [2], SparseVLM [25], and PyramidDrop [19]) or select/merge tokens before the lan-

guage model using encoder-side signals (VisionZip [20] and PruMerge [11]). A central design choice is the trade-off between coverage and relevance. DivPrune [1] treats selection as max–min diversity and is a strong question-agnostic baseline, whereas CDPruner [22] introduces question-conditioned diversity. DART [16] removes duplicated tokens, and VisPruner [23] combines encoder attention with redundancy removal. Recent methods use zeroth-order sensitivity (ZOO-Prune [7]), within-group redundancy (PruneSID [5]), attention with text-guided complements (VisionTrim [21]) or head-weighted text-to-image attention (HAWK [26]); SPARE [8] replaces cosine diversity by subspace reconstruction, and MetaCompress [15] learns the compression across regimes.

2.2. Training with Reduced Visual Tokens.

Training can recover part of the accuracy lost by aggressive inference-time reduction. LLaVA-Mini [24] compresses visual information into a small number of query tokens before the language model, while EPIC [17] progressively distills a compressed student from a full-token teacher. LearnPruner [13] learns a pruning module after the vision encoder and combines it with text-guided pruning inside the language model. LT-OPD [9], closest to our setting, distils a student on its own rollouts from a frozen full-token copy with a curriculum over the token budget; we use its recipe and study the selector and the encoder on top of it.

2.3. Evidence-Aware High-Resolution Selection.

Text-rich inputs make evidence placement critical. ETPrune [4] prunes inside the LLM at about half of the tokens, reading question-conditioned evidence from decoder attention, protecting text-like regions and setting a per-sample budget; DocPrune [3] removes background and question-irrelevant blocks of document images. ETD differs in operating at 5% before the LLM, taking evidence from a separately trained box-supervised scorer, and keeping a diversity term, and it is combined with a reduced-cost encoder. Visual CoT [12] supplies grounded boxes, which we use only to train the scorer, and V* [18] and divide–conquer systems [14] search high-resolution images over several steps; the glance encoder instead lowers the cost of a single pass through the encoder on large images.

3. The Glance-ETD Method

Glance-ETD reduces the visual input in two stages (Fig. 3 and Alg. S1): the glance encoder (Sec. 3.2) decides where to look at full resolution, and ETD (Sec. 3.3) chooses the K tokens passed to the language model. Section 3.4 describes the teacher-KL diagnostic, and Sec. 3.5 describes training.

3.1. Preliminaries and Budgets

Problem setup. An image is encoded into N visual tokens $X = \{x_i\}_{i=1}^{N}$ in the LLM embedding space, after the patch merger. Given a question q and a budget $K \ll N$, a selector returns an index set $S \subset \{1, \dots, N\}$ with $|S| = K$, and only X_S is fed to the language model, with its original positions. We use two budget rules. *Strict*: $K = \lceil rN \rceil$ with $r = 5\%$, so that $\sum K / \sum N = 5.0\%$ on every benchmark. *Floor-32*: $K = \max(32, \lceil 0.05N \rceil)$, the rule of the LT-OPD release, which keeps 7.1% of all tokens because small images hit the floor. All selectors compared in this paper keep the same K per image and differ only in which tokens they keep, not in how many visual tokens each image receives.

Diversity selection. DivPrune [1] starts from the token closest to the centroid and greedily adds

$$i^\star = \arg\max_{i \notin S} \; d_i, \qquad d_i = \min_{j \in S} \big(1 - \cos(x_i, x_j)\big), \quad (1)$$

until $|S| = K$. The max–min rule spreads the budget over the image and ignores q. We use it in both stages: in Stage 1 to choose which regions to see at native resolution, and in Stage 2 as the diversity term of ETD.

On-policy self-distillation. LT-OPD [9] trains a student MLLM that sees only X_S to match a frozen copy of itself that sees all of X. The student generates answers with the pruned input, and the teacher provides token-level supervision along these rollouts. We adopt this objective for every model trained in this paper as well.

3.2. Stage 1: Glance Encoder

Post-encoder selection leaves the encoder untouched, and for large images the encoder dominates: on HR-Bench 4K, 89% of its FLOPs are global attention over about 50k patches. The glance encoder decides, before full-resolution encoding, which parts of the image to see in detail.

Two-pass encoding. (i) *Glance.* The image is average-pooled 2×2 and encoded with full attention at a quarter of the patches, giving one coarse token per 2×2 cell of the native merged grid. (ii) *Look.* DivPrune (Eq. (1)) on the coarse tokens selects 25% of the cells; the native patches of all selected cells are encoded as one sequence with global attention among them, with the absolute and rotary position codes of their native positions. (iii) *Merge.* After the patch merger, selected cells take their second-pass tokens and every other cell copies its coarse token to its four native slots. The grid has the size of the dense encoder's, so any post-encoder selector applies unchanged, and the LLM gives every slot its own M-RoPE position. The four copies are identical features, so once one is kept the others have

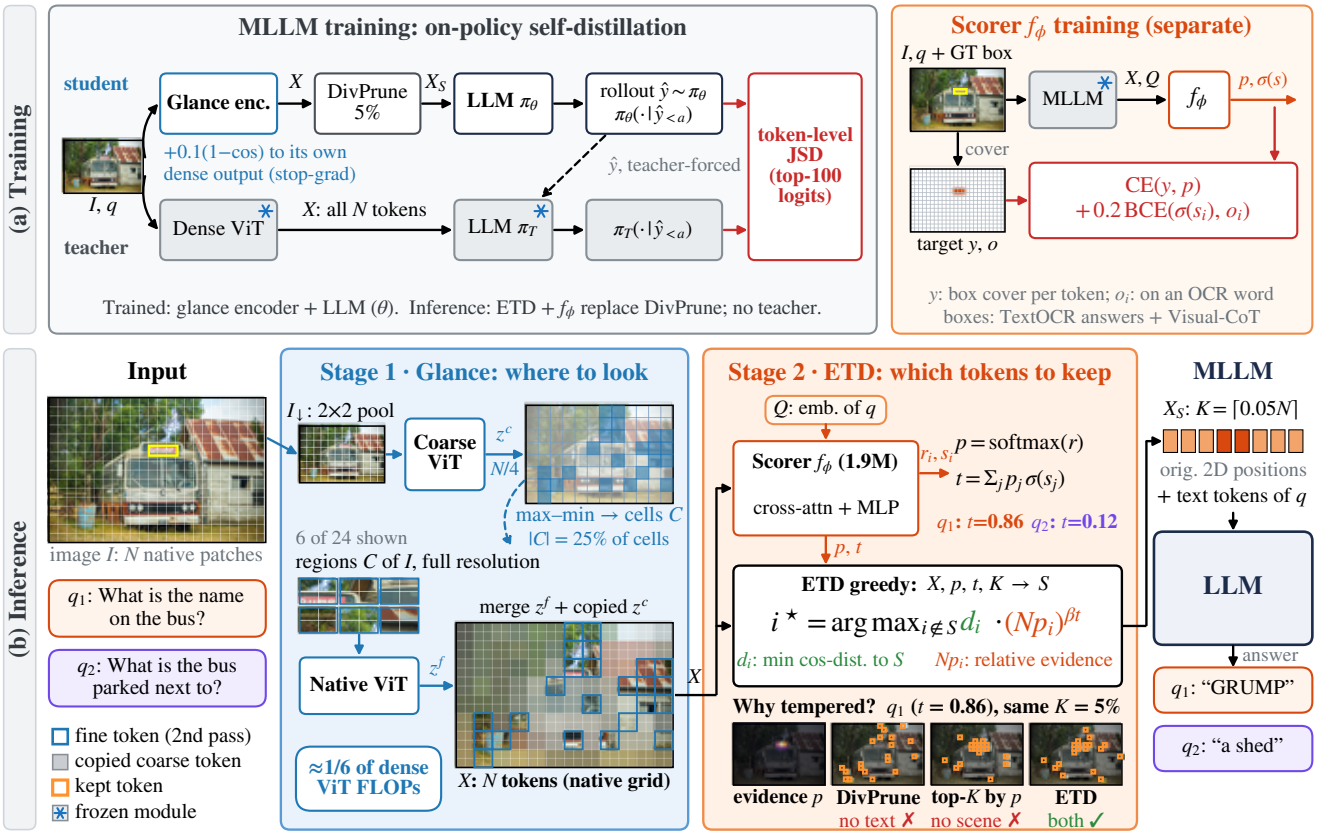


Figure 3. **Glance-ETD.** (a) Training: the student (glance encoder and LLM) selects with DivPrune, rolls out, and is distilled token by token towards a frozen full-token teacher; the scorer f_ϕ is trained separately on box targets with the MLLM frozen. (b) Inference: Stage 1 (Sec. 3.2) encodes the image at a quarter of the patches, picks 25% of the coarse cells by max–min diversity and re-encodes only these at native resolution, giving a grid X with all N native positions. Stage 2 (Sec. 3.3) maps X and the question to an evidence map p and a text confidence t, and ETD greedily selects K tokens by $d_i(Np_i)^{\beta t}$ (selections on an illustrative evidence map).

$d_i = 0$ and are not selected again. The encoder costs about a sixth of the dense FLOPs, and since the 5% budget is a fifth of the high-resolution pool, the kept tokens come mostly from regions seen at native resolution.

Training. The glance encoder is trained inside the MLLM by the self-distillation objective, with an auxiliary cosine loss (weight 0.1) towards the merger output of the dense encoder. The cell choice is a discrete max–min selection and receives no gradient: training adapts the encoder features along the chosen path, not which cells are chosen. It is applied to images with at least 4096 patches; smaller images are encoded densely at native resolution.

3.3. Stage 2: Evidence-Tempered Diversity

ETD keeps the diversity loop of Eq. (1) and tilts it towards query evidence by an amount set per question. It has two parts: a scorer that produces an evidence map and a confidence, and a greedy rule that uses them.

Evidence scorer. A small network f_ϕ reads the visual tokens and the question embeddings and outputs, for every visual token, a relevance logit r_i and a text logit s_i. Visual tokens are projected to $h = 256$ dimensions and given a 2D sine–cosine position code; one cross-attention layer lets them attend to the projected question tokens, producing a context c_i; an MLP on $[h_i, c_i, h_i \odot \bar{q}]$, with $\bar{q}$ the mean question feature, produces (r_i, s_i). The scorer has about 1.9M parameters and reads the MLLM's own token features, so no extra encoder is needed; with the glance encoder it reads the merged native grid, the same input that ETD selects from. From its outputs we define the evidence distribution and the *text-evidence confidence*

$$p = \text{softmax}(r), \qquad t = \sum_{j=1}^{N} p_j \, \sigma(s_j) \in [0, 1], \tag{2}$$

i.e. the probability, under the scorer's own evidence map, that the evidence lies on text tokens.

Selection rule. ETD replaces the greedy score of Eq. (1) with the evidence-tempered score below,

$$i^\star = \arg\max_{i \notin S} \; d_i \left(N p_i\right)^{\beta t}. \tag{3}$$

The factor Np_i is the evidence of token i relative to a uniform map, and βt acts as an inverse temperature. If the scorer finds no evidence (p uniform), every weight is one and ETD is exactly DivPrune. For scene and object questions t is small (median 0.12 on GQA and POPE) and the selection stays close to diversity; for reading questions t is large (median 0.84 on TextVQA, 0.91 on OCRBench, on the dense LT-OPD-recipe model) and high-evidence tokens dominate. No threshold or question classifier is involved, and we set $\beta = 1$ throughout. Because the distance term remains, a text region does not absorb the whole budget: once a few tokens of a word are kept, its neighbours have small d_i and the budget moves on. Hard splits and gates between the two criteria are compared in Sec. 4.4. We compute Eq. (3) in log space with the weights shifted by their maximum, which keeps the ranking exact.

Scorer supervision. The scorer is trained from boxes, not from benchmark labels. For TextVQA training questions we align answer words to TextOCR word boxes; for Visual-CoT [12] we use the annotated evidence boxes, which cover both text and general objects. The relevance target of token i is the fraction of its cell covered by evidence boxes, normalised over the image. The loss is the listwise cross-entropy between this target and p, plus a binary cross-entropy (weight 0.2) on $\sigma(s_i)$ for whether the cell touches an OCR word. Some images overlap evaluation sets (*e.g.* 491 of 5,349 DocVQA images); a scorer trained without them is in Sec. D.3. The 422K training records (28K TextVQA, the rest Visual-CoT) are seen for three epochs, with 60K training samples in each epoch.

3.4. Teacher-KL Diagnostic

Kept tokens stay at their native positions, and each new index maximises its cosine distance to that set, scaled by evidence (Eq. (3)). For a full-token greedy answer A of at most 32 tokens, the cost of a subset S is

$$\mathcal{L}(S) = \sum_{a=1}^{|A|} \mathrm{KL}\big(\pi(\cdot \mid X, q, A_{<a}) \,\big\|\, \pi(\cdot \mid X_S, q, A_{<a})\big), \tag{4}$$

computed with teacher forcing along A on the untrained model. On 8,000 training questions at a fixed K, ETD reaches half the $\mathcal{L}$ of DivPrune, almost all of it on text questions, and a power law in the kept fraction gives $K_m \propto N_m^{0.45}$ (Secs. D.1 and 4.4).

3.5. Training and Inference

MLLM. We train Qwen3.5-4B with on-policy self-distillation (Sec. 3.1); the student rolls out with 5% of the tokens selected by DivPrune. The training loss is the token-level generalised Jensen–Shannon divergence ($\alpha = 0.5$) between teacher and student along the student's rollouts, computed on a top-100 token support with an aggregated tail bucket; it is distinct from the KL of Eq. (4). We use one rollout per prompt and a 256-token answer cap, which shortens training from 17.4 h to 4.7 h on 8 A800 GPUs at the same accuracy (Sec. B.5). Training uses LT-OPD-14K with 1.5K spatial-reasoning samples replacing the same number of originals, for 175 steps. The vision encoder uses learning rate 2×10^{-6}, ten times the LT-OPD default, so that the glance encoder can adapt to its two-resolution input. This is a change of training recipe, not part of either component: it alone raises the dense model with DivPrune from 85.3% to 88.8%, and every component comparison in Sec. 4.3 holds this encoder learning rate fixed.

Scorer and inference. The scorer is trained once on the features of the untrained MLLM and shared by all Qwen models. Retraining it on the features of each trained model gives 91.5% on the dense-encoder model (91.9% with the shared scorer, seed 0) and $90.0 \pm 0.4\%$ on the glance model over three seeds ($90.3 \pm 0.8\%$), so the shared scorer is kept. At inference, the glance encoder produces the token grid, the scorer computes p and t, and ETD selects K tokens, which enter the LLM with their original positions. The MLLM is thus trained with DivPrune and evaluated with ETD. This keeps the MLLM independent of the scorer, so the same rule applies to the untrained model; training with ETD instead is compared in Sec. 4.3.

4. Experiments and Results

4.1. Setup, Benchmarks, and Budgets

Models. All experiments use Qwen3.5-4B [10] in non-thinking mode. **Glance-ETD** is the self-distilled model with the glance encoder (Sec. 3.5), evaluated with ETD; the *dense-encoder variant* is trained identically without the glance encoder; *untrained* is the released model.

Benchmarks and metric. We report the nine benchmarks of LT-OPD: V* [18], HR-Bench 4K [14], GQA, MMMU, MMBench (circular), MME, POPE, TextVQA and OCR-Bench, summarised by the *retention*, the mean ratio of each score to the same model's full-token score (first row of Tab. 1; repeated full-token runs in Sec. B.2). Eight benchmarks added after all design decisions serve as a held-out test (Sec. D.2).

Table 1. Strict 5% budget, same K per image for all methods. Rows are grouped by what is trained: nothing, an added module (MLLM frozen; ETD trains only its scorer), or the MLLM. DPP-r: CDPruner's DPP on our scorer. Self-distillation + DivPrune / + ETD: the same training recipe with either selector; + ETD is ours without glance. Ret.: mean ratio to full tokens over the nine LT-OPD benchmarks; the five held-out columns are excluded. Ours: mean±std of three seeds (Tab. S1). ‡: GQA lower-cased.

	General					Fine-grained high resolution					Text and document				
Method	GQA	MMMU	MMB	MME	POPE	V*	HR-4K	HR-8K	MME-RW	HRScene	TextVQA	OCRB	DocVQA	InfoVQA	Ret. (%)
Training-free								*Qwen3.5-4B*							
Full tokens (100%)	60.0	51.3	79.0	2218	89.9	80.6	78.0	72.6	48.9	62.5	80.5	848	94.6	79.4	100.0
DivPrune [1] (CVPR'25)	45.3	46.3	63.4	1695	77.4	70.2	73.4	70.5	38.1	49.6	49.2	264	32.1	21.7	75.8
DART [16] (EMNLP'25)	45.0	48.2	53.6	1529	77.6	49.7	61.5	60.4	30.1	45.0	49.6	125	24.9	21.8	67.7
CDPruner [22] (NeurIPS'25)	35.7	47.2	59.7	1605	74.3	53.9	62.7	56.4	29.7	43.3	43.3	215	21.8	20.5	67.6
PruneSID [5] (ICLR'26)	44.3	47.0	55.0	1602	77.9	69.6	73.8	68.8	35.0	49.2	25.1	108	22.3	20.7	68.7
VisionTrim [21] (ICLR'26)	41.8	45.8	55.4	1514	69.4	48.7	63.8	59.0	23.9	43.8	20.3	89	15.8	21.5	61.4
HAWK [26] (CVPR'26)	30.7	47.4	49.0	1336	70.9	49.2	60.4	56.2	22.5	44.3	16.5	72	9.7	18.9	56.9
ZOO-Prune [7] (CVPR'26)	46.2	46.4	60.1	1646	81.3	72.3	72.2	67.9	36.8	49.4	26.8	118	25.5	20.8	70.9
Frozen MLLM, learned module								*Qwen3.5-4B*							
LLaVA-Mini [24] (ICLR'25)	43.5	47.9	65.3	1579	75.7	56.5	68.6	61.7	32.4	50.1	34.9	236	34.7	27.4	70.3
LearnPruner [13] (ICLR'26)	43.3	45.6	55.1	1505	72.1	48.2	69.0	60.9	29.2	43.7	28.2	87	12.7	19.3	63.6
MetaCompress [15] (CVPR'26)	41.9	46.9	66.1	1576	69.8	63.4	70.4	62.7	27.8	49.2	41.2	241	28.5	22.9	71.3
DPP-r	40.6	46.1	64.3	1736	85.5	79.1	73.8	69.1	42.0	51.5	70.7	420	68.1	34.5	82.5
ETD (ours)	48.4	47.3	62.3	1750	84.3	77.0	76.4	73.0	41.8	51.8	72.2	427	74.3	39.2	**84.2**
MLLM adaptation								*Qwen3.5-4B*							
EPIC‡ [17] (NeurIPS'25)	42.1	44.4	57.0	1233	66.3	54.5	64.8	59.8	39.8	48.2	50.9	120	25.4	22.9	65.1
LT-OPD [9] (arXiv'26.09)	47.4	48.6	66.4	1771	81.9	67.0	69.6	65.1	40.5	51.4	55.8	341	40.6	25.6	79.0
Self-distillation + DivPrune	54.2	48.3	73.5	1988	85.9	75.0	77.1	73.5	45.9	56.2	70.4	489	68.4	36.0	88.8±0.2
Self-distillation + ETD (ours)	55.1	48.7	75.2	2000	87.8	77.1	76.2	74.4	46.1	55.9	76.0	579	79.7	45.8	91.7±0.2
Glance-ETD (ours)	54.5	48.2	72.8	1932	86.7	77.0	76.7	72.1	45.5	53.9	75.2	553	76.4	43.4	**90.3**±0.8
Training-free								*Gemma 4 E4B-it*							
Full tokens (100%)	41.3	46.9	73.6	1686	88.1	58.1	61.1	54.4	42.4	54.6	71.9	764	88.3	62.0	100.0
DivPrune [1] (CVPR'25)	37.1	42.9	60.5	1103	83.9	49.2	53.0	46.1	23.7	43.7	38.9	184	14.7	15.0	74.8
DART [16] (EMNLP'25)	33.2	41.3	52.5	934	72.4	34.6	40.2	35.8	17.9	34.7	19.9	83	9.1	12.3	60.2
CDPruner [22] (NeurIPS'25)	35.3	42.2	54.7	962	78.4	28.8	44.4	38.6	14.3	37.3	28.9	138	10.4	13.5	64.0
PruneSID [5] (ICLR'26)	36.8	41.1	60.2	1164	79.9	27.2	41.2	38.4	21.4	39.7	30.9	246	12.3	16.1	67.5
VisionTrim [21] (ICLR'26)	33.5	38.4	55.7	1046	78.0	41.4	54.4	47.5	22.7	39.6	24.0	138	9.6	15.6	66.7
HAWK [26] (CVPR'26)	30.4	40.2	50.4	874	69.2	21.5	30.3	26.9	10.8	31.3	9.6	85	7.4	13.6	52.1
ZOO-Prune [7] (CVPR'26)	36.5	43.1	60.1	1062	81.8	34.0	41.2	36.6	19.9	39.9	24.9	151	10.6	15.2	66.5
Frozen MLLM, learned module								*Gemma 4 E4B-it*							
LLaVA-Mini [24] (ICLR'25)	30.3	39.9	30.5	1018	65.5	38.7	32.5	35.4	29.2	34.9	11.9	62	8.5	18.3	53.2
LearnPruner [13] (ICLR'26)	32.6	40.9	47.9	952	72.7	28.3	36.3	37.3	14.6	35.3	13.3	25	4.1	12.0	55.6
MetaCompress [15] (CVPR'26)	35.6	40.6	59.4	1063	77.7	27.2	33.9	31.2	17.6	36.8	22.9	196	10.8	16.5	62.7
ETD (ours)	37.0	43.0	62.3	1230	82.2	54.5	60.2	54.0	31.1	41.7	61.0	440	42.3	23.2	**85.2**
MLLM adaptation								*Gemma 4 E4B-it*							
EPIC [17] (NeurIPS'25)	40.5	43.1	55.3	1362	78.2	44.5	47.6	42.2	34.5	41.5	24.5	171	13.7	17.9	71.7
LT-OPD [9] (arXiv'26.09)	39.3	43.1	63.5	1176	82.5	37.7	52.2	47.8	27.5	42.6	39.1	349	16.2	16.1	76.3
Self-distillation + DivPrune	41.3	42.9	70.8	1386	87.3	49.7	57.3	53.0	30.2	44.4	56.8	440	25.8	18.8	87.2±1.2
Self-distillation + ETD (ours)	40.9	43.9	70.8	1471	86.1	48.9	59.1	53.1	33.9	45.8	64.1	513	43.2	25.2	90.1±0.9
Glance-ETD (ours)	40.8	44.1	69.5	1528	86.1	45.9	57.5	47.4	31.8	45.0	60.8	553	35.8	24.7	**89.5**±1.1

Budgets and baselines. The main budget is strict 5% (Sec. 3.1); floor-32 is in Sec. E.1. Baselines are peer-reviewed methods from 2025–2026 (Tab. 1), re-run on Qwen3.5-4B with the same K per image as ETD. Training-free methods are ported from the official code to select after the vision merger (Sec. B.3). The modules of LLaVA-Mini, LearnPruner and MetaCompress are trained on LT-OPD-14K with the MLLM frozen. DPP-r keeps CDPruner's conditional DPP and replaces its question–token similarity by the min-max normalised relevance logits of the same scorer. LT-OPD is trained with its own recipe on each model (Sec. B.4).

Implementation. We use VLMEvalKit and lmms-eval, with greedy decoding on A800 GPUs. Open groups in MM-Bench are adjudicated by fixed rules without a judge model (Sec. B.2). Inference is not bit-exact; V* varies by up to 2 points between runs.

4.2. Main Results across Benchmarks

Frozen MLLM. With the MLLM untouched, ETD keeps 84.2% at strict 5%, 8.4 points above DivPrune, the strongest training-free baseline (Tab. 1). The learned modules of LLaVA-Mini and LearnPruner stay below DivPrune. DPP-r reaches 82.5%, 6.7 of ETD's 8.4 points over DivPrune, so most of that gain is the supervised evidence rather

Table 2. Component ablation at a strict 5% budget (same K per image). Glance: two-pass glance encoder (Sec. 3.2); without it the dense encoder is used. ETD: evidence-tempered selection (Sec. 3.3); without it DivPrune. The 10× columns share one recipe and differ only in the two components; 1× is the LT-OPD recipe; No train leaves the MLLM untouched. TFLOPs: prefill (encoder + LLM, attention included) on HR-Bench 4K, set by the encoder and K; the ETD scorer adds about 0.02. Trained entries are means over three training seeds, ± the standard deviation of Ret.

	10× LR				1× LR		No train	
Glance	–	–	✓	✓	–	–	–	–
ETD	–	✓	–	✓	–	✓	–	✓
V*	75.0	77.1	74.7	77.0	73.8	77.0	70.2	77.0
HR-4K	77.1	76.2	76.7	76.7	77.2	76.8	73.4	76.4
GQA	54.2	55.1	53.4	54.5	52.3	53.1	45.3	48.4
MMMU	48.3	48.7	49.1	48.2	48.9	48.9	46.3	47.3
MMB	73.5	75.2	71.7	72.8	70.6	71.3	63.4	62.3
MME	1988	2000	1902	1932	1865	1858	1695	1750
POPE	85.9	87.8	84.5	86.7	85.2	87.9	77.4	84.3
TextVQA	70.4	76.0	70.2	75.2	64.7	73.2	49.2	72.2
OCRB	489	579	474	553	394	488	264	427
Ret.	88.8 ±0.2	91.7 ±0.2	87.7 ±0.4	90.3 ±0.8	85.3 ±0.2	88.7 ±0.4	75.8	84.2
TFLOPs	373	373	65	65	373	373	373	373

than the selection rule. The other 1.7 points come from GQA (40.6 against 48.4), where DPP-r tilts every question while ETD stays close to DivPrune; V* moves the other way (79.1 against 77.0). On the trained glance model the same substitution costs 0.2 points (Tab. 3). The importance-based methods (ZOO-Prune, PruneSID, VisionTrim, HAWK; 56.9–70.9%) concentrate a 5% budget on a few salient regions and lose most of the text (TextVQA 17–27, OCRBench 72–118, against 49.2 and 264 for DivPrune and 72.2 and 427 for ETD). ETD also improves V*, HR-Bench and POPE over DivPrune (+6.8, +3.0, +6.9) and leads at every budget from 1% to 25% (Sec. E.2 reports that sweep).

MLLM adaptation. Trained rows are means over three training seeds (Tab. S1). LT-OPD retains 79.0%. Under our self-distillation recipe, which differs only in the selector, DivPrune retains 88.8% and ETD 91.7%; Glance-ETD retains 90.3±0.8%. Section 4.3 splits the gap to LT-OPD into recipe and components. EPIC (65.1%) stays below DivPrune, mostly on reading. At floor-32 (7.1% of the tokens) LT-OPD retains 86.1%, the LT-OPD recipe with DivPrune 92.2% and Glance-ETD 96.0±0.4% (Sec. E.1).

Gemma 4. On Gemma 4 E4B-it [6] (∼1,100 tokens per image) with the MLLM frozen, ETD keeps 85.2%, 10.4 points above DivPrune and above the other training-free methods (52.1–67.5%), mostly on reading (TextVQA 38.9→61.0, OCRBench 184→440; Sec. E.5). With MLLM adaptation, Glance-ETD retains 89.5±1.1% and ours without glance 90.1±0.9% (LT-OPD 76.3%).

4.3. Component Ablation Study

Table 2 removes each stage at strict 5%. In the 10× columns both encoders use the same recipe with a 10× encoder learning rate and differ only in the two components. The learning rate alone adds 3.5 points to the LT-OPD recipe with DivPrune (85.3→88.8%). At this recipe ETD adds 2.9 (91.7%), and the glance encoder costs 1.4 while cutting HR-Bench 4K prefill from 373 to 65 TFLOPs and time to first token from 3305 to 1376 ms on HR-Bench 4K.

ETD. ETD improves every model at the same K: +2.9 on the dense encoder, +2.6 on the glance encoder, +3.4 with the LT-OPD recipe and +8.4 without MLLM training, mostly on TextVQA and OCRBench (70.2→75.2 and 474→553 on the glance encoder). The MLLM is trained with DivPrune and evaluated with ETD. Training with ETD too adds 1.1 on the dense 1×-LR model (85.1 / 87.9 / 89.0 for DivPrune→DivPrune / DivPrune→ETD / ETD→ETD, one seed), and that model falls to 83.7 with DivPrune: the MLLM co-adapts with its selector. On the glance model training with ETD gives no gain (89.2 *vs*. 89.5 with DivPrune training, seed 0) and the model falls from 87.5 to 86.2 with DivPrune, so we keep DivPrune training, which also leaves the MLLM independent of the scorer.

Glance. At the same recipe the glance encoder costs 1.1 points with DivPrune and 1.4 with ETD. V* and HR-Bench 4K are unchanged (77.0 *vs*. 77.1, 76.7 *vs*. 76.2); the loss is on MMBench (−2.4) and MME (−68). MMBench images are below the glance threshold and encoded densely, so this loss comes from training the shared encoder (Sec. D.4). Choosing the cells with the scorer at the same latency raises the held-out mean from 53.1 to 55.0–55.5 (Tab. S7).

4.4. What matters in the selection rule

Table 3 changes the selection rule on Glance-ETD (Qwen, seed 0) and training-free ETD (Gemma); dense model results are reported in Tab. S5 as well.

Evidence and diversity do different jobs. Dropping the evidence term leaves DivPrune and costs 2.0 points on Qwen and 10.4 on Gemma, mostly on reading (Gemma TextVQA 61.0→38.9, OCRBench 440→184). Dropping diversity and keeping the top K tokens by p costs 2.4 and 3.9: it raises V* and HR-Bench on Qwen but lowers GQA from 54.4 to 51.2 and OCRBench from 552 to 459. Giving the scorer half of the budget costs 1.9 and 2.8, and a hard

Table 3. Selection-rule ablation at a strict 5% budget and the same K per image. Qwen is Glance-ETD; Gemma 4 is training-free ETD. Ret. is over nine benchmarks against that model's full-token score; six are shown. Δ is the difference in nine-benchmark retention. Qwen rows are single runs of training seed 0 (90.3 in Tab. 1 is the three-seed mean); three-seed comparisons are in Sec. 4.4. –: not run.

	Qwen3.5-4B, Glance-ETD								Gemma 4 E4B-it, ETD (training-free)							
Setting	V*	HR-4K	GQA	POPE	TVQA	OCRB	Ret.	Δ	V*	HR-4K	GQA	POPE	TVQA	OCRB	Ret.	Δ
Full pipeline	75.9	75.9	54.4	87.3	75.4	552	89.5	–	54.5	60.2	37.0	82.2	61.0	440	85.2	–
Stage 2: which tokens (same K per image)																
w/o evidence: DivPrune ($\beta t = 0$)	73.3	77.4	53.6	84.6	70.3	475	87.5	-2.0	49.2	53.0	37.1	83.9	38.9	184	74.8	-10.4
w/o diversity: top-K by p ($\beta = 64$, no t)	76.4	77.8	51.2	86.5	72.6	459	87.1	-2.4	52.4	57.2	37.3	82.9	51.7	358	81.3	-3.9
w/o confidence t: $d_i(Np_i)$	77.0	75.1	53.0	87.5	75.6	542	88.9	-0.6	50.3	65.4	35.7	79.6	60.7	434	84.7	-0.5
t replaced by constant 0.25	75.4	76.9	54.7	87.5	75.7	553	90.4	+0.9	55.0	62.0	37.7	83.3	59.9	396	84.8	-0.4
hard split: top-$K/2$ by p + DivPrune	73.3	75.2	52.6	85.3	73.1	505	87.6	-1.9	49.7	60.9	37.0	83.3	55.3	363	82.4	-2.8
hard gate: split only if $t > 0.5$	73.3	76.9	53.5	84.5	72.8	504	88.2	-1.3	50.3	54.1	37.0	83.3	55.0	362	80.9	-4.3
$\beta = 0.5$	74.9	76.8	54.3	86.6	75.7	554	89.9	+0.4	51.8	57.1	37.6	82.6	60.8	423	84.0	-1.2
$\beta = 2$	78.5	75.9	54.4	87.6	75.6	530	89.6	+0.1	52.4	62.4	37.1	80.7	60.2	426	84.6	-0.6
scorer given a wrong question	74.9	75.8	53.2	86.1	71.3	464	87.4	-2.1	54.5	57.2	36.4	81.8	54.7	319	81.5	-3.7
DPP-r instead of ETD	74.3	75.9	51.7	88.0	75.9	556	89.3	-0.2	–	–	–	–	–	–	–	–

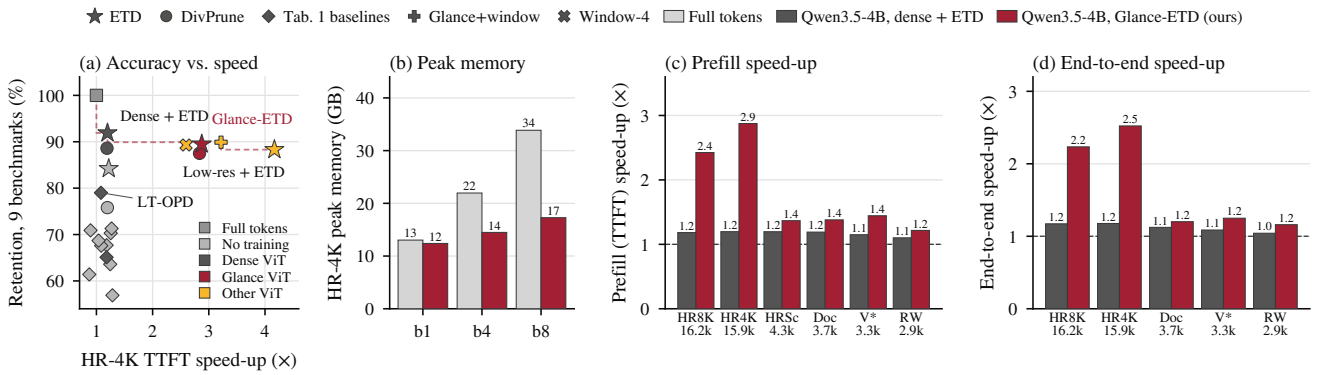


Figure 4. Efficiency of Qwen3.5-4B at strict 5% (one A800, bf16). (a) Retention (9 benchmarks, seed 0 of Tab. S4; baselines: Tab. 1) against HR-Bench 4K TTFT speed-up; stars ETD, circles DivPrune, diamonds the other Tab. 1 methods (Tab. S15), dashed line the Pareto front. (b) HR-Bench 4K peak memory at batch 1, 4 and 8. (c) Prefill-stage (TTFT: ViT, selection, LLM prefill) and (d) end-to-end speed-up over full tokens at batch 1 on many-token benchmarks at native size (visual tokens per image below each name; omitted when no bar exceeds 1.15×), 30 questions each, same run as Tab. S14. Resolution sweep, numbers and further views: Tab. S14 and Fig. S2.

gate at $t > 0.5$ costs 1.3 and 4.3; on Qwen the gate's V*, GQA and POPE are those of DivPrune.

Scorer or rule? Over three training seeds of the glance model, a constant exponent of 0.25 is 0.5 points above βt (paired bootstrap over questions, 95% interval $[0.1, 1.0]$), and DPP-r is 0.2 below ETD ($[-0.8, 0.5]$); $\beta \in \{0.5, 2\}$ and dropping t move retention by at most a point. As a detector, t separates reading questions (TextVQA, OCR-Bench) from perception questions with AUROC 0.90, but the best tilt does not grow with it: in each t quintile the best constant exponent is about 0.25, and choosing it per quintile gains only 0.1 over a single 0.25. The gain comes from p, not from t. Replacing p matters more: the ETD rule with question–token similarity gives 87.2 (0.3 below DivPrune), and a scorer fed a different question costs 2.1 points on Qwen and 3.7 on Gemma. ETD matches DPP-r without a kernel or a determinant.

Does ETD keep the answer? On TextVQA val (4,123 questions, strict 5%), a kept token counts when its cell overlaps the answer OCR box: 62.9% of those tokens and at least one on 99.0% of questions, against 10.4% and 80.2% for DivPrune and at most 11.5% for the other selectors (Fig. 2c, Sec. E.4).

4.5. Latency, Memory, and FLOPs

Figure 4 reports speed and memory (Sec. F). On HR-Bench 4K the ViT is 2.59 s of a 3.96 s TTFT, and dense-encoder selectors stay at 0.87–1.28× (LT-OPD 1.08×, ETD 1.22×; peak memory 13.0–13.3 GB, VisionTrim 14.7 GB). Glance cuts the ViT to 0.65 s (2.87× prefill, 2.52× end to end) for 1.4 points and halves batch-8 memory from 34.7 to 17.7 GB; low-resolution and windowed encoders lose 3–8

points on V* and HR-Bench (Fig. 4 and Tab. S15).

5. Conclusion and Limitations

Glance-ETD splits an extreme visual-token budget into two decisions: the glance encoder chooses where the image is seen at full resolution, removing encoder cost that token selection cannot touch, and ETD tilts diversity selection towards a supervised evidence map, which carries most of its gain on reading.

Limitations. The trained Stage 1 ignores the question (choosing its cells with the scorer repairs most of this, Tab. S7); the scorer needs box supervision per backbone; small images gain no encoder saving.

Statement on LLM Usage

We used Claude Code (Anthropic, Claude models) as a coding and writing assistant. It wrote and ran parts of the training, evaluation and analysis scripts under the authors' direction, generated the LaTeX tables from the evaluation outputs with scripts, and drafted and revised the manuscript text, which the authors edited and checked. It did not generate experimental measurements: every reported number comes from evaluation runs of the models, and MMBench adjudication is rule-based without a judge model. Citations were checked against the source papers by the authors. No generative model is an author of this paper.

References

[1] Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. DivPrune: Diversity-based visual token pruning for large multimodal models. In *CVPR*, 2025. 1, 3, 6

[2] 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 *ECCV*, 2024. 2

[3] Joonmyung Choi, Sanghyeok Lee, Jongha Kim, Sehyung Kim, Dohwan Ko, Jihyung Kil, and Hyunwoo J. Kim. DocPrune: Efficient document question answering via background, question, and comprehension-aware token pruning. In *CVPR*, 2026. 3

[4] Zizhong Ding, Junxian Li, Kai Liu, Shaoqiu Zhang, Xiao Xiao, Linghe Kong, and Yulun Zhang. ET-Prune: Evidence-aware dynamic budgeting for visual token pruning in text-rich MLLMs. *arXiv preprint arXiv:2608.01979*, 2026. 3

[5] Zhengyao Fang et al. Prune redundancy, preserve essence: Vision token compression in VLMs via synergistic importance-diversity. In *ICLR*, 2026. 3, 6

[6] Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026. 7

[7] Kim et al. ZOO-Prune: Training-free token pruning via zeroth-order gradient estimation in vision-language models. In *CVPR*, 2026. 3, 6

[8] Jaeyeon Lee, Shunjie Wen, and Dong-Wan Choi. Moving beyond diversity: Visual token pruning as subspace reconstruction for efficient VLMs. *arXiv preprint arXiv:2606.18681*, 2026. 3

[9] Junxian Li, Ruixuan Yang, Tianao Zhang, Tiange Xu, Weisheng Dong, and Yulun Zhang. Fewer tokens, more self-teaching: On-policy self-distillation for extreme visual token reduction. *arXiv preprint arXiv:2609.32353*, 2026. 1, 3, 6

[10] Qwen Team. Qwen3.5: Towards native multimodal agents. https://qwen.ai/blog?id=qwen3.5, 2026. 5

[11] Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive token reduction for efficient large multimodal models. *arXiv preprint arXiv:2403.15388*, 2024. 3

[12] Hao Shao et al. Visual CoT: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. In *NeurIPS Datasets and Benchmarks*, 2024. 3, 5

[13] Takezoe et al. LearnPruner: Rethinking attention-based token pruning in vision language models. In *ICLR*, 2026. 3, 6

[14] Wenbin Wang et al. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *AAAI*, 2025. 3, 5

[15] Yi Wang, Haofei Zhang, et al. Rethinking token reduction for large vision-language models. In *CVPR*, 2026. 3, 6

[16] 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 *EMNLP*, 2025. 3, 6

[17] Zichen Wen, Shaobo Wang, et al. Efficient multi-modal large language models via progressive consistency distillation. In *NeurIPS*, 2025. 3, 6

[18] Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal LLMs. In *CVPR*, 2024. 3, 5

[19] Long Xing et al. PyramidDrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. In *CVPR*, 2025. 2

[20] Senqiao Yang et al. VisionZip: Longer is better but not necessary in vision language models. In *CVPR*, 2025. 3

[21] Hanxun Yu et al. VisionTrim: Unified vision token compression for training-free MLLM acceleration. In *ICLR*, 2026. 3, 6

[22] Qizhe Zhang et al. Beyond attention or similarity: Maximizing conditional diversity for token pruning in MLLMs. In *NeurIPS*, 2025. 3, 6

[23] Qizhe Zhang et al. Beyond text-visual attention: Exploiting visual cues for effective token pruning in VLMs. In *ICCV*, 2025. 3

[24] Shaolei Zhang, Qingkai Fang, Zhe Yang, and Yang Feng. LLaVA-Mini: Efficient image and video large multimodal models with one vision token. In *ICLR*, 2025. 3, 6

[25] Yuan Zhang et al. SparseVLM: Visual token sparsification for efficient vision-language model inference. In *ICML*, 2025. 2

[26] Zhu et al. HAWK: Head importance-aware visual token pruning in multimodal models. In *CVPR*, 2026. 3, 6

Glance, then Select: Evidence-Tempered Visual Token Selection for Extremely Compressed Multimodal LLMs

Supplementary Material

Contents of the Supplement

A Glance-ETD inference 1

B Protocol and Baselines 1
- B.1. Training Seeds and Variance . . . 1
- B.2. Evaluation Protocol and Scores . . . 1
- B.3. Baseline Ports from Code . . . 2
- B.4. LT-OPD Baseline Reproduction . . . 2
- B.5. Self-Distillation Training Cost . . . 2

C Other Models and Encoders 3
- C.1. LLaVA-1.5 in the standard setting . . . 3
- C.2. Reinforcement Learning of the Scorer . . . 3
- C.3. Vision-encoder variants . . . 3

D Selection Behavior 4
- D.1. Teacher-KL Selection Analysis . . . 4
- D.2 Held-out benchmarks . . . 5
- D.3. Training-image overlap . . . 5
- D.4. Stage-1 evidence coverage . . . 6

E Budgets and Diagnostics 7
- E.1. The LT-OPD operating point . . . 7
- E.2. Sweep across Token Budgets . . . 7
- E.3. Allocating the budget across images . . . 7
- E.4. Evidence diagnostics . . . 7
- E.5. Gemma 4 across budgets . . . 7

F Latency and Memory 8

A. Glance-ETD inference

Algorithm S1 is the inference procedure. Stage 1 runs on images with at least 4096 patches; smaller images are encoded densely. The first token is the one closest to the centroid, so a uniform evidence map leaves the rest of the loop identical to DivPrune. Each later token maximises Eq. (3) with $\beta = 1$. Training does not use this rule: the MLLM is distilled with DivPrune, and the scorer is trained separately (Sec. 3.5 describes that training).

Algorithm S1 Glance-ETD inference.

Require: image, question q, rate r=0.05, β=1
Ensure: K visual tokens at their native positions
if at least 4096 patches **then**
  encode the 2×2-pooled image
  $C \leftarrow$ DivPrune, 25% of the coarse cells
  re-encode the native patches of C
  $X \leftarrow$ second-pass on C, coarse copies elsewhere
else
  $X \leftarrow$ dense encoder output
end if
$(r_i, s_i) \leftarrow f_\phi(X, q)$
$p \leftarrow \mathrm{softmax}(r)$
$t \leftarrow \sum_j p_j \, \sigma(s_j)$
$N \leftarrow |X|, \quad K \leftarrow \lceil rN \rceil$
$S \leftarrow \{i \text{ closest to the centroid of } X\}$
while $|S| < K$ **do**
  $d_i \leftarrow \min_{j \in S}(1 - \cos(x_i, x_j)), i \notin S$
  $i^\star \leftarrow \arg\max_{i \notin S} \; d_i (N p_i)^{\beta t}$
  $S \leftarrow S \cup \{i^\star\}$
end while
return X_S

B. Protocol and Baselines

The entries below fix the seeds, the scoring protocol, the baseline ports, and the cost of the shared training recipe.

B.1. Training Seeds and Variance

Table 1 reports, for our method, the mean over three training seeds. Table S1 lists all three. Seeds differ only in the training random seed of each run.

B.2. Evaluation Protocol and Scores

All models are evaluated without thinking: the chat template pre-fills an empty think block and the answer tag. The LT-OPD paper normalises by thinking-mode full-token scores from the model card (e.g. MMMU 77.6), which no non-thinking row can approach; we normalise by one fixed non-thinking full-token run of our own instead. MMMU uses the official 128-token answer cap.

Full-token repeats. Two further full-token runs under the same protocol give V* 80.6 / 80.6 / 80.6, HR-Bench 4K 78.0

Table S1. Three training seeds at the strict 5% budget of Tab. 1. The main table reports the mean over the three seeds; the best seed is in bold.

Method	Seed	V*	HR-4K	GQA	MMMU	MMB	MME	POPE	TextVQA	OCRB	Ret. (%)
Self-distillation + DivPrune	0	74.9	77.0	54.0	47.9	73.9	1975	86.3	70.5	481	88.6
	1	74.9	76.6	54.1	48.3	73.2	2033	85.3	70.0	500	88.9
	2	75.4	77.8	54.4	48.7	73.5	1957	86.1	70.6	486	**89.0**
Glance-ETD	0	75.9	75.9	54.4	46.1	72.3	1905	87.3	75.4	552	89.5
	1	78.0	77.5	54.3	49.3	74.0	1943	86.6	75.0	563	**91.1**
	2	77.0	76.8	54.7	49.2	72.0	1947	86.4	75.3	543	90.4
Dense-encoder variant	0	78.5	77.8	55.0	47.7	75.6	2014	87.9	75.8	572	**91.9**
	1	77.5	74.8	55.1	48.6	75.1	2011	87.8	76.2	575	91.6
	2	75.4	76.1	55.1	49.8	74.9	1976	87.8	76.1	590	91.7

/ 78.0 / 78.0, GQA 60.0 / 60.0 / 60.0, MMMU 51.3 / 51.3 / 51.2, MMBench 79.0 / 79.2 / 79.2, MME 2218 / 2228 / 2226, POPE 89.87 / 89.93 / 89.89, TextVQA 80.49 / 80.56 / 80.56 and OCRBench 848 / 850 / 847 (the first run is the denominator). Across runs the scores differ by at most 10 on MME, 3 on OCRBench and 0.2 elsewhere. Normalising Glance-ETD (seed 0) by any one run gives 89.47, 89.37 or 89.44% retention, and 89.43% with the mean of the three, so the choice of denominator moves retention by at most 0.1 points under every denominator we tried.

MMBench adjudication. Circular-evaluation groups that rule matching cannot resolve are decided by two fixed rules without a judge model: a group is correct if every answer names the gold letter unambiguously, and wrong if some unparsable answer contains no reading that names the gold letter. Groups matching neither rule would need a judge. No row of Tab. 1 needs a judge. The rules decide at most 8 of the 1,164 groups in any row (0.7% of the MMBench score), and Glance-ETD has at most one such group per seed.

Budget accounting. Table 1 and the original LT-OPD code do not use the same budget. The code sets $K = \max(32, \lceil 0.05N \rceil)$, a floor of 32 visual tokens on every image, so the keep rate is not strictly 5% and is greater than 5%. On the nine benchmarks that floor raises the total from 5.0% to 7.1%. The main table removes the floor, $K = \lceil 0.05N \rceil$, and every strict row has $\sum K / \sum N = 5.0\%$. Table S5 repeats the selection rules under both. Every selector writes a per-image log of (N, K).

B.3. Baseline Ports from Code

All training-free baselines are ported from the official code and select tokens after the vision merger and before the LLM. ZOO-Prune perturbs the inputs of Qwen's patch merger, its vision-to-language projector. VisionTrim ranks tokens by the attention they receive in the penultimate ViT block (Qwen's encoder has no CLS token) and fills the rest of its budget with the tokens closest to the question. HAWK scores tokens with the pre-RoPE queries and keys of the first full-attention LLM layer, since the first layers of Qwen3.5 use linear attention; its head weights are recomputed for this layer, following the procedure given in Sec. C.3.

B.4. LT-OPD Baseline Reproduction

The LT-OPD rows in Tabs. 1 and S10 are the LT-OPD-14K recipe with the learnable token merge turned off, trained on each model. CDPruner still selects the tokens. In the released checkpoint, each discarded token is assigned to its nearest retained token by cosine similarity, and an MLP adds a bounded residual, initialized at zero, onto that retained token. We disable this merge, so the language model receives the selected tokens alone. That is pure index pruning, the reduction in the original LT-OPD method, and the same strict 5% subset the other training-based rows pass. The released Qwen3.5-4B checkpoint keeps the merge. At strict 5% it retains 85.4% on the nine LT-OPD benchmarks (V* 73.3, HR-Bench 4K 71.5, TextVQA 63.3, OCRBench 419), and at floor-32 91.1% (V* 73.3, HR-Bench 4K 71.6, TextVQA 63.6, OCRBench 572); our reproduction with the merge retains 90.3% at floor-32. On the five added benchmarks at strict 5% it scores HR-Bench 8K 67.8, MME-RealWorld 44.4, HRScene 54.7, DocVQA 48.9 and InfoVQA 27.4; on documents it is below the LT-OPD recipe with DivPrune (DocVQA 52.5) and well below Glance-ETD on that benchmark (DocVQA score 76.4).

B.5. Self-Distillation Training Cost

Table S2 varies the number of rollouts per prompt and the answer cap. One rollout with a 256-token cap trains in 4.7 h instead of 17.4 h without a loss in retention, and we use it for all trained models. The amount of data is not a bottleneck either: training on coverage-selected subsets of LT-OPD-14K with the same recipe keeps 90.0–90.7% from 700 samples (5%) up to 7K samples, against 90.4% with all 14K.

Table S2. Training cost of the self-distillation recipe (LT-OPD-14K, 175 steps, 8 A800 GPUs; CDPruner selection at the floor-32 budget of the LT-OPD release). Retention is against an earlier full-token reference (MMMU 52.4 instead of 51.3), which lowers every value by about 0.2 points relative to the main tables; the comparison within the table is unaffected.

Rollouts per prompt, answer cap	Training time	Ret. (%)
8, 1024 (LT-OPD default)	17.4 h	90.1
2, 1024	8.7 h	91.1
1, 1024	7.3 h	90.5
1, 256 (ours)	4.7 h	90.7
1, 128	4.3 h	90.3

C. Other Models and Encoders

The same selectors are measured on LLaVA-1.5, with a reinforcement-learned scorer, and with other vision encoders.

C.1. LLaVA-1.5 in the standard setting

Table S3 evaluates the selectors on LLaVA-1.5-7B with its own evaluation protocol (vicuna_v1 prompt, padding to a square, fp16, Rosetta OCR tokens for TextVQA), where most token-pruning papers report results. Our full-token scores reproduce the published ones (GQA 61.9, MME perception 1507, POPE F1 85.9, TextVQA 58.3, MMBench 64.3, ScienceQA-IMG 69.6). The ETD scorer is trained on LLaVA-1.5 features with the same data and loss, and the MLLM is not trained. With 576 visual tokens per image and K between 64 and 192 (11–33%), ETD is 0.4–0.6 points above DivPrune, mostly on TextVQA (+1.2 to +1.4) and MME. At $K = 29$ (5%, as in the main tables) ETD keeps 91.7% against 91.2% for DivPrune (TextVQA 54.6 *vs*. 53.0, MME 1327 *vs*. 1280, POPE and MMBench about one point lower). The margin is much smaller than at 5% on Qwen3.5-4B (+8.4); LLaVA-1.5 encodes every image at 336 pixels, and four of its six benchmarks are scene questions. Question similarity in place of the scorer is below DivPrune at every K (83.7–96.0%). The rows marked § are the numbers reported by CDPruner [22], whose DivPrune runs agree with ours. On this backbone ETD is above VisionZip and DART but below CDPruner (97.3 vs. 97.9 at $K = 128$, 95.0 vs. 96.3 at $K = 64$); CDPruner's advantage is on POPE (87.5–87.7 vs. 84.4–86.1), while ETD is higher on TextVQA. Its relevance term works on LLaVA-1.5, while on Qwen3.5-4B at 5% the same method keeps 67.6% (Tab. 1, at that strict budget).

C.2. Reinforcement Learning of the Scorer

We trained the scorer with GRPO, using the negative teacher KL of the selected subset as a label-free reward and the split rule as the policy, together with a budget head. At strict 5% this gives 88.6% (88.7–88.8% with the budget head), against 87.7% for the supervised scorer under the same rule and 88.8% for ETD. Variants with group Plackett–Luce sampling, token-level credit and expert iteration reached 86.1–87.4%, and GRPO with the greedy subset as the baseline of the advantage reached 88.5%. The best of 16 sampled subsets, chosen by teacher KL, reaches 96.6%: the reward is informative, but online RL closes about one point of the 8.9-point gap. Distilling the best-of-16 subsets offline into the scorer gives 87.7%, and 87.3% when the scorer also reads query/key features of the frozen LLM, so offline distillation does not close the gap either.

C.3. Vision-encoder variants

Table S4 compares encoders trained with the same recipe and the same 10× encoder learning rate. Relative to glance, with ETD at strict 5%, encoding everything at low resolution loses 8.4 points on V* and 3.2 on HR-Bench, and window attention loses 7.6 on HR-Bench, where it cuts the global context of 4K images. A second-pass window (glance25w) reaches a similar nine-benchmark average (89.9% *vs*. 89.5% for the same glance run) but loses 4.2 on V* and 3.3 on HR-Bench. Over three seeds the dense encoder is 1.4 points above glance at strict 5% (91.7±0.2 *vs*. 90.3±0.8) and equal at floor-32 (96.0 each); the gap comes from MMBench (75.2 *vs*. 72.8) and MME (2000 *vs*. 1932), not from the high-resolution benchmarks. MMBench images are all below the 4096-patch threshold and are encoded densely by both models, so this gap is a training effect on the shared encoder of both models.

The encoder variants are trained with the same recipe as the glance encoder and differ only in the encoder path for images of at least 4096 patches. *Low resolution* encodes the 2×2-pooled image only and copies every coarse token to its four native positions (glance with no second pass). *Window-4* keeps the dense input but restricts attention to non-overlapping windows of 8×8 merged tokens in all blocks except four global ones (5, 11, 17, 23), as in Qwen2.5-VL. *Glance25w* is the glance encoder whose second pass uses the same windowed attention. All variants use the auxiliary cosine loss (weight 0.1). On MMMU the three-seed means are 48.2 (glance), 48.7 (dense, 10× LR) and 48.9 (dense, 1× LR), so the encoder choice does not change the MMMU result here.

HAWK head weights on Qwen3.5. The released HAWK head weights belong to Qwen2.5-VL-7B. We recompute them for the first full-attention layer of Qwen3.5-4B with HAWK's procedure (Eq. 3 of [26]): each of the 16 heads in turn is blocked from all visual tokens on the full-token model, and the drop in gold-answer log-likelihood is measured on 300 questions from each of five LT-OPD-14K training subsets. The weights range from 0.042 to 0.123;

Table S3. LLaVA-1.5-7B in its standard evaluation setting, training-free, K visual tokens of 576. MME is perception, POPE is F1. Ret. is the mean ratio to the full-token scores. The ETD scorer is trained on LLaVA-1.5 features with the same data; ETD-free replaces it with question similarity ($c = 1$). [§]: numbers reported in Table 1 of CDPruner [22] (its runs of each method; its DivPrune rows match ours within 0.8 points per benchmark and 6 points of MME), retention relative to the full-token row of that table.

K	Method	GQA	MME-P	POPE	TextVQA	MMB	SQA-I	Ret.
576	Full	61.9	1507	85.9	58.3	64.3	69.6	100.0
192	DivPrune	60.0	1425	87.2	56.6	62.6	68.7	97.7
	ETD-free	58.9	1381	86.5	54.0	62.1	69.3	96.0
	ETD	60.1	1434	87.0	57.9	63.1	68.2	98.2
128	DivPrune	59.4	1399	86.9	55.9	62.3	68.6	96.9
	ETD-free	57.4	1357	85.6	52.8	59.5	69.0	94.1
	ETD	59.3	1410	86.1	57.3	62.1	68.9	97.3
	VisionZip[§]	57.6	1437	83.3	56.9	62.1	68.7	96.3
	DART[§]	57.9	1409	80.4	56.3	60.7	69.1	95.1
	CDPruner[§]	59.9	1431	87.7	56.2	63.1	69.0	97.9
64	DivPrune	57.8	1335	85.2	54.9	59.5	68.6	94.4
	ETD-free	54.7	1277	82.4	49.7	53.5	69.4	89.6
	ETD	57.4	1373	84.4	56.1	59.6	68.8	95.0
	VisionZip[§]	55.1	1365	77.0	55.5	60.1	69.0	92.8
	DART[§]	54.7	1365	73.8	54.7	59.5	69.3	91.8
	CDPruner[§]	58.6	1415	87.5	55.3	61.1	68.1	96.3
29	DivPrune	54.7	1280	80.8	53.0	58.1	68.7	91.2
	ETD-free	50.9	1199	76.2	48.3	44.9	69.0	83.7
	ETD	54.6	1327	79.9	54.6	57.0	68.9	91.7

head 13 is the most important in three of the five subsets. At strict 5% the recomputed weights and uniform weights give the same retention (56.7% both), so the HAWK row of Tab. 1 does not depend on this choice.

D. Selection Behavior

These measurements separate which tokens are kept from how many, on the training images and on held-out benchmarks.

D.1. Teacher-KL Selection Analysis

Which tokens. At a fixed K on 8,000 training questions, the total cost $\mathcal{L}$ of Eq. (4) under ETD is 0.50 of DivPrune's: 0.46 on text questions and 0.94 on general ones. The hard split (top-$K/2$ by p, the rest by diversity) reaches 0.64, and ETD lowers the KL on 59% of the questions. Choosing the exponent of Eq. (3) per question by an oracle gives 0.28, and the oracle exponent grows with t (Spearman $\rho = 0.24$; mean 0.68 and 0.82 in the two lower quartiles of t, 1.39 in the two upper ones). ETD follows this trend with βt and no fitted parameters. Learned temperature heads on t, the concentration of p, $\log N$ and other features did not improve over it on the dense model at the floor-32 budget (Tab. S5).

How many tokens. On training questions the per-image KL follows a power law in the kept fraction, $\mathcal{L}_m(K) \approx c_m(K/N_m)^{-\alpha}$ with $\alpha = 0.89$. Minimising $\sum_m \mathcal{L}_m(K_m)$ under $\sum_m K_m = B$ equalises the marginal gains (water-filling); with c_m independent of N_m this gives

$$K_m \propto N_m^{\gamma}, \qquad \gamma = \alpha/(\alpha+1) \approx 0.47. \tag{S1}$$

Fitting γ directly on training questions gives 0.45, and this single power law matches an 8-bin lookup table fitted on the same data (validation KL 0.85 *vs*. 0.83). Large images receive a smaller fraction because their KL is already low at 5%. The confidence t does not help predict the budget; answer length does, and a factor of two for open-ended prompts, also fitted on training prompts, completes the rule used in Tab. S12 on this model.

Calibration of the text confidence. We check the scorer's text head on data it was not trained on: TextVQA val (4,123 questions, all TextOCR word boxes) and the held-out split of the scorer data (8,685 Visual-CoT and TextOCR questions, 13 sources). Per token, $\sigma(s_i)$ against "the cell touches an OCR word" (3.5M tokens, base rate 0.18) has an expected calibration error of 0.013 and a Brier score of 0.048; in the 0.3–0.8 bins it is slightly under-confident (*e.g.* 0.45 predicted, 0.53 observed). Per question, t estimates the share of the evidence map that lies on

Table S4. Vision-encoder variants, all trained with the same LT-OPD recipe and 10× ViT learning rate except the first row. Selection at inference only.

Method	V*	HR-4K	GQA	MMMU	MMB	MME	POPE	TextVQA	OCRB	Ret. (%)
DivPrune, strict 5%										
Dense ViT	73.8	76.2	52.4	50.1	71.0	1860	85.3	64.9	384	85.4
Dense ViT, 10× LR	74.9	77.0	54.0	47.9	73.9	1975	86.3	70.5	481	88.6
Low-res (1/4 patches)	70.2	75.1	53.5	46.9	70.4	1942	85.3	70.5	504	86.9
Glance 25% (ours)	73.3	77.4	53.6	48.1	72.6	1883	84.6	70.3	475	87.5
Glance 25% + window	70.2	72.5	53.7	50.4	73.2	1955	85.7	70.3	501	87.8
Window-4	75.9	68.3	53.6	49.4	71.1	1934	86.0	69.9	480	87.1
ETD, strict 5%										
Dense ViT	78.0	77.6	53.2	48.7	71.8	1842	87.7	73.3	486	88.8
Dense ViT, 10× LR	78.5	77.8	55.0	47.7	75.6	2014	87.9	75.8	572	91.9
Low-res (1/4 patches)	67.5	72.7	54.8	46.6	71.4	1933	87.4	75.3	571	88.3
Glance 25% (ours)	75.9	75.9	54.4	46.1	72.3	1905	87.3	75.4	552	89.5
Glance 25% + window	71.7	72.6	55.3	50.4	73.2	1955	87.0	75.6	551	89.9
Window-4	78.0	68.3	54.6	48.2	72.6	1911	87.1	74.7	566	89.3
ETD, $\max(32, \lceil 0.05N \rceil)$										
Dense ViT	78.5	78.1	57.7	51.8	80.8	2090	89.2	73.2	666	95.5
Dense ViT, 10× LR	78.5	78.4	58.2	49.1	80.2	2149	88.8	75.9	707	96.2
Low-res (1/4 patches)	68.1	73.0	57.9	49.1	79.8	2145	89.0	75.5	725	94.0
Glance 25% (ours)	77.0	76.1	58.2	50.0	80.6	2113	88.6	75.5	708	95.6
Glance 25% + window	71.2	72.4	58.3	52.6	80.9	2158	89.2	75.6	695	95.1
Window-4	78.0	67.8	58.3	50.8	80.3	2124	88.9	74.8	704	94.7

OCR cells, and on the 4,740 questions with word boxes it matches the measured share with a mean absolute error of 0.05 (Pearson 0.91). As a classifier of text-evidence questions t reaches an AUROC of 0.995 on the held-out sources (median 0.87–0.96 on the text sources, 0.08–0.13 on GQA, Flickr30k, OpenImages, Visual7W and VSR), with question-level ECE 0.11: it is bimodal, and the few questions with t between 0.4 and 0.6 are text questions 90% of the time. t does not depend on how much text the image contains (median 0.85–0.89 from 1–5 to over 50 OCR words on TextVQA). The scorer is therefore a well-calibrated text detector; the ablations in Sec. 4.4 show that this calibration is not what drives the retention gain.

D.2. Held-out benchmarks

The eight held-out benchmarks are in Tab. S8, with paired-bootstrap intervals for ETD minus DivPrune in Tab. S9. Documents suffer most at 5%: with DivPrune the LT-OPD model drops from 94.6 to 52.5 on DocVQA and from 79.4 to 30.2 on InfoVQA; ETD raises them to 74.5 and 40.8 (95% intervals of the gain $[20.7, 23.3]$ and $[8.8, 12.4]$), and the glance model to 76.4 and 43.4. On natural-image and multi-image benchmarks the median text confidence is 0.04–0.24 and ETD is within noise of DivPrune in 18 of 20 comparisons; the exception is RealWorldQA with the glance encoder (-2.5, $[-4.7, -0.3]$). The glance encoder loses about 3 points on HR-Bench 8K: Stage 1 does not read the question, and detail it does not re-encode cannot be recovered by Stage 2: diversity re-encodes the answer cells of 42% of glanced TextVQA answers, close to random cells (26%). Running ETD on the coarse tokens instead (80%), at inference and the same FLOPs, raises DocVQA from 76.4 to 80.9 and V* from 75.9 to 78.5 (Sec. D.4). Over three seeds, choosing the cells with the scorer raises the held-out mean by 1.9–2.4 points at the same time to first token (Tab. S7).

The document gap is not a budget effect: under the floor-32 rule DivPrune stays at 53.1 / 30.9 (LT-OPD) and 66.7 / 37.5 (glance) on DocVQA / InfoVQA, while the floor mainly helps MuirBench (about +5 for both selectors), whose multi-image inputs contain small images. The 5% models exceed the full-token model on HR-Bench 8K. Table S9 tests ETD against DivPrune question by question: the document gains hold for both models and both budget rules, and on benchmarks where the scorer's text confidence is low the two rules are mostly indistinguishable.

D.3. Training-image overlap

The scorer's training images (TextVQA train and Visual-CoT, 167K images) were matched against the images of 17 evaluation sets with 64-bit difference and perceptual hashes (a pair matches when the dHash distance is at most 6 and the pHash distance at most 10). Evaluation images with at least one match: POPE 2,376 / 9,000 (COCO images),

Table S5. Selection rule ablation on the LT-OPD + DivPrune model, identical K per image. Top: strict 5%. Bottom: $\max(32, \lceil 0.05N \rceil)$.

Method	V*	HR-4K	GQA	MMMU	MMB	MME	POPE	TextVQA	OCRB	Ret. (%)
Strict 5%										
DivPrune ($\beta t = 0$)	73.8	76.2	52.4	50.1	71.0	1860	85.3	64.9	384	85.4
Split: top-$K/2$ by p + DivPrune	78.0	75.9	52.4	50.0	70.9	1844	85.8	71.6	455	87.7
Split, scorer fine-tuned with GRPO	75.9	77.6	52.5	50.3	72.4	1887	86.1	71.7	481	88.6
ETD $d_i(Np_i)^t$	78.0	77.6	53.2	48.7	71.8	1842	87.7	73.3	486	**88.8**
$\max(32, \lceil 0.05N \rceil)$										
DivPrune ($\beta t = 0$)	73.8	76.9	57.5	51.1	80.8	2024	89.5	65.4	589	92.2
Split: top-$K/2$ by p + DivPrune	78.0	75.9	57.1	51.3	80.4	2034	88.8	71.8	652	94.2
Gate: split only if $t > 0.5$	74.3	76.4	57.4	51.7	80.9	2064	89.6	71.5	653	94.2
Soft tilt without t: $d_i(Np_i)$	78.0	77.4	55.5	51.0	79.6	2072	88.8	73.3	662	94.4
Constant temperature $\tau = 0.25$	78.0	77.4	57.2	51.9	81.4	2078	89.3	74.2	679	95.6
Teacher-calibrated temperature head	78.5	77.1	57.2	51.9	81.6	2077	89.2	74.0	680	95.7
ETD, $\beta = 0.5$	77.0	77.2	57.5	51.8	80.5	2087	88.8	73.7	677	95.3
ETD, $\beta = 2$	79.6	78.0	57.3	50.6	80.2	2082	89.3	72.8	648	94.9
ETD, $\beta = 1$	78.5	78.1	57.7	51.8	80.8	2090	89.2	73.2	666	**95.5**

Floor-32 rows other than ETD with $\beta = 1$ were evaluated in an earlier run of the same protocol.

DocVQA 491 / 5,349, MMBench 187 / 4,329, TextVQA 70 / 3,166, MME 50 / 2,374, OCRBench 35 / 1,000 and GQA 1 / 398. The deduplicated scorer is trained identically on data from which every training image matching an evaluation image is removed (851 images, 1,720 of 422K records, mostly Visual-CoT GQA, Visual7W and DocVQA; the removal rule also adds a pixel-correlation and a 256-bit dHash test). With the deduplicated scorer, ETD at strict 5% retains 83.9% on the untrained MLLM (84.2% with the original scorer) and $90.1 \pm 0.5\%$ on the glance model over three seeds ($90.3 \pm 0.8\%$). On the five held-out benchmarks it retains 79.1% (79.3%) on the untrained MLLM and 83.2% (82.8%) on the glance model (seed 0). The overlap does not inflate the reported results on these sets.

D.4. Stage-1 evidence coverage

Table S6 replaces the coarse-cell choice of the trained glance model (seed 0, ETD at strict 5%) at inference only; the zoom ratio, second pass and token budget are unchanged, so the encoder FLOPs are those of the trained glance. *Query-aware* runs ETD with the scorer on the coarse tokens; *oracle* first takes every coarse cell that touches an answer box. R_{fine} is the share of answer-box cells that are re-encoded at native resolution. Only 413 of the 5,000 TextVQA images exceed the 4096-patch threshold, so the TextVQA score barely tests Stage 1. Over the nine benchmarks 6.1% of the images (60% of the visual tokens) take the glance path: all of V* and HR-Bench 4K, 25% of MME, 15% of OCRBench, 10% of MMMU, 8% of TextVQA, and none of GQA, MMBench or POPE; the second block applies the glance to every image. Under diversity, questions whose evidence is fully re-encoded score 78.7 and partially covered ones 64.3. Diversity covers text evidence about as well as random cells, and on documents it is below random, which is the main weakness of a question-agnostic Stage 1 on documents.

Table S6. Stage-1 cell choice at inference on the glance model (seed 0, ETD, strict 5%). Top: TextVQA questions whose image takes the glance path and has answer boxes (345). Middle: glance forced on all images (4,123 with boxes). Bottom: accuracy on held-out and high-resolution benchmarks.

Stage-1 cells	diversity	random	Q-aware	oracle
R_{fine} (%)	42	26	80	99
TextVQA acc.	72.3	74.1	77.7	78.1
R_{fine}, forced (%)	42	25	79	99
TextVQA acc., forced	65.3	66.3	73.9	78.3
DocVQA	76.4	79.4	80.9	–
InfoVQA	43.4	45.9	44.3	–
HR-Bench 8K	72.1	67.1	71.6	–
V*	75.9	72.8	78.5	–
HR-Bench 4K	75.9	73.6	76.9	–

Stage-1 rules over three seeds. Table S7 repeats the comparison on all three training seeds of the glance model and adds two rules that use the scorer without diversity: *top-p* takes the coarse cells with the highest relevance, and *half top-p* takes half of them by relevance and the rest by max–min diversity; *query-aware* is ETD on the coarse tokens with a constant exponent of 0.25. Only Stage 1 changes at inference, with the trained weights, zoom ratio and budget; Stage 2 is ETD with the constant exponent 0.25 in every row, and the diversity row runs through the same code path as the others. Every rule that reads the scorer

raises the seven held-out benchmarks without HR-Bench 8K from 53.1 to 55.0–55.5, and the gain is again on documents (DocVQA 76.8 to 81.1–82.4, InfoVQA 43.9 to 46.4–48.2) and on MME-RealWorld and RealWorldQA; the seed-to-seed spread of this mean is at most 0.6. Random cells raise DocVQA too but lose 5 points on HR-Bench 8K, so coverage alone does not explain the gain. The scorer adds about 2 ms per image, and the time to first token on HR-Bench 4K stays at 1.28–1.42 s against 1.39 s for diversity (full tokens: 4.15 s). With the cells chosen query-aware, the glance model's DocVQA (81.1) exceeds the dense-encoder model with ETD (80.1, three seeds), so the document gap of the glance encoder in Tab. S8 comes from the cell choice and not from encoding a quarter of the image.

Zoom ratio and threshold. On seed 0, re-encoding 12.5%, 25% or 50% of the cells gives a held-out mean of 53.5 / 53.2 / 54.9 with diversity and 53.8 / 55.2 / 55.8 with query-aware cells, with a time to first token on HR-Bench 4K of about 1.2–1.3 s at 12.5% and 2.0 s at 50%. Query-aware cells at 25% thus match diversity at 50% for less than three quarters of its encoder time. Applying the glance to every image instead of those above 4096 patches lowers the nine-benchmark retention of the query-aware rule from 91.4% to 89.6% (TextVQA 71.3), so small images stay on the dense path.

E. Budgets and Diagnostics

The token budget is varied around the LT-OPD operating point, and the evidence checks include Gemma 4.

E.1. The LT-OPD operating point

The LT-OPD release sets $K = \max(32, \lceil 0.05N \rceil)$. Small images hit the floor of 32 tokens, so the nine benchmarks together keep 7.1% of the visual tokens rather than 5.0%. Table S10 evaluates each method at this budget. LT-OPD retains 86.1%; training with DivPrune on the same data reaches 92.2%. Glance-ETD retains $96.0 \pm 0.4\%$ over three seeds, 9.9 points above LT-OPD with the same number of tokens, and the dense-encoder variant reaches the same $96.0 \pm 0.2\%$. The released LT-OPD checkpoint, which keeps the learnable token merge, retains 91.1% (Sec. B.4).

E.2. Sweep across Token Budgets

Table S11 varies the strict budget from 1% to 25%. On the untrained model ETD is ahead of DivPrune at every measured budget (+9.5 at 1%, +8.4 at 5%, +7.1 at 10%, +3.7 at 25%) and ahead of DART by a wider margin. CDPruner, PruneSID and ZOO-Prune also stay below DivPrune at 10% (74.3–80.5) and 25% (84.4–90.3). ETD at 10% (90.8%) is within 1.2 points of DivPrune at 25% (92.0%) with 2.5× fewer tokens. After training, ETD at 3.5% matches DivPrune at 5%. On the glance model the gap falls from 6.8 points at 1% to 2.0 at 5% and below one point from 7.5% on: once a quarter of the tokens is kept, diversity alone already covers most evidence on text questions.

E.3. Allocating the budget across images

Table S12 applies Eq. (S1) on top of ETD on the dense LT-OPD model. With the same total number of tokens ($\sum K / \sum N$ of 5.00% and 4.98%), giving large images a smaller fraction and small images a larger one raises strict-5% retention by 4.9 points, and by 6.0 with the open-prompt factor. The gain comes from the small-image benchmarks (OCRBench 486→731, MMBench 71.8→79.5, MME 1842→2021), while V*, whose images are the largest, loses 5.2 points. At the floor-32 budget the gain is smaller (+1.5), because the floor already gives small images more than 5%. In an offline sweep of γ composed from the fixed-ratio runs, the training value 0.45 is also the best evaluation value. On Glance-ETD (seed 0) the same allocation raises strict-5% retention from 89.5 to 94.7, mostly on OCRBench (552→684) and GQA (54.4→57.9); on training-free ETD with Gemma 4 it is neutral (85.1 *vs*. 85.2).

E.4. Evidence diagnostics

Table S13 tests whether ETD's gain comes from the evidence the question asks for. Scoring tokens with a different question lowers TextVQA from 73.3 to 66.4 and OCRBench from 486 to 406 on the LT-OPD model, and from 75.4 to 71.3 and from 552 to 464 on the glance model, so the scorer uses the question rather than generic saliency. Removing the tokens ETD would have chosen and selecting from the rest is worse than DivPrune (TextVQA 58.3 *vs*. 64.9), so ETD's tokens carry information that the rest of the image does not replace. With 0.1% of the tokens the models fall to 52–54%, close to answering from the question alone. For answer-box recall, the untrained model gives the same numbers for ETD (99.2%, 63.2%), since the scorer does not depend on MLLM training at all.

E.5. Gemma 4 across budgets

Figure S1 extends the Gemma comparison past strict 5%. ETD retains 91.1% at 10%, against 82.5% for DivPrune, and 97.5% at 25% on the eight benchmarks scored so far (DivPrune 90.7% on the same eight; MMBench is still pending). At 5%, TextVQA rises from 38.9 to 61.0 and OCRBench from 184 to 440, against a full-token OCRBench of 764, and HR-Bench 4K rises from 53.0 to 60.2. GQA, MMMU and POPE stay within 1.7 points of DivPrune. ETD at 10% is above DivPrune at 25% on V*, HR-Bench, TextVQA and OCRBench. On Gemma, Glance-ETD uses the two-pass glance with Gemma's own vision tower: a quarter-resolution pass gives coarse features for every cell, and only the cells chosen by max–min diversity are re-encoded at native resolution before ETD se-

Table S7. Stage-1 cell rules at inference on the glance model at strict 5% (mean of three training seeds; Stage 2 is ETD with a constant exponent of 0.25 in every row). R_{fine}: share of answer-box cells of the 345 glanced TextVQA questions that are re-encoded at native resolution (seed 0). H7: mean of the seven held-out benchmarks without HR-Bench 8K. TTFT: HR-Bench 4K, batch 1, one A800 (full tokens 4.15 s). HR-Bench 8K of diversity, query-aware and top-p averages seeds 0 and 2. –: not measured.

Stage-1 cells	R_{fine}	HR-8K	DocVQA	InfoVQA	MME-RW	HRScene	RWQA	Muir	MMSI	H7	TTFT (s)
Diversity (trained)	42	71.8	76.8	43.9	45.2	55.1	64.9	55.0	30.8	53.1	1.39
Random	26	66.6	79.6	46.5	42.9	54.1	64.4	55.5	30.2	53.3	–
Query-aware	73	72.3	81.1	46.4	47.3	56.9	67.1	55.5	30.5	55.0	1.42
Half top-p	78	73.5	81.4	47.7	47.6	55.8	67.5	55.7	30.3	55.1	–
Top-p	88	75.0	82.5	48.2	47.8	55.6	68.5	55.6	30.2	55.5	1.28

Table S8. High-resolution, document and multi-image benchmarks not used for any design decision. Full-token row: untrained Qwen3.5-4B with all visual tokens; MuirBench and MMSI use a 256-token answer cap (the official 16-token cap truncates its answers before the option letter).

Method	HR-8K	DocVQA	InfoVQA	MME-RW	HRScene	RWQA	Muir	MMSI
Full tokens	72.6	94.6	79.4	48.9	62.5	73.7	62.5	22.7
LT-OPD, DivPrune 5%	75.2	52.5	30.2	44.0	55.3	65.6	54.7	30.0
LT-OPD, ETD 5%	75.0	74.5	40.8	45.9	54.6	66.7	55.3	30.3
LT-OPD, ETD floor-32	74.5	74.7	41.0	45.9	54.7	66.9	60.0	30.9
LT-OPD, DivPrune floor-32	75.0	53.1	30.9	43.8	55.9	66.9	60.3	30.2
Glance, DivPrune 5%	72.4	66.5	37.3	44.5	53.3	67.3	55.5	31.1
Glance, ETD 5%	72.1	76.4	43.4	45.5	53.9	64.8	55.0	30.5
Glance, DivPrune floor-32	71.9	66.7	37.5	44.3	54.1	66.5	60.2	31.3
Glance, ETD floor-32	72.0	76.5	43.8	45.7	54.1	64.2	60.0	30.7

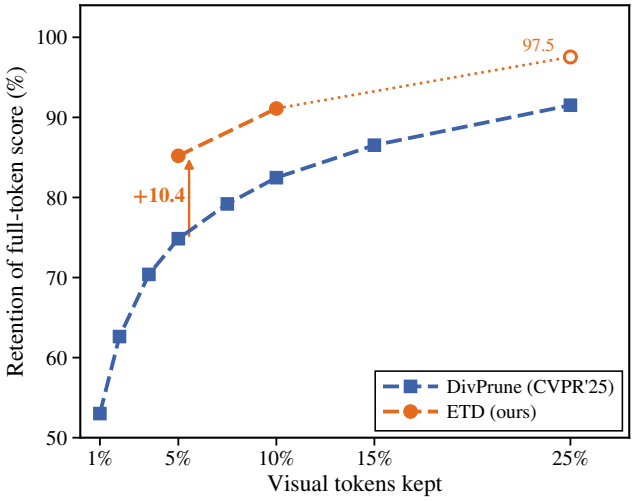


Figure S1. Retention of Gemma 4 E4B-it against its own full-token scores. ETD and DivPrune pass the same number of visual tokens per image. Filled markers average all nine benchmarks. The open marker at 25% omits MMBench; on those eight benchmarks DivPrune retains 90.7% on Gemma.

lects K tokens. It runs on the self-distilled checkpoints, which are trained without the glance, so unlike Qwen the encoder is not adapted to the glance features. Over three seeds it retains $89.5 \pm 1.1\%$ at strict 5%, 0.6 below ETD with the dense encoder ($90.1 \pm 0.9\%$) and 2.3 above DivPrune ($87.2 \pm 1.2\%$); with the constant exponent 0.25 it retains $89.7 \pm 1.2\%$. Unlike Qwen, where V* is unchanged, the untrained glance costs V* and TextVQA (48.9→45.9, 64.1→60.8) and gains OCRBench and MME (513→553 and 1471→1528 on those two).

F. Latency and Memory

Accuracy against cost. Matched to the benchmark that is timed (Fig. S2a,b), Glance-ETD keeps 75.9 on HR-Bench 4K and V* at 2.9× and 1.4× TTFT, while the low-resolution and windowed encoders give speed for detail (HR-Bench 4K 72.7, 72.6 and 68.3). Along the budget axis (Fig. S2c) a dense-encoder model moves by at most 30 of its 370–400 TFLOPs between 1% and 25%, because the encoder dominates the cost; the glance encoder moves the whole curve to 60–90 TFLOPs, and Glance-ETD at 7.5% (92.3%) uses a fifth of the FLOPs of the dense model at 5%.

Where the speed-up appears. On the twelve benchmarks the TTFT speed-up grows with the number of visual tokens, as the encoder takes a larger share of the cost: 1.5× on V* and DocVQA and 2.5–2.9× on HR-Bench (Fig. S2j).

Table S9. ETD minus DivPrune on the held-out benchmarks, same model and budget, with paired-bootstrap 95% confidence intervals over questions (10,000 resamples). Bold: interval excludes zero. HR-Bench 8K uses per-question accuracy. Last row: median text-evidence confidence t of the scorer (LT-OPD, 5%), on the untrained Qwen model.

Model, budget	DocVQA	InfoVQA	MME-RW	HR-8K	HRScene	RWQA	Muir	MMSI
LT-OPD 5%	**+22.0** [20.7, 23.3]	**+10.6** [8.8, 12.4]	**+1.8** [0.2, 3.5]	-0.2 [−2.0, 1.6]	-0.7 [−2.7, 1.3]	+1.0 [−1.7, 3.8]	+0.7 [−0.7, 2.0]	+0.3 [−1.3, 1.9]
LT-OPD floor-32	**+21.6** [20.3, 22.9]	**+10.0** [8.2, 11.8]	**+2.0** [0.5, 3.7]	-0.5 [−2.4, 1.5]	-1.2 [−3.3, 0.8]	+0.0 [−2.6, 2.6]	-0.3 [−1.5, 1.0]	+0.7 [−0.9, 2.3]
Glance 5%	**+9.9** [8.7, 11.0]	**+6.1** [4.4, 7.7]	+1.0 [−0.3, 2.4]	-0.2 [−1.5, 1.0]	+0.6 [−1.0, 2.3]	**-2.5** [−4.7, −0.3]	-0.5 [−1.8, 0.7]	-0.6 [−2.2, 1.0]
Glance floor-32	**+9.8** [8.7, 10.9]	**+6.3** [4.6, 8.0]	+1.4 [0.0, 2.7]	+0.1 [−1.1, 1.4]	+0.0 [−1.6, 1.7]	**-2.4** [−4.6, −0.1]	-0.1 [−1.1, 0.8]	-0.6 [−2.1, 0.9]
median t	0.94	0.94	0.31	0.19	0.24	0.20	0.15	0.04

Table S10. Operating point of the LT-OPD release, $K = \max(32, \lceil 0.05N \rceil)$ per image ($\sum K / \sum N = 7.1\%$). Rows above the line are baselines and LT-OPD-recipe models with DivPrune; rows below are ours.

Method	V*	HR-4K	GQA	MMMU	MMB	MME	POPE	TextVQA	OCRB	Ret. (%)
Full tokens	80.6	78.0	60.0	51.3	79.0	2218	89.9	80.5	848	100.0
DivPrune [1], training-free	70.7	73.9	53.4	48.8	75.5	1925	86.6	49.3	431	84.1
CDPruner [22], training-free	53.4	62.4	40.8	48.2	74.2	1862	81.9	43.5	398	75.4
LT-OPD [9] (arXiv'26.09)	67.0	69.6	51.7	49.4	77.8	2073	85.9	56.1	533	86.1
LT-OPD + DivPrune (ours, reproduced)	73.8	76.9	57.5	51.1	80.8	2024	89.5	65.4	589	92.2
Ours and its dense-encoder variants										
Glance-ETD (ours)	77.3	76.7	58.2	51.3	80.3	2132	88.8	75.3	704	**96.0**±0.4
ours, dense-encoder variant (no glance)	76.8	76.5	58.2	49.9	80.3	2168	88.9	76.1	708	96.0±0.2

The full-token row is read from the strict no-thinking scores in results/full; all retention denominators use those same scores.

Table S11. Retention (%) versus strict budget $\sum K / \sum N$.

	1%	2%	3.5%	5%	7.5%	10%	15%	25%
Untrained								
ETD	65.5	74.7	80.7	84.2	87.8	90.8	92.3	95.8
DivPrune	56.0	64.6	71.5	75.8	80.9	83.7	87.6	92.0
DART	–	–	–	67.7	–	77.2	–	91.0
CDPruner	–	–	–	67.6	–	74.3	–	84.4
PruneSID	–	–	–	68.7	–	80.5	–	86.0
ZOO-Prune	–	–	–	70.9	–	79.3	–	90.3
LT-OPD								
ETD	73.6	80.7	85.9	88.8	92.3	94.5	96.1	98.2
Glance								
ETD	74.1	82.0	87.1	89.5	92.3	94.4	96.3	97.7
DivPrune	67.3	76.6	84.1	87.5	91.8	93.7	95.4	98.2

Table S12. Global budget allocation with ETD on the LT-OPD model. γ and the open-prompt factor are fitted on training questions only; the scale is set to match the reference $\sum K / \sum N$.

Allocation	strict 5%	floor-32 budget (7.1%)
Same ratio for every image	88.8	95.5
$K \propto N^{0.45}$	93.8	95.8
$K \propto N^{0.45} \cdot 2^{[\text{open}]}$	**94.8**	**97.0**

End-to-end speed-ups are smaller because decoding takes the same 59 ms per token with or without pruning at batch 1 (Fig. S2e,k, same decode stage).

FLOPs. We count all matrix multiplications of one TTFT call, including the attention products, from the checkpoint shapes and the measured token counts of each question: ViT blocks $2(4d^2 + 2dd_{\text{ff}})$ per patch plus $4N^2d$ for global attention (d=1024, d_{ff}=4096, 24 blocks), the merger, and the LLM as two FLOPs per non-embedding parameter (3.57B) per token plus causal attention in its eight full-attention layers; the Gated-DeltaNet recurrence and the selector are below 0.5%. On HR-Bench 4K global attention is 332 of the 366 ViT TFLOPs (Fig. S2d). Counting only the linear layers gives 138, 41 and 24 TFLOPs for full tokens, dense + ETD and Glance-ETD, which hides the encoder's quadratic cost. Measured speed-ups stay well below the FLOPs ratio (2.51× against 7.5× at 8K, Fig. S2f): the LLM prefill alone is 9.8× faster, but the ViT only 4.0×, and host-side preprocessing does not shrink (Fig. S2e).

Table S13. Does ETD keep the evidence? Strict 5%. *Wrong question*: ETD scores tokens with the question of another sample, the model then answers the true question. *Complement*: ETD-selected tokens are removed from the pool before selecting. *Blind*: 0.1% of tokens.

Method	V*	HR-4K	GQA	MMMU	MMB	MME	POPE	TextVQA	OCRB	Ret. (%)
LT-OPD + DivPrune model										
DivPrune	73.8	76.2	52.4	50.1	71.0	1860	85.3	64.9	384	85.4
ETD	78.0	77.6	53.2	48.7	71.8	1842	87.7	73.3	486	88.8
ETD, wrong question	76.4	77.4	52.1	48.9	72.3	1846	86.8	66.4	406	86.4
ETD complement	75.4	75.6	51.8	49.4	72.3	1883	85.2	58.3	381	84.6
Blind (0.1%)	49.2	55.9	35.6	45.7	45.5	1338	61.6	10.8	54	54.1
Glance model										
DivPrune	73.3	77.4	53.6	48.1	72.6	1883	84.6	70.3	475	87.5
ETD	75.9	75.9	54.4	46.1	72.3	1905	87.3	75.4	552	89.5
ETD, wrong question	74.9	75.8	53.2	47.6	72.1	1883	86.1	71.3	464	87.4
ETD complement	74.3	75.9	51.9	48.4	72.6	1847	86.6	62.7	427	85.6
Blind (0.1%)	46.1	58.8	33.6	45.2	41.6	1273	56.3	10.5	50	52.0

Table S14. Efficiency on one A800-SXM4-80GB (bf16, median over 30 questions, configurations run one after another). HR-4K unless noted; batch 1 unless noted. Times are milliseconds. TFLOPs are ViT + LLM, attention included. Speed-ups over full tokens are in parentheses. Peak memory is batch 1 / batch 8.

	Full	Dense + ETD	Glance-ETD
TTFT	3955	3305 (1.20×)	1376 (**2.87**×)
V* TTFT	547	476 (1.15×)	379 (1.44×)
End-to-end	4246	3609 (1.18×)	1683 (2.52×)
TFLOPs	366 + 116	366 + 6	59 + 6
Batch-8 q/s	0.277	0.342 (1.23×)	0.815 (2.94×)
Mem. b1/b8 (GB)	13.3 / 34.7	13.7 / 35.1	12.7 / **17.7**

Memory. Weights take 9.1 GB. At batch 1 the remaining peak is set by ViT activations, so selection after the encoder does not lower it (selection itself adds about 0.4 GB on HR-Bench 4K). Per request, the decode cache is 32 KiB of KV per kept token in the eight full-attention layers plus a constant 49 MiB Gated-DeltaNet state (24 layers, fp32 state assumed); at 5% the KV part falls from 448 to 25 MiB on HR-Bench 4K, so the hybrid architecture limits the cache saving to 6.7× (Fig. S2l). Peak memory at batch 8 grows with the image for full tokens and the dense encoder, to 35 GB at 8K, and stays below 18 GB with the glance encoder (Fig. S2h).

Table S14 collects the numbers behind Fig. 4 on HR-Bench 4K and V*. Table S15 times every Tab. 1 method on Qwen3.5-4B in a separate run with the same protocol. All of them prune after the dense encoder, so the ViT bounds their speed-up; methods with a heavy selection step (CDPruner, PruneSID, ZOO-Prune and LT-OPD, whose DPP takes 0.6 s at 16k tokens) or extra ViT work (VisionTrim) fall to 0.87–1.09× against full tokens.

Table S15. Latency of the Tab. 1 methods on Qwen3.5-4B at strict 5% (one A800, bf16, batch 1, median over 30 questions, configurations run one after another; speed-ups over full tokens of the same run). Gain is the speed-up of TTFT over full tokens. TTFT and Sel. (selection) in ms, Mem. is HR-Bench 4K peak memory in GB. Sel. is the selection time on HR-Bench 4K (our implementations; DivPrune and ETD use a CUDA-graph greedy). VisionTrim recomputes a ViT attention map, which the fused attention kernel does not return (1.5 s of extra ViT time); ZOO-Prune spends 1.35 s on 64 zeroth-order probes of the projector. Ret. from Tab. 1.

		HR-Bench 4K			V*	
Method	Ret.	TTFT	Gain	Sel.	Gain	Mem.
Full tokens	100.0	4099	–	0	–	13.0
DivPrune	75.8	3430	1.19×	141	1.18×	13.2
DART	67.7	3469	1.18×	266	1.14×	13.1
CDPruner	67.6	3776	1.09×	592	1.05×	13.0
PruneSID	68.7	3948	1.04×	769	1.09×	13.3
VisionTrim	61.4	4696	0.87×	12	1.09×	14.7
HAWK	56.9	3190	1.28×	14	1.24×	13.2
ZOO-Prune	70.9	4576	0.90×	1352	0.76×	13.2
LLaVA-Mini	70.3	3267	1.25×	12	1.24×	13.1
LearnPruner	63.6	3290	1.25×	39	1.17×	13.1
MetaCompress	71.3	3228	1.27×	15	1.23×	13.1
EPIC	65.1	3471	1.18×	263	1.16×	13.1
LT-OPD	79.0	3791	1.08×	581	1.04×	13.1
ETD (ours)	84.2	3362	1.22×	148	1.24×	13.2
Glance-ETD (ours)	90.3	1413	2.90×	162	1.48×	12.2

Protocol. All latency numbers come from one run per image size on a single A800-SXM4-80GB (driver 550.54.15, PyTorch 2.10 with CUDA 12.6, FlashAttention, bf16), with the five configurations (full tokens, dense encoder with DivPrune or ETD, glance encoder with DivPrune or ETD) run one after another on the same otherwise idle GPU. Each configuration sees the same questions (30 per benchmark at native size, 24 HR-Bench 8K questions per resized size)

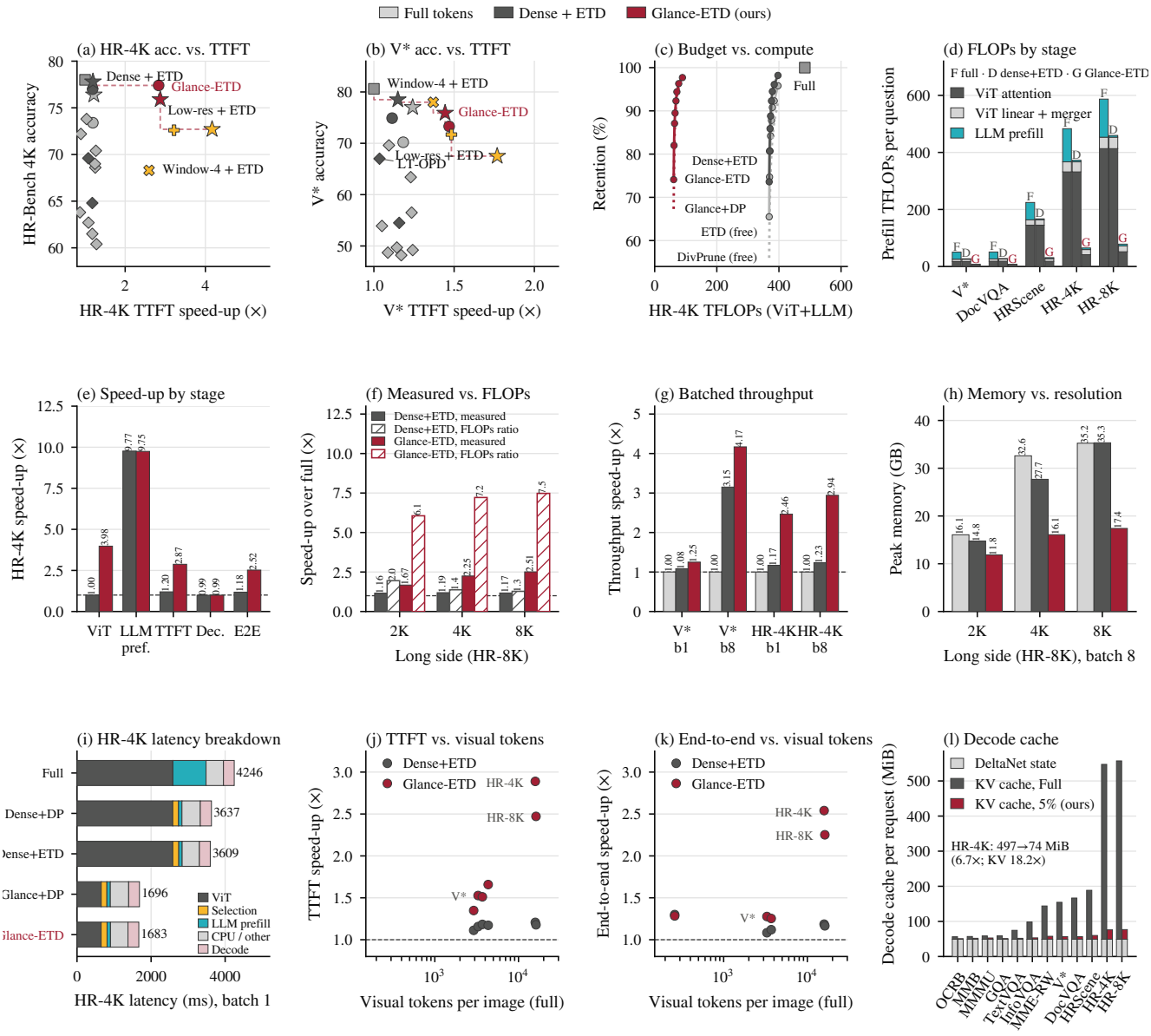


Figure S2. Further efficiency views (strict 5%, Qwen3.5-4B). (a)(b) Task-matched trade-off: HR-Bench 4K and V* accuracy against TTFT speed-up on the same benchmark (markers as in Fig. 4a). (c) Retention against HR-Bench 4K prefill TFLOPs as the budget goes from 1% to 25% (Tab. S11). (d) Prefill FLOPs per question by stage (F full, D dense + ETD, G Glance-ETD). (e) HR-Bench 4K speed-up of each stage (ViT, LLM prefill, TTFT, decoding, end to end). (f) Measured TTFT speed-up against the FLOPs ratio (hatched) on HR-Bench 8K resized to 2K–8K. (g) Throughput speed-up at batch 1 and 8 (Tab. S17). (h) Peak memory at batch 8 on resized HR-Bench 8K. (i) HR-Bench 4K latency by stage at batch 1 for all five configurations (end-to-end ms at the bar ends). (j)(k) TTFT and end-to-end speed-up against the mean number of visual tokens, on the benchmarks where Glance-ETD exceeds 1.15×. (l) Decode cache per request: KV of the eight full-attention layers plus the constant Gated-DeltaNet state. (a)–(c) and (e)–(i) use the run of Tab. S14 for full tokens and the dense and glance encoders; (d) and (j)–(l) use a 4-GPU run that covers all 12 benchmarks, with speed-ups taken against its own full-token run.

after three warm-up questions. TTFT is split by CUDA-synchronised hooks into ViT, scorer, selection, LLM prefill and the remainder (CPU preprocessing and other host work); FLOPs are counted analytically as described below. A second run of the 4K setting on another GPU reproduces every median within 2%. Full-token p95 on HR-Bench 4K is inflated by two questions whose prefill takes about 15 s, which we attribute to kernel compilation for unseen sequence lengths; the medians are unaffected.

Table S16 varies the image size. The speed-up grows

Table S16. Image size. HR-Bench 8K images resized to a long side of 2K–8K pixels (24 questions, batch 1). Glance-ETD TTFT is split into ViT, scorer plus selection (Sel.), LLM prefill and CPU; speed-up is full / Glance-ETD at the median. Memory is peak allocated GPU memory of that configuration on the single A800 used for the timing run. Times are in ms.

Long side	2K	4K	8K
Visual tokens N	2,688	10,752	16,224
Full TTFT p50/p95	385 / 532	2283 / –[‡]	4511 / 4719
Dense + ETD TTFT	331	1913	3845
Glance-ETD p50/p95	230 / 384	1015 / 1801	1797 / 1855
ViT/Sel./LLM/CPU	71/20/60/78	402/127/91/392	678/161/91/865
Speed-up	1.67×	2.25×	**2.51**×
Mem. full/glance (GB)	10.8 / 10.9	12.4 / 12.2	13.4 / 12.3

[‡] 15.9 s, from the compilation outliers described in the text.

Table S17. Throughput (questions/s, generation up to the official answer cap or 128 tokens) at batch 1, 4 and 8, and peak memory at batch 8; same run as Tab. S14. Speed-ups relative to full tokens in parentheses next to each throughput.

	Batch	Full	Dense + ETD	Glance-ETD
HR-4K	1	0.265	0.310 (1.17×)	0.653 (2.46×)
	4	0.280	0.343 (1.23×)	0.799 (2.85×)
	8	0.277	0.342 (1.23×)	0.815 (**2.94**×)
	mem. (GB)	34.7	35.1	17.7
V*	1	1.169	1.264 (1.08×)	1.462 (1.25×)
	4	0.910	2.229 (2.45×)	2.873 (3.16×)
	8	0.826	2.602 (3.15×)	3.444 (**4.17**×)
	mem. (GB)	17.5	16.0	12.5

with resolution, from 1.67× at 2K to 2.51× at 8K, as the encoder saving dominates; at 8K CPU preprocessing (865 ms) becomes the largest remaining cost. The scorer itself takes 2–3 ms at every size; most of the selection time is the greedy loop. Table S17 shows that batching does not help the full-token model at these sizes (its throughput stays flat or drops), while the pruned models gain from batching and Glance-ETD halves the batch-8 memory on HR-Bench 4K. These measurements replace an earlier profiling run (3918, 3272 and 1230 ms on HR-Bench 4K), which ran four configurations on four GPUs of one node at the same time.