RESIZEBENCH: REVEALING THE LIMITS OF RESIZING FOR VISUAL TOKEN COMPRESSION

Anonymous authors
Paper under double-blind review

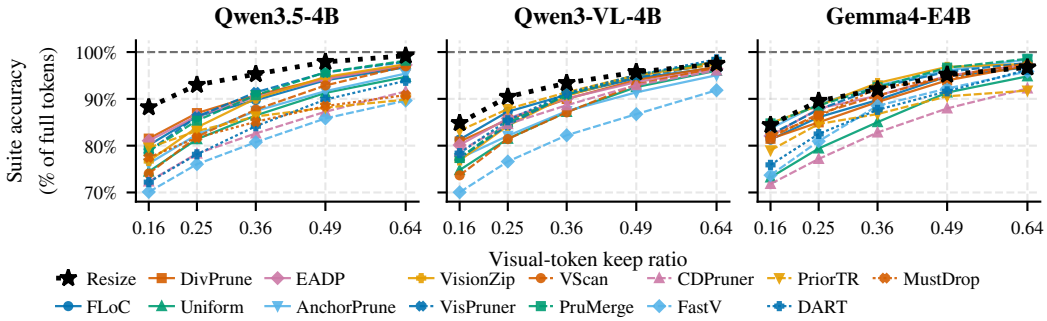

Figure 1: **Resizing vs. SOTA token-compression methods on the seven-benchmark standard suite** (relative to each model's own full-token run). The dotted black curve is naive pixel downsampling; each colored curve is a published token-compression method at the same visual-token keep ratio. *Resizing is a highly competitive baseline for recent token-compression methods*.

ABSTRACT

Vision-token compression methods are typically evaluated against one another on standard VQA benchmarks, without comparison to pixel-level resizing. Yet on modern variable-resolution VLMs, naive image resizing matches or outperforms methods published over the past two years, while reducing both vision-encoder computation and the number of visual tokens processed by the LLM. We show that this result reflects a limitation of existing evaluations: most benchmark questions remain answerable from coarse visual information and rarely test localized fine-grained evidence. We introduce **ResizeBench**, a benchmark of 1,458 resolution-critical samples designed to measure this missing regime. Each sample is associated with a verified *resolution barrier*, the image-area budget at which its answer-relevant evidence becomes unrecoverable through resizing. We mine candidate barriers from a sweep of 387K samples and verify them using two frontier VLMs, answer filtering, and factuality checking. Across 15 token-compression methods, three VLM backbones, and 7.2M generated answers, the standard-suite and ResizeBench leaderboards exhibit weak agreement. Resizing performs strongly on standard benchmarks but degrades sharply when answers depend on localized detail, where selective compression provides substantial gains. The highest-ranked compression method also varies across backbones, indicating that token compression remains an open problem with substantial method–model dependence and no universally effective strategy. These results establish resizing as an essential matched-budget baseline and resolution-critical evaluation as a necessary complement to standard VQA benchmarks.

1 INTRODUCTION

Vision-token compression methods are typically compared only against other token-selection approaches on standard VQA benchmarks. However, these approaches typically do not compare against standard pixel downsampling or resizing as a baseline. This is due to the fact that early

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.

methods such as FastV, DivPrune, and VisionZip were developed for fixed-resolution LLaVA models (Chen et al., 2024a; Alvar et al., 2025; Yang et al., 2025). A smaller input produced the same number of blurrier tokens rather than a smaller token sequence. This convention persisted as later work inherited the same model families and evaluation protocols (Cho et al., 2026; Wang et al., 2026; Oh & Kim, 2026; Zhang et al., 2025b; 2026; Shang et al., 2025; Zhang et al., 2025c; Chen et al., 2026b; Wen et al., 2025; Liu et al., 2024a). Modern Qwen-VL models (Bai et al., 2023; Wang et al., 2024b; Bai et al., 2025b;a) and the Gemma family (Team et al., 2024; Beyer et al., 2024) instead accept variable resolutions, making pixel-level resizing a valid zero-parameter compression baseline. Because resizing acts before the vision encoder, it can also reduce encoder computation, whereas post-ViT and in-LLM methods process the full-resolution image before removing tokens.

Resizing is highly competitive on standard benchmarks. We show in Figure 1 that naive resizing is quite competitive across OCRBench-v2 (Fu et al., 2025b), InfoVQA (Mathew et al., 2022), AI2D (Kembhavi et al., 2016), V* (Wu & Xie, 2024), RealWorldQA (RealWorldQA), SEED-Bench-IMG (Li et al., 2024a), and VQAv2 (Goyal et al., 2017) for every modern VLM tested. It ranks first in ten of fifteen model-budget settings, matching or outperforming methods published within the past two years at CVPR, NeurIPS, ICCV, ECCV, ICLR, EMNLP, and TMLR (Section 4). This exposes both a missing baseline and a benchmark limitation.

What this result means. Standard benchmarks are not inherently flawed: they measure broad VLM capability, but their questions mostly remain answerable from coarse global information after downsampling. This gives two practical takeaways. First, resizing should be a matched-budget baseline in every token-compression study. It acts before the vision encoder and can reduce both encoder and LLM computation, so equal final token counts need not imply equal end-to-end cost. Second, selective compression remains important when an answer depends on sparse, fine-grained evidence such as small text, symbols, or localized objects. Resizing degrades every region uniformly, whereas post-ViT and in-LLM methods first encode the full-resolution image and can retain tokens from the relevant region while removing redundant background content. They pay the full encoder cost but can outperform resizing when the answer depends on detail concentrated in a small part of the image.

ResizeBench evaluates the missing regime. These observations motivate ResizeBench, a benchmark comprising 1,458 resolution-critical samples. Each sample is associated with a *resolution barrier*: an image-area budget at which the visual evidence required to answer the corresponding question is no longer recoverable. Conventional benchmarks typically keep the sample set fixed and impose a single global compression ratio, despite the fact that most questions exhibit limited sensitivity to resolution. In contrast, ResizeBench calibrates difficulty at the level of each image–question pair: for any given budget, evaluation is performed on those samples whose answer-relevant evidence vanishes at that barrier. Consequently, the metric shifts from estimating average robustness over a fixed test suite to probing, in a targeted manner, which information a given compression method preserves. Section 3 details the dataset construction and verification procedures.

The two evaluation protocols yield divergent results. Across the full 7.2M-answer audit, rankings derived from the standard suite and from ResizeBench exhibit weak correspondence, with Kendall τ_b ranging from 0.15 to 0.34. Leaderboard positions diverge substantially: approaches that rank near the bottom under the standard suite can rank near the top on ResizeBench, whereas leading methods under the standard suite can fall to the bottom under ResizeBench. This pattern persists across model families and token-budget settings. These findings indicate that the standard suite is not simply a noisy proxy for resolution-critical evaluation; rather, it measures a distinct aspect of system behavior.

Our contributions are as follows:

- We perform a three-model benchmarking over 7.2M generated answers, demonstrating that input resizing constitutes a strong baseline within standard evaluation suites and can decrease computational demands for both the vision encoder and the LLM.
- We propose ResizeBench, a benchmarking protocol comprising 1,458 resolution-critical samples and per-sample barrier thresholds, designed to evaluate each token-budget setting at which task-relevant visual evidence becomes unavailable.
- We empirically show that rankings on standard benchmarks and resolution-critical benchmarks are largely independent, identify systematic reversals in method ordering, and argue for reporting both matched-budget resizing baselines and ResizeBench outcomes.

2 Related Work

ResizeBench connects four lines of work: variable-resolution VLMs, vision-token compression, VLM evaluation, and model-in-the-loop benchmark construction.

Native and variable-resolution vision-language models. Early VLMs such as LLaVA (Liu et al., 2023) used a fixed input grid, so a smaller image still produced the same number of blurrier tokens. Resizing could not serve as a compression baseline. Tiling systems such as LLaVA-NeXT, LLaVA-UHD, InternVL, and Idefics2 (Li et al., 2024b; Guo et al., 2024; Chen et al., 2024c; Laurençon et al., 2024) later allowed token count to grow with the number of fixed-size tiles, although every tile still uses one canonical scale. Native-resolution models go further. NaViT, Qwen-VL, Qwen2-VL, Qwen2.5-VL, Qwen3-VL, Gemma, PaliGemma, and Pixtral (Dehghani et al., 2023; Bai et al., 2023; Wang et al., 2024b; Bai et al., 2025b;a; Team et al., 2024; Beyer et al., 2024; Agrawal et al., 2024) tie token count directly to image resolution. Resizing is therefore comparable to learned pruning at the same token budget without changing the architecture or training the model. Evaluation protocols largely retain the assumptions of fixed-resolution models. This mismatch explains why resizing is both valid on current VLMs and largely absent from compression tables.

Vision-token compression. Vision-token compression reduces decoder attention and prefill cost by removing visual tokens. Post-ViT methods shorten the language-model prefill, but only after the vision encoder has processed the full-resolution image. This family includes FLoC, DivPrune, uniform sampling, EADP, AnchorPrune, VisionZip, VisPruner, VScan, PruMerge, and CDPruner (Cho et al., 2026; Alvar et al., 2025; Wang et al., 2026; Oh & Kim, 2026; Yang et al., 2025; Zhang et al., 2025b; 2026; Shang et al., 2025; Zhang et al., 2025c). In-LLM methods prune after one or more decoder layers, often using early attention as a relevance signal. These include FastV, PriorTR, DART, and MustDrop (Chen et al., 2024a; 2026b; Wen et al., 2025; Liu et al., 2024a). They remove less prefill cost because the initial layers still process the full sequence. The distinction is also mechanistic: PriorTR conditions selection on the instruction, DART uses duplicate-aware attention pivots, and MustDrop combines vision-side merging with later pruning. We retain the two families because pruning location determines both what evidence survives and when computation is saved. We evaluate them at scale (Section 4); preliminary runs also included SparseVLM and V2Drop (Zhang et al., 2025d; Chen et al., 2026a). Existing comparisons and surveys (Cao et al., 2025) do not evaluate these methods against pixel-level resizing, which reduces both encoder work and the LLM sequence, at matched token budgets.

Benchmarks for vision-language models. Compression studies rely on general-purpose VLM benchmarks spanning OCR and documents (Mathew et al., 2021; 2022; Masry et al., 2022; Chen et al., 2021; Liu et al., 2024c; Fu et al., 2025b; Biten et al., 2019; Kim et al., 2022), visual question answering (Goyal et al., 2017; Singh et al., 2019; Gurari et al., 2018; Li et al., 2023; 2024a; Fu et al., 2025a; Liu et al., 2024b; Chen et al., 2024b), and multimodal reasoning (Yue et al., 2024; Lu et al., 2024; Fu et al., 2024; Wang et al., 2024a; Lu et al., 2022; RealWorldQA; Wang et al., 2025; Cheng et al., 2024; Wu & Xie, 2024; Kembhavi et al., 2016). Our 387K-sample sweep draws from 39 such tasks, and our standard suite contains OCRBench-v2, InfoVQA, AI2D, V*, RealWorldQA, SEED-Bench-IMG, and VQAv2. This choice follows the evaluations used by the compression methods we audit. We standardize prompts and scoring with LMMS-Eval (Zhang et al., 2025a), enabling a 7.2M-answer comparison across three model families. These benchmarks report accuracy at a fixed dataset resolution; they do not measure the resolution sensitivity of individual samples. Section 5 shows that this missing axis prevents them from distinguishing selective compression from resizing.

Robustness benchmarks and model-in-the-loop construction. Corruption benchmarks such as ImageNet-C (Hendrycks & Dietterich, 2019) apply a common severity schedule to every image and average performance over that schedule. Difficulty-mining benchmarks such as Dynabench, ANLI, and ARC (Kiela et al., 2021; Nie et al., 2020; Clark et al., 2018) instead retain examples that defeat a reference model. ResizeBench combines these ideas but changes the unit of evaluation. Global corruption levels measure average robustness, while a resolution barrier identifies the specific budget at which one image-question pair fails. ResizeBench discovers that budget separately for every sample, then verifies the failure with independent frontier models and a factuality judge (Section 3). Per-sample barriers allow resizing and learned compression to be compared on the same budget while testing the evidence that each example was selected to lose.

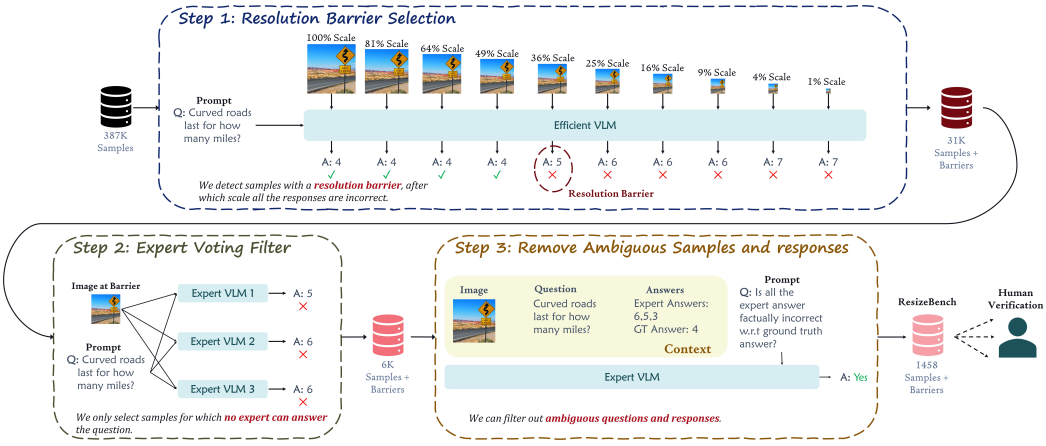

Figure 2: ResizeBench construction. **Step 1:** Sweep resolutions and detect clean barriers. **Step 2:** Verify barriers with frontier VLMs. **Step 3:** Filter ambiguous answers, label content, and finalize barriers 40–80 (1,458 samples). The final dataset is then human-verified on 100 random samples.

3 BUILDING RESIZEBENCH

ResizeBench is constructed through three automated stages (Figure 2): resolution sweeping, per-sample barrier detection, and frontier-model verification and filtering on the final 40–80 barrier grid. A final verification step audits the completed dataset.

Step 1: Resolution Sweep.

We begin from 39 image-understanding tasks totaling 387,513 samples, spanning document QA (DocVQA, InfoVQA, ChartQA, WebSRC, OCRBench, OCRBench-v2, ST-VQA, SynthDoG), general VQA (VQAv2, TextVQA, VizWiz-VQA, POPE, SEED-Bench, MME, MMBench-{en,cn,ko}, MMStar), and reasoning/perception (MMMU, MathVista, BLINK, MuirBench, ScienceQA, RealWorldQA, xLRS-lite, ScreenSpot). Every image is resized isotropically to ten side-scales $s \in \{100, 90, 80, 70, 60, 50, 40, 30, 20, 10\}$ percent of the original side length, and an efficient VLM (Qwen3-VL-4B-Instruct) is evaluated at each scale under the source task's own metric. We deliberately use an efficient 4B model for this $387{,}513 \times 10$ screening sweep. Its role is to identify candidate resolution flips at low cost, not to certify the final benchmark; only the retained candidates are subsequently sent to frontier models. This coarse-to-expert pipeline avoids running expensive verification over millions of image-scale pairs. Both image dimensions scale by $s/100$, so the fraction of image *area*, or in a multi-resolution VLM, of vision tokens which is retained at scale s is

$$\text{area_budget}(s) = (s/100)^2. \tag{1}$$

Area budget is the axis on which pixel resizing and token pruning are commensurable: a method keeping a fraction r of tokens and a resize keeping side-scale $\sqrt{r}$ deliver the same information budget to the language model, and every comparison with resizing is made at that matched budget.

Step 2: Barrier Detection.

For each sample we look for a single clean transition from a reliably-correct high-resolution regime to a reliably-incorrect low-resolution one. For candidate scale s, let $a(s)$ and $b(s)$ be the mean score strictly above and below s; we select the barrier s^* maximizing $a(s) - b(s)$ subject to $a(s) \geq 0.6$ and $b(s) \leq 0.4$, also requiring the sample to be correct at full resolution (so later failure is attributable to resolution, not ignorance) and that s^* is not at the extreme bottom of the sweep. Samples whose accuracy oscillates with resolution admit no valid s^* and are discarded. Restricting the retained flips to the final barriers 40, 50, 60, 70, and 80 leaves 16,011, 8,279, 4,079, 1,961, and 1,058 samples, respectively: 31,388 clean barrier flips (8.1% of the 387,513-sample pool).

Step 3: Frontier-Model Verification and Finalization.

A barrier inferred from one model can encode mislabeled ground truth or an ambiguous question rather than a resolution requirement. We therefore retain candidate barriers at side-scales 40, 50,

Table 1: ResizeBench composition: verified barrier structure, where area budget $= (s^*/100)^2$, and visual-content category distribution.

Barrier structure						**Visual-content categories**					
Barrier s^* (%)	40	50	60	70	80	Category	Natural scene	Text in a natural scene	Webpage/digital document text	Chart/figure	Real document text
Area budget	0.16	0.25	0.36	0.49	0.64						
# samples	477	498	265	132	86	# samples	846	248	227	96	41
Total: 1,458 samples											

60, 70, and 80, corresponding to area budgets 0.16, 0.25, 0.36, 0.49, and 0.64, and re-evaluate every candidate with two independent frontier VLMs (Claude, GPT) at exactly the sample's own barrier resolution, keeping only samples for which *both* answer incorrectly: a barrier that a stronger model clears is a property of the efficient model, not of the image–question pair. Both frontier models are explicitly instructed not to use tools to enhance the image resolution, to answer from the supplied image as is, and to return "Unanswerable" when degradation prevents an answer. This all-experts-fail criterion retains 3,223, approximately 1,410, 930, 500, and 310 samples at barriers 40–80, respectively, or approximately 6,370 in total. Two artifact filters follow: a lenient-match pass that drops nominally wrong answers matching any ground truth after normalization, and a GPT-vision factuality judge that drops answers correct in substance but different in form. Together these filters retain 1,127, 501, 270, 134, and 86 samples at barriers 40–80, respectively, or 2,118 in total. Each surviving image is tagged with a visual-content category by an LLM using five base categories, a free-form extension pass, and a consolidation pass. Requiring a single image field, category tag, and valid deduplicated QA record yields 477, 498, 265, 132, and 86 samples at barriers 40–80, respectively: 1,458 verified samples in total. Each carries the full-resolution image, question, deduplicated ground truths, category, and barrier area budget. Because Step 2 admits only samples verified correct at full resolution before the flip, failure at the barrier is attributable to resolution rather than question difficulty. The final benchmark statistics are shown in Table 1.

Verification step. After finalization, a human annotator reviews 100 uniformly sampled entries, comparing each query with its full-resolution and barrier images to verify that resizing materially distorts the information needed to answer.

What a barrier evaluation feeds the model. Evaluating ResizeBench at drop ratio r means: for every sample whose barrier budget is r, show the *resized* image (side-scale $\sqrt{r}$ of the original). The images themselves change with the drop ratio. i.e. we have a per-sample, per-barrier input, not a fixed input scored under a budget sweep. The area budget of a sample is its own; the compression ratio of a token-compression method is set per barrier to the same value (Equation 1), so resize and every selector are charged the identical per-barrier information budget.

4 Experimental Setup

Models. Three multi-resolution open VLMs: **Qwen3-VL-4B-Instruct** (the sweep model, and the family lineage ResizeBench's verification shares with), **Qwen3.5-4B** (hybrid-attention successor), and **Gemma-4-E4B** (a different model family, testing transfer of the axis beyond Qwen).

Methods. We compare published token-compression methods with two baselines (`full`: native resolution; `resize`: pixel downsampling). Post-ViT methods include FLoC (ICLR 2026), DivPrune and VisionZip (CVPR 2025), VisPruner and PruMerge (ICCV 2025), CDPruner (NeurIPS 2025), VScan (TMLR 2026), and EADP and AnchorPrune (ECCV 2026), alongside Uniform sampling as a control. In-LLM methods include FastV (ECCV 2024), PriorTR (ECCV 2026), DART (EMNLP 2025), and MustDrop (arXiv). Post-ViT methods shorten the LLM input after the full vision tower; in-LLM methods prune after the first decoder layers and therefore save less prefill. All methods use model-agnostic implementations in a shared harness. We match methods by final visual-token count rather than hardware-specific FLOPs or latency. This isolates information retention but is conservative for resizing, which also reduces vision-encoder computation. A compact method and venue summary is provided in Appendix A, and implementation details in Appendix B.

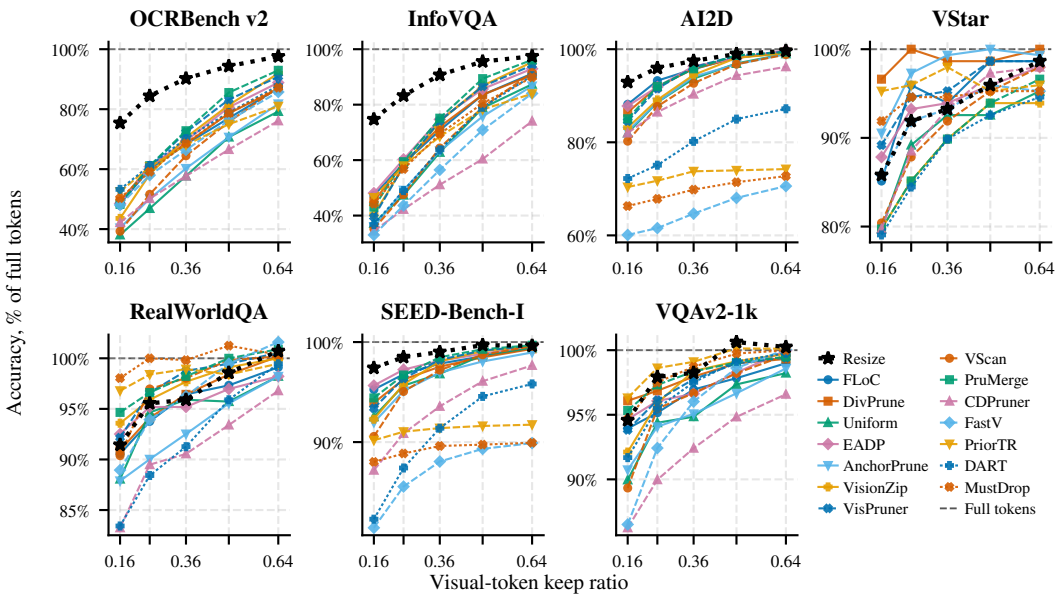

Figure 3: Seven-benchmark accuracy vs. keep ratio, all methods on Qwen3.5-4B (dashed line = the model's full-token run). *Resizing is a highly competitive baseline*.

Suites. The standard suite is a seven-benchmark set used by current compression papers: OCRBench-v2, InfoVQA, AI2D, V*, RealWorldQA, SEED-Bench-IMG, VQAv2 (VQAv2 capped at 1,000 samples; ≈32K samples per model per condition). Each method is run at keep ratios $\{0.16, 0.25, 0.36, 0.49, 0.64\}$ plus full, mirroring the ResizeBench barrier budgets, so suite and ResizeBench results share a token-price axis. We use each source task's native metric; ResizeBench uses one lenient text match against deduplicated ground truths, identical across models.

Scale. Across three models, fifteen methods, five budgets, plus full and resize baselines, the audit comprises 7,223,378 generated answers: 7,153,394 on the standard suite (2,437,928 each for Qwen3.5 and Gemma4, 2,277,538 for Qwen3-VL, whose suite run has no mustdrop) and 69,984 on ResizeBench (16 methods × 1,458 samples × 3 models) — to our knowledge the largest like-for-like token-compression comparison published, and the first in which every method is priced against resizing at the same budget.

5 Results

5.1 Seven-Benchmark Suite Results

Standard-Suite Performance. Figure 3 presents the seven-benchmark results for Qwen3.5-4B, with Qwen3-VL and Gemma-4 reported in Appendix C. On Qwen3.5-4B, all fifteen methods retain more than 85% of full accuracy at keep ratios of 0.49 and above. Resize retains 94.7% on average across budgets and reaches 99.2% at keep 0.64. On Qwen3-VL-4B, resize achieves the highest average retention at 92.4%, while twelve of thirteen compression methods remain within ten points of it. Gemma-4 separates the methods slightly more. VisionZip achieves 92.2%, PruMerge, AnchorPrune, and resize achieve approximately 92%, and CDPruner achieves 82.4%. Even at keep 0.25, the lowest-performing method retains approximately four-fifths of full accuracy. These results demonstrate that *the standard suite remains largely insensitive to substantial visual-token reduction*.

5.2 ResizeBench: Visual Samples

Figure 4 presents three examples before and at their sample-specific resolution barriers. Since each sample has a different failure boundary, the resized input also differs across samples. This directly tests whether a compression method preserves the visual evidence required to answer the question at the matched barrier budget.



Figure 4: **Examples of resizebench samples.** Each group shows one query above its original image and the image at the failing resize barrier.

	Qwen3-VL						Qwen3.5-VL						Gemma-4					
Method	b40	b50	b60	b70	b80	Mean	b40	b50	b60	b70	b80	Mean	b40	b50	b60	b70	b80	Mean
Post-ViT pruning																		
FLoC	50.3	59.6	64.5	53.8	65.1	58.7	45.9	51.2	51.3	40.9	44.2	46.7	35.2	39.8	36.2	25.8	23.3	32.0
DivPrune	52.2	61.4	61.9	52.3	66.3	58.8	47.6	53.8	50.6	47.0	50.0	49.8	38.4	41.0	38.5	29.5	22.1	33.9
Uniform	35.0	43.0	43.4	42.4	47.7	42.3	29.4	38.6	40.0	31.8	34.9	34.9	24.5	25.9	27.5	19.7	20.9	23.7
EADP	50.3	60.0	60.0	56.1	67.4	58.8	44.2	52.0	55.5	40.9	48.8	48.3	36.7	42.6	38.5	27.3	24.4	33.9
AnchorPrune	41.5	42.4	47.5	42.4	54.7	45.7	37.5	46.2	44.9	40.9	44.2	42.7	41.5	43.8	39.6	29.5	25.6	36.0
VisionZip	38.2	53.6	59.2	59.8	74.4	57.1	35.0	47.2	52.8	47.0	48.8	46.2	38.4	40.4	39.6	34.1	23.3	35.1
VisPruner	44.4	57.8	63.0	62.9	69.8	59.6	46.5	52.4	51.7	51.5	50.0	50.4	38.4	43.2	39.2	30.3	24.4	35.1
VScan	32.9	45.8	55.1	58.3	68.6	52.1	31.4	41.2	46.4	50.8	46.5	43.3	37.1	40.2	39.6	31.8	22.1	34.2
PruMerge	46.5	56.8	61.1	56.1	74.4	59.0	43.8	51.6	53.6	50.8	53.5	50.7	41.3	42.4	40.0	28.8	20.9	34.7
CDPruner	45.3	52.8	56.6	56.1	60.5	54.2	31.4	38.0	43.0	33.3	33.7	35.9	26.0	30.5	30.6	27.3	22.1	27.3
In-LLM pruning																		
FastV	29.1	37.6	35.8	43.2	39.5	37.1	36.1	44.4	50.6	40.2	41.9	42.6	32.5	36.9	38.1	26.5	23.3	31.5
PriorTR	45.5	52.8	52.8	48.5	51.2	50.2	49.1	57.2	54.0	49.2	50.0	51.9	28.1	28.1	26.0	15.9	16.3	22.9
DART	27.7	30.5	30.2	25.8	40.7	31.0	41.9	52.4	47.2	39.4	48.8	45.9	36.9	36.3	34.7	27.3	24.4	31.9
MustDrop	40.7	48.0	52.8	51.5	55.8	49.8	46.1	54.0	53.2	42.4	48.8	48.9	43.2	42.8	41.9	26.5	25.6	36.0

Table 2: ResizeBench accuracy (%) at each resize barrier across all three backbones. Colored divider rows separate post-ViT and in-LLM approaches. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

5.3 ResizeBench Results per Model

Table 2 reports per-barrier accuracy on ResizeBench. Each sample is evaluated at its own barrier, and every compression method is matched to the same area budget using Equation 1. We also report the mean across the five barriers.

Overall Results. Accuracy improves from barrier 40 to barrier 80 for every method. However, the ranking between methods is established at the lower barriers. Resize has the lowest mean accuracy on both Qwen models. It achieves 26.7% on Qwen3.5-4B compared with 51.9% for PriorTR, and 18.5% on Qwen3-VL-4B compared with 59.6% for VisPruner. This large difference occurs because each sample is evaluated at the resolution where its answer becomes unrecoverable through resizing. *At these sample-specific barriers, selective compression can substantially outperform matched-budget resizing*. PriorTR leads Qwen3.5 with 51.9% mean accuracy, appears in the middle on Qwen3-VL, and falls to 22.9%, the lowest compression-method mean on Gemma-4. *The highest-ranked token-compression method therefore varies across backbones*. Uniform dropping, as expected is the most consistent failure case. The category-level breakdown and analysis are provided in Appendix D.

Figure 5 directly compares average accuracy on the two suites. Across all five budgets and all three models, standard-suite accuracy remains higher than matched-budget ResizeBench accuracy, but its magnitude and shape vary by backbone. Compression methods act on model-specific representations: post-ViT selectors rank tokens using features produced by each vision encoder, while in-LLM methods depend on attention patterns and the decoder layer at which pruning occurs. A method that

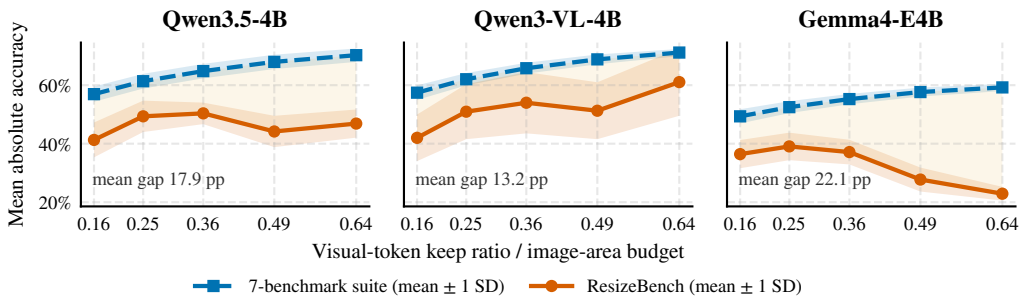

Figure 5: **Accuracy gap across matched pixel budgets.** Each panel shows that standard-suite accuracy remains above ResizeBench accuracy across matched budgets. Shading shows ± 1 standard deviation across methods. *The standard suite and ResizeBench probe distinct compression regimes.*

preserves fine-grained evidence on one backbone can therefore remove it on another, so its behavior does not transfer perfectly across models. Standard-suite curves remain comparatively smooth because most questions tolerate these differences in token selection. *ResizeBench exposes stronger method–backbone dependence because success requires preserving small, localized evidence.*

5.4 Leaderboard Divergence Across Evaluations

We next evaluate whether the standard suite and ResizeBench rank methods in the same order. For Qwen3.5-4B, we compare fourteen token-selection methods and resize at five matched budgets. Each ResizeBench barrier b is matched to keep ratio $(b/100)^2$, and scores are normalized to each model's own full-token run. The resulting Kendall τ_b values are 0.34, 0.27, 0.15, 0.26, and 0.25 at keep ratios $0.16, 0.25, 0.36, 0.49$, and 0.64 (Figure 6). These low values indicate weak agreement between the two leaderboards.

No individual standard benchmark consistently predicts the ResizeBench ranking. The highest agreement is $\tau_b = 0.54$ for VQAv2 at keep 0.16, while AI2D and SEEDBench-IMG have negative agreement at keep 0.25. The same analysis on Qwen3-VL and Gemma-4 is provided in Appendix F. Across all models, the highest single-benchmark agreement is $\tau_b = 0.73$ for OCRBench-v2 on Gemma-4. Moreover, the benchmark with the highest agreement changes across models. *No standard benchmark therefore serves as a reliable proxy for resolution-critical performance.* Full per-benchmark values are provided in Appendix F.1.

The complete keep-0.49 leaderboard and its detailed analysis are provided in Appendix F. The highest-ranked method differs across backbones, indicating that performance gains on one model do not establish a generally effective token-selection strategy. *Token compression therefore remains an open problem with substantial method–model dependence.*

5.5 Practical Implications

Across three model families, fifteen methods, and five matched budgets, the standard suite preserves high accuracy but does not predict performance at resolution barriers. Resize can lead the standard-suite evaluation while ranking last on ResizeBench, and the highest-ranked token-selection method differs across models. The tradeoff is direct: resizing is effective when coarse global information is sufficient and additionally saves vision-encoder computation, while selective pruning remains important when the answer depends on localized fine detail. Standard benchmarks are not inherently flawed, but their questions emphasize the former regime. We therefore recommend reporting resizing at the same visual-token budget and adding a resolution-critical split such as ResizeBench, where each budget is paired with samples known to fail at that resolution.

6 Conclusion

In this work, we discussed how pixel-level resizing is highly competitive baseline to published methods on token compression, specifically on standard benchmarks. It was strongest when coarse global

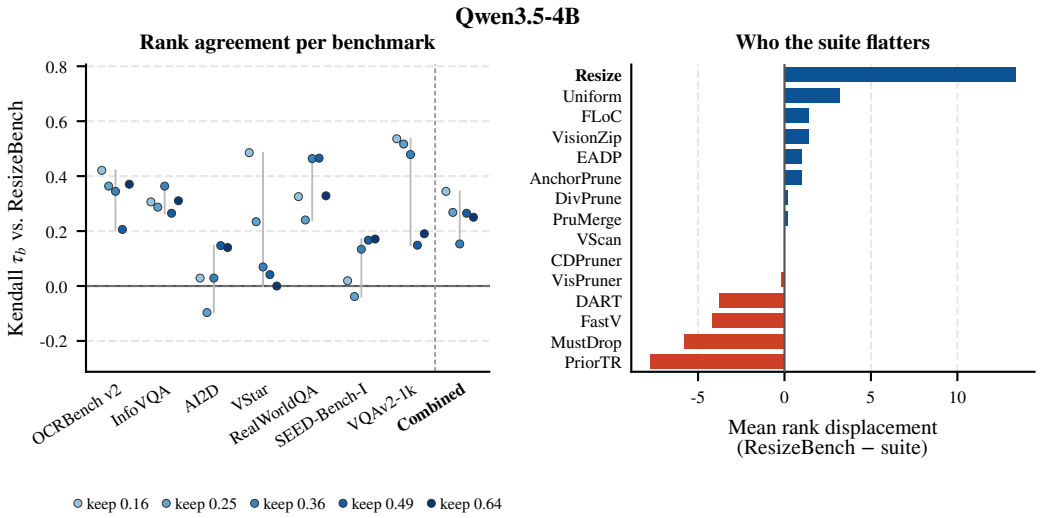

Figure 6: **Leaderboard agreement and rank displacement on Qwen3.5-4B. Left:** Kendall τ_b between each standard benchmark and ResizeBench at five matched budgets. *Combined* denotes the seven-benchmark macro-average. **Right:** mean ResizeBench rank minus standard-suite rank. Resize has the largest displacement at $+13.4$ ranks, while PriorTR moves in the opposite direction by -7.8 ranks. Results for Qwen3-VL and Gemma-4 are in Appendix F.

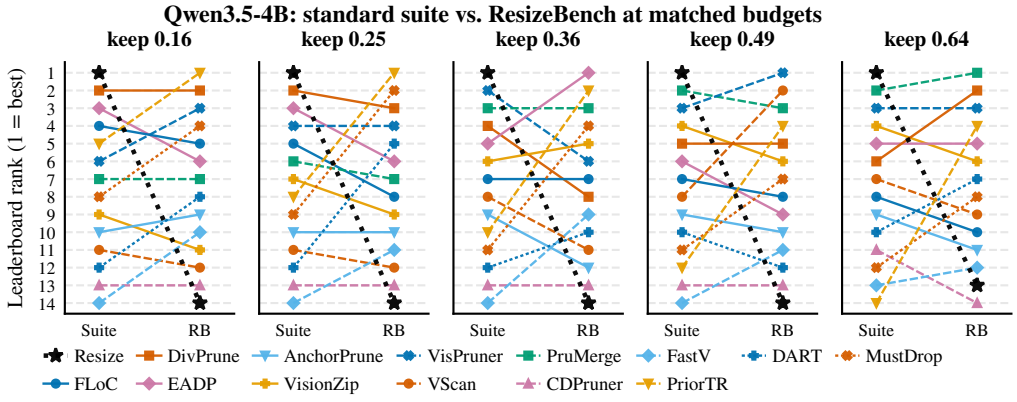

Figure 7: **Leaderboard re-ranking on Qwen3.5-4B.** Across budgets, 32–43% of approach pairs reverse order. *Standard-suite rankings do not reliably predict resolution-critical performance*.

information is sufficient while selective pruning remains necessary when an answer depends on localized fine detail that uniform downsampling removes. However, standard benchmarks were not designed to isolate this tradeoff and therefore favor the former regime. ResizeBench supplies the complementary measurement with 1,458 samples whose resolution barriers are verified by frontier models. Across three model families, 15 published methods, and 7.2M generated answers, standard-suite and ResizeBench rankings are close to independent, and resize falls from first on the suite to last on both Qwen models. Our analysis recommends that future studies on vision token compression report resizing at matched token budgets, recognize that such comparisons understate its encoder-side efficiency, and evaluate on resolution-critical data where fine-grained evidence matters. Most importantly, variation in the highest-ranked method across backbones indicates that token compression remains an open problem: performance gains on one model do not establish a generally effective selection strategy.

AI USE STATEMENT.

Generative AI tools were used to assist with language editing, code development, and figure and table formatting. All experimental decisions, analyses, claims, citations, and final manuscript content were performed and verified by the authors. The use of VLMs for benchmark construction, verification, and evaluation is part of the proposed methodology and is described explicitly in Section 3.

ETHICS STATEMENT.

ResizeBench is intended for research on VLM evaluation and computational efficiency and does not involve model deployment or consequential decision-making. As it is derived from existing research datasets, it may inherit biases, annotation errors etc. from its sources; the release will preserve source attribution and applicable usage terms. Model-based verification may also introduce systematic errors, which we attempt to mitigate through independent judges and a human verification step.

REPRODUCIBILITY STATEMENT.

The paper and appendix specify the evaluated models, token budgets, scoring protocol, and implementation settings for every compression method. As part of supplementary materials, we shared barriers and category metadata, along with scripts to obtain samples from source datasets, so that the reported evaluation can be reproduced.

REFERENCES

Pravesh Agrawal, Szymon Antoniak, Emma Bou Hanna, Baptiste Bout, Devendra Chaplot, Jessica Chudnovsky, Diogo Costa, Baudouin De Monicault, Saurabh Garg, Theophile Gervet, Soham Ghosh, Amélie Héliou, Paul Jacob, Albert Q. Jiang, Kartik Khandelwal, Timothée Lacroix, Guillaume Lample, Diego Las Casas, Thibaut Lavril, Teven Le Scao, Andy Lo, William Marshall, Louis Martin, Arthur Mensch, Pavankumar Muddireddy, Valera Nemychnikova, Marie Pellat, Patrick Von Platen, Nikhil Raghuraman, Baptiste Rozière, Alexandre Sablayrolles, Lucile Saulnier, Romain Sauvestre, Wendy Shang, Roman Soletskyi, Lawrence Stewart, Pierre Stock, Joachim Studnia, Sandeep Subramanian, Sagar Vaze, Thomas Wang, and Sophia Yang. Pixtral 12b, 2024. URL https://arxiv.org/abs/2410.07073.

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. doi: 10.1109/cvpr52734.2025.00877. URL http://dx.doi.org/10.1109/CVPR52734.2025.00877.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023. URL https://arxiv.org/abs/2308.12966.

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

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng,

Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b. URL https://arxiv.org/abs/2502.13923.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. Paligemma: A versatile 3b vlm for transfer, 2024. URL https://arxiv.org/abs/2407.07726.

Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marcal Rusinol, C.V. Jawahar, Ernest Valveny, and Dimosthenis Karatzas. Scene text visual question answering. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4290–4300. IEEE, October 2019. doi: 10.1109/iccv.2019.00439. URL http://dx.doi.org/10.1109/ICCV.2019.00439.

Weidong Cao, Zeqin Feng, Fangwei Sun, Danjun Liu, Yongqiang Xie, and Zhongbo Li. A survey on visual token compression for efficient vision-language models. In *2025 5th International Conference on Advanced Algorithms and Neural Networks (AANN)*, pp. 735–741. IEEE, August 2025. doi: 10.1109/aann66429.2025.11257670. URL http://dx.doi.org/10.1109/AANN66429.2025.11257670.

Junjie Chen, Xuyang Liu, Zichen Wen, Yiyu Wang, Siteng Huang, and Honggang Chen. Variation-aware vision token dropping for faster large vision-language models, 2026a. URL https://arxiv.org/abs/2509.01552.

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*, pp. 19–35. Springer Nature Switzerland, November 2024a. ISBN 9783031730047. doi: 10.1007/978-3-031-73004-7_2. URL http://dx.doi.org/10.1007/978-3-031-73004-7_2.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. Are we on the right way for evaluating large vision-language models? In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 27056–27087. Curran Associates, Inc., 2024b. doi: 10.52202/079017-0850. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/2f8ee6a3d766b426d2618e555b5aeb39-Paper-Conference.pdf.

Xingyu Chen, Zihan Zhao, Lu Chen, JiaBao Ji, Danyang Zhang, Ao Luo, Yuxuan Xiong, and Kai Yu. WebSRC: A dataset for web-based structural reading comprehension. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 4173–4185, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.343. URL https://aclanthology.org/2021.emnlp-main.343/.

Zengjie Chen, Yuxiang Cai, Jingcai Guo, Taotao Cai, Jianwei Yin, and Zhi Chen. *Accelerating Multimodal Large Language Models with Prior-Corrected Token Reduction*, pp. 213–233. Springer Nature Switzerland, 2026b. ISBN 9783032368393. doi: 10.1007/978-3-032-36839-3_13. URL http://dx.doi.org/10.1007/978-3-032-36839-3_13.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks, 2024c. URL https://arxiv.org/abs/2312.14238.

Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Li YanTao, Jianbing Zhang, and Zhiyong Wu. SeeClick: Harnessing GUI grounding for advanced visual GUI agents. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9313–9332, Bangkok, Thailand,

August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.505. URL https://aclanthology.org/2024.acl-long.505/.

Janghoon Cho, Jungsoo Lee, Munawar Hayat, Kyuwoong Hwang, Fatih Porikli, and Sungha Choi. Floc: Facility location-based efficient visual token compression for long video understanding, 2026. URL https://arxiv.org/abs/2511.00141.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Mostafa Dehghani, Basil Mustafa, Josip Djolonga, Jonathan Heek, Matthias Minderer, Mathilde Caron, Andreas Steiner, Joan Puigcerver, Robert Geirhos, Ibrahim M Alabdulmohsin, Avital Oliver, Piotr Padlewski, Alexey Gritsenko, Mario Lucic, and Neil Houlsby. Patch n' pack: Navit, a vision transformer for any aspect ratio and resolution. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 2252–2274. Curran Associates, Inc., 2023. doi: 10.52202/075280-0106. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/06ea400b9b7cfce6428ec27a371632eb-Paper-Conference.pdf.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, Rongrong Ji, Caifeng Shan, and Ran He. Mme: A comprehensive evaluation benchmark for multimodal large language models. In *Advances in Neural Information Processing Systems 38*, NeurIPS 2025, pp. 162549–162567. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2025a. doi: 10.52202/085713-4899. URL http://dx.doi.org/10.52202/085713-4899.

Ling Fu, Zhebin Kuang, Jiajun Song, Mingxin Huang, Biao Yang, Yuzhe Li, Linghao Zhu, Qidi Luo, Xinyu Wang, Hao Lu, Zhang Li, Guozhi Tang, Bin Shan, Chunhui Lin, Qi Liu, Binghong Wu, Hao Feng, Hao Liu, Can Huang, Jingqun Tang, Wei Chen, Lianwen Jin, Yuliang Liu, and Xiang Bai. Ocrbench v2: An improved benchmark for evaluating large multimodal models on visual text localization and reasoning. In *Advances in Neural Information Processing Systems 38*, NeurIPS 2025, pp. 107930–107981. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2025b. doi: 10.52202/085713-3253. URL http://dx.doi.org/10.52202/085713-3253.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. *BLINK: Multimodal Large Language Models Can See but Not Perceive*, pp. 148–166. Springer Nature Switzerland, October 2024. ISBN 9783031733376. doi: 10.1007/978-3-031-73337-6_9. URL http://dx.doi.org/10.1007/978-3-031-73337-6_9.

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 *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6325–6334. IEEE, 2017. doi: 10.1109/cvpr.2017.670. URL http://dx.doi.org/10.1109/CVPR.2017.670.

Zonghao Guo, Ruyi Xu, Yuan Yao, Junbo Cui, Zanlin Ni, Chunjiang Ge, Tat-Seng Chua, Zhiyuan Liu, and Gao Huang. *LLaVA-UHD: An LMM Perceiving Any Aspect Ratio and High-Resolution Images*, pp. 390–406. Springer Nature Switzerland, November 2024. ISBN 9783031730108. doi: 10.1007/978-3-031-73010-8_23. URL http://dx.doi.org/10.1007/978-3-031-73010-8_23.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3608–3617. IEEE, 2018. doi: 10.1109/cvpr.2018.00380. URL http://dx.doi.org/10.1109/CVPR.2018.00380.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations, 2019. URL https://arxiv.org/abs/1903.12261.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. *A Diagram is Worth a Dozen Images*, pp. 235–251. Springer International Publishing, 2016. ISBN 9783319464930. doi: 10.1007/978-3-319-46493-0_15. URL http://dx.doi.org/10.1007/978-3-319-46493-0_15.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in NLP. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4110–4124, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.324. URL https://aclanthology.org/2021.naacl-main.324/.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. *OCR-Free Document Understanding Transformer*, pp. 498–517. Springer Nature Switzerland, 2022. ISBN 9783031198151. doi: 10.1007/978-3-031-19815-1_29. URL http://dx.doi.org/10.1007/978-3-031-19815-1_29.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models?, 2024. URL https://arxiv.org/abs/2405.02246.

Bohao Li, Yuying Ge, Yixiao Ge, Guangzhi Wang, Rui Wang, Ruimao Zhang, and Ying Shan. Seed-bench: Benchmarking multimodal large language models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13299–13308. IEEE, 2024a. doi: 10.1109/cvpr52733.2024.01263. URL http://dx.doi.org/10.1109/CVPR52733.2024.01263.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models, 2024b. URL https://arxiv.org/abs/2407.07895.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 292–305, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.20. URL https://aclanthology.org/2023.emnlp-main.20/.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023. doi: 10.52202/075280-1516. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf.

Ting Liu, Liangtao Shi, Richang Hong, Yue Hu, Quanjun Yin, and Linfeng Zhang. Multi-stage vision token dropping: Towards efficient multimodal large language model, 2024a. URL https://arxiv.org/abs/2411.10803.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. *MMBench: Is Your Multi-modal Model an All-Around Player?*, pp. 216–233. Springer Nature Switzerland, October 2024b. ISBN 9783031726583. doi: 10.1007/978-3-031-72658-3_13. URL http://dx.doi.org/10.1007/978-3-031-72658-3_13.

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), December 2024c. ISSN

1869-1919. doi: 10.1007/s11432-024-4235-6. URL http://dx.doi.org/10.1007/s11432-024-4235-6.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 2507–2521. Curran Associates, Inc., 2022. doi: 10.52202/068431-0182. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/11332b6b6cf4485b84afadb1352d3a9a-Paper-Conference.pdf.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts, 2024. URL https://arxiv.org/abs/2310.02255.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.177. URL https://aclanthology.org/2022.findings-acl.177/.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for vqa on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 2199–2208. IEEE, January 2021. doi: 10.1109/wacv48630.2021.00225. URL http://dx.doi.org/10.1109/WACV48630.2021.00225.

Minesh Mathew, Viraj Bagal, Ruben Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. Infographicvqa. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2582–2591. IEEE, January 2022. doi: 10.1109/wacv51458.2022.00264. URL http://dx.doi.org/10.1109/WACV51458.2022.00264.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4885–4901, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.441. URL https://aclanthology.org/2020.acl-main.441/.

Kyuan Oh and Bumsoo Kim. *AnchorPrune: Relevance-Anchored Contextual Expansion for Visual Token Pruning*, pp. 595–612. Springer Nature Switzerland, 2026. ISBN 9783032372116. doi: 10.1007/978-3-032-37211-6_33. URL http://dx.doi.org/10.1007/978-3-032-37211-6_33.

RealWorldQA. https://x.ai/news/grok-1.5v/.

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, October 2025. doi: 10.1109/iccv51701.2025.02122. URL http://dx.doi.org/10.1109/ICCV51701.2025.02122.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8309–8318. IEEE, 2019. doi: 10.1109/cvpr.2019.00851. URL http://dx.doi.org/10.1109/CVPR.2019.00851.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo,

Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on gemini research and technology, 2024. URL https://arxiv.org/abs/2403.08295.

Fei Wang, Xingyu Fu, James Y. Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, Tianyi Lorena Yan, Wenjie Jacky Mo, Hsiang-Hui Liu, Pan Lu, Chunyuan Li, Chaowei Xiao, Kai-Wei Chang, Dan Roth, Sheng Zhang, Hoifung Poon, and Muhao Chen. Muirbench: A comprehensive benchmark for robust multi-image understanding, 2024a. URL https://arxiv.org/abs/2406.09411.

Fengxiang Wang, Hongzhen Wang, Zonghao Guo, Di Wang, Yulin Wang, Mingshuo Chen, Qiang Ma, Long Lan, Wenjing Yang, Jing Zhang, Zhiyuan Liu, and Maosong Sun. Xlrs-bench: Could your multimodal llms understand extremely large ultra-high-resolution remote sensing imagery? In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14325–14336. IEEE, 2025. doi: 10.1109/cvpr52734.2025.01336. URL http://dx.doi.org/10.1109/CVPR52734.2025.01336.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution, 2024b. URL https://arxiv.org/abs/2409.12191.

Xuehui Wang, Xuankun Yang, and Wei Shen. Combating textual noise and redundancy: Entropy-aware dense visual token pruning, 2026. URL https://arxiv.org/abs/2607.02484.

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 Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9961–9980, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.505. URL https://aclanthology.org/2025.emnlp-main.505/.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13084–13094. IEEE, 2024. doi: 10.1109/cvpr52733.2024.01243. URL http://dx.doi.org/10.1109/CVPR52733.2024.01243.

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. doi: 10.1109/cvpr52734.2025.01843. URL http://dx.doi.org/10.1109/CVPR52734.2025.01843.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning

benchmark for expert agi. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9556–9567. IEEE, 2024. doi: 10.1109/cvpr52733.2024.00913. URL http://dx.doi.org/10.1109/CVPR52733.2024.00913.

Ce Zhang, Kaixin Ma, Tianqing Fang, Wenhao Yu, Hongming Zhang, Zhisong Zhang, Haitao Mi, and Dong Yu. Vscan: Rethinking visual token reduction for efficient large vision-language models, 2026. URL https://arxiv.org/abs/2505.22654.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. LMMs-eval: Reality check on the evaluation of large multimodal models. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 881–916, Albuquerque, New Mexico, April 2025a. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.51. URL https://aclanthology.org/2025.findings-naacl.51/.

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, October 2025b. doi: 10.1109/iccv51701.2025.01939. URL http://dx.doi.org/10.1109/ICCV51701.2025.01939.

Qizhe Zhang, Mengzhen Liu, Lichen Li, Ming Lu, Yuan Zhang, Junwen Pan, Qi She, and Shanghang Zhang. Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms, 2025c. URL https://arxiv.org/abs/2506.10967.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. Sparsevlm: Visual token sparsification for efficient vision-language model inference, 2025d. URL https://arxiv.org/abs/2410.04417.

APPENDIX

CONTENTS

A Compression Methods and Venues **17**

B Baseline Implementation Details **17**

C Detailed Seven-Benchmark Numbers by Benchmark **20**

D Detailed ResizeBench Numbers by Category **20**

E ResizeBench Conditioned on Full-Resolution Correctness **20**

F Detailed Leaderboard Divergence **22**

F.1 Per-Benchmark Rank Agreement 22

F.2 Qwen3.5-4B at Keep Ratio 0.49 22

F.3 Method-Level Insights 22

F.4 Qwen3-VL-4B and Gemma4-E4B 22

G Additional ResizeBench Examples **32**

H Limitations **38**

A COMPRESSION METHODS AND VENUES

Table A1 summarizes the publication venue and pruning stage of each evaluated method.

Table A1: Publication venues of the evaluated token-compression methods. Uniform is a sampling control; Full and Resize are omitted.

Post-ViT pruning		**In-LLM pruning**	
Method	Venue	Method	Venue
FLoC Cho et al. (2026)	ICLR 2026	FastV Chen et al. (2024a)	ECCV 2024
DivPrune Alvar et al. (2025)	CVPR 2025	PriorTR Chen et al. (2026b)	ECCV 2026
EADP Wang et al. (2026)	ECCV 2026	DART Wen et al. (2025)	EMNLP 2025
AnchorPrune Oh & Kim (2026)	ECCV 2026	MustDrop Liu et al. (2024a)	arXiv
VisionZip Yang et al. (2025)	CVPR 2025		
VisPruner Zhang et al. (2025b)	ICCV 2025		
VScan Zhang et al. (2026)	TMLR 2026		
PruMerge Shang et al. (2025)	ICCV 2025		
CDPruner Zhang et al. (2025c)	NeurIPS 2025		
Uniform	Control		

B BASELINE IMPLEMENTATION DETAILS

Every method is implemented in a single shared evaluation harness and run against all three models with the same code paths, differing only in the model-side adapter. All methods are inference-only; no model is fine-tuned. Where a paper targets a different architecture than one of our three models, the implementation is a documented adaptation, not a bit-exact reproduction of the original repository; the required disclosures are stated per method below. These implementations were validated

for executable integration and cache/position consistency (keep-all outputs were compared against the unpruned path on fixed samples), not for numerical parity with each original codebase.

Common token contract. A visual token means one post-merger token for the Qwen models (whose vision encoders apply a native spatial merge before the LLM) and one projected soft token for Gemma-4-E4B (`max_soft_tokens`=1120, dynamic resolution). Standard preprocessing uses `max_pixels`=2,097,152 and `min_pixels`=1. Text tokens are never removed. Retained visual tokens are restored to their original spatial order before the LLM, and positional bookkeeping is compacted with them: M-RoPE coordinates for retained visual tokens are preserved and `rope_deltas` updated on the Qwen models; on Gemma, absolute position IDs, `mm_token_type_ids`, the per-layer vision inputs, and the attention mask are compacted to the same retained positions. For a request containing multiple images, a post-ViT selector allocates the total budget proportionally to each image's token count and selects independently within each image.

Budget semantics. For N visual tokens and requested keep ratio r, post-ViT methods retain $\max(1, \lfloor rN \rfloor)$ tokens; PriorTR and FastV use $\lfloor rN \rfloor$; DART uses $\mathrm{round}(rN)$. Because selection granularity can shift the realized count by one token, all comparisons in this paper are matched by the logged actual retained-token count rather than the requested ratio.

Shared selector implementations. Six of the ten post-ViT methods — EADP, AnchorPrune, CDPruner, VisPruner, VisionZip, and PruMerge — dispatch to a single model-agnostic selector module on both Qwen models and on Gemma; their scoring rules are therefore literally identical across the three models, differing only in the feature space they receive. FLoC, DivPrune, Uniform, and VScan are ported per model with identical semantics.

Uniform. Selects K evenly spaced `linspace` positions over $[0, N-1]$. Uses neither image content nor the question; a position-only control. Retained features are unchanged.

FLoC. Builds the pairwise cosine-similarity matrix of the visual tokens and maximizes the facility-location coverage objective $F(S) = \sum_j \max_{i \in S} \cos(f_i, f_j)$ with a lazy-greedy priority queue, returning exactly the requested budget. This is our local facility-location baseline control rather than a port of an external implementation.

DivPrune. Farthest-first traversal over pairwise cosine distance: the first token has the largest nearest-neighbor distance, and each subsequent token maximizes its minimum distance to the selected set. Deterministic, question-independent, no merging.

EADP. A port of the official EADP code (`SJTU-DeepVisionLab/EADP`, commit `e1a08801461c181871b9ff1cb0803b8e9966f083`) to post-vision feature space: global and token-level visual/text signals are computed against instruction-token embeddings, low-entropy instruction tokens provide the local signal, Gaussian spatial smoothing is applied when a valid 2D grid exists, and the resulting nonnegative weights drive weighted facility-location greedy selection. Query-aware.

AnchorPrune. Ranks visual tokens by maximum cosine similarity to instruction tokens, grows an anchor set under a novelty threshold with patience, and completes the budget with farthest-first selection weighted by token centrality. *Disclosure:* the original received-attention prior is unavailable at the post-vision interface, so post-vision feature centrality substitutes for it (with final-ViT received attention as the importance signal on Gemma). It must not be read as an unmodified official implementation.

CDPruner. Conditional-DPP selection: nonnegative quality weights modulate the visual cosine Gram matrix, and fixed-budget Fast MAP-DPP greedy decoding returns exactly the requested count. *Disclosure:* the official pipeline obtains image/text relevance from a CLIP dual encoder, which none of our three models exposes at this pruning point; we substitute mean cosine relevance between each visual token and the instruction-token embeddings in the model's own space. Reported as a documented architecture adaptation, query-aware.

VScan. Captures received-attention mass from an early and the final vision block (pre-merger scores are averaged over each merger group on Qwen). Half the budget comes from the early/local scores with a 4×4 window cap for spatial coverage; the other half from the final/global scores after

Table B1: Scoring/pruning layers for in-LLM methods. On both Qwen models every method scores at decoder layer 3 (the first full-attention layer) and prunes its output before layer 4.

Method	Qwen3-VL-4B / Qwen3.5-4B	Gemma-4-E4B
DART	before layer 4	layer 6
PriorTR	before layer 4	layer 5
FastV	before layer 4	layer 5
MustDrop	before layer 4	layer 5

masking locally selected tokens. On Gemma, dropped tokens are additionally averaged into their nearest retained token. Layer choices and grid handling are adapted per architecture.

VisPruner. Final-vision-block received attention, reduced over merger groups, fills 75% of the budget with the highest-attention tokens; the remainder is filled with redundancy-aware diversity selection in normalized feature space. Selected features are unchanged.

VisionZip. Uses final-vision-block received attention and mean key features: by default 90% of the budget goes to dominant high-attention tokens, contextual representatives are spaced across the rest, and every dropped token is cosine-assigned to a representative whose feature is replaced by the cluster mean. Unlike the fixed selectors, VisionZip can modify retained features through merging.

PruMerge. Neither model family exposes a usable vision CLS token at the pruning point, so importance is the final-vision-block received-attention mass; top-k tokens become representatives, and every dropped token is cosine-assigned to one and averaged into it. A PruMerge- style adaptation rather than the paper's CLS-attention path.

In-LLM methods. DART, PriorTR, FastV, and MustDrop begin prefill with the full visual prefix, score visual tokens at one full-attention decoder layer, then prune. Scoring layers are architecture-specific: Qwen3.5 uses a repeating layout of three linear-attention blocks followed by one full-attention block, so its scoring layers are chosen among full-attention layers rather than copied from LLaMA-era layer numbers; Gemma-4 has hybrid full/sliding-attention layers, and ranking is placed at full-attention layers so the visual span is never outside a sliding window. On both Qwen models all four methods score at layer 3 and prune before layer 4; Gemma defaults are listed in Table B1. Pruning occurs only during prefill: hidden states, attention masks, positions (and Gemma's per-layer vision inputs) and the already-created KV-cache entries are compacted together, and autoregressive decode continues on the compacted cache, so the LLM context length genuinely decreases. All in-LLM runs use batch size 1.

DART. Chooses up to four image and four text pivot states with the highest hidden-state L1 norm, retains visually dissimilar tokens around those pivots using negative cosine similarity (visual pivots first), and completes any unfilled budget by L1 norm. Single-stage final count follows the requested ratio. A deterministic hidden-state adaptation of DART's pivot-and-diversity principle; no attention weights required.

PriorTR. Captures eager attention at the scoring layer and scores each visual token with a directional term $q\log(q/\mathrm{prior})$, where the prior comes from the first post-image text position and q aggregates question-to-visual attention over question tokens; the top-scoring tokens are retained after head aggregation.

FastV. At the scoring layer, attention from the final prompt token to visual keys, averaged over heads, ranks visual tokens; the top k are compacted before the next layer. In our harness FastV is implemented exactly as PriorTR with last-token query aggregation, which is precisely its last-query mean-head rule.

MustDrop. Ranks visual keys by mean attention from all question-text positions after the image span and retains the requested top-k before the next decoder layer: PriorTR's prefill-compaction machinery with MustDrop's cross-modal scorer, as a single-stage query-aware adaptation.

Resize baseline. Resize rewrites the source image before the vision encoder: the requested ratio is a visual-area ratio, the model's native aspect-ratio-preserving target patch grid is computed first,

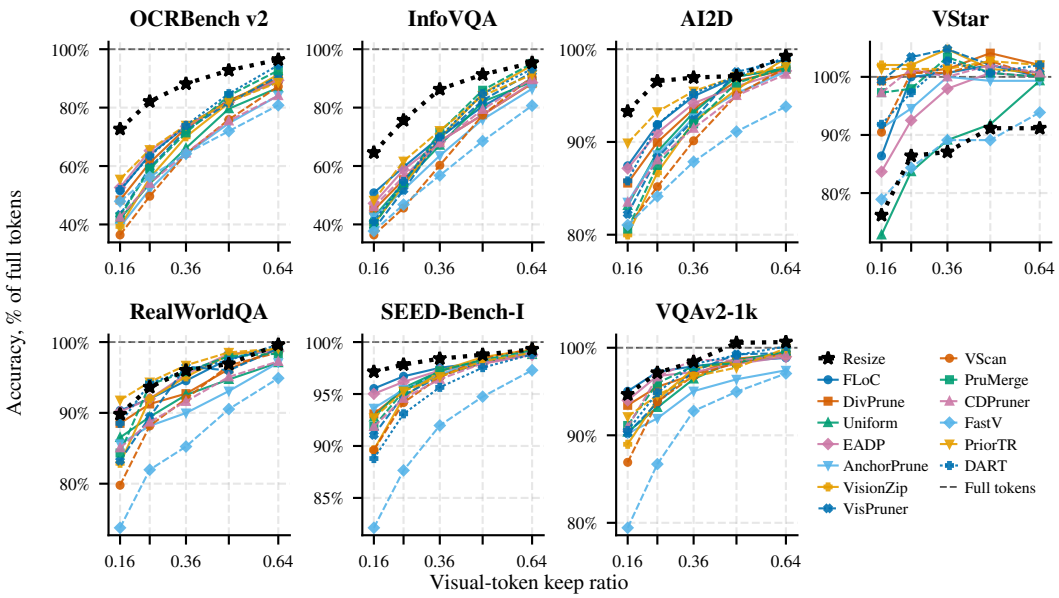

Figure C1: Seven-benchmark accuracy vs. keep ratio on Qwen3-VL-4B. MustDrop has no suite run and is omitted; dashed lines show full-token performance.

and its sides are scaled by $\sqrt{r}$, with target dimensions rounded to the vision pooler's alignment (this avoids encoders that would otherwise upscale a smaller image back to their token budget). Resize is therefore not a subset of the full-resolution token sequence, and it reduces vision-encoder computation as well as LLM prefill. Reports match runs by the logged effective token ratio, since grid alignment can deviate slightly from the requested area ratio.

Efficient evaluation. The ten post-ViT methods share a cached-variant runner: the vision encoder runs once per document, and all method/ratio variants are built from the immutable cache and batched into the LLM by equal compacted sequence length. In-LLM methods reuse the memoized vision pass but re-run the decoder per condition (their scores depend on intermediate LLM state); MustDrop and Resize use direct evaluation paths. The model adapters monkey-patch the Hugging Face classes before loading, so each process evaluates one registered pruning model. Post-ViT methods reduce LLM prefill but not vision-encoder cost, which is why ResizeBench and the suite both price them against resize at identical final-token budgets.

C Detailed Seven-Benchmark Numbers by Benchmark

Tables C1–C3 give absolute scores for every method at every keep ratio on each of the seven standard benchmarks, grouped by ratio; the Full row is the model's uncompressed run. Figures C1 and C2 show the corresponding accuracy curves for Qwen3-VL-4B and Gemma-4-E4B.

D Detailed ResizeBench Numbers by Category

Tables D1–D3 give ResizeBench accuracy per method, overall and split by the five visual-content categories (Table 1). Qwen3-VL-4B's MustDrop suite run is absent (Section 4); its ResizeBench run is complete.

E ResizeBench Conditioned on Full-Resolution Correctness

Throughout the paper we normalize each method to its own model's uncompressed run. That removes differences in overall model strength, but not differences in *composition*: a sample the model already fails at full resolution stays in the denominator, and a compressed variant that happens

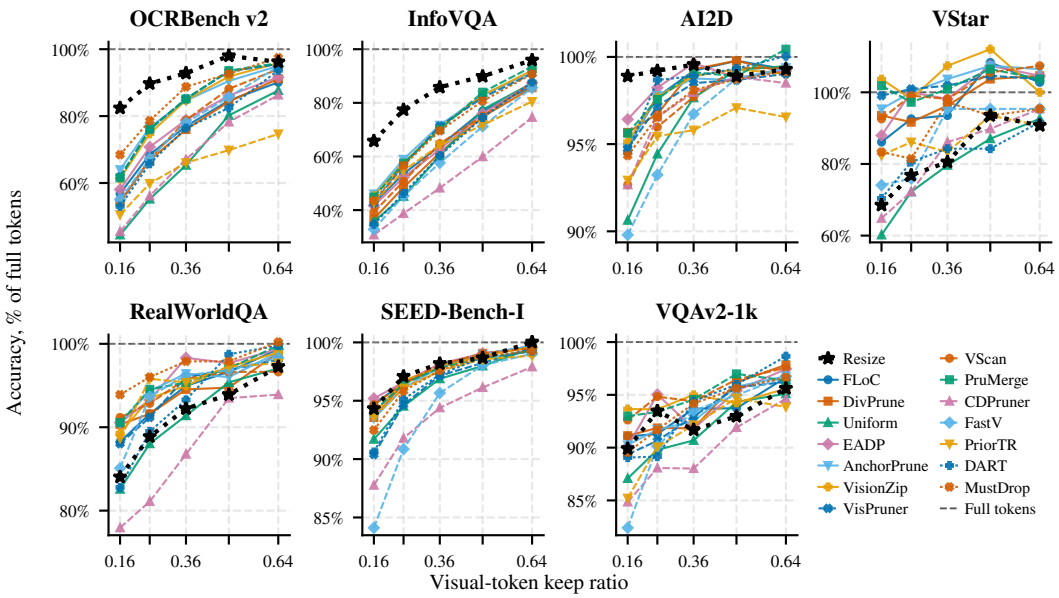

Figure C2: Seven-benchmark accuracy vs. keep ratio on Gemma-4-E4B; dashed lines show full-token performance.

to answer it correctly is credited for a capability the model does not have. The three backbones differ sharply here. Qwen3-VL-4B, the model used to mine the barriers, answers 92.2% of ResizeBench correctly with no compression; Qwen3.5-4B answers 70.9%; Gemma-4-E4B only 48.0%. For Gemma-4, in other words, slightly more than half of ResizeBench consists of questions it cannot answer with every pixel available.

Table E1 isolates that effect. For each model we form the subset A of samples it answers correctly uncompressed and re-score every method on A alone, where Full is 100% by construction. The two columns are related exactly by

$$\text{rel} \;=\; \text{cond} \;+\; \frac{|B|}{|A|}\,\text{acc}_B, \tag{2}$$

with B the complement of A and acc_B the method's accuracy on it, so the inflation is the product of how often the model fails at full resolution and how often a compressed variant lands on the answer anyway. Both factors are model-specific: $|B|/|A|$ is 0.08 for Qwen3-VL, 0.41 for Qwen3.5 and 1.08 for Gemma-4, and acc_B averages 2.1% (excluding the ResizeBench DART outlier discussed at the end of this section), 11.6% and 7.6% respectively. Mean retention over the fourteen selectors consequently falls by 0.3 pp on Qwen3-VL, 4.9 pp on Qwen3.5 and 8.2 pp on Gemma-4.

The consequences run in two opposite directions. First, the *ordering* of methods is untouched. Kendall τ_b between the two columns is 1.00 on Qwen3-VL, 0.99 on Gemma-4 and 0.96 on Qwen3.5, and no method moves by more than one rank: the only movement anywhere is two adjacent swaps on Qwen3.5 (MustDrop with VisPruner, EADP with PruMerge), and no pair involved is separated by more than 0.7 pp in either column. Every method-versus-method claim in Sections 5.3–5.4, including the re-ranking audit, is therefore independent of this choice of normalization.

Second, cross-model tax *magnitudes* do move, in the direction that matters for interpretation. Averaged over the thirteen selectors and the resize baseline that have a suite run on all three models (MustDrop has none on Qwen3-VL), the corresponding compression tax rises from 16.2 to 24.6 pp on Gemma-4, from 28.3 to 33.1 pp on Qwen3.5 and from 36.5 to 36.8 pp on Qwen3-VL. Gemma-4 still pays the smallest tax, but its margin over Qwen3.5 shrinks from 12.1 to 8.5 pp and over Qwen3-VL from 20.2 to 12.2 pp. We therefore read Gemma-4's comparatively gentle tax as limited headroom — it has fewer barrier-sensitive samples left to lose — rather than as evidence of intrinsic robustness to token compression.

Equation 2 also isolates one anomaly. DART on Qwen3-VL-4B scores 21.9% on the full-resolution-*wrong* subset B against 30.4% on A, whereas every other method on that model scores between 0.0% and 3.5% on B. Its corrected standard-suite retention is 89.7%, close to the other selectors, so this discrepancy is specific to ResizeBench rather than evidence of a general configuration failure. We report the anomaly but do not use it to infer DART's mechanism.

F Detailed Leaderboard Divergence

F.1 Per-Benchmark Rank Agreement

Table F1 gives the per-benchmark values behind the left panel of Figure 6. No standard benchmark consistently tracks ResizeBench: the largest agreement is $\tau_b = 0.54$, the combined agreement is non-monotone across budgets, and AI2D and SEEDBench-IMG become negative at keep 0.25. Six of the seven benchmarks recover at some later budget; only $V^\star$ declines throughout.

F.2 Qwen3.5-4B at Keep Ratio 0.49

Reading the rank reversals. The reversals are structured, not random. The standard suite's champion, resize, is worst on ResizeBench because a downsampled image still answers resolution-invariant questions — which is most of a standard suite — while at a sample's own barrier it uniformly destroys the small high-frequency evidence the answer rests on; on both Qwen models it slides from 1st on the suite to last of 16 on ResizeBench (Table 2). The mirror image on Qwen3.5, PriorTR (-9), is question-conditioned: it keeps what *the question* attends to, which on resolution-critical samples is precisely the region resizing smears. Between them the re-rankings split by content category (Appendix D): on webpage text the selector methods' usual advantage evaporates — resize matches or beats most of them on two of three models — while saliency concentrators collapse hardest on exactly this category; on natural scenes the ordering inverts. No strategy wins every category, and which strategy wins which category moves with the model. A single-number suite average hides all of this structure; ResizeBench exposes it because it prices each sample at its own breaking point.

F.3 Method-Level Insights

Backbone dependence. No pruning family is consistently barrier-robust across all three models. PriorTR leads ResizeBench on Qwen3.5 but is second-lowest on Gemma-4, whereas MustDrop leads Gemma-4 and sits below the leading post-ViT methods on both Qwen models. The in-LLM family likewise rises together on Qwen3.5 but splits on the other backbones. These reversals indicate a model–selector interaction rather than a universal advantage for one pruning stage.

Selection behavior. Uniform is last on Qwen3.5 and Gemma-4, while CDPruner also remains near the bottom on both. This suggests that spreading a tight budget broadly can miss the localized evidence that defines a resolution barrier. The category breakdown in Appendix D further shows that the preferred strategy changes with visual content. Because the same-model compression tax varies by roughly 3–5$\times$ across methods (Figure 5), these patterns should be read as interactions among the backbone, selector, and evidence type rather than as a fixed ranking of compression mechanisms.

F.4 Qwen3-VL-4B and Gemma4-E4B

Figures F2 and F3 repeat the Section 5.4 audit — Figure 6 — on the other two backbones, with an identical construction: fifteen approaches on Gemma4-E4B and fourteen on Qwen3-VL-4B, whose MustDrop suite run is absent (Section 4); ResizeBench barrier b matched to keep ratio $(b/100)^2$; every score relative to that model's own full-token run. The three τ_b panels share one axis, so the columns are directly comparable across figures. The displacement panels do not: resize moves 13 ranks on the Qwen models and 3 on Gemma-4, and one shared range would compress the Gemma-4 bars to slivers.

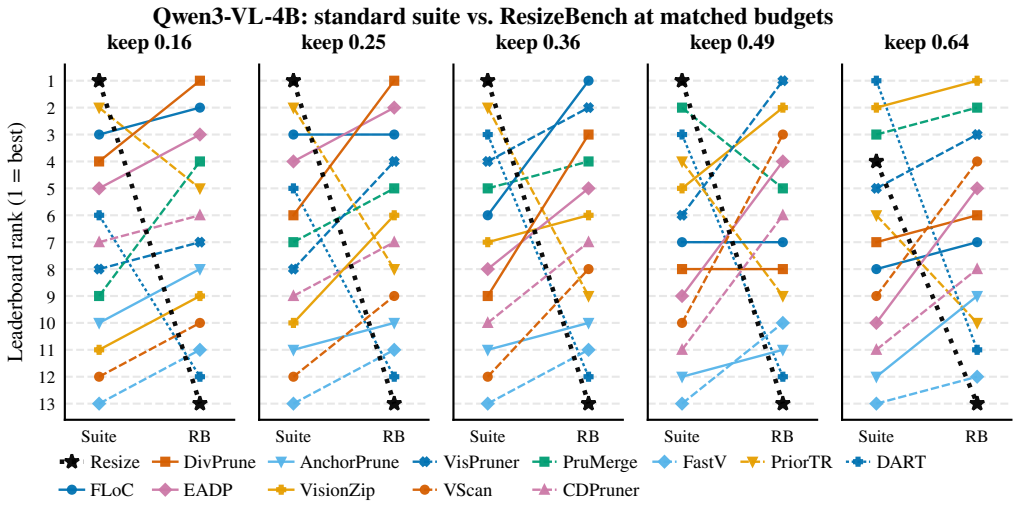

Figure F1: Re-ranking on Qwen3-VL-4B (thirteen approaches; MustDrop has no suite run). Panels and axes as in Figure 7. Resize is 13th of 13 on ResizeBench at every budget while leading the suite at four of five. DART ranks first–sixth on the suite but eleventh–twelfth on ResizeBench, a direct example of the model–selector interaction. 33–54% of the 78 approach pairs reverse (mean 43%). All method colors and legend entries match Figure 1.

Two things survive the model change and one does not. What survives is the weakness of the agreement — no benchmark on any model reaches $\tau_b = 0.75$, and the mean absolute agreement over the 35 benchmark-budget cells is 0.25 on Qwen3.5, 0.19 on Qwen3-VL and 0.36 on Gemma-4 — and resize's direction of travel, positive on all three. What does not survive is *which* approaches the standard suite flatters, and by how much. Resize is the largest mover on both Qwen models ($+13.4$, $+12.4$ ranks) but only the second largest on Gemma-4, at $+2.8$, behind PruMerge; the whole Gemma-4 displacement range is $[-2.8, +3.6]$, an order of magnitude tighter than Qwen3.5's $[-7.8, +13.4]$. PriorTR inverts outright: the largest climber on Qwen3.5 at -7.8, it becomes a *faller* on Qwen3-VL at $+5.0$. DART changes more sharply, climbing on Qwen3.5 and Gemma-4 but falling $+9.2$ ranks on Qwen3-VL. The in-LLM family, which climbs unanimously on Qwen3.5, therefore splits on the other two. The divergence reflects a model–selector interaction, not a fixed property of a pruning stage.

Figure F1 shows the Qwen3-VL re-ranking directly, as Figure 7 does for Qwen3.5. Averaged over the five budgets, 37% of 91 approach pairs reverse on Qwen3.5 and 43% of 78 reverse on Qwen3-VL. The Qwen3-VL fraction ranges from 33% to 54%, reinforcing that no single budget is representative.

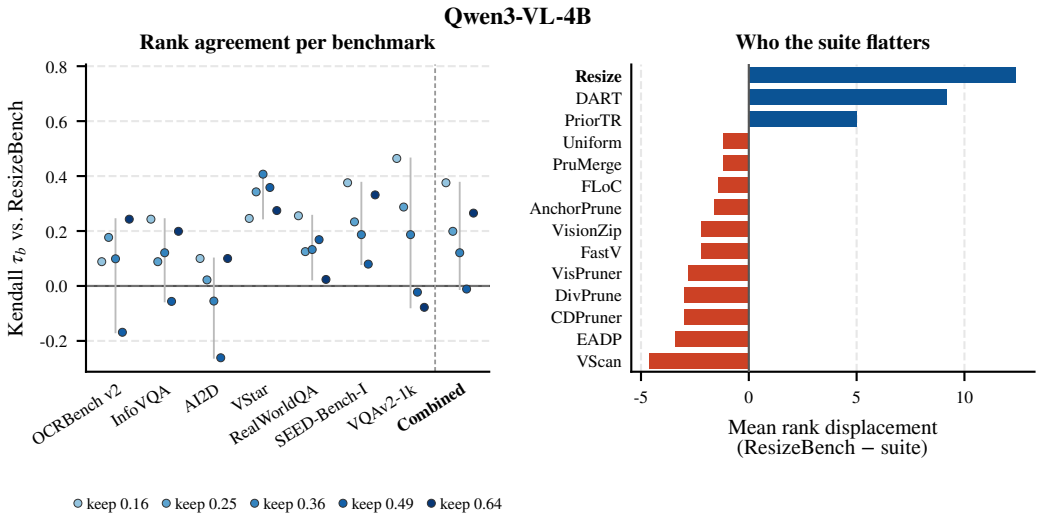

Figure F2: Leaderboard divergence on Qwen3-VL-4B (fourteen approaches; MustDrop has no suite run). Panels and axes as in Figure 6. Resize again moves furthest ($+12.4$ ranks), followed by DART ($+9.2$) and PriorTR ($+5.0$); the largest climber is the post-ViT selector VScan (-4.6).

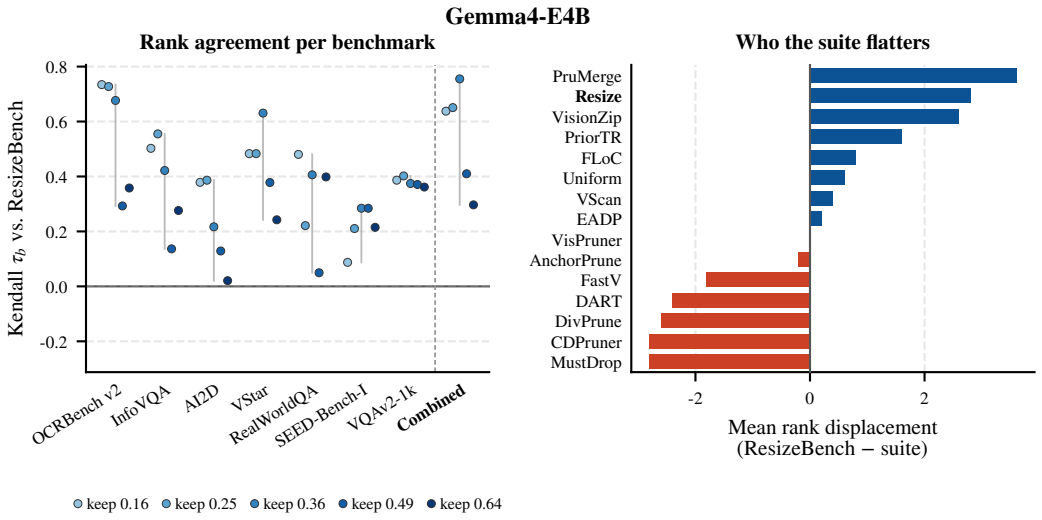

Figure F3: Leaderboard divergence on Gemma4-E4B (fifteen approaches). Panels and axes as in Figure 6. Agreement is the highest of the three models — OCRBench-v2 reaches $\tau_b = 0.73$ at the two tightest budgets — and the displacement range is the narrowest, $[-2.8, +3.6]$: consistent with Appendix E, where 52.0% of ResizeBench is already beyond this backbone at full resolution, leaving less room for the two leaderboards to disagree.

Method	Qwen3.5-VL							
	OCRBench v2	InfoVQA	AI2D	VStar	RealWorldQA	SEED-Bench-I	VQAv2-1k	Avg
keep ratio 0.16								
Resize	35.6	59.6	75.7	66.5	67.1	76.2	76.8	65.3
Post-ViT pruning								
FLoC	22.9	37.7	71.8	66.0	66.4	74.5	76.2	59.4
DivPrune	23.2	36.0	70.4	74.9	66.7	73.5	78.0	60.4
Uniform	18.0	29.6	66.8	61.8	64.6	72.4	73.0	55.2
EADP	22.9	38.4	71.5	68.1	67.8	74.8	76.8	60.1
AnchorPrune	19.1	28.6	67.1	70.2	64.4	72.0	73.6	56.4
VisionZip	20.6	32.2	67.7	61.8	68.6	72.2	74.8	56.8
VisPruner	22.7	31.5	68.9	69.1	67.6	73.1	76.1	58.4
VScan	18.5	28.1	65.4	62.3	66.3	70.9	72.5	54.8
PruMerge	22.8	34.3	69.3	61.8	69.4	73.9	77.4	58.4
CDPruner	19.9	27.5	66.8	61.8	61.0	68.2	70.0	53.6
In-LLM pruning								
FastV	22.7	26.3	49.0	66.5	65.2	63.7	70.2	51.9
PriorTR	24.8	37.1	57.4	73.8	71.0	70.6	78.2	59.0
DART	25.1	29.3	58.8	61.3	61.2	64.4	74.4	53.5
MustDrop	23.7	35.3	54.0	71.2	71.9	68.9	76.8	57.4
keep ratio 0.25								
Resize	39.8	66.4	78.2	71.2	70.1	77.1	79.5	68.9
Post-ViT pruning								
FLoC	28.3	46.9	76.0	74.3	68.8	75.7	77.2	63.9
DivPrune	28.8	46.9	74.9	77.5	69.0	75.3	78.6	64.4
Uniform	22.1	37.8	72.2	69.1	69.4	74.8	76.6	60.3
EADP	28.7	48.1	74.9	72.3	69.8	76.1	78.0	64.0
AnchorPrune	23.7	39.2	72.2	75.4	66.0	74.4	76.3	61.0
VisionZip	27.5	46.0	72.6	66.0	70.3	74.9	77.8	62.2
VisPruner	28.7	46.1	74.6	73.3	70.8	75.4	78.0	63.9
VScan	24.3	37.9	71.5	68.1	71.1	74.4	77.5	60.7
PruMerge	28.8	47.4	75.0	66.0	70.8	75.7	79.0	63.2
CDPruner	23.7	33.7	70.5	68.6	65.6	71.1	73.0	58.0
In-LLM pruning								
FastV	27.4	34.8	50.2	71.2	68.9	66.9	75.0	56.3
PriorTR	28.7	46.2	58.5	74.3	72.2	71.2	80.0	61.6
DART	28.9	39.1	61.2	65.4	64.8	68.4	77.5	57.9
MustDrop	27.9	45.3	55.3	73.3	73.3	69.5	79.1	60.5
keep ratio 0.36								
Resize	42.6	72.3	79.4	72.3	70.3	77.5	79.7	70.6
Post-ViT pruning								
FLoC	32.3	56.4	78.0	72.8	70.7	76.6	78.7	66.5
DivPrune	32.3	57.4	78.0	76.4	70.7	76.3	79.5	67.2
Uniform	27.2	50.1	76.3	71.7	70.3	75.8	77.0	64.1
EADP	32.8	59.6	78.5	72.8	69.8	76.8	78.4	66.9
AnchorPrune	28.4	50.5	76.7	77.0	67.8	75.9	77.2	64.8
VisionZip	33.3	59.0	77.0	69.6	71.6	76.8	79.8	66.7
VisPruner	34.1	59.1	78.3	73.8	72.4	76.9	79.3	67.7
VScan	30.4	51.3	75.5	71.2	71.9	76.1	78.5	65.0
PruMerge	34.4	59.9	78.0	69.6	72.0	77.1	79.8	67.3
CDPruner	27.2	40.7	73.6	72.3	66.4	73.2	75.0	61.2
In-LLM pruning								
FastV	31.3	45.0	52.7	72.3	70.6	68.9	77.9	59.8
PriorTR	32.1	54.6	60.1	75.9	72.5	71.5	80.4	63.9
DART	33.1	50.8	65.3	69.6	66.9	71.5	79.1	62.4
MustDrop	32.4	55.4	56.9	73.3	73.2	70.1	80.1	63.1
keep ratio 0.49								
Resize	44.5	76.2	80.6	74.3	72.3	78.0	81.7	72.5
Post-ViT pruning								
FLoC	36.3	66.6	80.1	76.4	71.4	77.1	79.4	69.6
DivPrune	37.1	66.6	80.0	76.4	72.2	77.1	80.3	70.0
Uniform	33.4	62.8	79.0	71.7	70.2	77.2	79.0	67.6
EADP	38.0	68.3	79.9	74.3	71.1	77.3	79.8	69.8
AnchorPrune	33.3	60.4	78.8	77.5	70.1	76.7	78.5	67.9
VisionZip	38.4	69.2	80.0	72.8	72.7	77.5	80.5	70.1
VisPruner	39.3	69.1	80.3	76.4	72.9	77.6	80.3	70.9
VScan	35.7	63.0	78.9	73.8	73.2	77.3	79.7	68.8
PruMerge	40.4	71.3	80.4	72.8	73.3	77.7	80.4	70.9
CDPruner	31.4	48.2	76.9	75.4	68.5	75.2	77.0	64.6
In-LLM pruning								
FastV	36.0	56.5	55.4	74.3	72.9	69.9	80.0	63.6
PriorTR	35.4	62.4	60.3	73.8	72.2	71.7	81.3	65.3
DART	37.2	63.0	69.2	71.7	70.3	74.0	80.5	66.6
MustDrop	37.1	64.2	58.2	73.8	74.2	70.2	81.0	65.5
keep ratio 0.64								
Resize	46.1	77.7	81.2	76.4	73.9	78.0	81.4	73.5
Post-ViT pruning								
FLoC	41.0	72.7	80.8	76.4	72.7	77.8	80.3	71.7
DivPrune	41.0	72.3	81.1	77.5	73.5	77.7	80.6	71.9
Uniform	37.4	69.6	80.7	73.8	72.0	77.8	79.7	70.2
EADP	41.9	74.1	81.3	75.9	72.0	77.9	81.1	72.1
AnchorPrune	38.6	69.2	80.6	77.0	72.0	77.4	80.0	70.7
VisionZip	43.4	76.0	80.8	72.8	73.3	77.9	80.9	72.2
VisPruner	42.8	75.1	81.1	76.4	73.7	77.9	80.7	72.5
VScan	41.2	73.0	80.5	75.9	73.5	77.9	80.8	71.8
PruMerge	43.9	76.7	81.0	74.9	74.0	78.0	80.7	72.7
CDPruner	35.9	59.0	78.4	75.9	71.0	76.4	78.4	67.9
In-LLM pruning								
FastV	40.5	67.0	57.5	73.8	74.5	70.3	81.2	66.4
PriorTR	38.2	66.9	60.5	74.3	72.9	71.8	81.2	66.6
DART	41.4	72.1	71.0	73.3	72.9	75.0	81.0	69.5
MustDrop	41.1	71.6	59.3	73.8	73.6	70.4	81.2	67.3
Full (no compression)	47.2	79.7	81.5	77.5	73.3	78.2	81.2	74.1

Table C1: Seven-benchmark suite accuracy (%) per method and keep ratio on Qwen3.5-4B. Avg is the macro average over the seven benchmarks. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Method	Qwen3-VL							
	OCRBench v2	InfoVQA	AI2D	VStar	RealWorldQA	SEED-Bench-I	VQAv2-1k	Avg
keep ratio 0.16								
Resize	33.7	50.8	76.2	58.6	64.4	76.6	77.2	62.5
Post-ViT pruning								
FLoC	23.9	40.0	71.4	66.5	64.8	75.3	77.5	59.9
DivPrune	22.4	35.3	69.9	76.4	63.5	73.4	76.2	59.6
Uniform	19.4	33.5	67.9	56.0	62.1	73.2	73.7	55.1
EADP	24.4	37.1	71.2	64.4	64.7	74.9	76.6	59.1
AnchorPrune	18.2	33.4	68.2	70.2	61.4	73.8	73.6	57.0
VisionZip	18.2	31.6	65.3	78.5	59.5	70.7	72.5	56.6
VisPruner	20.1	30.1	67.2	76.4	59.7	71.8	73.6	57.0
VScan	16.9	28.6	66.0	69.6	57.3	70.7	70.9	54.3
PruMerge	19.8	31.3	65.8	74.9	60.5	72.4	74.3	57.0
CDPruner	19.7	36.3	68.2	74.9	61.0	72.4	74.1	58.1
In-LLM pruning								
FastV	22.2	29.7	66.2	60.7	52.9	64.8	64.8	51.6
PriorTR	25.7	37.9	73.4	78.0	65.9	73.1	75.1	61.3
DART	24.1	32.2	70.1	70.7	63.5	70.0	73.8	57.8
keep ratio 0.25								
Resize	38.1	59.5	78.9	66.5	67.2	77.1	79.2	66.6
Post-ViT pruning								
FLoC	29.6	46.9	75.0	77.0	66.0	76.2	79.3	64.3
DivPrune	28.3	43.5	73.5	77.5	65.5	75.2	77.9	63.1
Uniform	25.0	42.6	72.6	64.4	64.2	75.4	76.0	60.0
EADP	30.2	46.2	74.3	71.2	66.0	75.8	78.9	63.2
AnchorPrune	24.1	40.8	72.4	72.8	63.3	75.2	75.0	60.5
VisionZip	25.9	42.0	70.8	78.5	66.0	74.6	76.7	62.1
VisPruner	27.6	40.7	72.1	79.6	64.2	74.8	76.4	62.2
VScan	23.0	35.9	69.6	77.5	63.3	74.3	76.5	60.0
PruMerge	27.3	42.7	71.4	75.4	67.5	75.1	78.2	62.5
CDPruner	25.1	44.3	71.9	78.0	63.7	74.6	78.8	62.3
In-LLM pruning								
FastV	26.1	36.8	68.7	64.9	58.8	69.1	70.7	56.5
PriorTR	30.4	48.5	76.2	78.0	67.7	75.1	77.1	64.7
DART	29.4	43.0	75.0	74.9	67.6	73.4	77.3	62.9
keep ratio 0.36								
Resize	40.9	67.9	79.2	67.0	68.9	77.6	80.2	68.8
Post-ViT pruning								
FLoC	34.1	55.0	77.5	77.5	67.8	76.9	79.8	67.0
DivPrune	33.8	53.9	76.4	78.0	66.5	76.4	79.0	66.3
Uniform	30.7	52.8	76.1	68.6	66.4	76.6	78.6	64.3
EADP	34.1	53.8	76.9	75.4	68.5	76.7	79.3	66.4
AnchorPrune	29.7	50.0	75.6	77.0	64.6	76.4	77.5	64.4
VisionZip	32.5	54.7	75.3	80.6	68.2	76.6	79.6	66.8
VisPruner	33.5	54.5	75.5	80.6	69.3	76.3	79.3	67.0
VScan	29.8	47.4	73.6	77.5	66.0	76.2	78.9	64.2
PruMerge	33.1	56.3	75.1	79.6	68.8	76.7	79.8	67.0
CDPruner	29.9	53.4	74.7	77.0	65.8	76.0	80.4	65.3
In-LLM pruning								
FastV	29.8	44.7	71.8	68.6	61.2	72.5	75.6	60.6
PriorTR	34.3	56.7	78.0	78.0	69.4	76.2	78.8	67.4
DART	34.3	55.0	77.8	79.1	68.9	75.4	80.0	67.2
keep ratio 0.49								
Resize	43.0	71.8	79.3	70.2	69.5	77.9	82.0	70.5
Post-ViT pruning								
FLoC	38.2	63.2	79.2	78.5	70.1	77.3	80.3	69.6
DivPrune	38.2	61.0	78.9	80.1	69.0	77.2	80.5	69.3
Uniform	36.9	64.5	79.3	70.7	68.0	77.5	80.4	68.2
EADP	38.0	61.0	78.4	77.5	70.6	77.2	80.0	69.0
AnchorPrune	34.6	59.7	77.8	76.4	66.8	77.4	78.6	67.3
VisionZip	37.8	66.7	78.1	78.0	70.5	77.7	80.3	69.9
VisPruner	37.9	64.9	79.7	78.5	68.9	77.3	80.9	69.7
VScan	35.2	60.9	77.6	79.1	69.4	77.6	80.1	68.6
PruMerge	39.0	67.6	78.7	77.5	70.3	77.5	80.1	70.1
CDPruner	35.0	62.4	77.6	78.5	68.2	77.2	80.3	68.5
In-LLM pruning								
FastV	33.4	53.9	74.4	68.6	65.0	74.7	77.4	63.9
PriorTR	38.0	66.1	79.3	79.1	70.7	77.4	79.7	70.0
DART	39.3	66.6	79.3	77.5	69.9	76.9	80.8	70.1
keep ratio 0.64								
Resize	44.7	75.0	81.1	70.2	71.5	78.3	82.1	71.8
Post-ViT pruning								
FLoC	41.3	70.7	80.0	77.0	71.1	78.0	80.6	71.2
DivPrune	41.6	69.4	79.9	78.5	71.1	78.0	80.9	71.4
Uniform	40.2	69.9	80.0	76.4	69.7	77.9	81.1	70.7
EADP	41.4	69.7	79.8	77.0	70.8	77.9	80.7	71.0
AnchorPrune	39.0	68.0	79.7	76.4	69.7	78.0	79.4	70.0
VisionZip	42.4	74.3	80.9	77.5	71.1	78.3	80.9	72.2
VisPruner	42.1	72.5	80.8	77.5	71.1	78.1	81.0	71.9
VScan	40.4	71.2	80.1	77.0	70.7	78.3	81.1	71.2
PruMerge	43.0	74.8	80.1	77.0	70.6	78.2	81.2	72.1
CDPruner	39.0	70.8	79.5	77.5	69.8	78.1	80.9	70.8
In-LLM pruning								
FastV	37.5	63.5	76.6	72.3	68.1	76.7	79.1	67.7
PriorTR	41.1	71.8	80.2	78.5	71.1	78.0	81.5	71.8
DART	43.8	73.3	81.0	78.5	71.5	77.9	81.7	72.5
Full (no compression)	46.4	78.7	81.7	77.0	71.8	78.8	81.5	73.7

Table C2: Seven-benchmark suite accuracy (%) per method and keep ratio on Qwen3-VL-4B. Avg is the macro average over the seven benchmarks. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Method	Gemma-4							
	OCRBench v2	InfoVQA	AI2D	VStar	RealWorldQA	SEED-Bench-I	VQAv2-1k	Avg
keep ratio 0.16								
Resize	29.9	41.4	72.5	38.7	52.3	71.6	56.4	51.8
Post-ViT pruning								
FLoC	20.6	26.8	69.5	48.7	54.9	71.9	57.1	49.9
DivPrune	19.7	23.1	70.1	52.9	56.2	71.0	57.1	50.0
Uniform	16.1	22.5	66.4	34.0	51.4	69.6	54.6	45.0
EADP	21.0	26.2	70.7	49.7	56.3	72.3	56.5	50.4
AnchorPrune	23.2	29.0	69.5	53.9	56.3	71.0	56.6	51.4
VisionZip	22.2	28.5	69.7	58.6	55.6	71.3	58.7	52.1
VisPruner	22.4	26.8	69.4	56.0	54.8	68.6	56.0	50.6
VScan	19.6	24.7	67.9	52.4	56.7	71.9	58.1	50.2
PruMerge	22.3	28.2	70.1	57.6	56.3	71.5	58.3	52.0
CDPruner	16.5	19.3	67.9	36.6	48.5	66.6	53.2	44.1
In-LLM pruning								
FastV	19.9	20.7	65.8	41.9	52.9	63.9	51.7	45.3
PriorTR	18.3	26.9	68.1	46.6	55.2	71.1	53.4	48.5
DART	19.2	21.8	69.3	39.8	51.5	68.8	55.8	46.6
MustDrop	24.8	27.4	69.1	47.1	58.4	70.2	56.1	50.5
keep ratio 0.25								
Resize	32.5	48.7	72.7	43.5	55.3	73.7	58.6	55.0
Post-ViT pruning								
FLoC	24.8	33.7	71.6	52.4	57.0	73.3	57.2	52.9
DivPrune	24.2	30.5	70.9	51.8	57.0	73.3	57.6	52.2
Uniform	20.0	28.4	69.2	40.8	54.8	71.8	56.3	48.8
EADP	25.6	32.8	72.0	56.5	58.0	73.3	59.6	54.0
AnchorPrune	27.8	37.1	71.7	56.5	58.7	73.0	57.5	54.6
VisionZip	27.1	36.2	71.6	55.5	58.2	73.5	58.7	54.4
VisPruner	27.5	35.7	71.2	57.1	56.7	71.9	56.8	53.9
VScan	24.3	31.9	70.3	56.0	57.8	73.3	59.5	53.3
PruMerge	27.6	36.2	71.5	55.0	58.8	73.2	58.5	54.4
CDPruner	20.3	24.4	70.8	40.8	50.5	69.7	55.2	47.4
In-LLM pruning								
FastV	24.4	28.5	68.3	42.9	58.3	69.0	56.5	49.7
PriorTR	21.7	34.4	69.9	48.7	59.6	73.0	56.5	52.0
DART	23.8	29.1	72.3	45.5	55.7	72.2	55.9	50.7
MustDrop	28.5	35.6	70.8	46.1	59.7	72.8	57.5	53.0
keep ratio 0.36								
Resize	33.6	54.0	73.0	45.5	57.4	74.6	57.5	56.5
Post-ViT pruning								
FLoC	27.9	40.2	72.7	52.9	59.1	74.3	58.7	55.1
DivPrune	27.5	38.6	72.8	54.5	58.8	74.3	57.6	54.9
Uniform	23.7	37.9	71.6	45.0	56.9	73.6	56.9	52.2
EADP	28.6	39.4	73.0	55.5	61.2	74.5	57.5	55.7
AnchorPrune	30.7	45.1	72.4	58.6	60.0	74.3	58.0	57.0
VisionZip	30.7	44.5	72.5	60.7	59.1	74.3	59.6	57.3
VisPruner	30.9	44.9	72.2	57.6	59.7	73.8	58.3	56.8
VScan	28.5	40.8	71.6	55.5	59.6	74.5	59.2	55.7
PruMerge	30.8	44.1	72.6	57.1	59.3	74.3	59.3	56.8
CDPruner	24.3	30.4	71.8	48.7	54.0	71.7	55.2	50.9
In-LLM pruning								
FastV	28.3	36.3	70.9	53.9	59.9	72.6	58.6	54.4
PriorTR	23.9	40.4	70.2	47.1	59.3	74.1	57.8	53.3
DART	27.7	37.9	72.5	47.6	58.0	73.8	59.2	53.8
MustDrop	32.1	43.8	71.9	55.0	60.9	74.1	59.0	56.7
keep ratio 0.49								
Resize	35.5	56.6	72.5	52.9	58.4	74.9	58.3	58.4
Post-ViT pruning								
FLoC	30.7	48.4	73.1	61.3	60.1	75.2	58.8	58.2
DivPrune	30.5	47.1	73.1	58.6	59.0	75.2	60.3	57.7
Uniform	29.1	48.8	72.7	49.2	59.3	74.4	59.1	56.1
EADP	31.2	47.8	72.4	60.7	60.8	75.2	59.8	58.3
AnchorPrune	32.8	52.1	72.3	60.7	59.7	75.2	60.0	59.0
VisionZip	33.3	52.1	72.8	63.4	60.5	75.1	59.1	59.5
VisPruner	33.8	51.6	72.3	59.2	60.3	74.6	60.0	58.8
VScan	31.9	47.2	72.8	59.7	60.1	75.2	60.5	58.2
PruMerge	33.9	52.8	72.5	60.2	60.4	74.9	60.8	59.4
CDPruner	28.3	37.7	72.4	50.8	58.2	73.0	57.6	54.0
In-LLM pruning								
FastV	31.0	44.8	72.4	53.9	60.3	74.5	59.5	56.6
PriorTR	25.3	45.1	71.1	52.9	60.5	74.8	59.4	55.6
DART	29.9	46.9	72.8	47.6	61.4	75.0	60.2	56.3
MustDrop	33.6	50.8	72.3	52.9	60.9	75.0	60.0	57.9
keep ratio 0.64								
Resize	34.8	60.4	72.8	51.3	60.5	76.0	59.9	59.4
Post-ViT pruning								
FLoC	32.6	54.2	72.8	58.1	61.8	75.7	60.6	59.4
DivPrune	33.0	53.9	72.6	59.2	61.6	75.6	61.3	59.6
Uniform	31.7	55.0	72.9	52.4	60.4	75.5	59.6	58.2
EADP	33.2	55.1	72.7	59.2	61.8	75.6	61.1	59.8
AnchorPrune	34.4	57.6	72.8	60.2	61.6	75.7	60.7	60.4
VisionZip	34.7	57.9	72.6	56.5	61.6	75.6	59.9	59.8
VisPruner	34.6	57.2	72.6	58.6	61.0	75.6	60.3	60.0
VScan	34.1	55.0	72.8	60.7	60.1	75.6	61.2	59.9
PruMerge	34.7	58.8	73.6	58.6	62.1	75.4	60.4	60.5
CDPruner	31.3	46.9	72.2	53.9	58.4	74.3	59.3	56.6
In-LLM pruning								
FastV	34.4	53.9	72.8	53.9	61.2	75.2	60.7	58.9
PriorTR	27.0	50.7	70.8	51.3	60.8	75.1	58.8	56.4
DART	33.9	55.2	73.3	51.8	62.1	75.4	61.9	59.1
MustDrop	35.3	57.1	72.7	53.9	62.4	75.4	60.6	59.6
Full (no compression)	36.2	62.9	73.3	56.5	62.2	75.9	62.7	61.4

Table C3: Seven-benchmark suite accuracy (%) per method and keep ratio on Gemma4-E4B. Avg is the macro average over the seven benchmarks. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Method	Qwen3.5-VL					
	Overall	Natural scene	Text-in-scene	Webpage	Chart/figure	Real doc
Post-ViT pruning						
FLoC	48.1	54.5	67.3	20.3	24.0	12.2
DivPrune	50.3	59.2	69.8	15.0	20.8	14.6
Uniform	35.0	41.8	44.0	11.9	20.8	0.0
EADP	48.9	55.9	67.7	19.4	24.0	12.2
AnchorPrune	42.5	49.9	56.9	15.9	16.7	12.2
VisionZip	44.3	49.8	60.5	21.6	26.0	2.4
VisPruner	50.1	57.9	66.1	22.0	20.8	17.1
VScan	40.1	45.0	56.9	18.9	20.8	0.0
PruMerge	49.5	56.4	63.7	24.7	28.1	7.3
CDPruner	36.1	43.0	52.8	7.0	13.5	4.9
In-LLM pruning						
FastV	42.2	48.2	60.5	18.5	12.5	9.8
PriorTR	52.8	64.4	66.1	14.5	22.9	14.6
DART	46.6	52.8	65.3	20.7	19.8	12.2
MustDrop	49.9	60.5	61.3	18.1	16.7	17.1

Table D1: ResizeBench accuracy (%) overall and by visual-content category on Qwen3.5-4B (all barriers pooled). Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Method	Qwen3-VL					
	Overall	Natural scene	Text-in-scene	Webpage	Chart/figure	Real doc
Post-ViT pruning						
FLoC	57.3	64.9	76.6	29.5	24.0	14.6
DivPrune	58.0	64.5	79.0	30.4	27.1	19.5
Uniform	40.7	47.8	46.0	21.1	24.0	9.8
EADP	56.9	63.5	74.6	32.6	25.0	24.4
AnchorPrune	43.8	53.0	48.8	17.6	27.1	7.3
VisionZip	51.4	58.6	68.1	22.0	30.2	12.2
VisPruner	55.6	64.3	76.6	22.5	18.8	17.1
VScan	45.7	52.0	63.3	18.9	24.0	9.8
PruMerge	55.2	63.6	74.2	23.8	24.0	14.6
CDPruner	51.8	60.4	69.0	18.5	24.0	19.5
In-LLM pruning						
FastV	35.1	42.4	36.3	19.4	15.6	9.8
PriorTR	49.9	61.5	59.3	14.1	21.9	19.5
DART	29.7	36.3	31.5	10.1	19.8	14.6
MustDrop	47.3	57.8	62.9	8.8	17.7	17.1

Table D2: ResizeBench accuracy (%) overall and by visual-content category on Qwen3-VL-4B (all barriers pooled). Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Method	Gemma-4					
	Overall	Natural scene	Text-in-scene	Webpage	Chart/figure	Real doc
Post-ViT pruning						
FLoC	35.4	36.9	49.2	25.6	24.0	2.4
DivPrune	37.5	40.0	51.2	23.3	27.1	7.3
Uniform	24.9	28.4	34.7	7.9	19.8	0.0
EADP	37.4	39.1	50.0	28.2	25.0	7.3
AnchorPrune	39.9	39.2	53.6	36.1	31.2	12.2
VisionZip	38.0	38.8	51.2	30.0	29.2	7.3
VisPruner	38.6	38.4	50.4	34.8	32.3	7.3
VScan	37.2	40.1	52.8	19.8	27.1	4.9
PruMerge	39.1	39.1	50.4	36.1	30.2	7.3
CDPruner	28.3	28.8	41.9	19.4	18.8	4.9
In-LLM pruning						
FastV	34.0	35.8	51.2	18.1	18.8	14.6
PriorTR	25.9	30.1	31.0	12.8	14.6	7.3
DART	34.7	35.3	48.8	28.2	19.8	7.3
MustDrop	40.3	41.7	53.2	31.3	27.1	12.2

Table D3: ResizeBench accuracy (%) overall and by visual-content category on Gemma4-E4B (all barriers pooled). Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Method	Qwen3-VL ($\lvert A\rvert = 1344$) Rel.	Cond.	Δr	Qwen3.5-VL ($\lvert A\rvert = 1034$) Rel.	Cond.	Δr	Gemma-4 ($\lvert A\rvert = 700$) Rel.	Cond.	Δr
Resize	17.6	17.6	0	31.0	27.5	0	79.4	70.0	0
Post-ViT pruning									
FLoC	62.1	61.9	0	67.9	64.1	0	73.7	66.9	0
DivPrune	62.9	62.9	0	71.0	66.1	0	78.1	69.1	0
Uniform	44.1	44.0	0	49.3	45.5	0	51.9	45.1	0
EADP	61.8	61.5	0	69.0	64.5	+1	78.0	69.1	0
AnchorPrune	47.5	47.3	0	60.0	55.4	0	83.1	74.3	0
VisionZip	55.7	55.4	0	62.5	56.8	0	79.1	70.0	0
VisPruner	60.3	60.1	0	70.7	65.1	-1	80.4	70.9	0
VScan	49.6	49.6	0	56.6	52.5	0	77.6	67.7	0
PruMerge	59.9	59.7	0	69.7	64.2	-1	81.4	73.1	0
CDPruner	56.2	56.0	0	50.9	46.2	0	58.9	50.7	0
In-LLM pruning									
FastV	38.1	37.9	0	59.6	54.2	0	70.7	63.7	0
PriorTR	54.2	54.0	0	74.5	68.9	0	54.0	46.6	0
DART	32.2	30.4	0	65.8	60.7	0	72.3	64.4	0
MustDrop	51.3	51.0	0	70.4	65.6	+1	83.9	77.1	0
Mean, 14 selectors	52.6	52.3	—	64.1	59.3	—	73.1	64.9	—

Table E1: **ResizeBench accuracy conditioned on full-resolution correctness.** For each model, A is the set of ResizeBench samples that model answers correctly with no compression, out of 1,458 total. *Rel.* is accuracy relative to the model's own uncompressed run over all samples, the normalization used throughout the main text; *Cond.* is accuracy on A alone, where Full scores 100% by construction; Δr is the change in rank among the 15 approaches when moving from the first column to the second. Conditioning lowers every retention figure, by an amount that depends on the backbone, but leaves the ordering essentially untouched (Qwen3-VL-4B $\tau_b = 1.00$; Qwen3.5-4B $\tau_b = 0.96$; Gemma4-E4B $\tau_b = 0.99$), so no comparison between methods in the main text depends on it. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Table F1: Kendall τ_b between each standard benchmark's leaderboard and the matched-budget ResizeBench leaderboard on Qwen3.5-4B. The evenly spaced sampling control is included here, while Figure 7 and Table F2 use only the 14 published approaches. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

Standard benchmark	keep 0.16	0.25	0.36	0.49	0.64
OCRBench-v2	0.42	0.36	0.34	0.21	0.37
InfoVQA	0.31	0.29	0.36	0.26	0.31
AI2D	0.03	−0.10	0.03	0.15	0.14
V*	0.49	0.23	0.07	0.04	0.00
RealWorldQA	0.33	0.24	0.46	0.47	0.33
SEEDBench-IMG	0.02	−0.04	0.13	0.17	0.17
VQAv2	0.54	0.52	0.48	0.15	0.19
Combined	0.34	0.27	0.15	0.26	0.25

Method	Std.	Std. rank	RB	RB rank	Displacement
Resize	97.7	**1**	52.5	14	**+13** [+8, +13]
Post-ViT pruning					
VisPruner	95.0	3	85.0	**1**	−2 [−2, +2]
VScan	91.9	8	83.7	2	−6 [−7, −2]
PruMerge	95.1	2	83.7	3	+1 [−1, +4]
DivPrune	93.6	5	77.5	5	+0 [−3, +3]
VisionZip	93.9	4	77.5	6	+2 [−2, +6]
FLoC	93.0	7	67.5	8	+1 [−1, +5]
EADP	93.5	6	67.5	9	+3 [+0, +7]
AnchorPrune	90.5	9	67.5	10	+1 [−4, +4]
CDPruner	86.1	13	55.0	13	+0 [−2, +1]
In-LLM pruning					
PriorTR	87.5	12	81.2	4	**−8** [−11, −5]
MustDrop	88.1	11	70.0	7	−4 [−6, +1]
FastV	85.5	14	66.2	11	−3 [−8, −1]
DART	89.3	10	65.0	12	+2 [−3, +3]

Table F2: Both leaderboards at keep ratio 0.49 (ResizeBench barrier-70 cohort, 132 samples; Qwen3.5-4B, the fourteen published approaches — the evenly spaced sampling control is excluded from both leaderboards, as in Figure 7), grouped by pruning stage. This is the keep-0.49 panel of Figure 7 with the scores behind it. Displayed ranks retain the global ResizeBench ordering within each group. Scores are accuracy relative to the full-token run of the same suite (ResizeBench full 60.6, suite full 74.1). Displacement = ResizeBench rank − suite rank. Intervals are 95% bootstrap over the 132 ResizeBench samples with the suite ranks held fixed: they cover ResizeBench sampling noise only, because the standard-suite runs are logged as aggregates and carry no per-sample records to resample against. Two cautions on reading single rows. One sample is 1.2 pp of the relative score here, so scores tie often — 3 groups of approaches share a displayed ResizeBench score — and ties are separated by the fixed method order Figures 6 and 7 use, so adjacent ranks within a tie are not a measured difference. And only 4 of the 14 intervals exclude zero (Resize, VScan, FastV, PriorTR); the section's claims rest on the displacement averaged over five budgets (Figure 6) and on τ_b, not on any single row of this table. Dark/light green marks the highest/second-highest; dark/light red marks the lowest/second-lowest among compared entries.

G ADDITIONAL RESIZEBENCH EXAMPLES

Figures G1–G5 show five additional samples at each resolution tier. The main-paper examples are excluded. Within each box, the original and barrier images have equal display size; the labels report their true pixel resolutions.

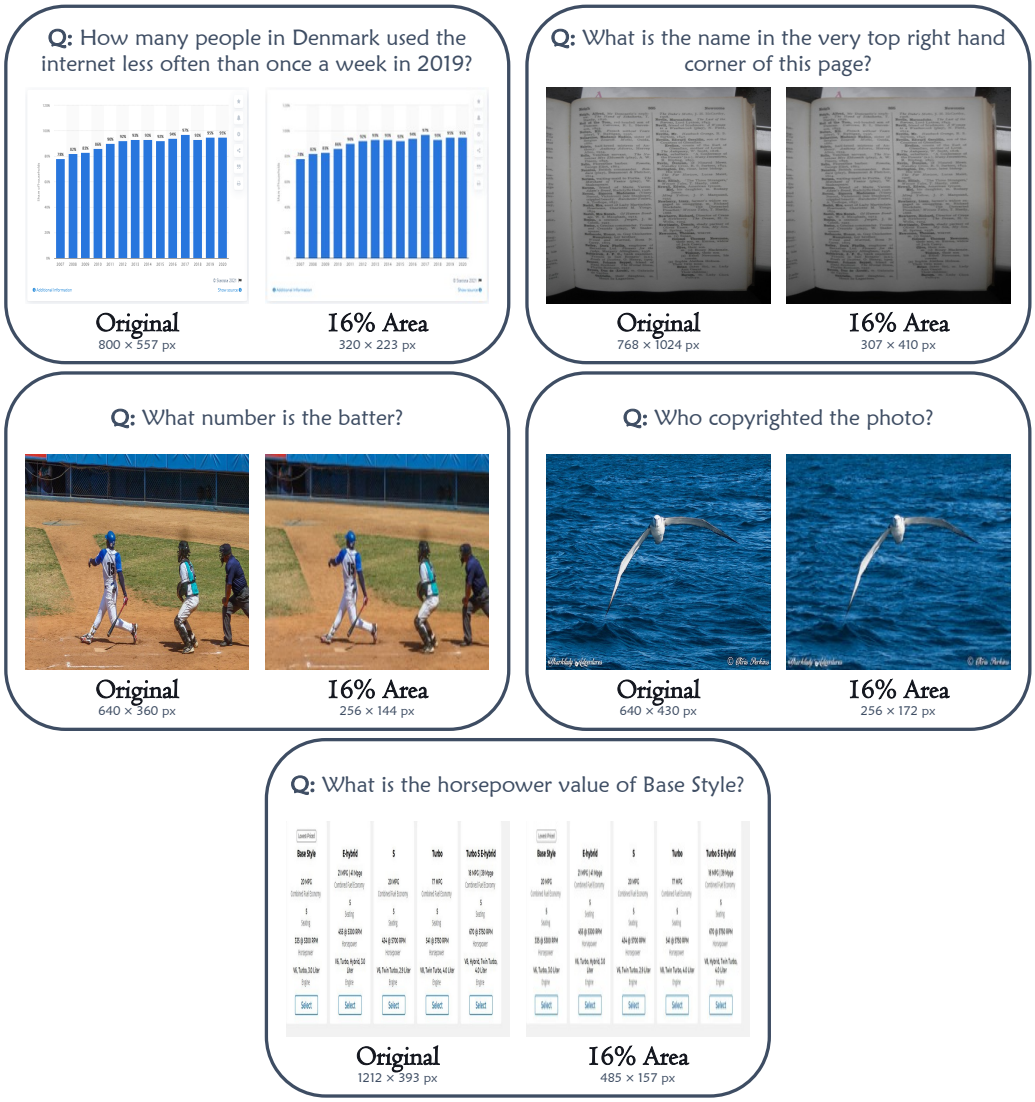

Figure G1: Additional ResizeBench samples at the 0.16 area budget, corresponding to a 40% side-scale barrier.



Figure G2: Additional ResizeBench samples at the 0.25 area budget, corresponding to a 50% side-scale barrier.

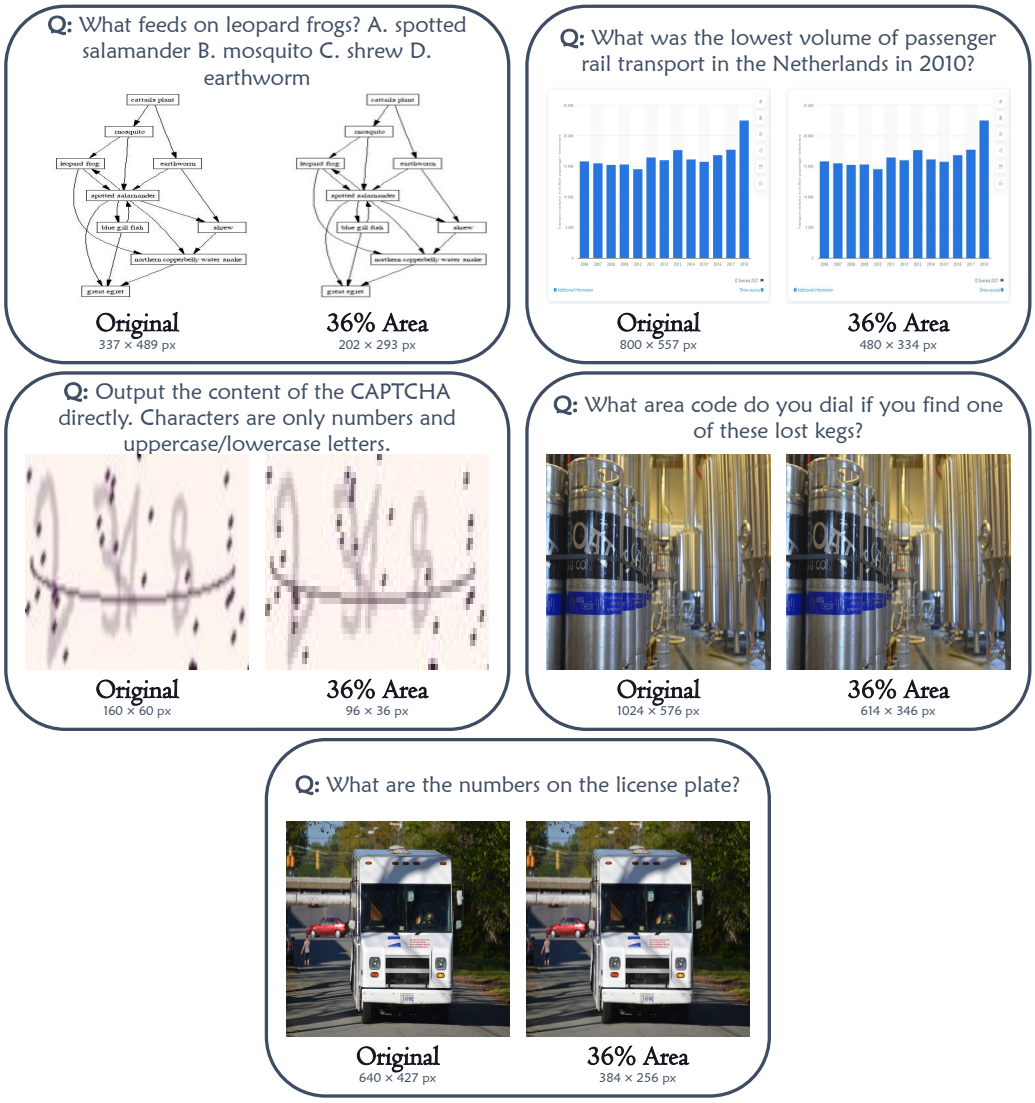

Figure G3: Additional ResizeBench samples at the 0.36 area budget, corresponding to a 60% side-scale barrier.

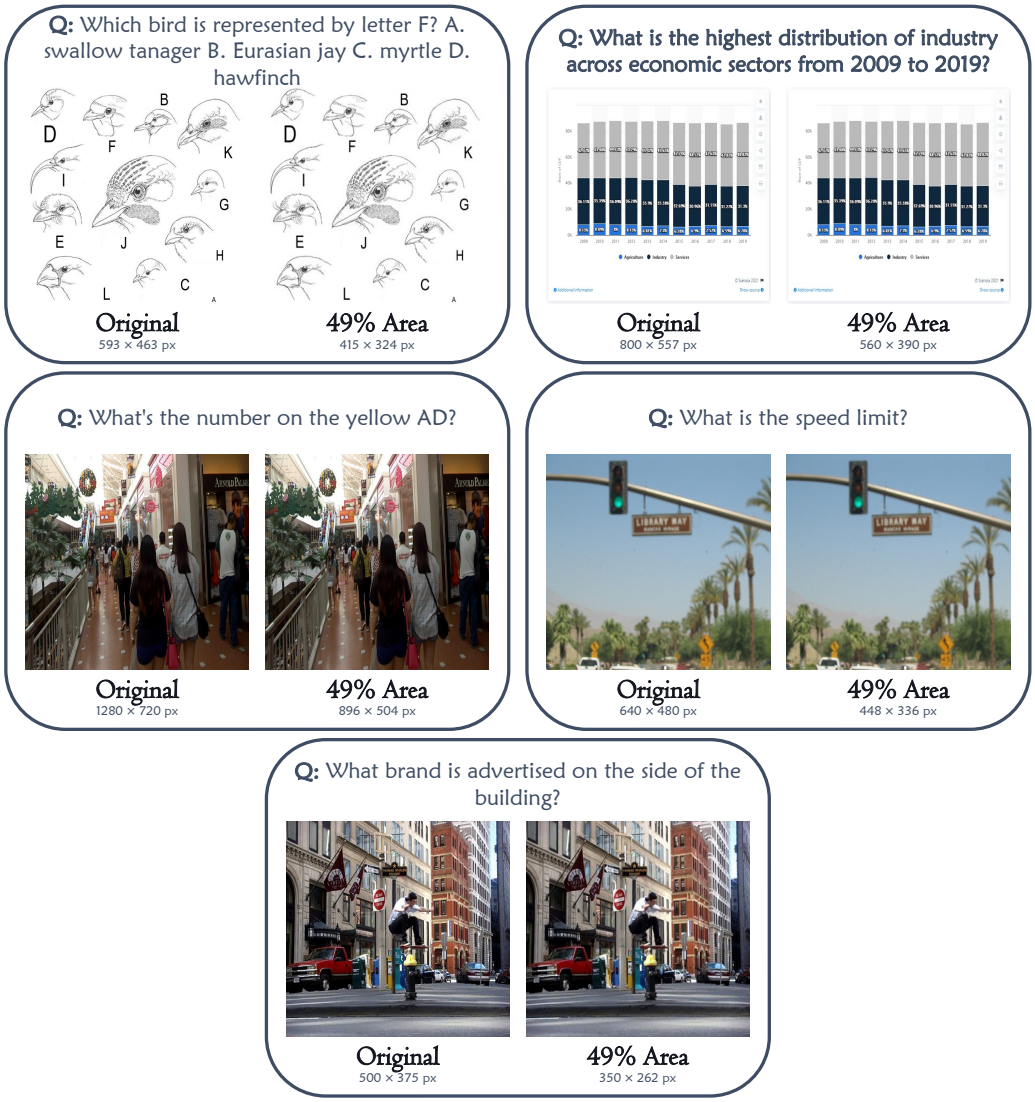

Figure G4: Additional ResizeBench samples at the 0.49 area budget, corresponding to a 70% side-scale barrier.



Figure G5: Additional ResizeBench samples at the 0.64 area budget, corresponding to an 80% side-scale barrier.

H LIMITATIONS

ResizeBench barriers are empirical, model-dependent failure points rather than universal thresholds at which resizing must achieve zero accuracy. They are set using Qwen3-VL and verified with frontier VLMs, but another model may recover an answer at the same resolution through a different visual encoder, prior knowledge, or a correct guess. This transfer variation is visible on Qwen3.5-4B and Gemma-4-E4B, neither of which was used to set the barriers. Nonzero resize accuracy on ResizeBench is therefore expected. A tool-enabled model that is not given the same restriction may invoke super-resolution or another image-enhancement tool. Evaluating such restoration is outside our scope: ResizeBench measures what a VLM can recover from the provided pixels. Similarly, the full-resolution accuracy is also not always perfectly accurate on ResizeBench. To counter this, we have an experiment in Appendix E which additionally conditions the evaluation on samples answered correctly at full resolution.