NOW YOU SEE IT, NOW YOU DON'T: RETHINKING VISUAL TOKEN ALLOCATION ACROSS VISION LANGUAGE MODEL LAYERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models (VLMs) can often localize relevant regions in an image, yet still fail to perceive the fine-grained visual evidence needed to answer correctly, leading to visual hallucinations. Using more tokens provides a finer representation of the image, but also increases the computation, memory, and latency required for inference. VLMs therefore operate under a limited visual-token budget: spending more of it on irrelevant details wastes computation without improving the answer. The central challenge is therefore to spend this limited budget where additional visual detail is most useful.

A common strategy is attention-guided cropping, however, existing approaches typically extract this attention from a manually selected layer or a fixed range of layers, without establishing which depth provides the most reliable signal for visual selection. We show that the choice of layer is a dominant factor in attention-guided visual selection. By systematically masking text-to-image attention across network depth, we show that the information relevant to the model's prediction is concentrated within an early-to-middle portion of the network. This boundary occurs substantially earlier than the layers used by existing attention-based selection methods. Based on this finding, we introduce two complementary strategies. Depth-Weighted Attention (DWA) combines attention information across the full depth of the model through a closed-form ridge regression, learning layer weights that suppress less informative late-layer contributions and produce more reliable spatial selection. Adaptive Visual Resolution (AVR) uses the identified depth boundary to remove visual tokens that no longer affect the prediction and redirects the saved computation toward higher-resolution encoding, allowing the model to represent important image regions with finer visual detail.

Across nine matched-compute comparisons, DWA consistently improves over existing layer-selection strategies, achieving gains of 7.9 and 8.9 percentage points over the strongest baseline and locating answer-relevant evidence in 63.9% of cases compared with 13.6%Àgainst native dynamic-resolution models operating at 5.5$\times$ the visual-token budget, DWA and AVR achieve statistically indistinguishable performance while using only 14-18% of the visual-token computation.

1 INTRODUCTION

A vision–language model asked "what material is the glove?" about a 2250×1500 photograph must first encode the image into visual tokens. At any finite budget the glove may occupy a fraction of a token, and the evidence is gone before the language model runs. The model still *localises* it, its attention concentrates on the right region but it cannot *perceive* it.

The standard remedy is attention-guided cropping: run the model once, read its attention over image positions, crop where the attention peaks, and re-encode that region at higher density (Zhang et al., 2025; Shen et al., 2025). However, existing methods in this family share a structurally flawed premise: they assume spatial signals can be reliably extracted from a manually selected layer or a fixed late-layer block.

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.

Rather than treating layer selection as an arbitrary design choice, we present a mechanistic study of how visual information is used across VLM decoder depth. We demonstrate that the decoder depth at which direct visual information affects prediction is not the same as the depth at which attention becomes most useful for spatial localization.

Based on this separation of depth-dependent roles, our findings and contributions are as follows:

- **Mechanistic finding on the Depth Boundary.** Layer-wise causal masking reveals a strict **depth boundary** where image-to-text transport concludes. Blocking direct image access from layer 16 (of 28) changes only 12% and 5% of predictions on the two core checkpoints; masking from layer 24 onward changes none.
- **Diagnostic finding on spatial readout.** To overcome the **sub-token encoding bottleneck**, models must localize evidence accurately. However, the layers providing the most useful spatial localization signal occur much later than the transport-sensitive region. This confirms that visual-information transport and spatial readout are two fundamentally distinct, depth-dependent roles.
- **DWA as a consequence (WHERE).** Because single-layer readout is unreliable and late layers collapse into structural artifacts (e.g., targeting grid edges in up to 84.8% of items), Depth-Weighted Attention (DWA) combines spatial signals across all decoder depths. This increases relevant spatial coverage from 46.6% to 63.9%, driving crop placement accuracy up by 7.9 and 8.9 points over the strongest adaptive baseline.
- **AVR as a consequence (WHEN).** Exploiting the discovered **depth boundary**, Adaptive Visual Resolution (AVR) restricts direct visual attention in the late layers where it is mechanically obsolete. Reallocating this nominal budget to encode a higher-resolution (900-token) image achieves 70.0% and 61.8% accuracy—matching native full-image baselines at a fraction of the native token allowance.
- **Task-dependent allocation.** Spatial cropping via DWA strictly benefits localized targets. When evidence is spatially separated or scene-level, cropping degrades accuracy by up to 14.1 points. In these distributed contexts, AVR's full-image allocation emerges as the necessary, robust policy.

2 Related Work

Attention-guided cropping: ViCrop (Zhang et al., 2025) uses the peak of a fixed mid-layer attention map to determine where to crop, while ZoomEye (Shen et al., 2025) searches a tree of candidate crops through repeated model queries. VisLens (He et al., 2026) takes a different approach and does not rely on attention. Instead, it decodes visual tokens using a tuned logit lens, matches them against the query, and feeds the selected crop back together with the original image in a single pass. A common limitation of these approaches is that read-out depth is fixed manually, and their evaluations do not control the compute that could instead be spent on simply increasing image resolution.

Layer-adaptive read-outs: LASER (Zhu et al., 2026) selects an attention layer separately for each sample by comparing attention maps obtained with and without the query. Its results suggest that there is no single "magic layer," a finding that we reach independently and investigate further from a causal perspective. ILVAD (Xie et al., 2026) aggregates newly activated tokens across neighboring layers. Our approach differs by measuring the effect of blocking visual access across decoder depth, combining read-outs across layers, and evaluating under matched compute budgets.

Visual-token pruning: FastV (Chen et al., 2025) prunes visual tokens at layer 2, PyramidDrop (Xing et al., 2025) progressively removes tokens using a fixed pruning ratio, and HiRED (Arif et al., 2025) determines the token budget before the language model processes the input. These methods are primarily evaluated in terms of how much compute can be saved while retaining performance. In contrast, we derive the pruning depth directly from our measurement of visual information transport rather than treating it as a tunable hyperparameter. At the identified depth, we find that which tokens are retained matters surprisingly little: randomly selected tokens perform comparably to attention-selected tokens. Before the measured transition, attention-ranked retention exceeds random retention by 8.4 and 8.9 points in the reported control. Instead, starting the restriction earlier than the boundary-based schedule loses 1.0 and 5.8 points (Appendix B.3).

3 METHODOLOGY

To move beyond heuristic crop placement, we must formally quantify how visual information propagates through a network. We first define the standard VLM perception pipeline and establish a controlled experimental protocol to isolate the physical limits of sub-token resolution. With this baseline, we introduce a causal layer-masking framework designed to precisely uncouple the depth at which a model transports visual evidence from the depth at which it forms readable spatial attention maps. Recognizing that visual transport and spatial readout operate at these distinct depths allows us to design two formal allocation algorithms: Depth-Weighted Attention (DWA) for robust spatial selection, and Adaptive Visual Resolution (AVR) for layer-wise compute reallocation.

3.1 WHAT DOES A VLM SEE, AND HOW?

We begin with a frozen VLM comprising a vision encoder E, a visual-to-language mapping P, and a language decoder F_θ with L decoder layers. Here, P represents visual projection or merging components of the model that produce embeddings for the language decoder. For a given image I and question prompt q:

$$V = P(E(I)) \in \mathbb{R}^{N \times d}, \tag{1}$$

$$\mathbf{z} = F_\theta(V, q) \in \mathbb{R}^K. \tag{2}$$

where V contains N visual-token embeddings, each with dimension d. The decoder F_θ, including its output head, produces the vocabulary-logit vector $\mathbf{z}$ at the final prompt position. The vocabulary size is K, and θ denotes the pretrained model parameters, which remain frozen. The prompt q includes the question, answer options, and answer instruction supplied with the example.

The decoder's logits define the next-token probability distribution through softmax. We use these logits to evaluate multiple-choice answers, with the scoring procedure described in Appendix A.

3.2 VLMS AND FINE-GRAINED PERCEPTION: VISUAL RESOLUTION AS AN ENCODING BOTTLENECK

A common assumption in multimodal evaluation is that VLMs fail on fine-grained queries because they struggle to isolate and reason over tiny details. However, sweeping the target extent in merged visual tokens for single-region questions (Appendix C) reveals a far more fundamental bottleneck: information is simply never encoded in the first place. We observe a sharp, architecture-invariant cliff exactly at 0.15-0.25 merged tokens, corresponding to barely 2-3$\%$ of image side. Below spatial threshold, accuracy collapses to 19.0%, yet it surges to 78.6% immediately above it. Because evaluating these exact items with high-resolution oracle crop yields a perfectly stable accuracy of 95.8% to 100.0%, this cliff cannot be attributed to a gradual degradation in reasoning capability. It represents a hard physical limit: sub-token visual evidence simply fails to enter model representations.

We next localise the source of this failure using layer-wise decoding and representation patching. If the model were capturing the visual detail but failing to process it, early layers would temporarily hold the correct answer. Instead, localisation alone provides little additional decodability (72.2% vs. 68.1%), and no layer shows evidence of encoding the answer and subsequently losing it. In contrast, providing sufficient visual detail through an oracle crop corrects 94.4% of localised-but-wrong cases, despite requiring a median of only 0.14 tokens on the target. Across models, 86.7%/84.0% of errors are therefore certified as encoding failures, confirming that the dominant bottleneck is extracting fine-grained visual evidence rather than reasoning over evidence that has already been encoded.

This raises a practical question: should we increase the overall visual-token budget, or allocate the same budget more selectively? We compare oracle-targeted allocation against the best uniform resolution each model can reach, across three architectures at native resolution and the HR-Bench boundary (Appendix C). Targeted allocation is substantially more efficient: a 300-token oracle crop reaches 97.4%/93.2%, outperforming any uniform full-image setting up to 7,776 tokens ($\geq$26$\times$/25.9$\times$ more tokens). Even at native resolution, a 600-token crop outperforms 3,290 uniform tokens by +11.6/+16.2 points.

3.3 IDENTIFYING THE DEPTH BOUNDARY

We examine where direct access to visual tokens affects answer prediction. The primary experiment uses Qwen3-VL-2B-Instruct (Qwen Team, 2025b) and Qwen2-VL-7B-Instruct (Qwen Team, 2024), each with $L = 28$ decoder layers. Indexing the decoder layers from zero, $\ell = 0, \ldots, L-1$, let $\mathcal{S}$ denote the specific set of layers selected for masking. At these candidate layers, we modify the additive attention mask to selectively blind post-image text tokens from attending to visual span:

$$\widetilde{M}^{(\ell)}_{u,v} = \begin{cases} m_{\min}, & \ell \in \mathcal{S},\ u \in \mathcal{T}_{\text{post}},\ v \in \mathcal{V}, \\ M^{(\ell)}_{u,v}, & \text{otherwise.} \end{cases} \tag{3}$$

Here, $M^{(\ell)}$ and $\widetilde{M}^{(\ell)}$ are the original and modified masks. The indices u and v denote query and key positions. The set $\mathcal{V}$ is the image-token span, identified from the input token IDs, and $\mathcal{T}_{\text{post}}$ contains all prompt positions after that span. The value $m_{\min}$ is the smallest finite value representable by the mask's data type. This replacement is applied across attention heads and suppresses the selected connections after softmax. We evaluate three masking schedules:

$$\mathcal{S}_{\text{single}}(b) = \{b\}, \tag{4}$$

$$\mathcal{S}_{\text{prefix}}(b) = \{0, \ldots, b\}, \tag{5}$$

$$\mathcal{S}_{\text{suffix}}(b) = \{b, \ldots, L-1\}. \tag{6}$$

Here, b is the candidate layer index. Single-layer masking is evaluated at every layer. Prefix and suffix masks are evaluated at $b \in \{0, 4, 8, 12, 16, 20, 24\}$alongside an all-layer control. To quantify the exact impact of these interventions, we measure the perturbation to the next-token probability distribution using Kullback–Leibler (KL) divergence (Kullback & Leibler, 1951):

$$\mathbf{p} = \text{softmax}(\mathbf{z}), \qquad \widetilde{\mathbf{p}}_{\mathcal{S}} = \text{softmax}(\widetilde{\mathbf{z}}_{\mathcal{S}}), \tag{7}$$

$$D_{\mathcal{S}} = \sum_{k=1}^{K} p_k \log \frac{p_k}{\widetilde{p}_{\mathcal{S},k}}. \tag{8}$$

Here, $\widetilde{\mathbf{z}}_{\mathcal{S}}$ is the masked logit vector. The vectors $\mathbf{p}$ and $\widetilde{\mathbf{p}}_{\mathcal{S}}$ contain baseline and masked probabilities over the vocabulary, and k indexes vocabulary tokens. The scalar $D_{\mathcal{S}}$ measures divergence from the baseline distribution to the masked distribution. We add 10^{-12} inside the baseline-probability logarithm for numerical stability. Moreover, We record the answer-flip indicator,

$$f_{\mathcal{S}} = \mathbf{1}[\hat{c}_{\text{base}} \neq \hat{c}_{\mathcal{S}}], \tag{9}$$

where $\hat{c}_{\text{base}}$ and $\hat{c}_{\mathcal{S}}$ are the baseline and masked option predictions under the boundary experiment's scoring rule in Appendix A. The indicator $\mathbf{1}[\cdot]$ equals one when its condition holds and zero otherwise. We average divergence and answer flips across examples. Answer flips measure changes in the prediction, not changes in correctness against the ground-truth answer.

Based on the above findings, We evaluate two complementary strategies:

(i) Depth-Weighted Attention (DWA) **(ii) Adaptive Visual Resolution (AVR)**

3.4 DWA : DEPTH WEIGHTED ATTENTION

DWA selects a region using information from the full model layer depth. We first encode the full image toward N_{vis} visual tokens and run the model with complete question prompt. At each layer, we extract the final prompt position's attention to image tokens and average it across attention heads.

Depth-aware attention features. Let $a_i^{(\ell)}$ denote the head-mean attention at image cell i and layer ℓ, where $i = 1, \ldots, N_{\text{vis}}$ and $\ell = 0, \ldots, L-1$, read from the final prompt token over the image cells. We normalise each attention map over the image cells:

$$A_i^{(\ell)} = \frac{a_i^{(\ell)}}{\max\left(\sum_{j=1}^{N_{\text{vis}}} a_j^{(\ell)}, \epsilon\right)}, \qquad \epsilon = 10^{-12}, \tag{10}$$

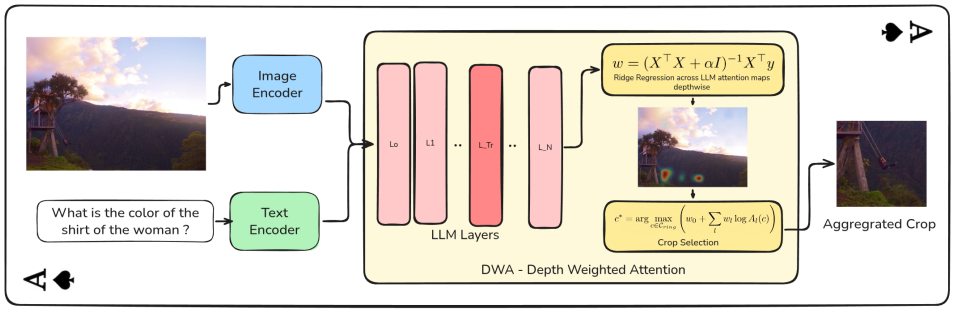

Figure 1: Overview of the proposed Depth-Weighted Attention method. Given an input image and question, image and text are encoded and processed through LLM layers. Attention maps extracted across the model depth are aggregated using ridge regression to learn depth-wise weights.

where j indexes image cells in the sum and ϵ prevents division by zero. Each cell is represented by its normalised depth profile:

$$\mathbf{x}_i = \left(\log(A_i^{(\ell)} + \epsilon)\right)_{\ell=0}^{L-1} \in \mathbb{R}^L, \tag{11}$$

so that the feature representation captures how attention to the cell evolves across the full model depth. The complete feature vector additionally includes rank, neighbourhood, and geometry features, specified in (Appendix B), giving $\mathbf{z}_i \in \mathbb{R}^{2L+7}$.

Learning crop scores. The supervision target measures how much of the annotated evidence a candidate crop would cover:

$$r_i = \frac{\text{area}(C_i \cap B)}{\max(\text{area}(B), \epsilon)}. \tag{12}$$

Here, C_i is the candidate crop centred at cell i, after shifting it inside the image boundaries. The box B is the smallest axis-aligned rectangle enclosing all annotated boxes for that example. The operator area measures area in normalised image coordinates, and r_i is the fraction of B covered by the crop. Thus, the target measures box coverage rather than intersection over union.

We standardise each feature using the training-fold mean and standard deviation, adding 10^{-9} to the standard deviation. Ridge regression (Hoerl & Kennard, 1970) then fits the crop-coverage target:

$$(\mathbf{w}^\star, \beta^\star) = \arg\min_{\mathbf{w},\beta} \left\{ \|X\mathbf{w} + \beta\mathbf{1} - \mathbf{r}\|_2^2 + \lambda\|\mathbf{w}\|_2^2 \right\}, \qquad \lambda = 1. \tag{13}$$

Here, $X \in \mathbb{R}^{M\times(2L+7)}$ contains the standardised features of the M training cells, and $\mathbf{r} \in \mathbb{R}^M$ contains their coverage targets. The vector $\mathbf{w}$ contains the regression coefficients, β is an unpenalised intercept, and $\mathbf{1}$ is a length-M vector of ones. The coefficient penalty has fixed strength $\lambda = 1$. We fit the regression coefficients and intercept by solving the corresponding linear system directly.

The resulting crop score for a candidate cell c is

$$s(c) = \mathbf{w}^{\star\top}\mathbf{z}_c + \beta^\star. \tag{14}$$

Selecting and re-encoding the crop. We select the eligible cell with the highest averaged score. The outer grid ring is excluded when both grid dimensions exceed two cells; otherwise, every cell is eligible. The crop spans 25% of original image width and 25% of its height. It is shifted inward when necessary to remain within the image, and its pixel coordinates are converted to integers. The crop is then re-encoded toward N_{vis} visual tokens and used with original question prompt for answering.

3.5 AVR: Adaptive Visual Resolution

AVR examines whether later visual-token processing can be reduced while retaining higher-resolution input in the earlier layers. The main configuration encodes the full image toward 900 visual tokens, retains unrestricted processing through layer 16, and restricts visual access from layer 17 onward. No crop is applied.

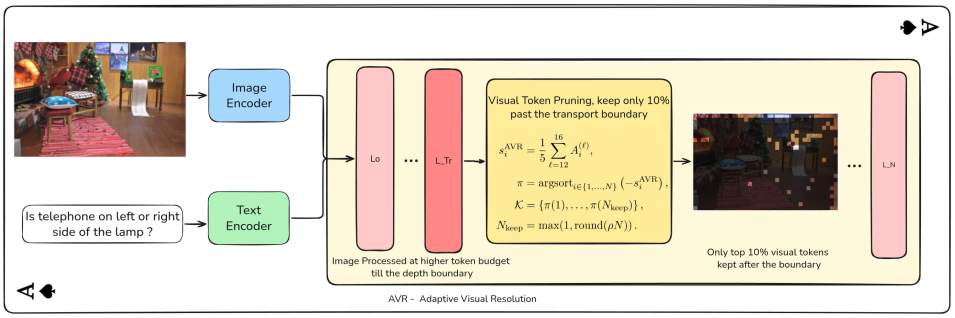

Figure 2: **Overview of Adaptive Visual Resolution (AVR).** Given an image and question, the image is encoded at a higher visual-token budget and processed through the transformer up to the discovered depth boundary. Attention from the final prompt position is aggregated across the preceding layers to score visual tokens. AVR ranks the tokens and retains only the top 10% beyond the boundary, reducing subsequent visual-token processing while preserving the full image context.

Selecting tokens for continued access. For a final unrestricted layer b, visual tokens are ranked:

$$s_i^{\mathrm{AVR}} = \frac{1}{b - \max(0, b-4) + 1} \sum_{\ell=\max(0,b-4)}^{b} A_i^{(\ell)}. \tag{15}$$

Here, s_i^{AVR} is the selection score for image token i, and $A_i^{(\ell)}$ is the normalised, head-averaged attention from the final prompt position, computed on the higher-resolution input. With $b = 16$, the score averages layers 12–16. We retain the highest-scoring tokens, with the count given by

$$N_{\mathrm{keep}} = \max(1, \mathrm{round}(\rho N)), \qquad \rho = 0.10. \tag{16}$$

Here, N is the realised input visual-token count, N_{keep} is the retained count, ρ is the retained fraction, and round denotes rounding to the nearest integer, with ties rounded to the nearest even integer. Additional configurations use layer-16-only attention or random selection at same retained fraction.

Evaluating the restriction. The evaluation first collects attention in an unmasked forward pass. It then runs the model again with an additive bias of -10^4 on discarded visual-key columns in layers $b+1, \ldots, L-1$. The bias applies to all query positions. It suppresses access to those visual keys while leaving the token tensors at their original lengths. Thus the measured accuracy tests the proposed allocation of direct visual access; no layer shortens its token tensors in this implementation. The corresponding token-layer allowance counts accessible visual keys across depth, rather than measured execution time or memory.

The input-token target, retained fraction, and cutoff are fixed for each evaluated configuration; the main experiment does not use a per-example rule to choose the input resolution.

Comparing visual-token budgets. DWA uses 300 input visual tokens for localisation and 300 for answering, so its total is compared with the native median input counts of 3,290 and 4,320 on the two checkpoints. For AVR, we account for the visual keys available to attention at each decoder layer. With N input visual tokens, N_{keep} accessible keys after layer b, and L layers, allowance is:

$$B_{\mathrm{AVR}} = N(b+1) + N_{\mathrm{keep}}(L - b - 1). \tag{17}$$

The 900-token configuration with $b = 16$, $L = 28$, and $N_{\mathrm{keep}} = 90$ has an allowance of 16,290 visual token-layers. Dividing by 28 times the respective native median input count gives approximately 18% and 14%. In this sense, AVR assigns more visual tokens to early layers and a smaller direct-access allowance to later layers within a comparable nominal budget. Both forward passes retain their token tensors, so these ratios describe allocation rather than measured runtime or memory.

4 RESULTS

4.1 VISUAL ACCESS BECOMES LESS INFLUENTIAL AT LATER DEPTHS

Causal interventions reveal a strict depth boundary where visual transport concludes. While single-layer masking produces its largest output change in the early-to-mid stack (layer 11 in Qwen3-VL-2B and layers 14–15 in Qwen2-VL-7B), masking direct image access in the late layers has almost no effect. When all visual access is restricted from layer 16 onward, the mean KL divergence collapses to just 0.014 for Qwen3-VL-2B and 0.029 for Qwen2-VL-7B. Correspondingly, the proportion of changed answers plummets to 12% and 5% at this $\ell = 16$ boundary. By layer 24, restricting visual access changes 0% of the answers for both core checkpoints, with Qwen3-VL-2B reaching this zero-impact threshold even earlier at layer 20. This confirms that late-layer attention is mechanically obsolete for transporting visual features.

The scope of this masking further confirms that visual transport is routed indirectly. Restricting visual access only at the final answer-emission position across all layers yields negligible KL divergences (0.025 and 0.034) and flips almost no answers (2.5% and 0%). Conversely, restricting all post-image text positions changes approximately 42% and 45% of answers. The answer token therefore does not attend to the image directly; instead, it relies entirely on preceding text tokens that successfully aggregated visual evidence before reaching the depth boundary.

Table 1: Prefix and suffix masks, all text rows, $n = 40$ per model, V*Bench at 300 tokens. The prefix mask $L_0..\ell$ saturates at the full-mask value once ℓ passes the window; the suffix mask $L_\ell..$end falls to zero once ℓ is past it. Both read the boundary at the same place, and the 36-layer checkpoint puts it at the same fraction of depth (ℓ=20 of 36).

	ℓ=4	ℓ=8	ℓ=12	ℓ=16	ℓ=20	ℓ=24
Qwen3 prefix $L_0..\ell$ (KL)	0.082	0.549	**1.278**	1.259	1.215	1.257
Qwen3 suffix $L_\ell..$ (KL)	0.967	0.880	0.320	**0.014**	0.002	0.001
Qwen3 suffix (answers flipped)	53%	50%	20%	12%	**0%**	**0%**
Qwen2 prefix $L_0..\ell$ (KL)	0.037	0.105	0.364	**0.547**	0.534	0.538
Qwen2 suffix $L_\ell..$ (KL)	0.518	0.465	0.429	**0.029**	0.004	0.002
Qwen2 suffix (answers flipped)	53%	47%	40%	5%	3%	**0%**
Qwen3-8B (36L) prefix $L_0..\ell$ (KL)	0.100	0.309	1.190	**1.740**	1.559	1.565
Qwen3-8B (36L) suffix $L_\ell..$ (KL)	1.671	1.538	1.278	0.413	**0.052**	0.009
Qwen3-8B (36L) suffix (answers flipped)	35%	45%	40%	15%	5%	**0%**

4.2 DWA IMPROVES ACCURACY THROUGH TARGETED CROP SELECTION

DWA improves accuracy when the task requires information that is spatially concentrated within the image. As shown in Figure 3, standard block-mean attention often assigns high importance to irrelevant image regions, whereas DWA combines attention signals across model depth to identify regions containing the relevant evidence. This improved localization is reflected in the bounding-box coverage: DWA covers at least half of the annotated bounding box in 63.9% and 55.0% of V*Bench examples for Qwen3-VL-2B and Qwen2-VL-7B, respectively, compared with 46.6% and 39.3% for the block-mean baseline.

The improved crop selection translates directly into higher accuracy on benchmarks where the required evidence is localized (Table 2). Under a fixed 600-token visual budget, DWA improves V*Bench accuracy by $+10.5$, $+8.9$, $+13.6$, and $+1.0$ percentage points for Qwen3-VL-2B, Qwen2-VL-7B, LLaVA-OV-7B, and InternVL3-8B, respectively. DWA also improves HR-4K across all four models, with gains ranging from $+2.1$ to $+8.2$ points. These results show that allocating the available visual tokens to regions selected by DWA can provide substantial accuracy gains over encoding the uncropped image at the same compute budget.

The gains, however, depend on the spatial characteristics of the task. On CV-Bench and RealworldQA, where relevant information can be distributed across the broader scene, restricting the visual input to a selected crop reduces accuracy relative to the full-image reference. On CV-Bench,

Table 2: **DWA accuracy across four benchmarks and models.** The reference is uncropped image at the equal-compute budget (600 visual tokens). Differences are paired and in percentage points.

Model	Benchmark	n	Reference (%)	DWA (%)	Δ
Qwen3-VL-2B	V*	191	62.3	72.8	+10.5
	HR-4K	800	59.0	62.5	+3.5
	CV-Bench	2,638	79.9	65.8	−14.1
	RealworldQA	765	62.6	56.9	−5.8
Qwen2-VL-7B	V*	191	59.2	68.1	+8.9
	HR-4K	800	57.0	59.1	+2.1
	CV-Bench	2,638	75.0	66.0	−9.0
	RealworldQA	765	64.7	57.5	−7.2
LLaVA-OV-7B	V*	191	51.8	65.4	+13.6
	HR-4K	800	48.2	56.5	+8.2
	CV-Bench	2,638	76.6	65.9	−10.7
	RealworldQA	765	61.6	59.2	−2.4
InternVL3-8B	V*	191	74.9	75.9	+1.0
	HR-4K	800	70.5	75.4	+4.9
	CV-Bench	2,638	81.7	68.6	−13.1
	RealworldQA	765	67.6	59.9	−7.7

the decrease ranges from -9.0 to -14.1 points, while RealworldQA shows decreases of -2.4 to -7.7 points. Thus, the results indicate that DWA is particularly effective when the answer depends on concentrated visual evidence, while full-image encoding remains advantageous for tasks requiring broader scene context.

Finally, DWA does not require extensive target-specific fitting to obtain these gains. As shown in Table 3, a DWA model fitted on V*Bench transfers directly to the evaluated target settings, with in-domain refitting changing performance by at most 1.5 percentage points. On V*Bench, zero-shot and fitted performance are identical by construction, while on the CV-Bench Depth+Distance subset, refitting improves accuracy by only $+0.3$ and $+1.5$ points for Qwen3-VL-2B and Qwen2-VL-7B, respectively. This suggests that the depth-weighted spatial selection learned from V*Bench is largely preserved when transferred to the evaluated settings.

Table 3: **DWA: zero-shot transfer versus in-domain fitting.** Fitting DWA on the target benchmark provides only marginal improvement over zero-shot transfer. Across the evaluated settings, refitting changes accuracy by at most 1.5 percentage points, while producing identical performance on V*Bench by construction. Δ denotes the fitted minus zero-shot accuracy.

Source (Fitted On)	Target (Evaluated On)	Zero-Shot (%)	Fitted (%)	Δ
Qwen3-VL-2B (V*Bench)	Qwen3-VL-2B (V*Bench)	72.8	72.8	0.0
Qwen3-VL-2B (V*Bench)	Qwen3-VL-2B (CV-Bench D+D, $n = 400$)	75.5	75.8	+0.3
Qwen2-VL-7B (V*Bench)	Qwen2-VL-7B (V*Bench)	60.0	60.0	0.0
Qwen2-VL-7B (V*Bench)	Qwen2-VL-7B (CV-Bench D+D, $n = 400$)	76.5	78.0	+1.5

4.3 AVR Achieves Higher Accuracy Via Token Reallocation

By restricting direct visual access after the measured depth boundary, AVR frees the token-layer budget to encode a higher-resolution full image without applying any spatial crops. On tasks demanding high-resolution localized evidence, this increased input density drives consistent improvements. Evaluated against a budget-matched uncropped reference, AVR yields gains of $+5.2$ points on Qwen3-VL-2B, $+5.8$ points on Qwen2-VL-7B, and $+6.8$ points on LLaVA-OV-7B on the V*Bench evaluation.

However, the critical advantage of AVR emerges on tasks requiring distributed evidence. Spatial cropping severely penalizes scene-level reasoning: concentrating resolution spatially via methods like DWA reduces cross-instance accuracy on HR-Bench 4K by 7.5 and 6.8 points. On scene-level benchmarks, these losses cascade further, driving CV-Bench drops of 14.1 and 9.0 points, and reducing pooled RealworldQA performance by 5.8 and 7.2 points on the core checkpoints.

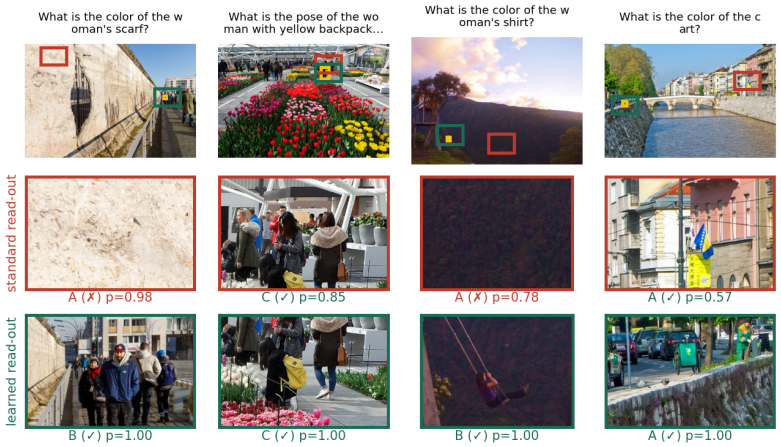

Figure 3: **Crop placement for a sub-token target.** The input and evidence box are shown at left. Block-mean attention selects a misaligned crop near a grid artifact, whereas DWA selects the target region and yields the correct answer.

Table 4: **AVR: coverage kept, resolution gained.** AVR never crops. It keeps the whole image and removes visual tokens only at the depth where they no longer influence the answer, spending the freed computation on a finer input encoding; scene-level questions therefore keep their full context while local detail gains resolution. Shown is accuracy against the uncropped image, with paired differences in percentage points.

Model	Benchmark	n	Reference (%)	AVR (%)	Δ
Qwen3-VL-2B	V*	191	62.3	67.5	+5.2
	HR-4K	800	59.0	59.9	+0.9
	CV-Bench	2,638	79.9	80.4	+0.5
	RealworldQA	765	62.6	63.3	+0.7
Qwen2-VL-7B	V*	191	59.2	64.9	+5.8
	HR-4K	800	57.0	57.1	+0.1
	CV-Bench	2,638	75.0	71.5	−3.5
	RealworldQA	765	64.7	62.6	−2.1
LLaVA-OV-7B	V*	191	51.8	58.6	+6.8
	HR-4K	800	48.2	46.2	−2.0
	CV-Bench	2,638	76.6	76.0	−0.6
	RealworldQA	765	61.6	60.0	−1.6

Because AVR allocates visual access across depth rather than space, it completely avoids these catastrophic drops. By retaining the full uncropped image, AVR preserves the context required for relational reasoning, keeping CV-Bench and RealworldQA performance highly stable. Its performance on these benchmarks ranges from slight positive gains (+0.5 and +0.7 points) on Qwen3-VL-2B to only modest reductions (−3.5 and −2.1 points) on Qwen2-VL-7B.

5 Conclusion

Decoder depth determines both when direct image access influences prediction and where attention provides a useful spatial read-out. Our interventions and layer sweeps separate these roles. DWA uses fitted depth weights to improve crop placement, while AVR preserves full-image coverage with higher input resolution and restricts direct visual access after the measured depth boundary.

DWA improves local-detail answering, and its advantage over block-mean crop selection extends across the evaluated architectures. AVR better preserves performance on tasks requiring evidence from multiple regions. Together, these findings provide an empirical basis for choosing spatial concentration or full-image coverage according to the required evidence.

AI USE STATEMENT

AI tools assisted with drafting and revising portions of the manuscript, improving language, and checking LaTeX cross-references. The authors reviewed the resulting text and take responsibility for the methods, reported results, citations, and conclusions.

REFERENCES

Kazi Hasan Ibn Arif, JinYi Yoon, Dimitrios S. Nikolopoulos, Hans Vandierendonck, Deepu John, and Bo Ji. HiRED: Attention-guided token dropping for efficient inference of high-resolution vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 1773–1781, 2025. URL https://ojs.aaai.org/index.php/AAAI/article/view/32171.

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *Computer Vision – ECCV 2024*, pp. 19–35, 2025. doi: 10.1007/978-3-031-73004-7_2. URL https://link.springer.com/chapter/10.1007/978-3-031-73004-7_2.

Google DeepMind. Gemma 3 model card. Official model documentation, 2025. URL https://ai.google.dev/gemma/docs/core/model_card_3. Accessed 23 September 2026.

Jingyi He, Sanghwan Kim, and Zeynep Akata. VisLens: Single-pass interpretable visual search for multimodal LLMs. In *ECCV 2026 Workshop on Explainable Computer Vision (eXCV)*, 2026. URL https://openreview.net/forum?id=I5RAChaCG. Accepted proceedings-track paper.

Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970. doi: 10.1080/00401706.1970.10488634. URL https://doi.org/10.1080/00401706.1970.10488634.

Solomon Kullback and Richard A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951. doi: 10.1214/aoms/1177729694. URL https://doi.org/10.1214/aoms/1177729694.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=zKv8qULV6n.

OpenGVLab. Introduction of InternVL3.0 series. Official model documentation, 2025. URL https://internvl.readthedocs.io/en/latest/internvl3.0/introduction.html. Accessed 23 September 2026.

Qwen Team. Qwen2-VL-7B-Instruct: Official model card. Official model documentation, 2024. URL https://huggingface.co/Qwen/Qwen2-VL-7B-Instruct. Accessed 23 September 2026.

Qwen Team. Qwen2.5-VL-7B-Instruct: Official model card. Official model documentation, 2025a. URL https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct. Accessed 23 September 2026.

Qwen Team. Qwen3-VL-2B-Instruct: Official model card. Official model documentation, 2025b. URL https://huggingface.co/Qwen/Qwen3-VL-2B-Instruct. Accessed 23 September 2026.

Haozhan Shen, Kangjia Zhao, Tiancheng Zhao, Ruochen Xu, Zilun Zhang, Mingwei Zhu, and Jianwei Yin. ZoomEye: Enhancing multimodal LLMs with human-like zooming capabilities through tree-based image exploration. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 6602–6618, 2025. doi: 10.18653/v1/2025.emnlp-main.335. URL https://aclanthology.org/2025.emnlp-main.335/.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Ziteng Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/9ee3a664ccfeabc0da16ac6f1f1cfe59-Paper-Conference.pdf.

Wenbin Wang, Liang Ding, Minyan Zeng, Xiabin Zhou, Li Shen, Yong Luo, Wei Yu, and Dacheng Tao. Divide, conquer and combine: A training-free framework for high-resolution image perception in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 7907–7915, 2025. URL https://ojs.aaai.org/index.php/AAAI/article/view/32852.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In *CVPR*, pp. 13084–13094, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wu_V_Guided_Visual_Search_as_a_Core_Mechanism_in_Multimodal_CVPR_2024_paper.html.

xAI. RealWorldQA. Dataset, 2024. URL https://huggingface.co/datasets/xai-org/RealworldQA.

Yutong Xie, Zhenglin Hua, Ran Wang, Wing W. Y. Ng, Xizhao Wang, and Yuheng Jia. Finding the correct visual evidence without forgetting: Mitigating hallucination in LVLMs via inter-layer visual attention discrepancy. In *International Conference on Machine Learning*, 2026. URL https://icml.cc/Downloads/2026. Accepted conference paper.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, and Dahua Lin. Conical visual concentration for efficient large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14593–14603, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Xing_Conical_Visual_Concentration_for_Efficient_Large_Vision-Language_Models_CVPR_2025_paper.html.

Jiarui Zhang, Mahyar Khayatkhoei, Prateek Chhikara, and Filip Ilievski. MLLMs know where to look: Training-free perception of small visual details with multimodal LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=DgaY5mDdmT.

Zipeng Zhu, Zhanghao Hu, Qinglin Zhu, Jingyong Su, Yulan He, and Lin Gui. Beyond static cropping: Layer-adaptive visual localization and decoding enhancement. In *ACL 2026 Student Research Workshop*, 2026. URL https://2026.aclweb.org/program/srw_papers/. Accepted workshop paper.

APPENDIX

A EXPERIMENTAL SETUP

Models and benchmarks. The core comparison uses Qwen3-VL-2B-Instruct (Qwen Team, 2025b) and Qwen2-VL-7B-Instruct (Qwen Team, 2024). Additional V*Bench comparisons use Gemma-3-4B (Google DeepMind, 2025), InternVL3-8B (OpenGVLab, 2025), and LLaVA-OneVision-7B (Li et al., 2025). The resolution analysis also includes Qwen2.5-VL-7B (Qwen Team, 2025a) and Qwen3-VL-8B. Table 5 summarises the evaluation sets. We retain the supplied question categories; RealworldQA's answer-format split is not treated as an annotation of whether a question requires local or scene-level evidence.

Comparisons and transfer. We compare DWA with an uncropped image at the reference budget, a fixed-layer crop at layer 14, and a block-mean attention crop. The latter two are placement controls within the same crop-only pipeline, rather than reproductions of the complete ViCrop method (Zhang et al., 2025). The LASER-based control (Zhu et al., 2026) uses contrastive attention

Table 5: Evaluation sets. Counts refer to the full sets; the core Qwen3-VL-2B placement comparison uses the 190 items available for every baseline. The layer sweep and architecture comparison use all 191 V*Bench items.

BENCHMARK	ITEMS	REPORTED PARTITION OR USE
V*Bench (Wu & Xie, 2024)	191	115 single-instance; 76 cross-instance
HR-Bench 4K (Wang et al., 2025)	800	400 single-instance; 400 cross-instance
HR-Bench 8K (Wang et al., 2025)	800	Resolution analysis; 400 per stratum
CV-Bench (Tong et al., 2024)	2,638	Count, relation, distance, and depth tasks
RealworldQA (xAI, 2024)	765	Multiple-choice and open-answer questions

and per-example layer selection, but retains our quarter-width crop and omits LASER's contrastive decoding. It selects the layer with the largest Euclidean norm of the positive difference between normalised head-mean query and no-query maps. The no-query map requires an additional 300-token pass, which is excluded from the nominal 600-token placement accounting. Ground-truth-guided crops serve as an oracle-placement reference, not a deployable method.

For cross-benchmark evaluations, ridge coefficients are fitted on V*Bench and transferred within the same checkpoint without target bounding-box supervision. The dedicated HR-Bench transfer protocol standardises features over the unlabelled target set; the extended benchmark protocol standardises them within each image. These transfer runs are reported separately from the grouped, out-of-fold V*Bench comparison. The extended AVR runs use random retention of 10% of visual tokens, with the first restricted layer at $\mathrm{round}(0.57L)$; this differs from the attention-ranked core schedule in Sec. 3.5.

Statistical comparisons. We use paired item bootstrap intervals with 8,000 resamples and report 95% confidence intervals for accuracy differences. All differences are in percentage points. We identify a difference as statistically significant when its interval excludes zero. The repeated grouped splits for DWA are described in Sec. 3.4.

Answer scoring. We evaluate DWA and AVR using next-token logits at the final prompt position. For each answer option, we combine the scores of the token IDs obtained from its plain and space-prefixed letter:

$$g_c = \log \sum_{k \in \mathcal{J}_c} \exp(z_k), \tag{18}$$

$$\pi_c = \frac{\exp(g_c)}{\sum_{c' \in \mathcal{C}} \exp(g_{c'})}, \qquad \hat{c} = \arg\max_{c \in \mathcal{C}} \pi_c. \tag{19}$$

Here, $\mathcal{C} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}\}$ is the set of answer options, indexed by c and c'. The set $\mathcal{J}_c$ contains the distinct final token IDs returned when tokenising the plain and space-prefixed forms of letter c without special tokens. The quantity z_k is the logit for vocabulary token k, g_c is the combined option score, π_c is its scalar probability after normalisation over the four options, and $\hat{c}$ is the predicted option.

The depth-masking experiment instead uses one token ID per option: the first ID returned when the plain letter is encoded without special tokens. Its answer prediction is the option with the largest logit among those four IDs. Its primary divergence measure covers the full vocabulary; it also records four-option divergence for the all-layer and single-layer interventions.

For CV-Bench and the multiple-choice portion of RealworldQA, the option-letter score is evaluated over the supplied choices and also includes the final token ID of the opening-parenthesis form, such as `(A`. Open-answer RealworldQA items use greedy generation of at most eight tokens and exact matching after lowercasing and removing punctuation, articles, and extra whitespace.

Performance and data use. Answer accuracy measures agreement between the predicted option and the ground-truth label. Crop localisation is evaluated using the box coverage defined in Sec. 3.4. Where a binary coverage hit is reported, a crop must cover at least half of the enclosing box. We

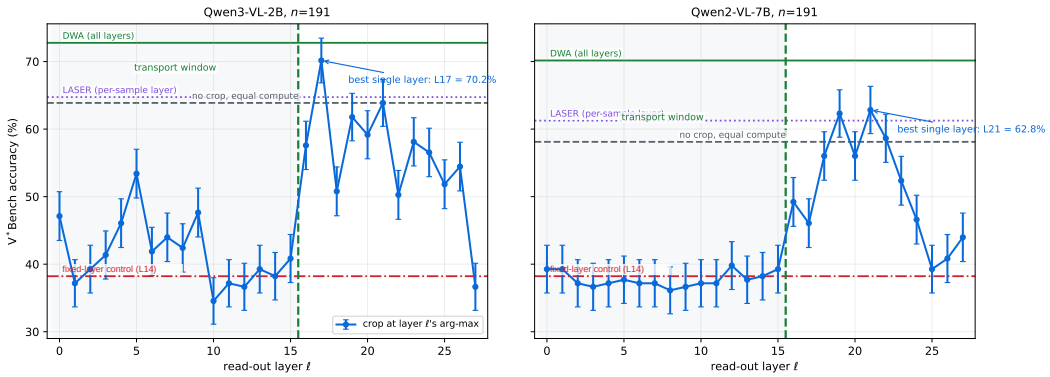

Figure 4: **Accuracy when only the crop read-out layer changes.** The layer sweep uses 191 V*Bench examples per checkpoint; error bars show one standard error. The fixed-layer and LASER (Zhu et al., 2026) lines denote the placement controls defined in Sec. A. The dashed boundary is the approximate transition identified by masking.

draw the boundary-analysis examples from V*Bench without a separate held-out split for boundary selection.

Resolution targets and computation accounting. We resize images with bicubic interpolation toward a target visual-token count and check the count using the model's image processor. We allow up to six resizing attempts and keep the closest candidate. The relative stopping tolerance is 8% for the depth analysis and DWA, and 6% for AVR. Target counts therefore need not equal realised counts.

The reference full-image input targets 600 visual tokens. DWA's budget includes both passes described in Sec. 3.4. For an AVR token-removal schedule, the visual token-layer proxy is

$$C_{\mathrm{TL}} = N(b+1) + N_{\mathrm{keep}}(L-b-1). \tag{20}$$

Here, C_{TL} counts visual tokens multiplied by the number of layers through which they would be processed. The first term represents full-width processing through layer b, and the second represents processing of retained tokens afterward. For exactly 900 input tokens, $b = 16$, $L = 28$, and $N_{\mathrm{keep}} = 90$, this gives $16{,}290$ visual token-layers, compared with $16{,}800$ for exactly 600 tokens across all 28 layers. We compute this proxy using the realised token counts.

This proxy quantifies the visual-token processing of the specified token-removal schedule. Vision-encoder and text-token processing lie outside its scope. The masking evaluation in Sec. 3.5 measures the schedule's effect on accuracy while retaining full-length token tensors; C_{TL} is the cost assigned to physical token removal.

B Ablations

B.1 Choice of Read-out Depth

We vary the attention layer while keeping the crop size, answering procedure, and visual-token budget fixed. Fig. 4 shows a 35.6-point accuracy range on Qwen3-VL-2B and a 26.7-point range on Qwen2-VL-7B. The best layers are 17 and 21, respectively. Every tested read-out before layer 16 performs below the uncropped reference on both checkpoints. This separates the depth at which image access affects the answer from the depth at which the final prompt position supplies a useful spatial map.

DWA exceeds the best single layer by 2.6 points on Qwen3-VL-2B and 7.3 points on Qwen2-VL-7B. Only the latter difference has a paired interval excluding zero. Since the best layer is selected using benchmark accuracy, this is an optimistic single-layer reference.

B.2 What Does the Fitted Read-out Contribute?

Weight signs. Constraining the 56 log-attention and rank coefficients to be non-negative produces no statistically significant end-task loss relative to DWA on either core checkpoint. The spatial coefficients and intercept remain unconstrained. The corresponding localisation fits set 44 and 43 of the 56 layer coefficients to zero, concentrating weight on a sparse subset of layers.

A training-free aggregation control. Taking the maximum over layers 17–20 on Qwen3-VL-2B and 19–22 on Qwen2-VL-7B gives 68.9% and 60.7% accuracy, compared with DWA's 72.6% and 70.2% in the matched comparison. Fitting gives higher point accuracy on both checkpoints, with a significant advantage on Qwen2-VL-7B.

Fold assignment. Refitting with disjoint fold seeds changes pooled accuracy by at most one point on each checkpoint, with both paired intervals including zero. DWA is therefore stable across the tested grouped partitions.

B.3 AVR Cutoff, Retained Tokens, and Input Resolution

Cutoff and token ranking. Before the measured transition, attention-ranked retention exceeds random retention by 8.4 and 8.9 points in the reported comparison. At later cutoffs, the ranking advantage decreases. In the full Qwen3-VL-2B V*Bench run, attention and random retention give 70.2% and 68.6% accuracy and agree on 96.3% of answers. A schedule that begins reducing visual access within the sensitive depth range loses 1.0 and 5.8 points relative to the boundary-based schedule. These controls support retaining visual access through the sensitive range and applying the restriction after the transition.

Higher resolution versus restriction alone. We compare AVR with the unmodified model at the same 900-token input resolution. Across the ten evaluated benchmark–checkpoint–stratum combinations, seven accuracy differences are within 0.5 points. Two are positive by 1.3 and 2.6 points, while HR-Bench 8K cross-instance shows a significant loss of 1.8 points $[-3.2, -0.5]$. Later restriction thus keeps accuracy close to the unrestricted higher-resolution model across the evaluated settings.

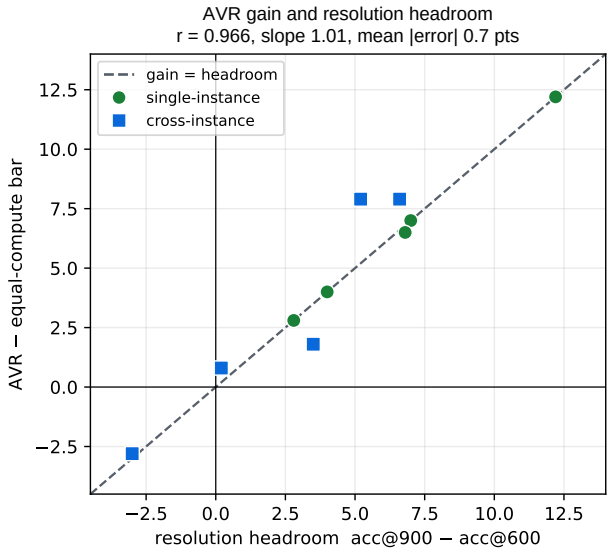

Figure 5: **AVR gain and the benefit of higher input resolution.** Each point is one of ten reported benchmark–checkpoint–stratum combinations with an unmodified 900-token reference. The horizontal axis measures its accuracy gain over 600 tokens; the vertical axis measures AVR's gain over that same reference budget. The diagonal indicates equal gains. Both axes share the same 600-token reference.

Table 6: **The encoding cliff.** Single-region localised items (n=63, Qwen3-VL-2B). The largest below/above step occurs at $t = 0.15$.

t	n below	acc below	acc above	step
0.10	16	18.8%	72.3%	+53.6
0.15	21	**19.0%**	**78.6%**	**+59.5**
0.25	33	33.3%	86.7%	+53.3
0.50	46	47.8%	88.2%	+40.4
1.00	57	56.1%	83.3%	+27.2

Fig. 5 shows that AVR's gain tracks the benefit of increasing input resolution, with a reported correlation of 0.966 and a mean absolute difference of 0.7 points. Together with the same-resolution control, this identifies additional input detail as the main source of AVR's accuracy gain. The later restriction sets the nominal visual-access allowance that permits that larger input under the token-layer accounting.

C THE SIZE CLIFF

Protocol. All measurements use with ground-truth boxes on Qwen3-VL-2B and Qwen2-VL-7B unless stated otherwise. A *merged visual token* is one post-merge patch embedding. At a 300-token budget, a target with image-area fraction S occupies SB merged tokens, where B is the realised token count. The *oracle crop* replaces the model's window with the ground-truth box at the same token budget and is used only as a control. Thresholds are obtained by a sliding below/above sweep, with at least ten items per side; intervals are paired item bootstraps with 8000 resamples.

The encoding cliff. Under uniform encoding, accuracy is near chance for the smallest targets, falls *below* the 25% chance level, and then rises sharply through the 0.15–0.25 merged-token range. The oracle crop remains flat at 95.8%/100.0% across the same range. On the cleanly localised single-region subset, the largest step occurs at $t = 0.15$; across all single-region items, both models exhibit a transition at approximately 0.25 merged tokens.

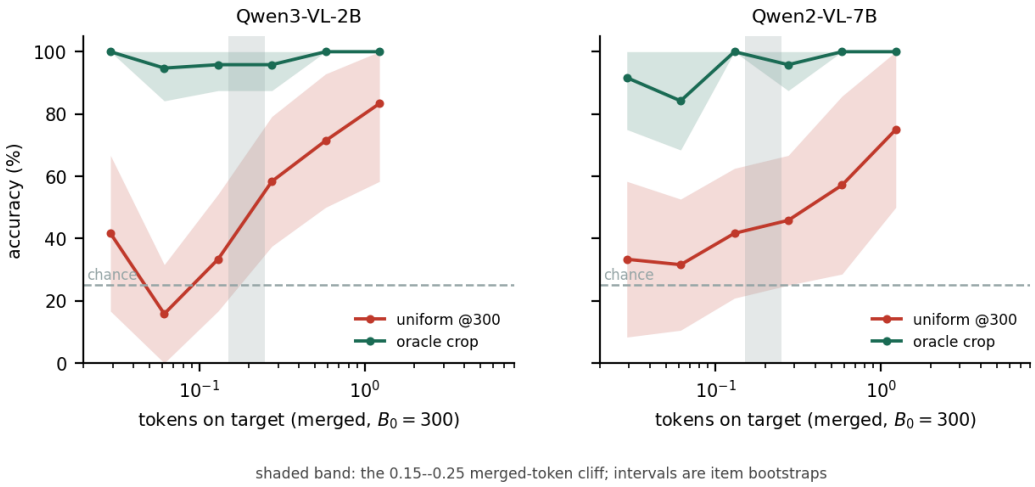

Figure 6: **The encoding cliff.** Accuracy vs. target extent in merged tokens (B_0=300). Red: uniform encoding; green: oracle crop at the same budget. The uniform curve falls below chance before rising sharply through the 0.15–0.25 merged-token range, while the oracle remains flat.

The cliff is architecture-invariant. Across all single-region items (n=115 per model), both Qwen3-VL-2B and Qwen2-VL-7B exhibit a 0.25-token transition. Uniform accuracy changes from 32.4%

Table 7: **The cliff is architecture-invariant.** All single-region items, n=115 each. The same 0.25-token cliff appears in both models, while the oracle crop remains flat across the split.

model	cliff	uniform below	uniform above	oracle below	oracle above
Qwen3-VL-2B	0.25 tok	32.4%	77.3%	95.8%	100.0%
Qwen2-VL-7B	0.25 tok	36.6%	63.6%	95.8%	100.0%

to 77.3% on Qwen3 and from 36.6% to 63.6% on Qwen2, while the oracle crop remains at 95.8% below and 100.0% above the threshold.

The threshold is relative, not budget-limited. Increasing the uniform budget from 300 to 588 tokens shifts the Qwen3 threshold from 0.25 to 0.55 merged tokens, but its image-side fraction remains nearly unchanged (2.89% to 3.06%). Qwen2 is noisier but follows the same direction. Accuracy also does not collapse when indexed by tokens-on-target: in the $[0.15, 0.25)$ bin it changes from 33.3% to 53.8% on Qwen3 and from 41.7% to 61.5% on Qwen2. Thus the limiting quantity is the target's *relative size*, not its absolute token count.

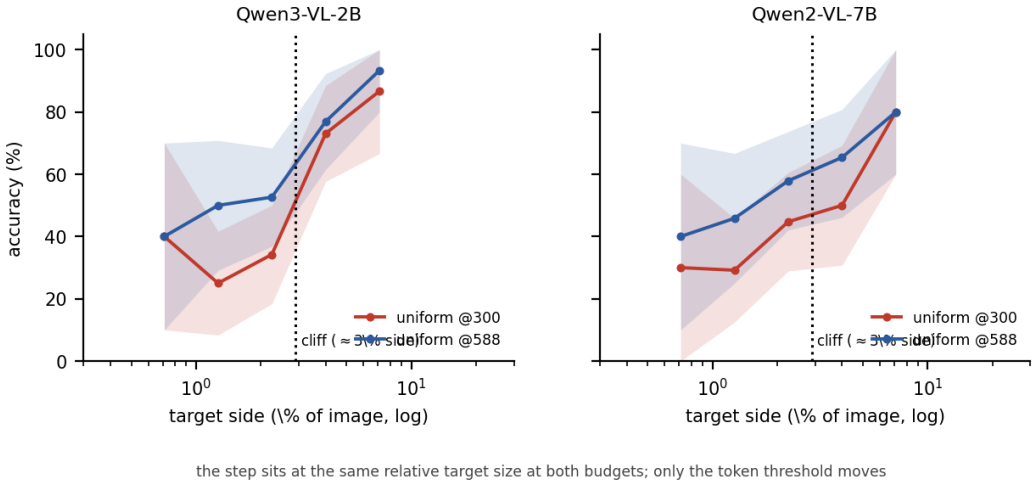

Figure 7: **The cliff does not move with budget.** Increasing the uniform budget shifts the threshold in merged-token units but leaves the critical target size at approximately 3% of the image side.

At W=0.15, the oracle resolves 100% of items in every size quartile, showing that once the region is isolated, magnification is sufficient. The bottleneck is therefore the uniform encoding of targets whose relative extent falls below the cliff.

D The Information Is Never Encoded

The answer is not in the stack. We define a *certified encoding failure* as an item that is wrong under uniform encoding but correct under the oracle crop at the same budget. Layer-wise logit-lens read-outs test whether the missing answer appears at some intermediate depth and is subsequently lost. Under uniform encoding, no layer exceeds 52.9% accuracy. On localised-but-wrong items (n=36), the correct option is the layer-wise argmax on 72.2% of items, compared with 68.1% when the window misses the target. The apparent signal at L20 (41.7%) collapses to 0% from L22. There is therefore no layer at which a reliable answer is present and subsequently lost.

The oracle produces the opposite pattern: the layer-wise read-out reaches 92.6%, with the answer appearing abruptly at L22 and accounting for 90% of the gain in a single transition. The same transition occurs one layer later on Qwen2-VL-7B and only when the crop covers the target (+23.8

points when covered, exactly 0.0 when it misses). The crop thus supplies information that the fixed-depth language model can use; it does not merely improve downstream reasoning.

Representation patching provides an independent check. Replacing image tokens with those from another image at L16 changes the output no more than replacing irrelevant early text, with KL values of 0.009 vs. 0.009 on Qwen3 and 0.005 vs. 0.006 on Qwen2. Across the benchmark, certified encoding failures account for 37.7%/41.4% of all items and 86.7%/84.0% of all errors; at W=0.15, these fractions rise to 94.0%/89.4%.

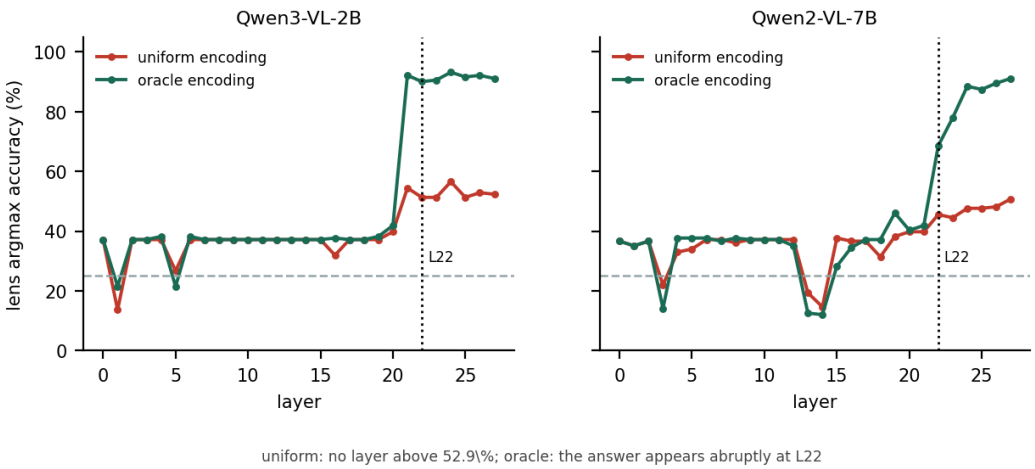

Figure 8: **The answer is not in the stack.** Under uniform encoding, no layer exceeds 52.9% and no intermediate layer contains an answer that is subsequently lost. Under the oracle crop, the answer becomes decodable abruptly at L22, accounting for most of the gain.

At the 300-token pass, the median single-region target has a side of 0.210 merged tokens and occupies 0.176 tokens of area; 100% of targets are smaller than one merged token and 62% fall below the 0.25-token cliff. The target is therefore averaged with its surrounding background before the language model receives it, providing the geometric explanation for the encoding failure.

E Allocation Beats Budget

Allocation vs. budget. We compare an oracle crop at the base budget against the best uniform allocation available to each model. A random crop of the same size measures the placement contribution. An oracle crop at 300 tokens reaches 97.4% on Qwen3-VL-2B and 93.2% on Qwen2-VL-7B, while no uniform rung up to 7,776 tokens reaches the crop arm; Qwen2's uniform ceiling is 75.4%. The resulting exchange rates are therefore at least 26× and 25.9×, respectively. The placement effect alone is +40.0 and +59.2 points.

The same effect persists at native resolution: a perfectly placed 600-token crop beats native dynamic resolution at 3,290 tokens by +11.6/+16.2 pooled. The benefit has a clear boundary. Allocation helps when the target is smaller than the uniform budget can resolve: 71% of targets are sub-token and oracle allocation yields +37.2 points. When targets are already resolvable, discarded context can dominate instead; on HR-Bench 4K, allocation loses 8.8 points. Thus the value of allocation depends on whether the evidence lies below the resolution supported by the uniform encoding.

We also tested multi-crop extensions under matched compute. Adding the original image costs 1.5× the bar and reduces the pooled gain from +10.5 to +4.2 points while reaching −10.5 on the cross stratum. A funded two-crop variant improves the single stratum (80.9 vs. 62.6) but does not pass the pre-registered cross criterion (68.4 vs. 65.8, 0 of 2 criteria). Thus additional context does not recover the single-crop advantage under the same compute constraint.

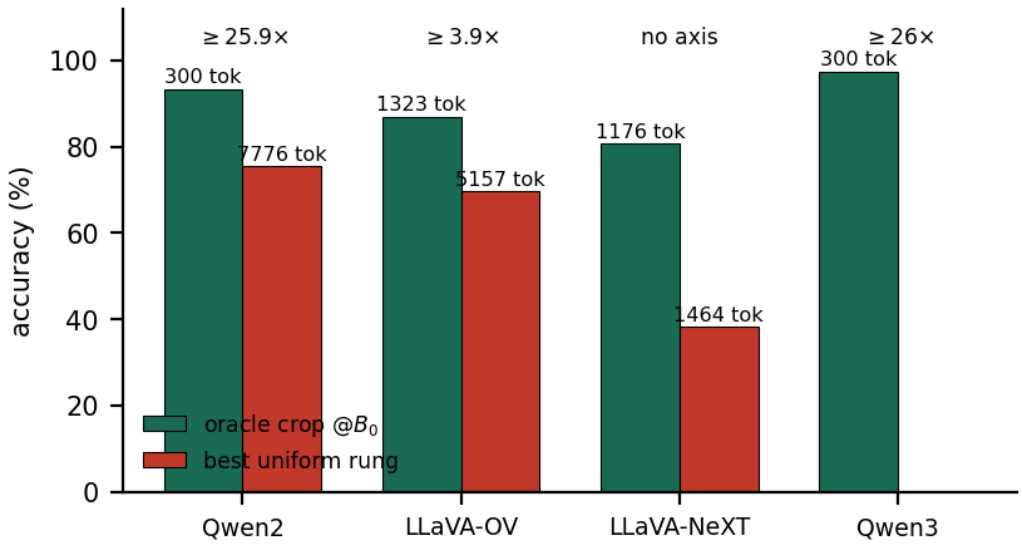

Figure 9: **Allocation vs. budget.** Oracle crop at the base budget against the best uniform rung reached by each model. No uniform rung matches the crop, making each exchange rate a lower bound.

Table 8: **Allocation vs. budget.** Placement effect is oracle crop minus random crop.

model	oracle crop	random crop	placement effect	uniform ceiling	exchange rate
Qwen2-VL-7B	93.2% @300	34.0%	+59.2	75.4% @7,776	$\geq 25.9\times$
LLaVA-OneVision-7B	86.9% @1,323	37.7%	+49.2	69.6% @5,157	$\geq 3.9\times$
LLaVA-NeXT-7B	80.6% @1,176	38.2%	+42.4	38.2% @1,464	no axis
Qwen3-VL-2B	97.4% @300	—	+40.0	—	$\geq 26\times$

F THE TRANSPORT MEASUREMENT IN FULL

A forward pre-hook on every decoder layer adds -10^4 to the attention logits of chosen query rows over the image-token columns, at chosen layers only. Masking *all* text rows at *all* layers destroys the answer (KL 1.266 / 0.532, 42% / 45% of answers flip, n=40); masking only the answer token's row at all layers does not (KL 0.025, 3% flip). Table 9 localises the window.

Table 9: Prefix and suffix masks, all text rows, $n = 40$ per model, V*Bench at 300 tokens. The prefix mask $L_0..\ell$ saturates at the full-mask value once ℓ passes the window; the suffix mask $L_\ell..$end falls to zero once ℓ is past it. Both read the boundary at the same place, and the 36-layer checkpoint puts it at the same fraction of depth (ℓ=20 of 36).

	ℓ=4	ℓ=8	ℓ=12	ℓ=16	ℓ=20	ℓ=24
Qwen3 prefix $L_0..\ell$ (KL)	0.082	0.549	**1.278**	1.259	1.215	1.257
Qwen3 suffix $L_\ell..$ (KL)	0.967	0.880	0.320	**0.014**	0.002	0.001
Qwen3 suffix (answers flipped)	53%	50%	20%	12%	**0%**	**0%**
Qwen2 prefix $L_0..\ell$ (KL)	0.037	0.105	0.364	**0.547**	0.534	0.538
Qwen2 suffix $L_\ell..$ (KL)	0.518	0.465	0.429	**0.029**	0.004	0.002
Qwen2 suffix (answers flipped)	53%	47%	40%	5%	3%	**0%**
Qwen3-8B (36L) prefix $L_0..\ell$ (KL)	0.100	0.309	1.190	**1.740**	1.559	1.565
Qwen3-8B (36L) suffix $L_\ell..$ (KL)	1.671	1.538	1.278	0.413	**0.052**	0.009
Qwen3-8B (36L) suffix (answers flipped)	35%	45%	40%	15%	5%	**0%**

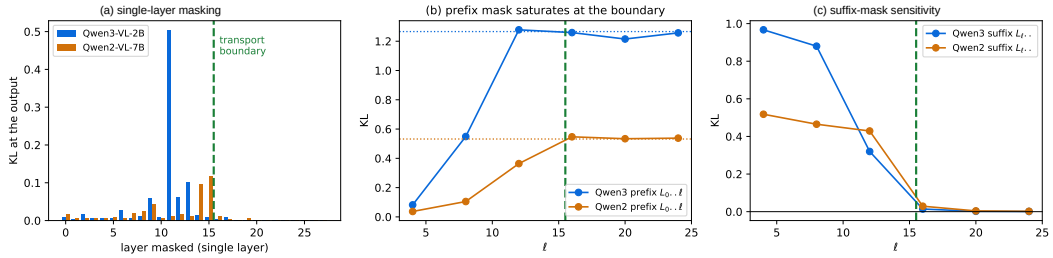

Figure 10: **The transport measurement.** (a) Masking one layer's image attention at a time: the effect is concentrated well before the boundary and is at or below 0.0023 from L20 on both models. (b) Masking a prefix $L_0..\ell$ reaches the full-mask value (dotted) once ℓ passes the window. (c) Masking a suffix $L_\ell..$end costs nothing once ℓ is past it. Three readings, one boundary.

Single-layer masks peak at L11 on Qwen3 (KL 0.504; next highest L13 at 0.101) and at L15/L14 on Qwen2 (0.118/0.096). From L20 onward every layer is at or below 0.0023 on both models. On Qwen3-VL-8B, which has 36 layers, the same behaviour appears at L21, i.e. the same 0.57 fraction of depth.

Three independent readings of the same boundary. The masks are one. Pruning is the second: discarding 90% of visual tokens at the boundary costs $+0.0$ to $+0.5$ points across ten stratum-cells, whereas moving the same cut to ℓ=4 costs 5.2 more on one model (0.5 on the other), and a taper that removes half the tokens at ℓ=9–12 loses 3.5 points against not pruning at matched tokens while buying nothing over a single cut (-1.0). Read-out depth is the third: restricting the localiser to layers $\leq$ L20 costs -0.5 and $+1.6$ points of coverage, while restricting it to $\leq$ L16 costs 9.4 and 22.0. Transport finishes at the boundary; the signal the ranking reads is produced after it.