CELLGAIN: ONE-SHOT ANSWER-METRIC PRICING FOR HARD-BUDGET VISUAL TOKEN ALLOCATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Fine-detail visual question answering presents a discrete acquisition problem before expensive encoding: a 6×6 page lattice offers 36 cheap 16-token refinements and 36 detailed 64-token refinements, but a hard cap admits only a question-dependent subset. CellGain assigns these 72 choices their expected answer-metric gain across partial visual states. Offline reveal-and-redecode experiments shape the target with the residual between privileged dense and matched partial-view regressors; deployment compresses that computation into one query-conditioned map from a 22M-parameter scout and compiles the map into a feasible mixed-scale set. With InternVL2-8B and 1,008 optional fine tokens, CellGain reaches 82.7/52.1/74.8/63.8 on DocVQA, InfographicVQA, ChartQA, and ScreenQA—within 0.4/0.7/0.4/0.9 points of dense 2,304-token inference. At the same mean allocation, it exceeds Query-Window by 0.8/1.2/1.7/1.3 points; all four Holm-corrected paired intervals exclude zero. Under an identical cell interface, residual supervision leads observed utility, dense-teacher-only, and non-privileged pairwise targets, while the original map retains $\rho = 0.31$ correlation with remaining realized values after 16 reveals and leaves only 0.1–0.3 points to sequential re-scoring. The resulting 56.2% reduction in optional fine tokens lowers model-side latency from 912 to 468 ms/image, and retraining the valuation mechanism on Qwen2-VL-7B-Instruct preserves the four-task advantage.

1 INTRODUCTION

Consider the token menu created by a 6×6 page grid. Every cell can remain coarse, receive 16 Level-1 tokens, or receive 64 Level-2 tokens, with at most one refinement level per cell. The dense menu costs 2,304 optional tokens; the operating point studied here enforces a 1,024-token per-example cap, with validation-selected admission thresholds producing a 1,008-token test mean for matched comparisons. A feasible allocation must therefore resolve two coupled choices—*where* to spend and *at what scale*—before the main vision encoder constructs the fine features. This geometry recurs in document, infographic, chart, and interface QA (Singh et al., 2019; Methani et al., 2020; Chen et al., 2021; Mathew et al., 2021; Masry et al., 2022; Hsiao et al., 2025), where the answer may depend on one small label, tick, icon, or field.

The decision variable required by this menu is the marginal answer value of a cell–scale pair. Visual saliency emphasizes conspicuous content; query relevance emphasizes correspondence; both can prefer evidence already resolved by the coarse view. The allocation geometry instead asks whether adding a particular 16- or 64-token group to the current state changes the decoded answer. This places the technical boundary before fine encoding. Post-encoding mergers pay the complete fine-feature acquisition cost, while early within-ViT pruning still admits every fine patch to initial vision layers (Bolya et al., 2023; Rao et al., 2021; Liang et al., 2022; Chen et al., 2024a; Gao et al., 2026); global selectors choose one resolution for an input (Kimhi et al., 2026); and crop policies search over a different, often overlapping proposal space (Wu & Xie, 2024; Shi et al., 2026; Shabtay et al., 2026).

CellGain turns the 72-way menu into a supervised valuation problem. For a partially revealed page, it measures the official benchmark-score change caused by adding one legal cell refinement. Sampling prefixes at four depths exposes the same action under different amounts of already available evidence. A privileged regressor that sees the dense candidate feature and a matched regressor restricted to the

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.

partial state supply a visibility residual around that measured change. The resulting target represents answer value across the rollout states in which an action may be considered.

Deployment then collapses the offline experiments into one pass. A ViT-S/16 scout reads the coarse page and question, predicts all 72 values simultaneously, and ranks positive value per token. A compiler admits a mutually exclusive mixed-scale set under the cap; the selected crops enter the frozen vision tower in level-homogeneous batches. Dense candidate features, answer labels, and counterfactual re-decoding are training-only. Thus the deployed mechanism is a single pre-encoding map rather than an iterative crop-and-answer loop.

The experiments follow this geometry from valuation to system outcome. At the policy level, final cell coverage gives every implemented pre-encoding allocator the same 16/64-token prices and separates initially encoded tokens from later retained tokens. At the mechanism level, four targets share the same 72 actions, deployed inputs, architecture, tuples, compiler, and evaluation examples. At the amortization level, realized action values are remeasured after 4, 8, and 16 reveals and compared with sequential re-scoring. The operating point uses 1,008 tokens, reaches 82.7/52.1/74.8/63.8 on the four task metrics, and improves over the strongest matched window allocator by 0.8–1.7 points with Holm-corrected paired intervals excluding zero. It also removes 56.2% of optional fine tokens, cuts InternVL2-8B model-side latency by 48.7%, and preserves the four-task ordering when the full valuation mechanism is retrained with Qwen2-VL-7B-Instruct.

The primary contribution is the train-to-deployment transformation: partial-state, answer-metric counterfactuals become a one-shot pre-encoding value map over a priced multi-scale lattice. Its residual target transfers candidate visibility available during training while retaining the measured reveal effect for each tuple. A controlled empirical package establishes the consequence of that design through same-interface target isolation, matched cell-price comparisons on four benchmarks, a five-budget frontier, rollout diagnostics, full latency accounting, and replication across two VLM families.

2 Related Work

Adaptive visual computation controls token survival, depth, or sampling (Ryoo et al., 2021; Yin et al., 2022; Fayyaz et al., 2022; Bolya et al., 2023). ToMe, FastV, and LLaVA-PruMerge reduce an activation sequence after dense tokenization (Bolya et al., 2023; Chen et al., 2024a; Shang et al., 2025). QuietPrune moves query-guided pruning into early ViT layers and groups adjacent tokens to preserve structure (Gao et al., 2026); these tokens still enter the vision encoder before pruning. Geometry-driven systems such as LLaVA-NeXT, InternVL2, and Monkey determine coverage from aspect ratio or tiling rules (Liu et al., 2024; Chen et al., 2024c; Li et al., 2024). Document-specialized models such as LayoutLMv3, Donut, mPLUG-DocOwl 1.5, and Pix2Struct improve structured visual-text representations (Huang et al., 2022; Kim et al., 2022; Hu et al., 2024; Lee et al., 2023). CellGain acts at a distinct cost boundary: it decides which fine groups the vision encoder will create.

Input-side methods expose complementary acquisition units. CARES predicts one resolution for an image–query pair from multi-resolution convergence labels (Kimhi et al., 2026), while ResAdapt distributes query-aware resolution across video frames using accuracy–cost reward (Liao et al., 2026). V* searches proposed windows, Q-Zoom localizes an adaptive region through a self-distilled gate, and AwaRes retrieves high-resolution crops through a learned multi-turn tool policy (Wu & Xie, 2024; Shi et al., 2026; Shabtay et al., 2026). CROP localizes a contiguous region and then compresses or prunes post-encoder features (Guo et al., 2025). CellGain exposes all cell–level choices simultaneously, assigns each a fixed token price, and learns a separate answer effect for every choice across partial states.

Counterfactual cell values connect the method to value-of-information and active perception (Howard, 1966; Bajcsy, 1988; Foster et al., 2021; Mnih et al., 2014; Ba et al., 2015). The dense candidate branch is also related to learning with privileged information and teacher–student transfer (Vapnik & Vashist, 2009; Hinton et al., 2015). CellGain uses that privilege specifically as a residual between matched regressors around an observed task-metric delta, rather than as a deployment input or an output-distribution target. Exact cell value requires another VLM decode for each state–action pair, so the technical contribution is to amortize many such experiments into one complete map. Query-Window provides the closest finite-window control in the shared implementation: it observes

Table 1: Where adjacent methods make their efficiency decision. Implemented pre-encoding comparators pay for final cell coverage; published methods with native global, frame, crop, or within-ViT actions retain their original interface. "Pre-ViT" means before any fine patch enters the main vision encoder.

Method family	Query-cond.	Pre-ViT	Unit	Budget rule	Signal
ToMe / FastV	No	No	token	retention	No
QuietPrune	Yes	No	group	retention	self-distill
Geometry tilers	No	Yes	window	fixed	No
CARES	Yes	Yes	resolution	adaptive	convergence
ResAdapt	Yes	Yes	frame	adaptive	reward
V* / AwaRes	Yes	Yes	crop	search/tool	search/RL
CROP PLC/ILP	Yes	No	region/token	retention	No
Q-Zoom	Yes	Yes	ROI	adaptive	self-distill
Crop-Rerank	Yes	Yes	proposal	hard	No
Query-Window	Yes	Yes	window	hard	observed
CellGain	Yes	Yes	cell	hard	residual

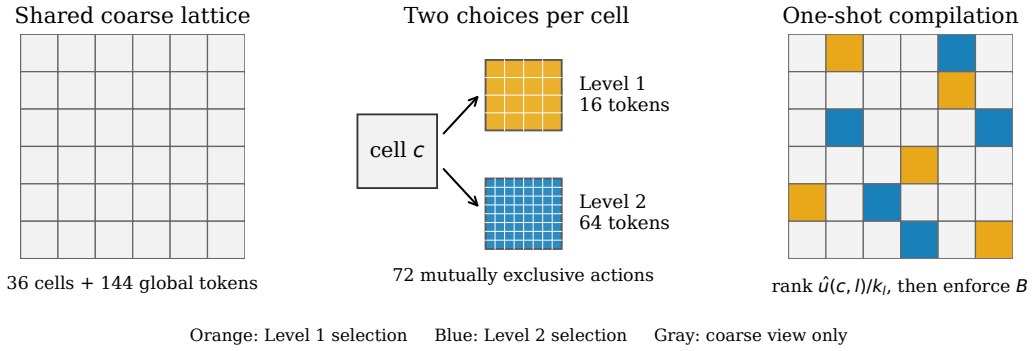

Figure 1: CellGain's decision geometry. A shared coarse view defines 36 cells; each cell offers one mutually exclusive 16-token Level-1 or 64-token Level-2 action. The scout values all 72 actions jointly, and the compiler selects mixed-scale coverage before fine patches enter the frozen vision tower.

coarse inputs and answer deltas but ranks overlapping windows. Crop-Rerank begins from Edge Boxes (Zitnick & Dollár, 2014), exposing the combined effect of proposal coverage and ranking under the same final cell prices.

3 CellGain: Pricing Cellwise Evidence Before Encoding

3.1 Action Lattice, Tokenization, and Answer Value

InternVL2-8B combines InternViT-300M-448px, a frozen MLP projector, and InternLM2.5-7B (Chen et al., 2024b). We disable native dynamic tiling and apply one deterministic transform to every method: convert to RGB, resize directly to each square canvas with bicubic interpolation, and apply the checkpoint channel normalization. The three canvases are Level 0 (336×336), Level 1 (672×672), and Level 2 (1344×1344). With 14-pixel patches, they contain 24^2, 48^2, and 96^2 patch positions. The checkpoint's 0.5-ratio pixel-unshuffle reduces these grids to $12^2 = 144$, $24^2 = 576$, and $48^2 = 2{,}304$ projected visual tokens.

The 6×6 boundaries align exactly with every canvas. A Level-1 action crops a 112×112 half-open cell region and yields a 4×4 post-unshuffle grid; a Level-2 action crops the corresponding 224×224 region and yields 8×8. Crops receive no extra context or padding. We interpolate the vision positional grid to the crop's local patch grid, encode all selected crops in at most two level-homogeneous batches, and pass them through the frozen projector. Their projected tokens then receive learned global row,

column, and level embeddings, are ordered by cell raster position and level, and are concatenated after the cached 144 Level-0 tokens. Thus a Level-1 cell costs 16 optional tokens and a Level-2 cell costs 64; sparse budgets exclude the mandatory Level-0 view.

An action $a = (c, l)$ retains Level 0 and adds one fine representation of cell c at level $l \in \{1, 2\}$. Fine-level choices are mutually exclusive: a policy selects at most one level for each cell, and a Level-2 action costs 64 tokens rather than a cumulative $16 + 64$. For feasible set $\mathcal{S}$,

$$\sum_{(c,l)\in\mathcal{S}} k_l \le B, \qquad \sum_{l\in\{1,2\}} \mathbb{1}[(c, l) \in \mathcal{S}] \le 1 \quad \forall c, \tag{1}$$

where $k_1 = 16$ and $k_2 = 64$. This definition aligns configured caps, realized tokens, and patch encoding.

Let state $s = I_0 \cup \mathcal{P}$ contain the global view and an already revealed prefix. The frozen VLM greedily decodes $\hat{y}_\theta(q, s)$, and each benchmark metric $m_t \in [0, 1]$ defines $\ell_t(q, s) = 1 - m_t(\hat{y}_\theta(q, s), y^*)$. The exact state-action utility is

$$u(a \mid q, s) = \ell_t(q, s) - \ell_t(q, s \oplus a). \tag{2}$$

Positive utility is therefore the measured improvement from one reveal, with ANLS used for DocVQA and InfographicVQA, relaxed accuracy for ChartQA, and exact match for ScreenQA. Tables report $100m_t$, so score differences are percentage points. Deployment requires one score per action although training tuples span partial states. CellGain therefore learns the prefix-marginalized, residual-corrected surrogate

$$\bar{u}_{\mathrm{RC}}(a \mid q, I_0) = \mathbb{E}_{s\sim P_{\mathrm{prefix}}(\cdot|q,I_0)}[y_{\mathrm{RC}}(a, s)], \tag{3}$$

where the expectation is over sampled training prefixes and y_{RC} is defined in Eq. 5. This is a one-step reveal value marginalized over rollout states, rather than a cumulative planning objective. The observed-only control replaces the corrected target with exact $u(a \mid q, s)$. Inference sorts positive $\hat{u}_\phi(a \mid q, I_0)/k_l$ and greedily admits feasible actions while satisfying Eq. 1.

3.2 One-Shot Value Map and Budget Compilation

CellGain produces the complete reveal-value map in one scout pass. A ViT-S/16 scout (22M parameters) maps the 336×336 view to a 21×21 patch grid and adaptive-average-pools it to 6×6 cell features. A learned linear layer projects each pooled feature to 256 dimensions. The frozen VLM's mean-pooled question-token representation is projected from 768 to 256 dimensions; learned row, column, and level embeddings identify the action coordinate and scale. The MLP in Eq. 4 is a two-layer 256-unit GELU regressor with scalar output and is reused unchanged in every same-interface target control:

$$\hat{u}_\phi(c, l) = \mathrm{MLP}([f_{\mathrm{scout}}(c); e_q; e_{\mathrm{pos}}(c); e_l; e_{\mathrm{aux}}(c)]) . \tag{4}$$

Selected patches are encoded by the frozen vision tower, tagged with their original two-dimensional cell coordinates and level embedding, ordered in raster order, and concatenated with Level-0 tokens through the backbone's existing vision-to-language projector.

The auxiliary vector has four operational components. Answer entropy is the Level-0 decoder entropy multiplied by normalized coarse query-to-cell attention, making it cell-specific without a fine encode. OCR contributes box geometry only: text strings are discarded, boxes are mapped through the same square transform, and n_c counts boxes whose centers fall in cell c. The density feature is $\log(1 + n_c)/\log(1 + \max_j n_j)$ and is zero when OCR is disabled or returns no boxes. Edge complexity is mean Sobel magnitude per cell, normalized within the page. Query-cell attention is the mean coarse cross-attention from question tokens to pooled scout patches. Each scalar passes through its own two-layer projection to 16 dimensions; concatenating the four outputs gives the 64-dimensional $e_{\mathrm{aux}}(c)$. Every signal is available from the coarse pass or preprocessing; the no-OCR variant in Section 4.5 isolates the OCR contribution.

3.3 Learning Across Partial Visual States

Training amortizes counterfactual experiments performed across many rollout prefixes. To sample $s \sim P_{\mathrm{prefix}}$, we draw a prefix length uniformly from $\{0, 4, 8, 16\}$, draw that many distinct cells

without replacement, assign each drawn cell Level 1 or 2 with equal probability, and uniformly permute the resulting feasible actions. For each state, eight candidate actions are drawn from unrevealed cells with equal mass over the two levels. Offline tuple generation greedily decodes before and after each candidate reveal and evaluates Eq. 2.

Let $h(s)$ contain the frozen query vector, pooled Level-0 cell sequence, mean-pooled projected features of cells already present in the prefix, the 36-cell level/feasibility mask, and the mean-pooled hidden state of the current draft decode. Each sequence-valued component is projected to 256 dimensions before pooling. For candidate $a = (c, l)$, a privileged regressor g_ψ receives $h(s)$, action position and level embeddings, and the dense Level-2 feature at c; a partial-view regressor $g_{\bar{\psi}}$ has the identical two-layer 256-unit GELU head but replaces that feature with the representation available in s (Level 0 for an unrevealed cell). Both regressors predict exact utility by least squares, use the example-disjoint training/validation partition described below, and are selected by validation MSE before their weights are frozen. Their detached outputs construct

$$y_{\mathrm{RC}}(a, s) = u(a \mid q, s) + \mathrm{sg}\big[g_\psi(a, q, s; I_{\mathrm{full}}) - g_{\bar{\psi}}(a, q, s)\big] . \tag{5}$$

Because the two regressors share state inputs and prediction architecture, $g_\psi - g_{\bar{\psi}}$ isolates the value associated with candidate evidence visible only to the dense branch. Adding this visibility residual to u preserves the measured answer effect for every tuple while supplying information hidden from the partial representation. Its prefix expectation is the target in Eq. 3. CellGain minimizes MSE between $\hat{u}_\phi(a \mid q, I_0)$ and y_{RC}. Predictions and targets are clipped to $[-1, 1]$ before loss evaluation. A calibration term partitions predictions into 15 equal-width bins and averages the squared difference between mean prediction and mean target over nonempty bins; the total loss is MSE plus $\lambda_{\mathrm{cal}}\mathcal{L}_{\mathrm{cal}}$, with $\lambda_{\mathrm{cal}} = 0.1$. Dense features, both regressors, and ground-truth answers are confined to offline training.

The objective controls retain the same 72 actions, scout inputs, architecture, training tuples, budget compiler, and evaluation examples. Observed-only supervision uses u directly; dense-teacher-only replaces y_{RC} with $\mathrm{sg}[g_\psi(a, q, s; I_{\mathrm{full}})]$; and the non-privileged pairwise control minimizes $\log(1 + \exp[-z(\hat{u}_a - \hat{u}_b)])$ for non-tied same-prefix action pairs, where $z = \mathrm{sign}(u_a - u_b)$, without dense features or teacher targets. The separate feature controls retain residual supervision while removing auxiliaries or query conditioning. Table 3 therefore distinguishes target effects within a fixed interface from feature effects within the residual objective.

The sampler forms 1,800,000 training and 200,000 validation tuples from official training splits. Examples are partitioned 90/10 before any tuple generation, preventing one source image from contributing to both partitions. Tuples are mixed uniformly over the four tasks and then uniformly over source examples within each task. The two regressors are optimized first with AdamW at 3×10^{-4}, cosine decay, batch size 128, and a 20K-step maximum; validation MSE checked every 1K steps selects each head. We then freeze them and optimize the scout with the same optimizer, learning rate, schedule, batch size, and 20K-step horizon. Validation mean Spearman correlation selects the scout checkpoint. Training uses $8\times$A100 80GB GPUs with CUDA 12.4; InternVL2-8B remains frozen, and benchmark test examples are evaluation-only.

The sequential re-scoring ablation uses the same tuple targets and head width but conditions on the current partial-state summary used by $g_{\bar{\psi}}$ and an updated draft-decoder state. It recomputes remaining-action values after every admitted reveal. This state-conditioned variant measures the accuracy and latency recovered by replacing one-shot amortization with iterative inference; it is separate from the deployed map $\hat{u}_\phi(a \mid q, I_0)$.

4 Experiments

4.1 Evaluation Along the Allocation Pipeline

Evaluation mirrors Figure 1. The policy comparison measures complete allocators after all outputs are charged for final cell coverage. The target study then freezes the 72-action interface and changes only supervision or deployed features. Rollout analysis remeasures the original map after the state has changed; systems and backbone experiments test whether the selected geometry translates into realized computation and a second representation family.

Algorithm 1 CellGain inference

Require: Query q, image I, scout π_ϕ, frozen VLM f_θ, budget B
1: Encode $I_0 \leftarrow \mathrm{encode}(I, 336 \times 336)$ and cache coarse tokens
2: Run the Level-0 draft decode and form query, position, level, and auxiliary features
3: **for** each $(c, l) \in \{1, \ldots, 36\} \times \{1, 2\}$ **do**
4: $\quad \hat{u}(c, l) \leftarrow \pi_\phi(c, l, q, I_0)$
5: **end for**
6: Sort positive actions by $\hat{u}(c, l)/k_l$
7: Admit actions greedily subject to budget and one-level-per-cell feasibility
8: Encode selected patches in level-homogeneous batches and decode with f_θ

We evaluate DocVQA (ANLS, 5,188 test examples) (Mathew et al., 2021), InfographicVQA (ANLS, 3,288) (Mathew et al., 2022), ChartQA (relaxed accuracy, 2,500) (Masry et al., 2022), and ScreenQA (exact match, 4,612) (Hsiao et al., 2025). Reported task scores are $100m_t$, and differences are percentage points. The primary checkpoint is InternVL2-8B with InternViT-300M-448px (Chen et al., 2024b). Every method uses the same frozen checkpoint and projector, task prompt, greedy decoder, square normalization, test examples, and 6×6 accounting grid.

Full-Res adds Level-2 representations for all cells. Uniform-9 adds nine evenly spaced Level-2 cells. Entropy-Alloc and Saliency-Alloc rank the same two-level cell candidates by Level-0 entropy-attention and Sobel energy, apply validation-selected level calibration and score thresholds, and greedily admit score per token without exceeding the 1,024-token cap. ToMe and FastV begin from all 2,304 fine patch tokens and reduce the active sequence to 960 and 864 tokens for later multimodal processing; Table 2 records both initial and retained counts. Crop-Rerank creates 128 Edge Boxes proposals from the Level-0 image, pools coarse vision features inside each proposal, and scores each with a two-layer 256-unit GELU query cross-attention reranker. Non-maximum suppression at IoU 0.7 precedes admission. Each admitted proposal reveals intersected cells at Level 2, proposal unions are charged by their final cell coverage, and a score threshold selected on the held-out validation partition controls the operating point. Both native and matched variants skip a proposal whose dynamic cost exceeds the remaining 1,024-token cap; the matched threshold is separately selected for a 1,008-token mean.

Query-Window enumerates all 36 one-cell and 25 adjacent 2×2 windows at both refinement levels (122 candidates). Its two-layer 256-unit cross-attention scorer uses the same query and Level-0 visual features as CellGain, without CellGain's OCR, entropy, edge, or residual-teacher inputs, and is trained with pairwise ranking on observed one-window task-metric deltas. Admission recomputes score per dynamic marginal cost after every selection: each window updates covered cells to the higher of their current and proposed levels, pays only the increase in final cell prices, skips zero-cost or over-budget candidates, and stops at the same 1,024-token per-example cap. This rule charges overlaps once and leaves one final level per cell.

Allocator and reranker components use seeds 11, 23, 37, 53, and 71. Table 2 and the learned transfer rows report controller-seed means and standard deviations; the target study reports five-seed means under its fixed interface. For the primary CellGain–Query-Window contrasts, paired 95% confidence intervals use 10,000 shared example-level bootstrap resamples after averaging each example over matched seeds, and Holm correction covers the four tasks. Seed SD and paired intervals therefore expose controller and test-example variation separately. Token columns distinguish fine tokens entering the visual encoder from tokens retained for language processing. Latency uses one A100 80GB GPU, batch size one, synchronized CUDA timing, 100 warm-up examples, and at least 1,000 timed examples per dataset.

4.2 Shared-Price Frontier: Near-Dense Quality at 43.8% of Fine Tokens

CellGain is the strongest sparse operating point on every task. Against Query-Window at the same 1,008-token mean, its gains are 0.8, 1.2, 1.7, and 1.3 points; paired 95% intervals are [0.4,1.2], [0.6,1.8], [0.9,2.5], and [0.7,1.9]. Every comparison remains significant after Holm correction (largest corrected $p = 0.008$). Against Crop-Rerank retuned to the same mean, CellGain gains 1.1/1.5/2.3/1.7 points. Relative to Full-Res, it removes 1,296 of 2,304 optional fine tokens while preserving all but

Table 2: Frontier on InternVL2-8B. Sparse pre-encoding allocators use a 1,024-token per-example cap; Full-Res, ToMe, and FastV are accounting references outside that cap. "Enc./Ret." are dataset-average optional fine tokens initially encoded and retained for later multimodal processing; the common 144 Level-0 tokens are excluded. Learned rows show mean±controller-seed SD over five seeds. Scores are $100m_t$: ANLS, ANLS, relaxed accuracy, and exact match. Bold marks the strongest budgeted pre-encoding result.

Method	Enc.	Ret.	DocVQA	InfoVQA	ChartQA	ScreenQA
Full-Res	2304	2304	83.1	52.8	75.2	64.7
Uniform-9	576	576	77.8	44.6	66.8	57.3
Entropy-Alloc	720	720	79.9	47.8	69.8	59.1
Saliency-Alloc	684	684	79.1	47.1	68.2	58.4
ToMe	2304	960	80.4	49.1	70.3	60.8
FastV	2304	864	79.8	48.2	69.4	59.6
Crop-Rerank (native)	792	792	80.8±0.19	49.4±0.25	71.1±0.23	61.2±0.28
Crop-Rerank (matched)	1008	1008	81.6±0.18	50.6±0.24	72.5±0.22	62.1±0.27
Query-Window (matched)	1008	1008	81.9±0.15	50.9±0.23	73.1±0.20	62.5±0.24
CellGain	1008	1008	**82.7±0.12**	**52.1±0.20**	**74.8±0.17**	**63.8±0.23**
CellGain-Light	648	648	81.4±0.14	50.3±0.21	72.9±0.18	61.7±0.25

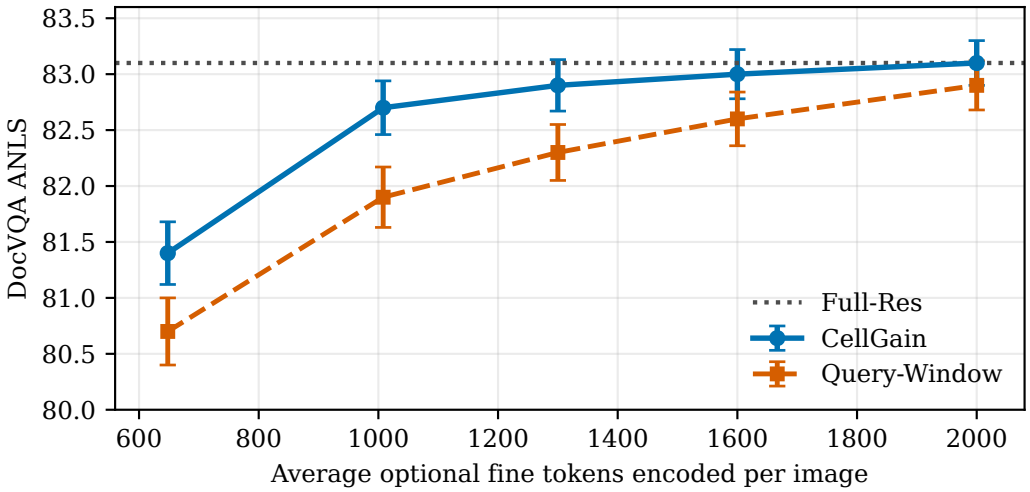

Figure 2: DocVQA frontier over optional fine tokens. Curves show five-seed means with example-level 95% bootstrap intervals; the horizontal reference is Full-Res. CellGain dominates Query-Window at all five evaluated budgets and matches the rounded 83.1 dense score at 2,000 rather than 2,304 optional tokens.

0.4/0.7/0.4/0.9 task points. ToMe and FastV shorten later activation sequences but begin from all dense fine patches, separating retention efficiency from the acquisition saving measured here.

Figure 2 shows a consistent frontier shift rather than a single tuned crossing. CellGain leads Query-Window by 0.7 points at 648 tokens, 0.8 at 1,008, and 0.2 at 2,000, where it matches the rounded 83.1 dense reference with 304 fewer optional tokens. Additional tokens reduce the cost of allocation error, but the ordering persists throughout the measured range. Raw paired outcomes and plotted coordinates are provided in Appendix A.

4.3 Residual Supervision Identifies High-Value Cells

Table 3 supplies two controlled contrasts. In the first four rows, the 72 actions, deployed inputs, scout, optimization, tuples, compiler, and examples are identical; only supervision changes. Residual correction improves over observed utility by 0.9/1.1/0.9/1.0 points and over the non-privileged pairwise

Table 3: Value-map component isolation under the shared 1,024-token cap. The first four rows hold cell actions, deployed features, scout architecture, tuples, and compiler fixed while changing supervision. The next two retain residual supervision and remove deployed features. Scores are five-seed means; tokens are test-example mean±SD; ρ is Spearman correlation with realized one-patch reveal value.

Target / variant	Doc	Info	Chart	Screen	Tokens	ρ
CellGain full	**82.7**	**52.1**	**74.8**	**63.8**	1008±11.6	**0.43**
Observed utility	81.8	51.0	73.9	62.8	998±14.1	0.37
Dense teacher only	82.1	51.2	74.0	63.1	1006±12.9	0.39
Pairwise, non-privileged	82.2	51.4	74.1	63.0	1004±13.7	0.40
Residual, no auxiliaries	81.9	50.7	73.4	62.4	1015±8.2	0.38
No query conditioning	80.8	49.1	71.8	60.9	1022±3.1	0.31
Random allocation	78.4	46.3	67.2	57.8	1010±10.4	—

objective by 0.5/0.7/0.7/0.8. It also improves over the dense-teacher-only target by 0.6/0.9/0.8/0.7 points and raises realized-value correlation from 0.39 to 0.43, showing the benefit of anchoring privileged transfer to the measured reveal delta. The next block fixes residual supervision: removing auxiliaries costs 0.8/1.4/1.4/1.4 points, while removing query conditioning costs 1.9/3.0/3.0/2.9 and lowers ρ to 0.31. Random allocation loses 4.3/5.8/7.6/6.0 points at comparable token use. The consistent four-task margins link the final map to both the visibility residual and question-conditioned coarse evidence.

4.4 One Map Retains Value Through 16 Reveals

The one-shot ranking remains informative as selected evidence changes the visual state: ρ moves from 0.43 before selection to 0.40, 0.36, and 0.31 after 4, 8, and 16 refinements. At each depth, we compute Spearman correlation over feasible remaining actions per example and macro-average over examples, tasks, and controller seeds. The state-conditioned sequential variant adds 31 ms/image and changes DocVQA, InfographicVQA, ChartQA, and ScreenQA by +0.2, +0.3, +0.2, and +0.1 points. The resulting prefix-marginalized map leaves only 0.1–0.3 points to that iterative variant while preserving a single controller pass.

4.5 Token Savings Halve Model-Side Latency

Token reduction matters operationally only when it shortens the realized model path. Table 4 separates routing, visual processing, and decoding, then restores preprocessing and OCR in the end-to-end total.

CellGain's OCR-enabled end-to-end median is 534 ms/image versus 936 for Full-Res, while its model-side path is 468 versus 912 ms/image, a 48.7% reduction. Model-side Crop-Rerank and Query-Window require 548 and 506 ms/image at 1,008 mean tokens. Disabling OCR reduces end-to-end latency to 486 ms/image while retaining 82.5/51.8/74.4/63.4 on the four task metrics. OCR contributes 0.2–0.4 points to CellGain, while the no-OCR map remains ahead of Query-Window on every task.

4.6 The Ordering Replicates with Qwen2-VL

We retrain the complete valuation mechanism on a second VLM family rather than reusing an InternVL controller.

Applying the same action topology, objective, and pricing rule to Qwen2-VL-7B-Instruct reproduces the ordering. CellGain stays 0.6/0.8/0.7/0.9 points from dense and improves over Query-Window by 0.8/1.3/1.5/1.3 points. The distinct absolute scores and task gaps show that the frontier effect persists under a different representation family.

Table 4: Operational cost on A100 80GB at batch size one, pooled over all four test sets. The upper block reports median ms/image; Enc./Ret. exclude Level-0, Visual includes the tower, reduction, projector, and prefill, and model-side latency is Route+Visual+Decode. The lower block reports end-to-end median [95% bootstrap CI], p95, and peak memory including preprocessing and OCR. CellGain routing includes its Level-0 draft pass.

Method	Enc./Ret.	Pre.	OCR	Route	Visual path	Decode
Full-Res	2304/2304	24	—	—	651	261
ToMe	2304/960	24	—	—	330	148
FastV	2304/864	24	—	—	312	130
Crop-Rerank	1008/1008	27	—	55	340	153
Query-Window	1008/1008	32	—	39	322	145
CellGain	1008/1008	9	57	31	295	142
CellGain, no OCR	1008/1008	18	—	31	293	144
CellGain-Light	648/648	9	56	27	213	114

Method	End-to-end median [95% CI]	p95	Peak GB
Full-Res	936 [927,945]	1088	19.8
ToMe	502 [495,509]	590	18.9
FastV	466 [459,473]	552	18.5
Crop-Rerank	575 [568,582]	681	14.4
Query-Window	538 [531,545]	632	13.6
CellGain	534 [528,540]	621	12.9
CellGain, no OCR	486 [480,492]	572	12.7
CellGain-Light	419 [414,424]	492	10.8

Table 5: Replication on Qwen2-VL-7B-Instruct (Wang et al., 2024). Its effective 28-pixel stride yields 144/576/2,304 tokens for the three levels; fine-token counts exclude common Level-0, so the sparse point is 43.8% of dense. Controllers are retrained with Qwen features using their original objectives and the shared topology; learned rows show five-seed mean±SD.

Method	Fine tokens	DocVQA	InfoVQA	ChartQA	ScreenQA
Full-Res	2304	84.6	55.2	77.0	66.5
Query-Window	1008	83.2±0.17	53.1±0.24	74.8±0.21	64.3±0.27
CellGain	1008	**84.0±0.14**	**54.4±0.21**	**76.3±0.18**	**65.6±0.24**

5 Conclusion

CellGain translates partial-state reveal experiments into a one-shot answer-value map over 72 priced cell refinements. At 1,008 optional fine tokens, it stays within 0.4–0.9 points of dense InternVL2-8B, exceeds the matched window allocator by 0.8–1.7 points on four benchmarks, and reduces model-side latency from 912 to 468 ms/image. Same-interface target isolation, rollout re-scoring, and retraining on Qwen2-VL attribute this outcome to query-conditioned, residual-supervised pre-encoding valuation. Within the evaluated fixed-grid regime, 1,296 fine tokens per image can therefore be withheld before the vision tower while retaining near-dense answer quality.

Reproducibility Statement

Sections 3–4 specify the action distribution, image and token path, objectives, model heads, training schedule, datasets, seeds, comparators, statistical procedure, hardware, and accounting rules. The appendix records paired counts, plotted frontier coordinates, and category slices.

AI USE STATEMENT

Generative AI tools assisted with manuscript preparation, including prose editing, formatting, and bibliographic consistency checks. The authors reviewed all AI-assisted material and take responsibility for the final manuscript and its claims.

REFERENCES

Jimmy Lei Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.7755.

Ruzena Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, 1988. doi: 10.1109/5.5968. URL https://doi.org/10.1109/5.5968.

Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JXg7fW9D3n.

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

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 *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 4173–4185, 2021. doi: 10.18653/v1/2021.emnlp-main.343. URL https://aclanthology.org/2021.emnlp-main.343/.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, Lixin Gu, Xuehui Wang, Qingyun Li, Yimin Ren, Zixuan Chen, Jiapeng Luo, Jiahao Wang, Tan Jiang, Bo Wang, Conghui He, Botian Shi, Xingcheng Zhang, Han Lv, Yi Wang, Wenqi Shao, Pei Chu, Zhongying Tu, Tong He, Zhiyong Wu, Huipeng Deng, Jiaye Ge, Kai Chen, Kaipeng Zhang, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024b. doi: 10.48550/arXiv.2412.05271. URL https://arxiv.org/abs/2412.05271.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024c. URL https://openaccess.thecvf.com/content/CVPR2024/html/Chen_InternVL_Scaling_up_Vision_Foundation_Models_and_Aligning_for_Generic_CVPR_2024_paper.html.

Mohsen Fayyaz, Soroush Abbasi Koohpayegani, Farnoush Rezaei Jafari, Sunando Sengupta, Hamid Reza Vaezi Joze, Eric Sommerlade, Hamed Pirsiavash, and Juergen Gall. Adaptive token sampling for efficient vision transformers. In *Proceedings of the European Conference on Computer Vision*, pp. 396–414, 2022. doi: 10.1007/978-3-031-20083-0_24. URL https://doi.org/10.1007/978-3-031-20083-0_24.

Adam Foster, Desi R. Ivanova, Ilyas Malik, and Tom Rainforth. Deep adaptive design: Amortizing sequential Bayesian experimental design. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 3384–3395, 2021. URL https://proceedings.mlr.press/v139/foster21a.html.

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

URL https://openaccess.thecvf.com/content/CVPR2026/html/Gao_QuietPrune_Query-Guided_Early_Token_Pruning_for_Vision-Language_Models_CVPR_2026_paper.html.

Jiawei Guo, Feifei Zhai, Pu Jian, Qianrun Wei, and Yu Zhou. CROP: Contextual region-oriented visual token pruning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9767–9783. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.492. URL https://aclanthology.org/2025.emnlp-main.492/.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. doi: 10.48550/arXiv.1503.02531. URL https://arxiv.org/abs/1503.02531.

Ronald A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, 2(1):22–26, 1966. doi: 10.1109/TSSC.1966.300074. URL https://doi.org/10.1109/TSSC.1966.300074.

Yu-Chung Hsiao, Fedir Zubach, Gilles Baechler, Srinivas Sunkara, Victor Carbune, Jason Lin, Maria Wang, Yun Zhu, and Jindong Chen. ScreenQA: Large-scale question-answer pairs over mobile app screenshots. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 9427–9452, 2025. doi: 10.18653/v1/2025.naacl-long.477. URL https://aclanthology.org/2025.naacl-long.477/.

Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. mPLUG-DocOwl 1.5: Unified structure learning for OCR-free document understanding. In *Findings of the Association for Computational Linguistics: EMNLP*, pp. 3096–3120, 2024. doi: 10.18653/v1/2024.findings-emnlp.175. URL https://aclanthology.org/2024.findings-emnlp.175/.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. LayoutLMv3: Pre-training for document AI with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 4083–4091, 2022. doi: 10.1145/3503161.3548112. URL https://doi.org/10.1145/3503161.3548112.

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. In *Proceedings of the European Conference on Computer Vision*, 2022. doi: 10.1007/978-3-031-19815-1_29. URL https://doi.org/10.1007/978-3-031-19815-1_29.

Moshe Kimhi, Nimrod Shabtay, Raja Giryes, Chaim Baskin, and Eli Schwartz. CARES: Context-aware resolution selector for VLMs. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2243–2256. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.102.

Kenton Lee, Mandar Joshi, Iulia Raluca Turc, Hexiang Hu, Fangyu Liu, Julian Martin Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2Struct: Screenshot parsing as pretraining for visual language understanding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 18893–18912, 2023. URL https://proceedings.mlr.press/v202/lee23g.html.

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26753–26763, 2024. doi: 10.1109/CVPR52733.2024.02527. URL https://doi.org/10.1109/CVPR52733.2024.02527.

Youwei Liang, Chongjian Ge, Zhan Tong, Yibing Song, Jue Wang, and Pengtao Xie. Not all patches are what you need: Expediting vision transformers via token reorganizations. In *International Conference on Learning Representations*, 2022. URL https://iclr.cc/virtual/2022/poster/6168.

Huanxuan Liao, Zhongtao Jiang, Yupu Hao, Yuqiao Tan, Shizhu He, Ben Wang, Jun Zhao, Kun Xu, and Kang Liu. ResAdapt: Adaptive resolution for efficient multimodal reasoning. *arXiv preprint arXiv:2603.28610*, 2026. doi: 10.48550/arXiv.2603.28610. URL https://arxiv.org/abs/2603.28610.

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

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 *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, 2022. 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 *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pp. 2199–2208, 2021. doi: 10.1109/WACV48630.2021.00225. URL https://doi.org/10.1109/WACV48630.2021.00225.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and C.V. Jawahar. InfographicVQA. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2582–2591, 2022. doi: 10.1109/WACV51458.2022.00264. URL https://doi.org/10.1109/WACV51458.2022.00264.

Nitesh Methani, Pritha Ganguly, Mitesh M. Khapra, and Pratyush Kumar. PlotQA: Reasoning over scientific plots. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pp. 1516–1525, 2020. doi: 10.1109/WACV45572.2020.9093523. URL https://doi.org/10.1109/WACV45572.2020.9093523.

Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, volume 27, 2014. URL https://papers.nips.cc/paper_files/paper/2014/hash/3e456b31302cf8210edd4029292a40ad-Abstract.html.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. In *Advances in Neural Information Processing Systems*, volume 34, pp. 13937–13949, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/747d3443e319a22747fbb873e8b2f9f2-Abstract.html.

Michael S. Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. TokenLearner: Adaptive space-time tokenization for videos. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12786–12797, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/6a30e32e56fce5cf381895dfe6ca7b6f-Abstract.html.

Nimrod Shabtay, Moshe Kimhi, Artem Spector, Sivan Haray, Ehud Rivlin, Chaim Baskin, Raja Giryes, and Eli Schwartz. Look where it matters: High-resolution crops retrieval for efficient VLMs. *arXiv preprint arXiv:2603.16932*, 2026. doi: 10.48550/arXiv.2603.16932. URL https://arxiv.org/abs/2603.16932.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive token reduction for efficient large multimodal models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22857–22867, 2025. doi: 10.1109/ICCV51701.2025.02122. URL https://doi.org/10.1109/ICCV51701.2025.02122.

Yuheng Shi, Xiaohuan Pei, Linfeng Wen, Minjing Dong, and Chang Xu. Q-Zoom: Query-aware adaptive perception for efficient multimodal large language models. *arXiv preprint arXiv:2604.06912*, 2026. doi: 10.48550/arXiv.2604.06912. URL https://arxiv.org/abs/2604.06912.

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

Vladimir Vapnik and Akshay Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 22(5–6):544–557, 2009. doi: 10.1016/j.neunet.2009.06.042. URL https://doi.org/10.1016/j.neunet.2009.06.042.

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. *arXiv preprint arXiv:2409.12191*, 2024. doi: 10.48550/arXiv.2409.12191. URL https://arxiv.org/abs/2409.12191.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13084–13094, 2024. doi: 10.1109/CVPR52733.2024.01243. URL https://doi.org/10.1109/CVPR52733.2024.01243.

Hongxu Yin, Arash Vahdat, Jose M. Alvarez, Arun Mallya, Jan Kautz, and Pavlo Molchanov. A-ViT: Adaptive tokens for efficient vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10799–10808, 2022. doi: 10.1109/CVPR52688.2022.01054. URL https://doi.org/10.1109/CVPR52688.2022.01054.

C. Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, pp. 391–405, 2014. doi: 10.1007/978-3-319-10602-1_26.

A Paired Uncertainty and Frontier Coordinates

Table 6 reports answer-level paired outcomes for CellGain versus Query-Window. Ties include examples on which both systems receive the same metric value; bootstrap intervals operate on the underlying continuous ANLS/relaxed-accuracy differences rather than on win counts.

Table 6: CellGain wins/losses/ties against Query-Window at 1,008 mean optional fine tokens. Denominators are the complete test sets.

Dataset (denominator)	Wins	Losses	Ties
DocVQA (5,188)	392	347	4449
InfographicVQA (3,288)	301	260	2727
ChartQA (2,500)	276	233	1991
ScreenQA (4,612)	423	363	3826

Table 7: DocVQA values used in Figure 2. CI entries are symmetric half-widths from example-level 95% bootstrap intervals.

Optional tokens	648	1008	1300	1600	2000
CellGain mean	81.4	82.7	82.9	83.0	83.1
CellGain CI half-width	0.28	0.24	0.23	0.22	0.20
Query-Window mean	80.7	81.9	82.3	82.6	82.9
Query-Window CI half-width	0.30	0.27	0.25	0.24	0.22

B Category Slices

The aggregate advantage also appears across semantically distinct question groups. Figure 3 compares the nearest methods on chart forms and interface elements at the shared operating point, exposing whether the mean is dominated by one visual category.

CellGain's gains over Query-Window are 0.8, 0.9, 1.1, 1.0, and 0.9 points across the five ChartQA categories, and 1.1, 1.2, 1.3, 1.3, and 1.3 points across the five ScreenQA categories.

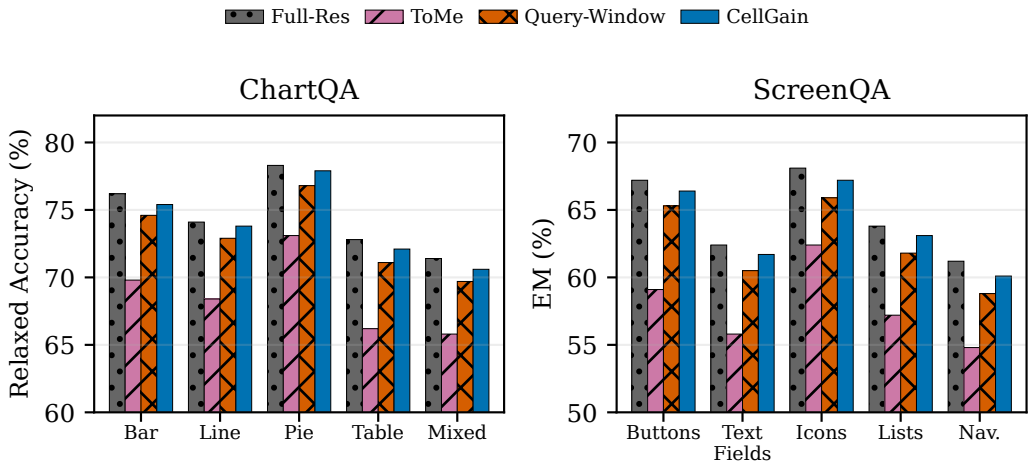

Figure 3: ChartQA relaxed accuracy and ScreenQA exact match by category. Full-Res uses 2,304 initial/retained fine tokens, ToMe uses 2,304/960, and Query-Window and CellGain use 1,008/1,008. CellGain exceeds the matched Query-Window comparator in every displayed category.