QUERY-ADAPTIVE RECURRENT TOKENIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual information worth extracting from an image and the token budget it warrants depend on the text query. Yet modern vision and vision-language models largely encode images independently of the query and allocate a fixed visual token budget to every input. We introduce the Query-Adaptive Recurrent Tokenizer (QART), a backbone-agnostic module that makes visual representations both query-*steerable* and budget-*adaptive*. QART steers the visual representation toward query-relevant evidence while adapting its token budget to the amount of information required. Its key principle is simple. Additional tokens are allocated until the visual features relevant to the query are represented with sufficient fidelity. Starting from frozen encoder features, QART recurrently constructs a compact representation and uses reconstruction quality over the query-relevant visual support to control token allocation. On Visual7W and VQAv2 , QART achieves **81.96%** and **76.37%** accuracy using only about 29 visual tokens, reducing token counts by **20–25**$\times$ relative to dense VLM inputs. It further reaches 92.0% P@1 on CORE and 36.3% PR-AUC on PODS, outperforming SteerViT on PODS by 8.4%. Together, these results show that visual representations need not be fixed in either content or capacity and can instead adapt to what each query requires.

1 INTRODUCTION

> *"A wealth of information creates a poverty of attention and a need to allocate that attention efficiently among the overabundance of information sources that might consume it."*
>
> — Herbert A. Simon (1971)

An image contains evidence for many possible questions, and the visual information that matters depends on what is being asked Anderson et al. (2018); Hudson & Manning (2019); Pantazopoulos & Özyiğit (2025). In the same street scene, identifying the color of a traffic light requires evidence from a small region, whereas counting vehicles turning left requires information distributed across objects and their spatial relationships (Acharya et al., 2018; Rajabi & Košecká, 2024; Ishihara et al., 2025; Huang et al., 2026; Sadeghi et al., 2026). Yet SOTA vision encoders (Tschannen et al., 2025; Radford et al., 2021; Oquab et al., 2023; He et al., 2022) produce the same representation of an image regardless of the query (Figure 1(a)). These representations capture broadly useful visual information but cannot explicitly prioritize the requested evidence (Ruthardt et al., 2026). More importantly, visual salience is not akin to query relevance: a prominent object may be irrelevant to the question, while a small detail may determine the answer. A useful visual representation should therefore be *steerable*: changing the query should change which visual information the representation emphasizes.

A separate limitation concerns how much representational capacity is allocated to that evidence. Visual tokenizers commonly produce a fixed-size sequence; ViT-based encoders (Oquab et al., 2023; Tschannen et al., 2025) provides a fixed set of patch tokens, while the Q-Former uses a fixed number of latent tokens regardless of the image or the evidence required by the downstream task (Li et al., 2023). A large token budget increases downstream computation for every example (Yin et al., 2025; 2022; Li et al., 2026a), while a small budget may discard necessary details (Kim et al., 2026). These considerations expose two distinct axes of visual tokenization: which evidence to preserve and how much capacity to allocate (*adaptivity*), as shown in Fig. 1. These properties are increasingly relevant to embodied systems, where agents repeatedly interpret visual observations under changing goals,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

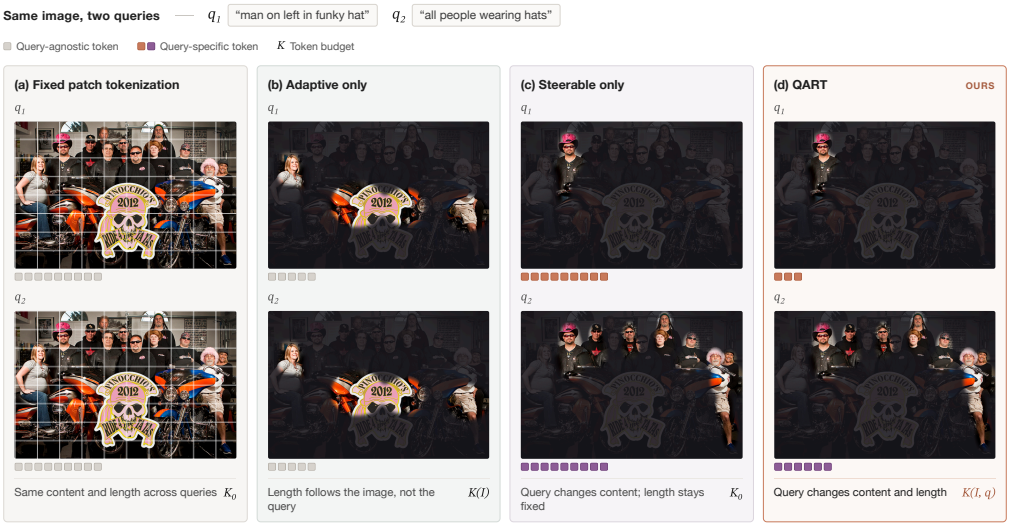


Figure 1: **QART combines query steerability with budget adaptivity.** Fixed patch tokenizers use the same representation and token budget for every query. Adaptive tokenizers change *how many* tokens are allocated, while steerable methods change *what* visual information is emphasized. QART does both, steering the representation toward the query while adapting its capacity to the required visual evidence. The clear regions represent the tokenizers' focus.

and efficient processing matters over long interaction horizons (Saxena et al., 2025; Cheng et al., 2024). We seek a tokenizer that connects both axes, allocating capacity according to how well the current representation preserves query-relevant visual evidence.

Prior work addresses parts of this problem. Learned resamplers create compact latent representations, steerable encoders condition visual features on language (Ruthardt et al., 2026), and token-pruning methods remove redundant or query-irrelevant patches (Li et al., 2026a; Takezoe et al., 2026). Adaptive image tokenizers additionally vary their sequence length according to image complexity (Duggal et al., 2025b; Yu et al., 2024). What remains less explored is an explicit mechanism that couples these choices: rather than prescribing a token count or predicting one directly, can a tokenizer continue allocating capacity until the evidence relevant to the current query has been represented with sufficient fidelity?

We introduce the Query-Adaptive Recurrent Tokenizer (QART), a visual tokenizer that converts dense encoder features into a variable-length set of query-conditioned visual tokens. Given frozen image features and a text query, a cross-modal relevance module first identifies the visual evidence relevant to the query. QART begins with a small set of tokens and recurrently refines them using this evidence and the query. After each round, a shared decoder reconstructs the relevant visual features, and reconstruction quality determines whether more tokens are added or computation halts. The resulting tokens form a compact representation whose content and capacity are both shaped by the query, and can be used by a downstream task head or language model.

We evaluate QART on prompt-conditioned retrieval (CORE (Ruthardt et al., 2026), GeneCIS (Vaze et al., 2023), PODS (Sundaram et al., 2025)), compositional matching (Winoground (Thrush et al., 2022)), Visual Question Answering (VQAv2 (Goyal et al., 2017)), and grounding (Visual7W pointing (Zhu et al., 2016). QART learns non-uniform allocations and compresses dense patch tokens by 25×. On CORE and GeneCIS, it improves over the query-agnostic SigLIP2 representation by 23.5% and 2.7% respectively. On Visual7W pointing, it outperforms the strongest token-reduction baseline DART Wen et al. (2025) by 22.1% under a matched protocol; on VQAv2, it retains 93.8% of dense accuracy while reducing tokens by 25× and outperforms a matched 32-token Q-Former, while improving end-to-end throughput by 1.53×. Together, these results highlight the significance of query-adaptive visual representations.

2 Related Work

2.1 Compact and Variable-Length Visual Tokenization

Vision encoders commonly represent an image as a dense grid of patch embeddings, motivating learned bottlenecks that summarize this grid with a smaller set of visual tokens (Bachmann et al., 2025). TokenLearner learns a compact set of content-adaptive tokens (Ryoo et al., 2021), while BLIP-2 uses a Querying Transformer (Q-Former) to map frozen image features into a fixed-size latent sequence for a downstream language model (Li et al., 2023). InstructBLIP further conditions the Q-Former on the input instruction, but retains a fixed latent budget (Dai et al., 2023). Matryoshka Query Transformer (MQT) supports multiple output lengths within one model by training nested prefixes of latent queries; however, the desired length is selected externally at inference time rather than inferred from the visual evidence required by each query (Hu et al., 2024).

Adaptive-length image tokenizers vary representation capacity with image content (Yan et al., 2024; Duggal et al., 2025a; Chen et al., 2026; Lu et al., 2026). ALIT, the closest architectural precedent to our work, recurrently updates a latent sequence, adds tokens over successive rounds, and uses reconstruction as a learning signal for variable-length image representations (Duggal et al., 2025b). QART changes the unit of adaptation from the image alone to the image–query pair. Its text-conditioned relevance model determines which patch features must be preserved, and the recurrent tokenizer halts when those query-relevant features have been reconstructed sufficiently well. Consequently, the same image can produce different token contents and lengths under different queries.

2.2 Query-Conditioned and Steerable Visual Representations

Other work studies how language can steer visual representation (Mao et al., 2025). InstructBLIP extracts instruction-specific features through an instruction-aware Q-Former (Dai et al., 2023); FLAIR applies text-conditioned attention pooling to local image features for fine-grained retrieval (Xiao et al., 2025); and TIE conditions the image encoder on the text query (Thirukovalluru et al., 2025). SteerViT injects language into intermediate vision-transformer layers to produce dense visual features that can be directed toward a requested concept (Ruthardt et al., 2026). QMoP uses the query to route visual features among pooling, resampling, and pruning branches (Li et al., 2026b).

These methods establish that useful visual evidence depends on the language context; QART is therefore not the first query-conditioned visual representation. Its distinction lies in coupling query conditioning to recurrent capacity allocation. Rather than only changing which information is emphasized within a dense or prescribed-size representation, QART jointly determines what information to encode and how many latent tokens are required. The token count follows from an explicit representation-adequacy criterion instead of a fixed budget or a separately predicted length.

2.3 Visual Token Pruning and Merging

Visual-token reduction is also widely studied through pruning and merging (Chen et al., 2024; Zhang et al., 2025b; Yang et al., 2025; Zhang et al., 2025a; Shang et al., 2025). ToMe reduces vision-transformer sequences by progressively merging similar tokens (Bolya et al., 2023). In vision-language systems, QG-VTC uses question–patch relevance to progressively compress visual tokens within the vision encoder (Li et al., 2025), while QuietPrune introduces query guidance early in the vision transformer so that pruning preserves prompt-relevant regions (Gao et al., 2026). ATP-LLaVA learns instance- and layer-dependent pruning ratios inside the language model (Ye et al., 2025); LearnPruner combines learned post-encoder reduction with query-guided pruning in intermediate language-model layers (Takezoe et al., 2026); and OccamToken derives image- and query-adaptive pruning thresholds without additional training (Li et al., 2026a).

These approaches primarily retain, merge, or discard existing tokens. QART instead constructs and recurrently expands a slot-based representation from frozen patch features until a query-conditioned reconstruction criterion is met, which supplies a shorter visual sequence to downstream models.

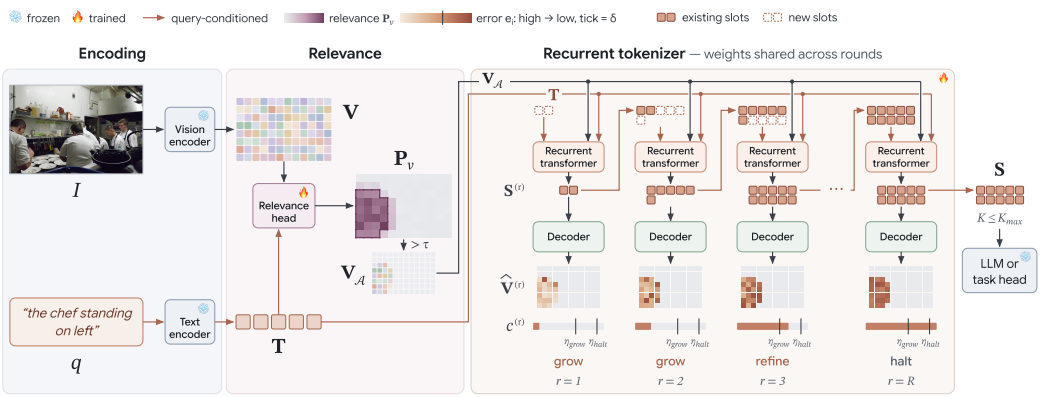


Figure 2: **Overview of QART.** Given an image and a text query, frozen encoders provide visual and linguistic features $\mathbf{V}$ and $\mathbf{T}$ respectively. A relevance module H_ϕ selects query-relevant patches $\mathbf{V}_\mathcal{A}$; QART then recurrently refines a small set of query-conditioned slots $\mathbf{S}$ by passing them along with $\mathbf{V}_\mathcal{A}$ and $\mathbf{T}$ to the shared transformer F_θ. After each round r, a shared decoder D_ψ reconstructs the visual features, and reconstruction quality determines whether more slots are added or recurrence halts. Only slot states propagate across rounds.

3 METHOD

QART converts dense encoder features into a variable-length sequence of query-conditioned visual tokens in three stages (Figure 2). A cross-modal relevance module selects query-relevant visual features (§3.1); a weight-shared recurrent Transformer then builds a slot-based representation from them, beginning with K_0 query-initialized slots and adding ΔK new ones when reconstruction coverage of the selected features falls below a coverage criterion (§3.2–3.3). The trainable components are optimized jointly using query-weighted reconstruction and relevance-prediction objectives (§3.4).

3.1 IDENTIFYING QUERY-RELEVANT EVIDENCE

Given an image $I \in \mathbb{R}^{H\times W\times 3}$ and text query q, QART represents the image–query pair as a variable-length set of steered visual tokens, or *slots*:

$$G_{\theta,\phi,\psi}(I, q) = \mathbf{S} \in \mathbb{R}^{K\times D}, \qquad K \le K_{\max}. \tag{1}$$

Frozen vision and text encoders produce visual features $\mathbf{V} = E_{\mathrm{v}}(I) \in \mathbb{R}^{N\times D_v}$ and text tokens $\mathbf{T} = E_{\mathrm{t}}(q) \in \mathbb{R}^{L\times D_t}$, projected to a shared dimension D. A cross-modal relevance module then predicts a relevance map $\mathbf{P}_{\mathrm{v}}$ which indicates how strongly each patch $\mathbf{v}_i$ corresponds to the query:

$$\mathbf{P}_{\mathrm{v}} = [p_1, p_2, ..., p_N]^T \in [0,1]^N, \qquad p_i = H_\phi(\mathbf{v}_i, \mathbf{T}), \qquad p_i \in [0,1]. \tag{2}$$

This relevance map is used to identify the query-relevant support $\mathcal{A} = \{i \mid p_i > \tau\}$ and the corresponding features $\mathbf{V}_\mathcal{A} = \{\mathbf{v}_i\}_{i\in\mathcal{A}}$. Having identified which evidence to preserve, the remaining problem is how to represent $\mathbf{V}_\mathcal{A}$ compactly without prescribing a fixed token count.

3.2 RECURRENT QUERY-CONDITIONED REPRESENTATION

To answer the question, QART builds on progressive token allocation (Duggal et al., 2025b). Specifically, a pooled query representation $\bar{\mathbf{q}}$ seeds K_0 initial slots through a learned projector, $\mathbf{S}^{(0)} = f_{\text{init}}(\bar{\mathbf{q}}) \in \mathbb{R}^{K_0\times D}$;these slots are then recurrently refined to gather evidence from $\mathbf{V}_\mathcal{A}$.

At each round r, the slots $\mathbf{S}^{(r-1)}$ interact with the selected visual features $\mathbf{V}_\mathcal{A}$ and text through a shared Transformer F_θ, followed by a cross-attention readout over $\mathbf{V}_\mathcal{A}$:

$$\widetilde{\mathbf{S}}^{(r)} = F_\theta\Big(\mathbf{S}^{(r-1)}, \mathbf{V}_\mathcal{A}, \mathbf{T}\Big), \qquad \mathbf{S}^{(r)} = \mathrm{LN}\Big(\widetilde{\mathbf{S}}^{(r)} + \mathrm{MHA}\Big(\widetilde{\mathbf{S}}^{(r)}, \mathbf{V}_\mathcal{A}, \mathbf{V}_\mathcal{A}\Big)\Big). \tag{3}$$

When the current set of slots cannot represent $\mathbf{V}_{\mathcal{A}}$ with sufficient fidelity , ΔK new slots are initialized from $\bar{\mathbf{q}}$ via f_{init} and concatenated to the existing set before the next round, growing the representation from K_r to $\min(K_r + \Delta K,\ K_{\max})$.

3.3 Reconstruction-Guided Token Allocation

QART ties representation sufficiency to feature reconstruction. Specifically, after each round, a shared decoder D_ψ reconstructs the full visual feature grid from the current slots, producing $\widehat{\mathbf{V}}^{(r)} = D_\psi(\mathbf{S}^{(r)})$. The per-patch reconstruction error is $|\widehat{\mathbf{v}}_i^{(r)} - \mathbf{v}_i\|_2^2/D$, evaluated only over $\mathcal{A}$. The fraction of support patches whose error has dropped below a threshold δ by round r defines the *reconstruction coverage*:

$$c^{(r)} = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \mathbb{I}\Big[|\widehat{\mathbf{v}}_i^{(r)} - \mathbf{v}_i\|_2^2/D < \delta\Big]. \tag{4}$$

If $c^{(r)} < \eta_{\text{grow}}$, ΔK new slots are added before round $r{+}1$:

$$K_{r+1} = \begin{cases} \min(K_r + \Delta K,\ K_{\max}), & c^{(r)} < \eta_{\text{grow}}, \\ K_r, & \text{otherwise.} \end{cases} \tag{5}$$

Recurrence halts once $c^{(r)} \geq \eta_{\text{halt}}$. Setting $\eta_{\text{grow}} < \eta_{\text{halt}}$ decouples the two decisions: slot addition can stop while the existing slots continue to refine. Because $\mathcal{A}$ depends on the query, the same image follows different allocation trajectories under different queries (App. C.3).

3.4 Learning Objective

Given soft relevance targets $\mathbf{h} \in [0,1]^N$ derived from segmentation or alignment annotations (App. A.1), we define per-patch importance weights $\rho_i^* = h_i/(\sum_j h_j + \epsilon)$ and the target support $\mathcal{A}^* = \{i \mid h_i > 0\}$. The reconstruction loss weights each patch by its query relevance, averaged across all R rounds:

$$\mathcal{L}_{\text{rec}} = \frac{1}{R} \sum_{r=1}^{R} \sum_{i \in \mathcal{A}^*} \rho_i^* \, |\widehat{\mathbf{v}}_i^{(r)} - \mathbf{v}_i\|_2^2/D. \tag{6}$$

The relevance module is trained with soft binary cross-entropy against $\mathbf{h}$, denoted $\mathcal{L}_{\text{rel}}$. The combined objective is

$$\mathcal{L} = \lambda_{\text{rec}}\, \mathcal{L}_{\text{rec}} + \lambda_{\text{rel}}\, \mathcal{L}_{\text{rel}}. \tag{7}$$

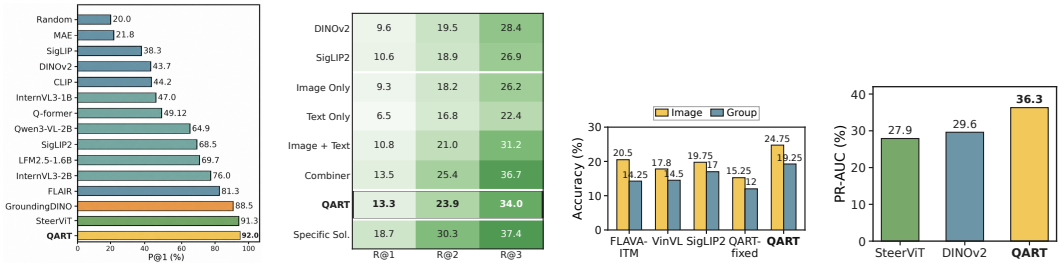
At inference, no relevance supervision is needed; the predicted relevance map and reconstruction coverage jointly determine the token count and number of recurrent rounds for each image–query pair.

4 Experiments

Our experiments probe the central properties of QART rather than performance on individual benchmarks alone. We ask whether query conditioning can steer a frozen visual representation toward the requested content, whether the resulting compact representation preserves downstream reasoning, whether its behavior genuinely depends on the query, whether it retains information beyond what the query explicitly specifies, and whether its compactness translates into practical inference gains.

4.1 Experimental Setup

Unless stated otherwise, QART uses frozen SigLIP2 vision and text encoders and produces at most 32 visual tokens from 729 patch features. A single pretrained QART checkpoint is used across the representation benchmarks without task-specific finetuning. For VQA, QART replaces the visual tokens of LLaVA-1.5-7B while both QART and the language model remain frozen; only a lightweight visual projector is trained. Full architecture, training, and evaluation details are provided in the appendix A. We additionally evaluate backbone generalization by using DINOv2 in App. B.



(a) Steering on CORE. (b) Zero-shot transfer on GeneCIS. (c) Query dependence on Winoground. (d) Fine-grained retention on PODS.

Figure 3: **QART steers visual representations while preserving transferable visual information.** (a) On CORE, QART redirects visual representations toward queried content. (b) The same representation transfers zero-shot to natural images on GeneCIS without dataset-specific tuning. (c) Replacing the sample-specific query with fixed conditioning substantially degrades Winoground performance. (d) On PODS, coarse conditioning preserves fine-grained instance information beyond the prompt.

4.2 Does QART Redirect a Pretrained Visual Representation?

We first ask whether QART changes what a frozen visual representation encodes. If query conditioning is effective, the resulting representation should emphasize the visual evidence requested by the query rather than simply inherit the dominant semantics of the pretrained backbone.

Controlled evaluation on CORE. CORE Ruthardt et al. (2026) [1] directly tests this behavior by requiring retrieval according to a queried object that may occupy only a small part of the image. QART reaches 92.0% P@1, compared with 68.5% for the native SigLIP2 global representation (Fig. 3a). Because both representations originate from the same frozen vision encoder, this improvement isolates the effect of query-conditioned representation construction. QART therefore does more than compress the underlying patch features; it changes which visual evidence dominates the representation. QART also outperforms open-vocabulary localization methods like Grounding-DINO (Liu et al., 2024) and SteerViT (Ruthardt et al., 2026) showcasing the benefit of query-relevant adaptive representations.

Does the same behavior transfer to natural images? We next test whether this steering behavior transfers beyond the controlled CORE setting. On GeneCIS Focus Object, QART uses the same pretrained checkpoint without GeneCIS-specific tuning. It improves over native SigLIP2 representations by 2.7, 5.0, and 7.1% at R@1, R@2, and R@3, respectively (Fig. 3b).

The gain increases with retrieval depth, suggesting that query conditioning consistently promotes condition-compatible images throughout the ranked list rather than affecting only the top prediction. QART also improves over simple Image + Text composition at every rank and approaches Combiner, despite Combiner being designed specifically for composed image-text retrieval.

Insight

QART can steer a frozen visual representation toward query-relevant content, and this behavior transfers from controlled scenes to natural images without task-specific adaptation.

4.3 Can QART Preserve Reasoning and Grounding Under Extreme Visual Compression?

Steering a representation is useful only if the resulting compact tokens retain the information required downstream. We test this in two complementary settings. VQAv2 evaluates general visual reasoning, while Visual7W places greater emphasis on grounding and localized visual evidence.

[1] We follow SteerViT's protocol for CORE.

(a) VQAv2 results. Many baselines use a different vision tower, so we use relative accuracies with the corresponding dense baselines to compare performance.

Method	Base Model	Visual tokens ↓	Accuracy ↑	RelAcc
LLaVA-1.5-7B	CLIP	576	78.5	100%
LLaVA-1.5-7B+SigLIP2	SigLIP2	729	81.4	100%
Q-Former	SigLIP2	32	74.67	91.73%
QG-VTC	CLIP	36	71.4	90.96%
QMoP	CLIP	36	74.1	94.39%
DivPrune	CLIP	32	71.2	90.7%
VisPruner	CLIP	32	67.7	86.24%
VisionZip	CLIP	32	67.1	85.48%
DART	CLIP	32	67.1	85.48%
FastV	CLIP	32	43.4	55.29%
SparseVLM	CLIP	32	58.6	74.65%
OccamToken	CLIP	32	75.5	**96.18%**
LearnPruner	CLIP	32	74.0	94.27%
PruMerge+	CLIP	32	65.6	83.57%
QART	SigLIP2	**29.4**	**76.37**	93.82%

(b) Visual7W results. Here, all token-reduction methods are reproduced using the SigLIP2 vision tower so direct accuracy comparisons stand.

Method	Visual tokens ↓	Layer avg. ↓	Accuracy ↑
LLaVA-1.5-7B	576	576	83.21
LLaVA-1.5-7B+SigLIP2	729	729	88.49
DivPrune	32	32	52.16
VisPruner	32	32	49.35
VisionZip	32	32	46.82
DART	32	75.6	59.86
FastV	32	75.6	52.12
SparseVLM	32	118.0	51.92
OccamToken	32	129.1	55.61
LearnPruner	32	56.0	56.29
QART	**29**	**29**	**81.96**

Table 1: **Reasoning and grounding under aggressive visual compression.** QART preserves strong downstream performance with approximately 29 visual tokens on both VQAv2 and Visual7W. Visual7W additionally reports the average token count processed across language-model layers for methods that prune within the LLM.

General reasoning on VQAv2. Replacing the dense visual tokens of LLaVA-1.5-7B+SigLIP2 with QART tokens, QART reaches 76.37% acc., retaining 93.8% of dense accuracy (81.4%) with 24.8× fewer tokens, and outperforms a matched 32-token Q-Former by 1.7%. Pruning baselines use a different base model, so we compare relative accuracies (Tab. 1a); strong pruners retain slightly more performance here, as VQAv2's general questions limit the benefit of query-directed representation.

Grounded reasoning on Visual7W. We next test whether this compact representation preserves the spatially localized evidence required for grounding in Tab. 1b. We specifically focus on the pointing task as it requires selecting the box that answers a question, so evidence at specific locations must be preserved. Under a matched pipeline (App. A.3), QART reaches 81.96% with 29 tokens, retaining 92.6% of dense model's accuracy and outperforming the strongest baseline, DART, by 22.1%. This larger gap indicates that aggressive compression is more demanding when predictions depend on localized visual evidence. Nevertheless, QART retains substantially more grounding information than existing low-token pruning methods. This suggests that query-conditioned token construction preserves the spatial evidence needed to associate the query with relevant image regions.

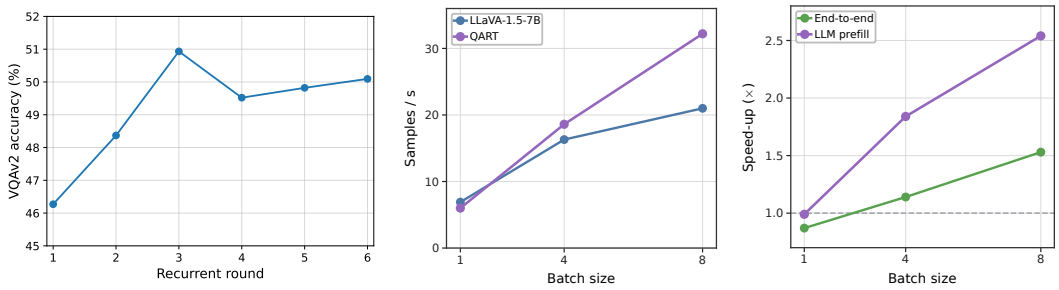
Does recurrence refine the representation? To isolate recurrent refinement from token growth, we evaluate the same QART checkpoint while fixing the representation to two slots throughout recurrence, such that no additional slots are added. VQAv2 accuracy increases from 46.27% after one round to 50.93% after three rounds and remains around that number thereafter (Fig. 4a). Since both the query-relevant support and token count remain fixed, this improvement shows that recurrent updates can refine the representation independently of increasing its capacity. Additional qualitative results are provided in App. C.2.

Insight

QART preserves general reasoning and excels at grounding under severe compression. With 29 tokens, it retains over 92% of dense tokens' accuracy on both benchmarks, with the largest gains when answers depend on localized evidence.

4.4 Does the Conditioning Query Actually Control the Representation?

We use Winoground (Thrush et al., 2022) to test whether QART responds to the compositional meaning of the conditioning query. The benchmark pairs two images with two captions that contain the same words but differ in composition, and evaluates whether the model matches each caption to the correct image. We emphasize Image accuracy, where both candidate images are conditioned on the same caption, and Group accuracy, which additionally requires the Text judgment to be correct.



(a) VQAv2 accuracy with a fixed $K = 2$ token budget across rounds.

(b) End-to-end VLM throughput across batch sizes.

(c) End-to-end and language-model prefill speed-up over LLaVA-1.5-7B.

Figure 4: **Recurrent refinement and inference efficiency.** (a) QART's recurrence refines representations even upon disabling token growth. (b–c) The compact visual sequence increasingly improves throughput and language-model prefill as batch size grows.

QART reaches 24.75% Image and 19.25% Group accuracy, compared with 19.75% and 17.00% for native SigLIP2. It also has 36.25% text accuracy. Further, it outperforms stronger models like FLAVA (Singh et al., 2022) and Vin-VL (Hu et al., 2022) comfortably. More importantly, replacing the sample-specific query with a fixed query reduces Image accuracy to 15.25% and Group accuracy to 12.00% (Fig. 3c).

This controlled intervention isolates the effect of conditioning from the tokenizer itself, showing that QART changes its visual representation according to the compositional meaning of the query rather than only the words it contains. Sec. 4.7. further probes the impact of query conditioning in representation steering and adaptive token allocation.

4.5 Does Query Conditioning Preserve Information Beyond the Prompt?

Query-conditioned compression introduces a complementary risk. A representation may become highly focused on the prompt while discarding visual detail that the prompt does not explicitly mention. We test this on PODS, where QART receives only a coarse supercategory prompt, such as "mug" or "shoe", but must distinguish a particular object from visually similar instances of the same category.

QART reaches 36.3% PR-AUC and 61.3% NDCG, improving PR-AUC by 8.4% over SteerViT under the same supercategory prompting setting and by 6.7% over unconditioned DINOv2 (Fig. 3d).

The query itself cannot identify the correct instance. QART must therefore use it to determine which region or semantic content is relevant while still preserving visual cues that are more specific than the language supervision. The improvement on PODS shows that steering does not collapse the representation to only what is explicitly stated in the prompt. Instead, the query determines where representational capacity is focused while fine-grained visual information within that region is retained.

4.6 Does Compactness Translate into Real Inference Gains?

A shorter visual sequence is useful only if the reduction offsets the cost of constructing it. We therefore measure end-to-end VLM inference as batch size increases.

At batch size one, QART's recurrent tokenization introduces more overhead than is recovered by the shorter language-model sequence, resulting in a $0.87\times$ end-to-end speed ratio. The balance changes as batch size increases and processing the visual sequence inside the language model becomes more expensive. At batch size four, QART reaches $1.14\times$ end-to-end and $1.84\times$ prefill speed-up. At batch size eight, these gains increase to $1.53\times$ and $2.54\times$, respectively, with throughput increasing from 21.0 to 32.2 samples/s (Fig. 4).

Table 2: Ablation of the two roles of query conditioning in QART on CORE. Query conditioning helps steer the visual support used during recurrence and independently control the evidence used for slot growth and halting.

Query-steered support	Query-aware budget	P@1 ↑	Mean K ↓	Mean rounds ↓
✓	✓	92.01	17.25	4.25
✓	✗	92.08	32.00	6.00
✗	✗	79.7	32	6.00

The different scaling of tokenization and language-model processing explains this trend. QART incurs a largely fixed cost to construct its representation, while reducing the sequence subsequently processed by the 7B language model. Its advantage therefore becomes larger when downstream sequence processing dominates the computation.

Insight

QART trades recurrent tokenization overhead for substantially cheaper downstream sequence processing. The benefit grows under batched inference, reaching $1.53\times$ end-to-end and $2.54\times$ prefill speed-up at batch size eight.

4.7 What Does Query Conditioning Control?

Query conditioning enters QART in two places. It determines the visual support available during recurrent refinement, and it defines the evidence used by the reconstruction coverage $c^{(r)}$ to control slot growth and halting. We ablate these two roles on CORE, and illustrate the results in Tab. 2.

Making the budget decisions query-agnostic has almost no effect on retrieval accuracy, but removes the computational adaptivity of QART. The model reaches 92.08% P@1, yet activates all 32 slots and runs all six rounds. Using query-relevant evidence for growth and halting preserves essentially the same accuracy while reducing the average budget to 17.25 slots and 4.25 rounds. This corresponds to roughly 46% fewer active tokens and 29% fewer recurrent rounds.

The query-relevant support itself plays a different role. When the relevance threshold τ is set to zero, all patches participate rather than focusing recurrence on the queried evidence. When there is no query-conditioning for the visual support, P@1 drops to 79.7%, still outperforming the SigLIP2 number. Thus, query conditioning of the visual support is important for steering the representation, whereas query conditioning of the reconstruction coverage enables efficient allocation of capacity.

Insight

Query conditioning serves two complementary roles in QART. It enhances steering of the representation toward the requested visual evidence, while query-aware growth and halting avoid spending capacity on information that the query does not require.

5 Conclusion

We introduced QART, a recurrent visual tokenizer that combines query steerability with budget adaptivity. QART steers its representation toward query-relevant evidence and allocates additional visual tokens only when that evidence is not yet represented with sufficient fidelity. Across retrieval, grounding, compositional matching, and visual question answering, QART produces compact representations with substantially fewer tokens while preserving strong downstream performance. Its gains are strongest when predictions depend on localized visual evidence, while on general VQA it remains competitive with existing token-reduction methods. Overall, these results show that visual representations need not be fixed in either content or capacity, and can instead be shaped by what each query requires.

AI USE STATEMENT

We have used generative AI tools for providing feedback on experiments, reformatting datasets, designing some experiments, modifying some figures, editing parts of the research paper and editing some parts of the software code. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design research methodology, implement methods, assist with translation, support qualitative and thematic data analysis, interpret results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work uses publicly available datasets and pretrained models and does not involve new human-subject data collection. As with other vision and vision–language systems, QART may inherit biases and limitations from its training data and pretrained encoders. Any downstream deployment should therefore consider the provenance and biases of the underlying data and models, as well as the requirements of the target application.

REPRODUCIBILITY STATEMENT

To the best of my knowledge, all the necessary details needed to reproduce the results have been provided in the main paper and appendix. However, the code cannot be released as this projected was conducted with an industrial partner which limits us. However, if absolutely necessary, we can try to release the necessary supplementary materials.

REFERENCES

Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions. *arXiv preprint arXiv:1810.12440*, 2018. URL https://www.alphaxiv.org/abs/1810.12440.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6077–6086, 2018. URL https://www.alphaxiv.org/abs/1707.07998.

Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. FlexTok: Resampling images into 1D token sequences of flexible length. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 2241–2292. PMLR, 2025. URL https://proceedings.mlr.press/v267/bachmann25a.html.

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://arxiv.org/abs/2210.09461.

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*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/10478_ECCV_2024_paper.php.

Zeyuan Chen, Kai Zhang, Zhuowen Tu, and Yuanjun Xiong. Soft tail-dropping for adaptive visual tokenization, 2026. URL https://arxiv.org/abs/2601.14246.

Kai Cheng, Zhengyuan Li, Xingpeng Sun, Byung-Cheol Min, Amrit Singh Bedi, and Aniket Bera. Efficienteqa: An efficient approach for open vocabulary embodied question answering. *arXiv preprint arXiv:2410.20263*, 2024.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven C. H. Hoi. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2142. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/9a6a435e75419a836fe47ab6793623e6-Abstract-Conference.html.

Shivam Duggal, Sanghyun Byun, William T. Freeman, Antonio Torralba, and Phillip Isola. Single-pass adaptive image tokenization for minimum program search, 2025a. URL https://arxiv.org/abs/2507.07995.

Shivam Duggal, Phillip Isola, Antonio Torralba, and William T. Freeman. Adaptive length image tokenization via recurrent allocation. In *International Conference on Learning Representations*, 2025b. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/534ec8dfc0312d96e1c1b357304fadc2-Abstract-Conference.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.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6904–6913, 2017. URL https://openaccess.thecvf.com/content_cvpr_2017/html/Goyal_Making_the_v_CVPR_2017_paper.html.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 15979–15988. IEEE, 2022.

Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, Amita Kamath, Nanyun Peng, and Kai-Wei Chang. Matryoshka query transformer for large vision-language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 50168–50188, 2024. doi: 10.52202/079017-1588. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/59c147c7d4fdb732daea3064eab949bf-Abstract-Conference.html.

Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. Scaling up vision-language pretraining for image captioning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17959–17968. IEEE, 2022.

Haojie Huang, Xinlei Yu, Chengming Xu, Zhangquan Chen, Cheng Yang, Qingdong He, Yu Yang, Jiangning Zhang, and Xiaobin Hu. Evidence-rl: Towards evidence-intensive visual reasoning, 2026. URL https://arxiv.org/pdf/2608.08021v1.

Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6700–6709, 2019. URL https://www.alphaxiv.org/abs/1902.09506.

Keishi Ishihara, Kento Sasaki, Tsubasa Takahashi, Daiki Shiono, and Yu Yamaguchi. Stride-qa: Visual question answering dataset for spatiotemporal reasoning in urban driving scenes. *arXiv preprint arXiv:2508.10427*, 2025. URL https://www.alphaxiv.org/abs/2508.10427.

Jiwan Kim, Kibum Kim, Wonjoong Kim, Byung-Kwan Lee, and Chanyoung Park. Why and when visual token pruning fails? a study on relevant visual information shift in mllms decoding. In *European Conference on Computer Vision*, pp. 244–263. Springer, 2026.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

Geng Li, Guohao Chen, Ting Chen, Shilin Shan, Kuangji Zuo, Bofan Lyu, Tuo An, Gen Li, and Jianfei Yang. Occamtoken: Efficient vlm inference with training-free and budget-adaptive token pruning. *arXiv preprint arXiv:2605.29657*, 2026a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19730–19742. PMLR, 2023. URL https://proceedings.mlr.press/v202/li23q.html.

Shuai Li, Jian Xu, Xiao-Hui Li, Chao Deng, and Lin-Lin Huang. QG-VTC: Question-guided visual token compression in MLLMs for efficient VQA. *arXiv preprint arXiv:2504.00654*, 2025. URL https://arxiv.org/abs/2504.00654.

Zhongyang Li, Yaqian Li, Faming Fang, Rinyoichi Takezoe, Zi-Hao Bo, Cheng Qian, Mo Guang, Guixu Zhang, and Kaiwen Long. QMoP: Query guided mixture-of-projector for efficient visual token compression. *arXiv preprint arXiv:2603.21232*, 2026b. URL https://arxiv.org/abs/2603.21232.

Chang Liu, Henghui Ding, and Xudong Jiang. Gres: Generalized referring expression segmentation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23592–23601. IEEE, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023b.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Xiaocheng Lu, Yuxi Chen, Jie Zhang, Jian Liu, Jingcai Guo, Fangqi Zhu, Tao Han, and Song Guo. AdaTok: Self-budgeting image tokenization with quality-preserving dynamic tokens, 2026. URL https://arxiv.org/abs/2606.07185.

Lingjun Mao, Rodolfo Corona, Xin Liang, Wenhao Yan, and Zineng Tang. Images are worth variable length of representations, 2025. URL https://arxiv.org/abs/2506.03643.

Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE international conference on computer vision*, pp. 4990–4999, 2017.

Toan Q Nguyen and Julian Salazar. Transformers without tears: Improving the normalization of self-attention. In *Proceedings of the 16th international conference on spoken language translation*, 2019.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Georgios Pantazopoulos and Eda B. Özyiğit. Towards understanding visual grounding in vision-language models. *arXiv preprint arXiv:2509.10345*, 2025. URL https://www.alphaxiv.org/abs/2509.10345.

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

Navid Rajabi and Jana Košecká. Gsr-bench: A benchmark for grounded spatial reasoning evaluation via multimodal llms. In *NeurIPS 2024 Workshop on Compositional Learning: Perspectives, Methods, and Paths Forward*, 2024. URL https://www.alphaxiv.org/abs/2406.13246.

Jona Ruthardt, Manu Gaur, Deva Ramanan, Makarand Tapaswi, and Yuki M. Asano. Steerable visual representations. *arXiv preprint arXiv:2604.02327*, 2026. URL https://arxiv.org/abs/2604.02327.

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

Pouya Sadeghi, Shawn He, Pedro Pablo Guerrero Vela, C. Thomas, Alex Wong, and Sirisha Rambhatla. Pinpoint: Prompting with informative interior points. 2026. URL https://arxiv.org/abs/2605.26689.

Saumya Saxena, Blake Buchanan, Chris Paxton, Bingqing Chen, Narunas Vaskevicius, Luigi Palmieri, Jonathan Francis, and Oliver Kroemer. Grapheqa: Using 3d semantic scene graphs for real-time embodied question answering. In *Proceedings of the Conference on Robot Learning*, 2025.

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. URL https://openaccess.thecvf.com/content/ICCV2025/html/Shang_LLaVA-PruMerge_Adaptive_Token_Reduction_for_Efficient_Large_Multimodal_Models_ICCV_2025_paper.html.

Herbert A. Simon. Designing organizations for an information-rich world. In Martin Greenberger (ed.), *Computers, Communications, and the Public Interest*, pp. 37–72. The Johns Hopkins Press, Baltimore, MD, 1971.

Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15617–15629. IEEE, 2022.

Shobhita Sundaram, Julia Chae, Yonglong Tian, Sara Beery, and Phillip Isola. Personalized representation from personalized generation. In *International Conference on Learning Representations*, volume 2025, pp. 92572–92606, 2025.

Rinyoichi Takezoe, Yaqian Li, Zihao Bo, Anzhou Hou, Mo Guang, and Kaiwen Long. LearnPruner: Rethinking attention-based token pruning in vision language models. *arXiv preprint arXiv:2604.23950*, 2026. URL https://arxiv.org/abs/2604.23950.

Raghuveer Thirukovalluru, Xiaochuang Han, Bhuwan Dhingra, Emily Dinan, and Maha Elbayad. Text-guided semantic image encoder. *arXiv preprint arXiv:2511.20770*, 2025. URL https://arxiv.org/abs/2511.20770.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5228–5238. IEEE, 2022.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. URL https://arxiv.org/abs/2502.14786.

Sagar Vaze, Nicolas Carion, and Ishan Misra. Genecis: A benchmark for general conditional image similarity. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6862–6872. IEEE, 2023.

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

Rui Xiao, Sanghwan Kim, Mariana-Iuliana Georgescu, Zeynep Akata, and Stephan Alaniz. FLAIR: VLM with fine-grained language-informed image representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24884–24894, 2025. doi: 10.1109/CVPR52734.2025.02317. URL https://openaccess.thecvf.com/content/CVPR2025/html/Xiao_FLAIR_VLM_with_Fine-grained_Language-informed_Image_Representations_CVPR_2025_paper.html.

Wilson Yan, Volodymyr Mnih, Aleksandra Faust, Matei Zaharia, Pieter Abbeel, and Hao Liu. ElasticTok: Adaptive tokenization for image and video, 2024. URL https://arxiv.org/abs/2410.08368.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. VisionZip: Longer is better but not necessary in vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19792–19802, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Yang_VisionZip_Longer_is_Better_but_Not_Necessary_in_Vision_Language_CVPR_2025_paper.html.

Xubing Ye, Yukang Gan, Yixiao Ge, Xiao-Ping Zhang, and Yansong Tang. ATP-LLaVA: Adaptive token pruning for large vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24972–24982, 2025. doi: 10.1109/CVPR52734.2025.02325. URL https://openaccess.thecvf.com/content/CVPR2025/html/Ye_ATP-LLaVA_Adaptive_Token_Pruning_for_Large_Vision_Language_Models_CVPR_2025_paper.html.

Hongxu Yin, Arash Vahdat, Jose 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*, 2022.

Shicheng Yin, Kaixuan Yin, Yang Liu, Weixing Chen, and Liang Lin. Dart: Differentiable adaptive region tokenizer for vision foundation models. *arXiv preprint arXiv:2506.10390*, 2025.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions. In *European Conference on Computer Vision*, pp. 69–85, 2016. doi: 10.1007/978-3-319-46475-6_5. URL https://arxiv.org/abs/1608.00272.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966, 2024.

Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond text-visual attention: Exploiting visual cues for effective token pruning in VLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20857–20867, 2025a. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhang_Beyond_Text-Visual_Attention_Exploiting_Visual_Cues_for_Effective_Token_Pruning_ICCV_2025_paper.html.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis A Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. SparseVLM: Visual token sparsification for efficient vision-language model inference. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 74840–74857. PMLR, 2025b. URL https://proceedings.mlr.press/v267/zhang25s.html.

Table 3: Data mixture used for QART pretraining. VG is separated into native Visual Genome images and images overlapping with COCO.

Source	Sampling probability	Validation records
RefCOCO	0.10	10,834
RefCOCO+	0.10	10,758
RefCOCOg	0.10	9,536
gRefCOCO	0.20	14,229
Mapillary Vistas	0.15	3,053
VG, native images	0.33	3,003
VG, COCO images	0.02	381

Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7W: Grounded Question Answering in Images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

A Implementation Details

A.1 Datasets

Tokenizer pretraining data. We pretrain QART on a mixture of referring-expression, scene-understanding, and region-description datasets comprising RefCOCO Yu et al. (2016), RefCOCO+ Yu et al. (2016), RefCOCOg Yu et al. (2016), gRefCOCO Liu et al. (2023a), Mapillary Vistas Neuhold et al. (2017), and Visual Genome (VG) Krishna et al. (2017). The complete mixture contains 2,078,367 image–text annotation records; this number counts annotations rather than unique images. Table 3 summarizes the sampling distribution. Training examples are sampled with replacement according to the specified mixture weights.

Query-relevance supervision. During tokenizer pretraining, we use dense query-region targets to supervise the relevance predictor, following Ruthardt et al. (Ruthardt et al., 2026). For the COCO-family referring-expression datasets (Yu et al., 2016), relevance targets are derived from segmentation annotations. For Mapillary Vistas (Neuhold et al., 2017) and Visual Genome (Krishna et al., 2017), we use cached dense query-region alignment maps. The resulting heatmaps are retained as soft targets in $[0, 1]$ rather than being converted to binary bounding boxes. These targets provide training supervision for the relevance predictor and the query-weighted reconstruction objective, but are not used to define the visual support or the recurrent growth and halting decisions. The recurrent tokenizer instead operates on the support predicted by the relevance module, $\mathcal{A} = \{i \mid p_i > \tau\}$, both during training and inference.

Vision–language alignment data. For integration with LLaVA-1.5-7B Liu et al. (2023b), we train the visual projector in two stages. The first stage performs one epoch of Visual Genome-based alignment using native-image VG region descriptions. The region phrase conditions QART, while the language model receives a generic region-description instruction and predicts the corresponding phrase.

The second stage trains the projector for one epoch on the full LLaVA-Instruct-665K mixture Liu et al. (2023b). QART and the language model remain frozen throughout both stages, and only the visual projector is optimized.

A.2 Setup

Backbone and input representations. Our main configuration uses the frozen vision and text encoders from `google/siglip2-so400m-patch14-384`. Images are processed at 384×384 resolution. The visual encoder produces a 27×27 grid of $N = 729$ patch features, each with dimension 1152. We precompute and cache these frozen visual features in FP16.

A bias-free linear layer projects the visual features to the tokenizer dimension $D = 768$.

The frozen text encoder provides token-level query representations, which are projected to the same 768-dimensional space when necessary. Padding tokens are masked, and a masked mean of the query features is used to initialize the slot bank. Tokenizer pretraining uses a maximum text length of 32 tokens.

The reconstruction target is the projected visual feature map used by the tokenizer. Specifically, the frozen backbone features are first projected to the tokenizer dimension, and the decoder reconstructs these projected visual features rather than RGB pixels.

Relevance predictor. The relevance predictor consists of five visual-to-text cross-attention layers with eight attention heads. Visual patches act as queries and text tokens as keys and values. Each layer uses pre-layer normalization (Nguyen & Salazar, 2019) and a residual connection without an intermediate feed-forward block. A final linear projection followed by a sigmoid predicts one relevance score $p_i \in [0, 1]$ for each visual patch.

Queries and keys are normalized independently within each attention head after their learned projections. At inference, the query-relevant visual support is defined as

$$\mathcal{A} = \{i \mid p_i > \tau\}. \tag{8}$$

The relevance set is computed once before recurrence and remains fixed across recurrent rounds. For tasks in which a visual target is guaranteed to exist, an empty selection falls back to the 100 patches with the highest relevance scores.

Recurrent tokenizer. QART maintains at most $K_{\max} = 32$ active latent slots. A masked mean of the query-token representations produces a pooled query representation $\bar{\mathbf{q}}$. At initialization, a learned slot projector f_{init} maps $\bar{\mathbf{q}}$ to $K_0 = 2$ query-conditioned slots,

$$\mathbf{S}^{(0)} = f_{\text{init}}(\bar{\mathbf{q}}) \in \mathbb{R}^{K_0 \times D}. \tag{9}$$

The projector produces distinct slot embeddings from the shared query representation, followed by per-slot layer normalization.

Each recurrent round uses a three-layer pre-normalized Transformer with eight attention heads and a feed-forward width of 3072. The complete three-layer Transformer is shared across recurrent rounds, while the three layers within the block use distinct parameters. Learned modality embeddings distinguish slot, visual, and text tokens, and a learned round embedding is added to the active slots.

At round r, the shared Transformer jointly processes the current slots $\mathbf{S}^{(r-1)}$, the selected query-relevant visual features $\mathbf{V}_{\mathcal{A}}$, and the text features $\mathbf{T}$:

$$\tilde{\mathbf{S}}^{(r)} = F_\theta\left(\mathbf{S}^{(r-1)}, \mathbf{V}_{\mathcal{A}}, \mathbf{T}\right). \tag{10}$$

A subsequent slot-to-patch cross-attention readout allows the resulting slots to gather additional information from the selected visual evidence:

$$\mathbf{S}^{(r)} = \mathrm{LN}\left(\tilde{\mathbf{S}}^{(r)} + \mathrm{MHA}\left(\tilde{\mathbf{S}}^{(r)}, \mathbf{V}_{\mathcal{A}}, \mathbf{V}_{\mathcal{A}}\right)\right). \tag{11}$$

Only the slot states $\mathbf{S}^{(r)}$ are propagated recurrently. The selected visual features $\mathbf{V}_{\mathcal{A}}$ and text features $\mathbf{T}$ remain fixed and are supplied again at every round.

The joint self-attention uses per-head QK normalization. We additionally use shared learned residual-scaling parameters

$$\mathbf{U} = \alpha\mathbf{X} + \beta\,\mathrm{SA}(\mathrm{LN}(\mathbf{X})), \tag{12}$$

$$\mathbf{X}' = \alpha\mathbf{U} + \beta\,\mathrm{FFN}(\mathrm{LN}(\mathbf{U})), \tag{13}$$

initialized to $\alpha = 0.75$ and $\beta = 0.25$.

When the current slot set does not represent the selected visual evidence with sufficient fidelity, $\Delta K = 6$ additional slots are initialized from $\bar{\mathbf{q}}$ using f_{init} and concatenated with the existing slots before the next recurrent round. The active slot count therefore follows

$$K_{r+1} = \begin{cases} \min(K_r + \Delta K, K_{\max}), & c^{(r)} < \eta_{\text{grow}}, \\ K_r, & \text{otherwise.} \end{cases} \tag{14}$$

Reconstruction and adaptive computation. A shared decoder reconstructs the dense visual feature grid from the active slots after each recurrent round. The decoder uses N learned positional queries and four pre-normalized cross-attention/FFN blocks, with the active slots supplying keys and values. Each block uses eight attention heads and a feed-forward width of $4D$. A final layer normalization and linear projection produce the reconstructed visual features $\hat{\mathbf{V}}^{(r)} \in \mathbb{R}^{N \times D}$.

For visual feature $\mathbf{v}_i \in \mathbb{R}^D$ and its reconstruction $\hat{\mathbf{v}}_i^{(r)}$ at round r, we compute the normalized squared reconstruction error as

$$\frac{1}{D} \left\| \hat{\mathbf{v}}_i^{(r)} - \mathbf{v}_i \right\|_2^2. \tag{15}$$

Representation completion at round r is measured over the selected query-relevant support,

$$c^{(r)} = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \mathbb{I} \left[\frac{1}{D} \left\| \hat{\mathbf{v}}_i^{(r)} - \mathbf{v}_i \right\|_2^2 < \delta \right]. \tag{16}$$

We use $\delta = 0.30$, $\eta_{\text{grow}} = 0.85$, and $\eta_{\text{halt}} = 0.95$. If $c^{(r)} < \eta_{\text{grow}}$, up to six additional slots are activated, subject to the 32-slot limit. The resulting maximum-capacity sequence is $2, 8, 14, 20, 26, 32$. Inference may halt after a minimum of two recurrent rounds once $c^{(r)} \geq \eta_{\text{halt}}$, and otherwise continues for at most six rounds. During tokenizer pretraining, all six rounds are executed while slot growth remains adaptive.

Tokenizer optimization. We optimize the trainable QART components using AdamW with learning rate 10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, and weight decay 0.01. The learning rate increases linearly for 4,000 optimizer steps and then follows a cosine schedule over 25 epochs. Training uses a global batch size of 256, BF16 mixed precision, and global-norm gradient clipping at 0.5.

The training objective combines query-weighted feature reconstruction and relevance prediction as described in Sec. 3.4. We use $\lambda_{\text{rec}} = 1$ and $\lambda_{\text{rel}} = 0.5$. The implementation additionally includes a slot-count term with weight 0.01. Because slot allocation is discrete, this term does not propagate gradients through the allocation decisions.

Vision–language projector. For LLaVA-1.5-7B, the active QART slots are mapped to the language-model embedding dimension using a two-layer projector

$$768 \to 4096 \to 4096, \tag{17}$$

with input layer normalization, GELU activation, and output layer normalization.

Both projector-training stages use AdamW with learning rate 2×10^{-4}, zero weight decay, a 500-step linear warmup followed by cosine decay, and gradient-norm clipping at 1.0. The global batch size is 32. The first Visual Genome alignment stage is run for one epoch, followed by one epoch on LLaVA-Instruct-665K. Only the projector parameters are updated in both stages.

A.3 Evaluation Protocols

Shared representation evaluation. CORE, GeneCIS, PODS, and Winoground use the frozen pretrained QART tokenizer without benchmark-specific fitting. For retrieval-style evaluations, we mean-pool the active slots, compute the representation in FP32, apply ℓ_2 normalization, and use cosine similarity.

Unless stated otherwise, transfer experiments use the same SigLIP2 QART checkpoint and the relevance threshold $\tau = 0.25$. GeneCIS, PODS, and Winoground execute each image–query pair independently so that recurrent halting is determined per example.

CORE. We evaluate on the `JonaRuthardt/CORE` test split using the provided scene-specific galleries and object prompts. For each target object, all images in its gallery are encoded under the same query. Each positive image is used as a retrieval query, while itself and variants of the same source image are excluded from the gallery. We report precision at rank one (P@1) over the eligible retrieval queries.

GeneCIS. We evaluate the GeneCIS Vaze et al. (2023) *Focus Object* split. For every query, the reference image and its 15 candidate images are encoded using the same object condition. Candidate membership from the benchmark is preserved without introducing additional negatives. Candidates are ranked using cosine similarity between normalized mean-slot representations, and we report R@1, R@2, and R@3.

VQAv2. For VQAv2 Goyal et al. (2017), we use the QART–LLaVA-1.5-7B model obtained after the two-stage projector training described above. QART and the 7B language model remain frozen, and the relevance threshold is set to $\tau = 0.05$. Only active QART slots are projected into the language-model embedding space and inserted before the textual question.

Answers are generated autoregressively using greedy decoding. We use at most 16 generated tokens and evaluate predictions using the standard VQAv2 evaluation protocol.

For VQAv2, we reproduce LLaVA+SigLIP2 where the visual representations are replaced with siglip tokens, and Q-Former Li et al. (2023) by following the same protocol as QART. The other numbers are reported from their corresponding papers.

Visual7W. For Visual7W, we evaluate it on the pointing task. For the pointing task, the 4 candidate bounding boxes are given in [x1, y1, x2, y2] format. Also, the 4 candidates are given as A, B, C and D to the LLM, and the answer is always one of the 4 alphabets. Evaluation is normal accuracy checking for a perfect match.

We reproduce all compared methods under a common evaluation pipeline. We use the LLaVA+SigLIP2 configuration as the base model for all the pruning baselines which is finetuned with the same protocol as qart, which is one epoch of vg-alignment and 6 epochs of Visual7W projector finetuning. And for all learnable baselines, we additionally finetune for 4 epochs on Visual7W data to adapt to the compressed representations.

Winoground. We evaluate all 400 official Winoground Thrush et al. (2022) examples and score all four image–caption pairings. For an image I_i conditioned on caption c_j, QART reconstructs the visual feature grid from its active slots. We then apply the frozen SigLIP2 attention pooler and compute cosine similarity with the native SigLIP2 text representation,

$$S_{ij} = \cos\Big(H_v\big(\hat{V}(I_i, c_j)\big), H_t(c_j)\Big). \tag{18}$$

We compute the official Text, Image, and Group accuracies from the resulting 2×2 similarity matrix. For the fixed-query control, the same generic text query is used to condition QART for every image while the official Winoground scoring protocol remains unchanged.

PODS. We follow the PODS Sundaram et al. (2025) evaluation protocol over 100 target objects, 300 reference images, and 10,888 test images. For the main experiment, the dataset's coarse super-category label is used as the conditioning query for all reference and test images associated with a target object.

For classification, each test image is scored by its maximum cosine similarity to the three positive reference representations. We compute average precision independently for each target object and report the macro average. For retrieval, each positive test image ranks all 300 references, with the three references belonging to the same target object treated as relevant. We compute binary NDCG over the complete ranking and average first within each target object and then across objects.

B Evaluating Backbone generalization with DINOv2

B.1 Data and Setup

We additionally train QART using frozen DINOv2-B/14 visual features and a frozen RoBERTa-large text encoder. Images are resized to 378×378, producing a 27×27 grid of 729 patch features with dimension 768 after removing the CLS token. RoBERTa-large provides the query features, which are projected to the same 768-dimensional tokenizer space.

Table 4: VQAv2 results for QART with SigLIP2 and DINOv2 visual backbones. Both variants use the same LLaVA-1.5-7B integration and projector-training protocol.

Metric	SigLIP2 QART	DINOv2 QART
Overall accuracy	**76.37%**	72.93%
Yes/No	**91.20%**	89.87%
Number	**60.80%**	54.16%
Other	**69.21%**	65.01%
Mean active slots	**29.45**	30.96

The remaining QART architecture is unchanged from the SigLIP2 configuration. We use the same relevance predictor, recurrent tokenizer, reconstruction decoder, slot-allocation policy, training objective, data mixture, optimizer, and training schedule described in Sec. A. The model uses two initial slots, a maximum of 32 slots, and at most six recurrent rounds.

The DINOv2 configuration differs from the main SigLIP2 model in an important respect. SigLIP2 jointly pretrains its visual and textual representations, whereas DINOv2 and RoBERTa are independently pretrained and do not share a cross-modal embedding space. Consequently, the DINOv2–RoBERTa configuration provides a weaker initialization for query-conditioned visual representation learning despite using the same QART architecture.

VQAv2 setup. For VQAv2 Goyal et al. (2017), we follow the same protocol as followed for the SigLIP2 QART variant.

B.2 Results

CORE. We evaluate DINOv2 QART using the same CORE protocol as the SigLIP2 model. The relevance threshold is applied to the predicted patch-relevance scores, and image representations are obtained by mean pooling the active slots, followed by ℓ_2 normalization and cosine similarity.

At $\tau = 0.25$, DINOv2 QART reaches 90.33% P@1 with 23.34 active slots on average. Its best recorded operating point reaches 90.59% P@1 at $\tau = 0.20$. For comparison, the SigLIP2 QART configuration reaches 92.01% P@1 at $\tau = 0.25$.

The relatively small gap on CORE indicates that the adaptive tokenization mechanism transfers effectively to DINOv2 features, despite the absence of a jointly pretrained vision–language representation.

VQAv2. We evaluate the DINOv2 variant using the same LLaVA-1.5-7B integration and projector-training protocol as the SigLIP2 QART model. The results are shown in Tab. 4.

Under the matched training protocol, the DINOv2 variant reaches 72.93% overall accuracy compared with 76.37% for SigLIP2 QART. The largest difference appears on number questions, while the gap is smaller for yes/no questions. DINOv2 also uses slightly more active slots on average.

Although the DINOv2 variant trails SigLIP2, its VQAv2 performance remains competitive with prior visual-token pruning and reduction methods operating at similarly aggressive token budgets, as reported in the main paper. This indicates that the QART formulation remains effective even when the visual and textual encoders are not jointly pretrained.

We attribute the remaining gap primarily to the weaker cross-modal initialization of the independently pretrained DINOv2 and RoBERTa encoders, which lack the native vision–language alignment available in SigLIP2.

B.3 Discussion

The DINOv2 results show that QART is not tied to a particular visual backbone. The variant remains close to SigLIP2 on CORE and retains competitive VQAv2 performance at a compact visual-token budget, despite using independently pretrained vision and text encoders.

Table 5: Sensitivity to the relevance threshold. Lower thresholds generally increase the selected visual support, but performance does not vary monotonically with the resulting token budget. Mean K denotes the average number of active QART slots.

Task	τ	Primary metric ↑	Mean K
CORE	0.10	85.61	–
	0.20	90.96	–
	0.25	**92.01**	17.25
	0.35	91.92	–
	0.50	89.79	–
GeneCIS	0.05	12.81	30.04
	0.10	13.32	28.04
	0.20	**15.10**	25.38
	0.25	13.32	24.43
	0.35	14.18	22.63
	0.50	12.91	21.52
Winoground	0.10	15.25	31.07
	0.20	15.25	28.84
	0.25	**19.25**	27.49
	0.35	17.00	25.40
	0.50	15.00	22.18

The stronger SigLIP2 results are consistent with the benefit of native cross-modal alignment. SigLIP2 jointly learns visual and textual representations, whereas DINOv2 and RoBERTa are pretrained independently and therefore provide a weaker starting point for query-conditioned representation learning. We therefore view the DINOv2 experiments as evidence that QART generalizes across backbone families, while also highlighting the importance of aligned vision–language features for its strongest performance.

C Additional Results and Analyses for QART

C.1 Threshold Sensitivity and Adaptive Budgeting

The relevance threshold τ controls the spatial support selected for recurrent tokenization. Changing this threshold therefore affects both *which* image regions are represented and the amount of representational capacity subsequently allocated by QART. We analyze this interaction on CORE, GeneCIS, and Winoground using the same SigLIP2 QART model.

Table 5 reports the primary metric for each task together with the resulting mean active-slot count when available. For CORE, we report P@1; for GeneCIS, R@1; and for Winoground, Group accuracy.

The optimal operating point differs across tasks. CORE performs best at $\tau = 0.25$, whereas GeneCIS reaches its highest R@1 at $\tau = 0.20$. Winoground also favors an intermediate threshold rather than either extreme.

More importantly, increasing the selected visual support and the resulting token budget does not guarantee improved performance. On GeneCIS, lowering the threshold from 0.20 to 0.05 increases the mean slot count from 25.38 to 30.04, yet R@1 decreases from 15.10% to 12.81%. A similar pattern appears on Winoground, where $\tau = 0.10$ uses 31.07 slots on average but obtains lower Group accuracy than $\tau = 0.25$, which uses only 27.49 slots.

These results highlight that QART's budget should not be interpreted as a simple "more tokens is better" trade-off. The relevance threshold jointly controls the spatial extent and composition of the selected evidence, while the recurrent tokenizer allocates capacity to represent that evidence. Different tasks therefore favor different combinations of visual coverage and token budget.

Insight

Effective visual representations require adapting both spatial support and representational capacity.

Broader coverage generally increases the available token budget, but additional tokens do not monotonically improve downstream performance. The task-dependent optima suggest that a single fixed visual budget or uniformly holistic representation is unlikely to be optimal across different visual reasoning settings.

C.2 Does Recurrence Refine the Representation?

QART does not construct its representation in a single step. The active slots are repeatedly refined using the same query and visual evidence, allowing their spatial support to change across recurrent rounds. Figure 5 shows representative examples of this behavior.

The progression is particularly clear when the initial representation attends to a nearby distractor or only coarsely identifies the relevant region. For the query "person in foreground with backpack," the early support is spread over nearby foreground content before progressively concentrating on the queried person and backpack. A similar shift occurs for "guy in plaid," where the support moves toward the queried individual over successive rounds. For "number 13," the representation initially responds to nearby players before becoming localized around the jersey number of the target player.

These examples suggest that recurrence provides more than additional representational capacity. Later rounds can revise where the active slots attend, progressively moving their support toward evidence that is better aligned with the query. This is useful when the relevant region is visually ambiguous or surrounded by similar content, where a single tokenization step may commit too early to an incorrect or overly broad region.

Insight

Recurrence allows QART to revise, rather than simply expand, its representation. Across rounds, slot support can move from coarse or distracting regions toward the queried visual evidence, providing a mechanism for iterative query-conditioned refinement.

C.3 Is QART query-steerable and budget-adaptive?

Figure 6 qualitatively illustrates how the conditioning query changes the visual representation. For the same input image, QART shifts its support from the left pizza to the right pizza when the query changes. Importantly, the representation does not only move spatially: the active token budget also changes from $K = 8$ to $K = 20$. This provides a direct example of query steerability and budget adaptivity within the same image.

The allocated capacity also varies substantially across different image–query pairs. Figure 7 shows representative examples spanning $K = 2$ to the full $K = 32$ slot budget. Localized targets such as "zebra on far right" and "top right bowl" require only a small representation, whereas other queries receive substantially more capacity. Queries involving broader or spatially distributed evidence can activate most or all of the available slots.

The token count is not determined by the number of queried objects alone. Instead, it reflects the visual evidence that QART needs to preserve for a particular image–query pair, including its spatial extent, appearance, and surrounding context. Together, Figs. 6 and 7 show that QART adapts both the content and the capacity of its representation rather than applying a fixed visual budget to every query.

Insight

QART adapts both where it represents and how much it represents. Changing the query can redirect the visual support even for the same image, while the active token budget varies from a few slots to the full capacity depending on the evidence required by the query.

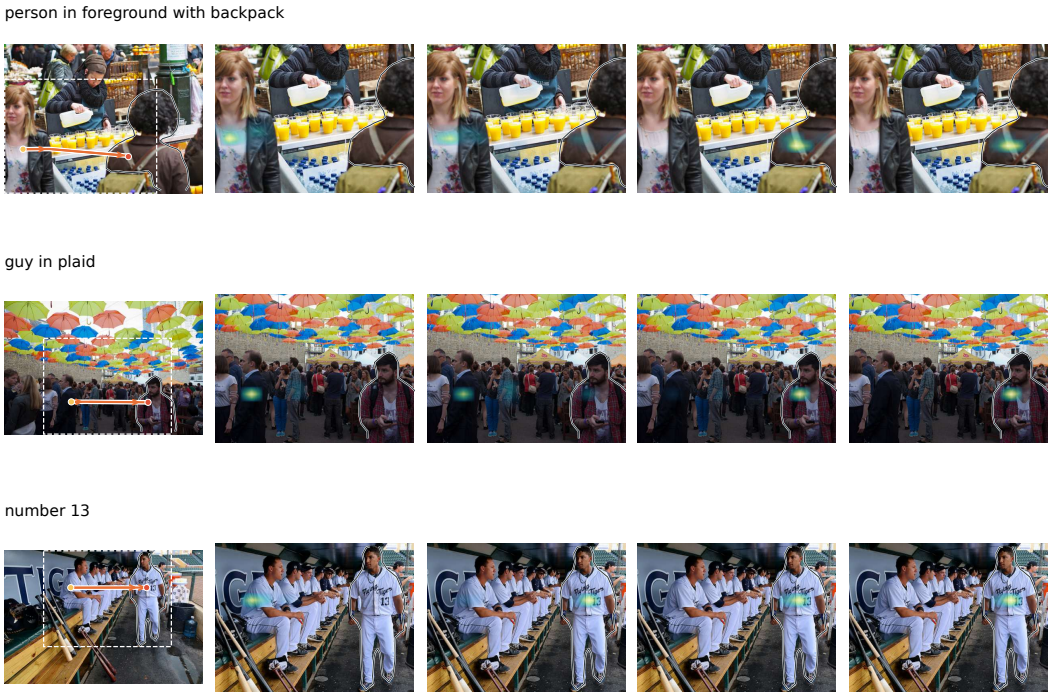


Figure 5: **Representation refinement across recurrent rounds.** Each row shows the spatial support of an active QART slot over successive rounds for a different query. The support progressively shifts toward the query-relevant region, including the foreground backpack, the person in plaid, and jersey number 13.



Figure 6: **Query-dependent visual support and token allocation.** For the same input image, changing the query from "left pizza" to "right pizza" redirects the visual support to the requested region and changes the active token budget from $K = 8$ to $K = 20$.

C.4 Limitations

While QART is evaluated across retrieval, compositional matching, visual question answering, and grounding, our experiments focus on single-image settings. Extending query-adaptive token allocation to video, multi-turn interaction, and temporally evolving visual observations remains an interesting direction for future work.

The current formulation also uses a small number of thresholds to control relevance selection, slot growth, and halting. As shown in our sensitivity analysis, the preferred operating point can vary across tasks. Exploring fully learned or calibration-free allocation criteria could further simplify deployment across heterogeneous downstream settings.

Finally, QART uses dense query-region supervision during tokenizer pretraining to train the relevance predictor and reconstruction objective. This supervision is not required at inference, but



Figure 7: **QART adapts its token budget across image–query pairs.** Representative examples span the allocation range from $K = 2$ to $K = 32$. Localized queries can be represented with only a few active tokens, while broader or more spatially distributed visual evidence receives a larger representation.

future work could investigate weaker or self-supervised relevance signals to reduce the annotation requirements of tokenizer pretraining.