RECO:TRAINING-FREE VISUAL TOKEN PRUNING VIA RELATIONAL TRANSITION CORESET IN LARGE VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large Vision-Language Models (LVLMs) incur substantial inference overhead and KV cache costs because visual inputs are represented as long, dense token sequences. Among existing pruning methods, attention-based approaches compress high-dimensional interactions into scalar saliency scores that tend to exhibit instability across layers and attention heads, whereas redundancy elimination methods often equate unimodal feature similarity with cross-modal semantic substitutability, risking the loss of fine-grained visual evidence. Both paradigms typically evaluate tokens within an isolated representation space either prior to or following cross-modal projection, failing to utilize the information conveyed by pairwise relational transitions across the projection. We propose ReCo, a training-free visual token pruning framework based on relational transition coresets. ReCo jointly characterizes pre-projection visual relations, post-projection language-aligned relations, and projection-induced relational shifts, integrating them into a unified relational transition kernel. Visual token pruning is subsequently formulated as a submodular facility-location problem, where a greedy selector constructs a compact coreset by maximizing marginal relational coverage. Experiments on LLaVA-1.5, LLaVA-NeXT, and Qwen2.5-VL demonstrate that ReCo achieves competitive performance across both fixed-length and dynamic-resolution inputs. It maintains 91.02% of the performance on Qwen2.5-VL even under an aggressive 90% visual token reduction.

1 INTRODUCTION

Large Vision-Language Models (LVLMs) have demonstrated strong multimodal capabilities by projecting visual representations into dense sequences of visual tokens fed into language models Liu et al. (2023); Bai et al. (2023). However, processing high-resolution inputs typically generates hundreds or even thousands of visual tokens, which substantially increases the KV cache footprint and incurs notable inference latency Zhang et al. (2023); Dao (2024); Pathak & Han (2026). Consequently, visual token reduction via token merging or pruning has emerged as an effective approach to accelerating LVLM inference Zeng et al. (2022); Yu et al. (2024).

Existing visual token reduction methods primarily rely on attention-guided saliency selection and similarity-driven redundancy elimination. For attention-based approaches, methods such as FastV Chen et al. (2024), MADTP Cao et al. (2024), and SparseVLM Zhang et al. (2025) filter tokens via shallow-layer scores, cross-modal alignment, and adaptive strategies, respectively, whereas PyramidDrop Xing et al. (2025) and VisionDrop Xu et al. (2026) turn to inter-layer scheduling or self-attention priors. However, reducing high-dimensional interactions to scalar scores makes these methods susceptible to distributional biases across layers and attention heads. This tends to over-concentrate the retained subset on high-weight regions, inadvertently discarding diffuse yet critical visual context. Regarding redundancy elimination, ToMe Bolya et al. (2023), VisionZip Yang et al. (2025), and DivPrune Alvar et al. (2025) explore bipartite matching, feature decoupling, and diversity criteria, respectively. Recent studies such as CrisPrune Liu et al. (2026) and AdaPrune Zhu et al. (2026) further introduce spatial repulsion mechanisms and orthogonal feature decoupling. Nevertheless, unimodal feature similarity does not equate to cross-modal semantic substitutability. Solely pursuing representational diversity tends to bias the token budget toward background outliers,

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.

whereas coarse-grained merging often blurs feature boundaries, thereby degrading fine-grained evidence concerning object counts, spatial locations, and localized attributes.

Although existing methods diversify the selection criteria for token pruning, they mostly evaluate tokens within an isolated feature space either prior to or following cross-modal projection, overlooking the transition of inter-token relations across modalities. Pre- and post-projection feature relations characterize visual appearance similarity and linguistic semantic consistency, respectively, with their discrepancy reflecting the cross-modal reorganization of interaction affinities. As demonstrated in Figure 1, pairwise token relations undergo shifts across projection. To this end, we formalize the shifts in token dependencies induced by cross-modal projection as a relational transition, which encompasses pre-projection relations, post-projection relations, and their transitional shifts, and construct composite transition representations to quantify this process. If two tokens exhibit consistency across all three relational dimensions, one can substitute for the other with minimal information loss; conversely, discrepancies in relational structures or transition trajectories indicate complementary information, precluding their concurrent pruning. Accordingly, we recast visual token pruning as a coreset selection problem over these relational transition representations, identifying a compact subset that is both representative and complementary by optimizing global relational coverage.

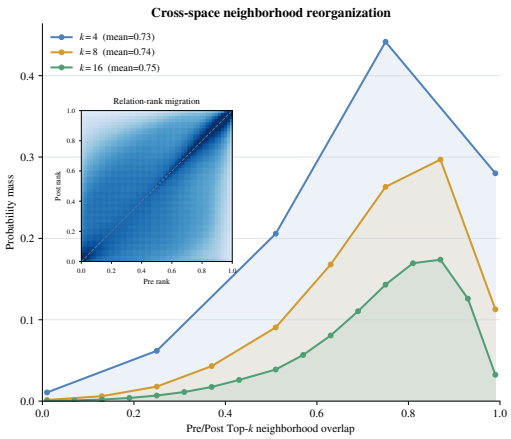


Figure 1: **Relational shifts across projection.** Top-k neighborhood overlap (main) and rank migration (inset) demonstrate that cross-modal projection reshapes pairwise token relations. Additional analyses are provided in Appendix A.7.

Thus, we propose ReCo, a training-free visual token pruning framework. Via parameter-free scale alignment, this framework integrates pre-projection associations, post-projection relations, and their transitional shifts into a unified relational transition kernel to quantify inter-token substitutability. Building upon this, ReCo incorporates a facility-location objective for set-level selection, progressively optimizing the coreset based on the marginal coverage gain contributed by each candidate token, thereby promoting both representativeness and complementarity within the retained subset. Extensive experiments validate the effectiveness of ReCo: it retains 95.48% of the performance on LLaVA-1.5 while pruning 88.9% of visual tokens, and maintains 91.02% performance on Qwen2.5-VL with 90% tokens pruned, outperforming the second-best method by 3.47%.

The main contributions of this paper are summarized as follows:

- By characterizing token relational transitions across cross-modal projection, we formalize token pruning as a coreset selection problem over relational transition representations.
- We propose ReCo, a training-free framework that constructs a unified relational transition kernel to quantify cross-modal substitutability, and achieves set-level pruning that maximizes marginal coverage via a facility-location objective.
- Extensive experiments across multiple benchmarks and LVLMs, including LLaVA-1.5 and Qwen2.5-VL, demonstrate that ReCo maintains stable task performance under varying token budgets, verifying its effectiveness.

2 Related Work

2.1 Token Reduction in Large Vision-Language Models

To alleviate the computational and KV cache overhead in LVLMs, attention-driven approaches identify critical tokens through layer-wise statistics. Specifically, FastV Chen et al. (2024) prunes tokens

using shallow attention scores, SparseVLM Zhang et al. (2025) adaptively determines sparsity via textual guidance, PyramidDrop Xing et al. (2025) performs progressive inter-layer dropping, FitPrune Ye et al. (2025) optimizes discrete parameter spaces, and VisionDrop Xu et al. (2026) adopts visual self-attention to mitigate cross-modal bias. From the perspective of redundancy elimination, ToMe Bolya et al. (2023) merges similar tokens via bipartite matching, LLaVA-PruMerge Shang et al. (2025) pools local neighborhood features, and VisionZip Yang et al. (2025) decouples core features from compact context. Further developments optimize token allocation by removing prompt dependence Wang et al. (2026), scheduling pruning via evidence awareness Choi et al. (2026), or exploring graph topologies Han et al. (2026a) and multi-objective trade-offs Ji et al. (2026a). Nevertheless, these approaches assess tokens solely within an isolated feature space either before or after cross-modal projection. Unlike these approaches, ReCo explicitly models the projection-induced relational shift, tracking the evolution of inter-token relationships to preserve a globally representative and complementary subset.

2.2 Submodular Coreset Selection for Visual Representations

Coreset selection aims to extract a compact subset from dense data to maximize space coverage. Wei et al. formulated this process as cardinality-constrained submodular maximization with theoretical approximation guarantees Wei et al. (2015), Kaushal et al. introduced facility location functions to visual tasks Kaushal et al. (2019), and Pyramid-BERT validated the effectiveness of set-level coreset downsampling in Transformers Huang et al. (2022). However, relying solely on static similarity matrices tends to incur local homogenization or outlier sampling, prompting recent studies to jointly optimize feature quality and diversity. ICCTP clusters tokens via noun anchors Yu et al. (2026), TrimTokenator couples mutual information with manifold divergence Zhang et al. (2026), CrisPrune enforces spatial repulsion using determinantal point processes Liu et al. (2026), and AdaPrune alongside ID-Pruner balances both criteria via two-stage filtering or orthogonal decoupling Zhu et al. (2026); Ji et al. (2026b). Drawing upon submodular optimization principles, ReCo formulates visual token reduction as a facility-location coreset selection problem governed by a relational transition kernel. Under this formulation, ReCo greedily selects tokens yielding maximal marginal coverage gains, thereby promoting comprehensive coverage of global relational representations while mitigating token redundancy.

3 Methodology

ReCo (Figure 2) is a training-free visual token pruning framework. It constructs a unified relational kernel by integrating pre- and post-projection relations with their transition dynamics (attenuation, preservation, and enhancement). Subsequently, a max-coverage selector identifies a compact subset of representative tokens.

3.1 Problem Formulation

Given an input image, an LVLM vision encoder first extracts a visual token sequence:

$$X = [x_1, x_2, \ldots, x_N]^\top \in \mathbb{R}^{N \times d_v}, \tag{1}$$

where d_v denotes the visual feature dimension. A multimodal projector $\mathcal{P}(\cdot)$ subsequently maps these representations into the LLM embedding space:

$$Y = [y_1, y_2, \ldots, y_N]^\top \in \mathbb{R}^{N \times d_l}, \quad y_i = \mathcal{P}(x_i), \tag{2}$$

with d_l being the language hidden dimension.

While pairwise relations in X characterize visual appearance similarity, those in Y capture linguistic semantic consistency; their discrepancy reflects the cross-modal reorganization of interaction affinities. ReCo leverages this relational transition to formalize visual token pruning as a coreset selection problem over token indices $\mathcal{V} = \{1, \ldots, N\}$. Given a retention budget $K < N$, our objective is to select a subset S^* that maximizes relational coverage:

$$S^* = \arg \max_{S \subseteq \mathcal{V}, |S| = K} \mathcal{F}(S; X, Y), \tag{3}$$

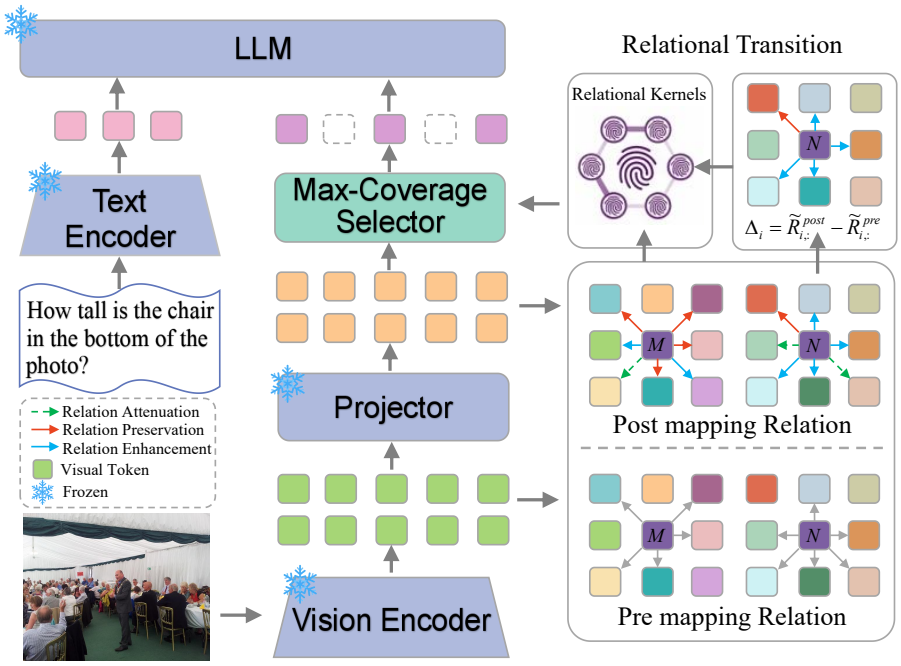


Figure 2: Overview of the proposed ReCo framework. ReCo extracts pairwise token relation matrices prior to and following the cross-modal projector. By measuring relational shifts (Δ), it characterizes three transition regimes: relation attenuation, preservation, and enhancement, forming composite relational kernels. A max-coverage selector is then deployed to greedily select a compact, representative visual token subset to be fed into the LLM alongside text tokens.

where $\mathcal{F}(S; X, Y)$ measures how comprehensively S preserves the overall relational transition across the multimodal interface. The selected tokens $Y_{S^*} = [y_j]_{j \in S^*}$ are retained in their original raster-scan order before being fed into the LLM.

3.2 Characterizing Cross-Modal Relational Transitions

The cross-modal projector bridges the visual representation space and the language embedding space. ReCo captures the collective organization of visual tokens through their pairwise relationship matrices before and after projection, as well as the transition dynamics between them.

Pre- and Post-Mapping Relation Matrices. We first normalize the individual token representations to unit hyperspheres:

$$\hat{x}_i = \frac{x_i}{\|x_i\|_2}, \qquad \hat{y}_i = \frac{y_i}{\|y_i\|_2}, \quad \forall i \in \{1, \dots, N\}. \tag{4}$$

The pairwise relation matrices in the pre-projection visual space and post-projection language-aligned space are computed via cosine similarity:

$$R^{pre}_{ij} = \hat{x}_i^\top \hat{x}_j, \qquad R^{post}_{ij} = \hat{y}_i^\top \hat{y}_j, \quad \forall i, j \in \{1, \dots, N\}, \tag{5}$$

where $R^{pre}, R^{post} \in [-1, 1]^{N \times N}$.

Transition-Induced Relational Shifts. Directly computing the element-wise difference $R^{post} - R^{pre}$ is confounded by token-specific similarity offsets: certain tokens naturally exhibit higher baseline affinities across the sequence due to shared background contexts or representation anisotropy. To isolate the genuine relational reorganization induced by cross-modal projection, we eliminate diagonal self-affinities and row-center the matrices by their off-diagonal means:

$$\widetilde{R}^*_{ij} = \begin{cases} R^*_{ij} - \mu^*_i, & i \neq j, \\ 0, & i = j, \end{cases} \quad \text{where } \mu^*_i = \frac{1}{N-1} \sum_{k \neq i} R^*_{ik}, \quad * \in \{pre, post\}. \tag{6}$$

For each token i, its relational shift descriptor is formulated as the profile difference across all visual contexts:

$$\Delta_i = \widetilde{R}^{post}_{i,:} - \widetilde{R}^{pre}_{i,:} \in \mathbb{R}^N. \tag{7}$$

Consequently, Δ_i encapsulates the cross-modal reorganization of token i's interaction affinities. Specifically, individual entries in Δ_i naturally delineate three transition regimes: positive shifts ($\Delta_{ij} > 0$), near-zero variations ($\Delta_{ij} \approx 0$), and negative shifts ($\Delta_{ij} < 0$) capture relation enhancement, preservation, and attenuation between tokens i and j, respectively.

Relational Kernel Construction. To establish metrics for evaluating inter-token substitutability, we map these relational representations onto positive semi-definite affinity kernels bounded within $[0, 1]$ via linear scaling:

$$K^{pre}_{ij} = \frac{1 + R^{pre}_{ij}}{2}, \quad K^{post}_{ij} = \frac{1 + R^{post}_{ij}}{2}, \quad K^{shift}_{ij} = \frac{1 + \widehat{\Delta}_i^\top \widehat{\Delta}_j}{2}, \tag{8}$$

where $\widehat{\Delta}_i = \Delta_i / \|\Delta_i\|_2$ denotes the normalized relational shift vector. Here, K^{shift}_{ij} measures the trajectory alignment of relational reorganization between tokens i and j. Consequently, $\{K^{pre}, K^{post}, K^{shift}\}$ quantify pairwise token substitutability across visual appearance, linguistic semantics, and relational transition dynamics.

3.3 Scale-Aligned Relational Kernel Fusion

To align the dispersion scales across visual, linguistic, and transition dynamics, we standardize each relational kernel $K \in \{K^{pre}, K^{post}, K^{shift}\}$. Specifically, we apply double centering to each kernel:

$$K_c = HKH, \quad \text{where} \quad H = I_N - \frac{1}{N}\mathbf{1}_N\mathbf{1}_N^\top, \tag{9}$$

and $\mathbf{1}_N \in \mathbb{R}^N$ is the all-ones vector. We then normalize the variation scale using the root-mean-square magnitude of K_c:

$$\mathrm{Align}(K) = \frac{K_c}{\frac{1}{N}\|K_c\|_F + \varepsilon}, \tag{10}$$

where $\|K_c\|_F$ denotes the Frobenius norm and $\varepsilon = 10^{-6}$ ensures numerical stability.

With all three kernels centered and standardized to comparable scales, we construct the unified transition kernel via unweighted summation followed by non-negative rectification:

$$K^{fused} = \mathrm{ReLU}\left(\mathrm{Align}(K^{pre}) + \mathrm{Align}(K^{post}) + \mathrm{Align}(K^{shift})\right). \tag{11}$$

Here, the $\mathrm{ReLU}(\cdot)$ operator guarantees $K^{fused}_{ij} \geq 0$, retaining non-negative affinity values required by the subsequent facility location objective.

3.4 Facility-Location Coreset Selection

Submodular Coverage Objective. Using the unified transition kernel K^{fused}, we instantiate the relational coverage utility $\mathcal{F}(S)$ from Section 3.1 via a facility location objective:

$$\max_{S \subseteq \mathcal{V},\, |S|=K} \mathcal{F}(S) \triangleq \sum_{i \in \mathcal{V}} \max_{j \in S} K^{\text{fused}}_{ij}. \tag{12}$$

Since $K^{\text{fused}}_{ij} \geq 0$, $\mathcal{F}(S)$ is a monotone submodular function, which inherently promotes diversity via diminishing marginal gains as relational profiles become covered.

Max-Coverage Selection. We optimize Eq. equation 12 via a greedy max-coverage selector. Starting with $S = \emptyset$ and coverage $c_i = 0$, each iteration incorporates the token j^* yielding the maximal marginal gain:

$$j^* = \arg\max_{j \in \mathcal{V} \setminus S} \sum_{i \in \mathcal{V}} \max\left(K^{\text{fused}}_{ij} - c_i,\, 0\right), \tag{13}$$

where $c_i = \max_{j \in S} K^{\text{fused}}_{ij}$ tracks accumulated coverage. After K iterations, the selected tokens Y_S are restored to their original raster-scan order and fed into the LLM.

Table 1: Performance on LLaVA-1.5-7B. Bold/underlined values denote the best/second-best results per budget; Avg†. denotes mean performance relative to the unpruned 576-token model.

Method	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	VQAv2 Acc. ↑	TextVQA Acc. ↑	Avg.† ↑
Total 576 Tokens								
LLaVA-1.5-7B	61.90	64.70	1862.00	85.90	69.50	78.50	58.20	100%
Retain 192 Tokens ↓ 66.7%								
ToMe (ICLR 2023)	54.30	60.50	1563.00	72.40	65.20	68.00	52.10	88.49%
FastV (ECCV 2024)	52.70	61.20	1612.00	64.80	67.30	67.10	52.50	87.75%
HiRED (AAAI 2025)	58.70	62.80	1737.00	82.80	68.40	74.90	47.40	93.84%
DivPrune (CVPR 2025)	59.97	62.54	1762.23	87.00	68.91	76.87	56.97	97.78%
PyramidDrop (CVPR 2025)	57.10	63.20	1766.00	82.30	68.80	75.10	56.10	95.95%
SparseVLM (ICML 2025)	57.60	62.50	1721.00	83.60	69.10	75.60	56.10	95.93%
PruneSID (ICLR 2026)	60.10	**63.70**	**1791.00**	86.90	68.50	76.80	56.70	98.10%
FiCoCo-V (AAAI 2026)	58.50	62.30	1732.00	82.50	67.80	68.69	55.70	94.37%
ZOO-Prune (CVPR 2026)	60.03	62.89	1781.66	87.24	69.16	77.34	**57.30**	98.27%
ReCo (Ours)	**60.17**	63.32	1783.85	**87.69**	**69.46**	**77.52**	56.82	**98.47%**
Retain 128 Tokens ↓ 77.8%								
ToMe (ICLR 2023)	52.40	53.30	1343.00	62.80	59.60	63.00	49.10	80.38%
FastV (ECCV 2024)	49.60	56.10	1490.00	59.60	60.20	61.80	50.60	81.22%
HiRED (AAAI 2025)	57.20	61.50	1710.00	79.80	68.10	73.40	46.10	91.84%
DivPrune (CVPR 2025)	59.25	62.03	1718.22	86.72	**68.96**	75.96	56.06	96.73%
PyramidDrop (CVPR 2025)	56.00	61.00	1644.00	82.30	68.30	72.90	55.10	93.52%
SparseVLM (ICML 2025)	56.00	60.00	1696.00	80.50	67.10	73.80	54.90	93.27%
PruneSID (ICLR 2026)	58.80	62.10	1749.00	86.50	68.30	75.30	54.70	96.26%
FiCoCo-V (AAAI 2026)	57.60	61.10	1711.00	82.20	68.30	65.94	55.60	93.27%
ZOO-Prune (CVPR 2026)	59.49	61.86	1751.60	87.13	68.91	**76.57**	**57.87**	97.62%
ReCo (Ours)	**59.79**	**62.97**	**1762.00**	**87.44**	68.77	76.42	56.53	**97.68%**
Retain 64 Tokens ↓ 88.9%								
ToMe (ICLR 2023)	48.60	43.70	1138.00	52.50	50.00	57.10	45.30	70.12%
FastV (ECCV 2024)	46.10	48.00	1256.00	48.00	51.10	55.00	47.80	71.10%
HiRED (AAAI 2025)	54.60	60.20	1599.00	73.60	68.20	69.70	44.20	87.95%
DivPrune (CVPR 2025)	57.78	59.28	1674.40	85.56	68.17	74.11	54.69	94.42%
PyramidDrop (CVPR 2025)	41.90	33.30	1092.00	55.90	68.60	69.20	45.90	72.66%
SparseVLM (ICML 2025)	52.70	56.20	1505.00	75.10	62.20	68.20	51.80	86.52%
PruneSID (ICLR 2026)	57.10	58.80	**1733.00**	83.80	67.80	73.70	54.20	94.05%
FiCoCo-V (AAAI 2026)	52.40	60.30	1591.00	76.00	68.10	59.75	53.60	88.28%
ZOO-Prune (CVPR 2026)	58.47	60.22	1675.59	85.86	68.27	75.02	55.35	95.20%
ReCo (Ours)	**58.69**	**61.58**	1644.12	**85.92**	**68.67**	**75.09**	**55.64**	**95.48%**

4 Experiments

4.1 Experimental Setup

We evaluate ReCo across different LVLMs under matched budgets: LLaVA-1.5-7B (retaining $\{192, 128, 64\}$ of 576 tokens), AnyRes LLaVA-NeXT-7B (retaining $\{640, 320, 160\}$ of $\leq 2{,}880$ tokens), and Qwen2.5-VL-7B-Instruct with dynamic-resolution inputs (10% and 20% retention). We compare against existing methods, including ToMe Bolya et al. (2023), FastV Chen et al. (2024), HiRED Arif et al. (2025), DivPrune Alvar et al. (2025), PyramidDrop Xing et al. (2025), SparseVLM Zhang et al. (2025), PruneSID Fang et al. (2026), FiCoCo-V Han et al. (2026b), ApET Ma et al. (2026), and ZOO-Prune Kim et al. (2026). More details are provided in Appendix A.2.

Table 2: Performance on LLaVA-NeXT-7B. "–" denotes missing results.

Method	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	VQAv2 Acc. ↑	TextVQA Acc. ↑	Avg.† ↑
Total 2,880 Tokens								
LLaVA-NeXT-7B	64.20	67.90	1842.00	86.40	70.20	80.10	61.30	100%
Retain 640 Tokens ↓ 77.8%								
SparseVLM (ICML 2025)	60.30	65.70	1772.00	–	67.70	77.10	57.80	–
DivPrune (CVPR 2025)	61.58	65.38	1773.04	85.51	67.82	78.94	55.41	96.14%
PruneSID (ICLR 2026)	61.60	64.20	1795.00	86.30	68.30	78.50	54.57	96.02%
ApET (CVPR 2026)	63.00	65.30	1815.00	87.20	68.22	79.20	57.90	97.75%
ZOO-Prune (CVPR 2026)	62.19	65.21	**1816.45**	86.75	68.02	**79.64**	**57.98**	97.55%
ReCo (Ours)	**63.61**	**66.41**	1796.54	**88.42**	**68.42**	79.36	56.45	**97.91%**
Retain 320 Tokens ↓ 88.9%								
SparseVLM (ICML 2025)	57.70	64.30	1694.00	–	67.30	73.40	55.90	–
DivPrune (CVPR 2025)	59.63	63.66	1731.04	83.47	67.82	76.64	53.84	93.91%
PruneSID (ICLR 2026)	60.50	63.00	1754.00	83.10	67.30	76.60	54.25	94.06%
ApET (CVPR 2026)	61.00	63.50	1783.00	85.60	67.28	75.80	54.40	94.80%
ZOO-Prune (CVPR 2026)	60.97	64.86	**1787.68**	85.47	67.77	**78.08**	**57.28**	**96.28%**
ReCo (Ours)	**61.53**	**65.03**	1774.18	**87.23**	**68.02**	76.94	55.26	96.00%
Retain 160 Tokens ↓ 94.4%								
SparseVLM (ICML 2025)	51.20	63.10	1542.00	–	67.50	66.30	46.40	–
DivPrune (CVPR 2025)	57.79	62.29	1658.25	79.36	68.02	73.92	52.42	91.19%
PruneSID (ICLR 2026)	58.90	60.80	1704.00	76.90	67.10	73.80	52.82	90.96%
ApET (CVPR 2026)	58.40	60.80	1680.00	82.60	67.67	72.70	53.80	91.75%
ZOO-Prune (CVPR 2026)	59.93	**64.18**	**1738.64**	83.05	**68.42**	**76.12**	**55.42**	**94.47%**
ReCo (Ours)	**60.04**	63.23	1663.67	**85.42**	67.18	74.88	52.82	93.02%

4.2 Performance Across Diverse Tasks

LLaVA-1.5-7B. Table 1 shows that ReCo achieves the highest average performance retention across all three budgets (retaining 95.48% with 64 tokens), exhibiting a gradual performance degradation trend. Under the 64-token budget, ReCo achieves the best performance on six out of seven benchmarks. Its POPE score (85.92) is on par with the unpruned model, indicating that modeling cross-modal relational transitions helps identify and retain complementary localized evidence, thereby mitigating object hallucination while filtering out homogeneous visual redundancy. **LLaVA-NeXT-7B.** As shown in Table 2, on the high-resolution sequences produced by AnyRes, ReCo retains 93.02%–97.91% of the performance across budgets ranging from 160 to 640 tokens, achieving the highest scores on GQA and POPE under all budget settings. Under the stringent budget of 160 tokens, however, TextVQA performance exhibits a noticeable decrease. This reflects an inherent trade-off: a task-agnostic global coverage strategy struggles to fully preserve localized details, such as small text, under highly constrained token quotas. **Qwen2.5-VL-7B-Instruct.** As shown in Table 3, under the dynamic resolution setting, ReCo maintains 91.02% of the performance at a 10% retention budget (outperforming the second-best method by 3.47 percentage points), while exceeding the second-best method by 8.01 on TextVQA. Across three model architectures, ReCo operates effectively under both fixed-length and dynamic-resolution settings, demonstrating the generalizability and adaptability of the relational transition-based coreset selection strategy across diverse visual representations.

4.3 Component Ablation

Table 4 evaluates the individual contributions of pre-projection relations (Pre), post-projection relations (Post), and relational shifts (Shift). Full ReCo demonstrates consistently balanced performance across budgets, achieving the highest scores across all six benchmarks at 128 tokens. Ablating any

Table 3: Performance comparison on Qwen2.5-VL-7B-Instruct.

Method	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	VQAv2 Acc. ↑	TextVQA Acc. ↑	Avg.† ↑
Retain 100% visual tokens								
Qwen2.5-VL-7B-Instruct	60.85	83.42	2306.64	86.35	87.80	83.96	77.73	100%
Retain 20% visual tokens ↓ 80.0%								
PruneSID (ICLR 2026)	58.75	79.81	2146.54	84.86	83.89	79.69	63.84	93.74%
FiCoCo-V (AAAI 2026)	55.11	77.49	2060.64	77.61	80.22	76.33	54.87	87.93%
ZOO-Prune (CVPR 2026)	**58.86**	77.32	2046.86	**84.90**	84.68	78.98	46.48	89.54%
ReCo (Ours)	58.62	**80.24**	**2234.47**	82.98	**85.03**	**80.14**	**70.48**	**95.49%**
Retain 10% visual tokens ↓ 90.0%								
PruneSID (ICLR 2026)	54.63	75.43	2008.25	79.66	82.10	75.33	54.47	87.55%
FiCoCo-V (AAAI 2026)	51.14	72.34	1898.65	71.31	78.28	70.94	43.69	80.79%
ZOO-Prune (CVPR 2026)	55.26	73.63	1945.55	**80.52**	81.90	74.61	32.63	82.97%
ReCo (Ours)	**56.05**	**77.15**	**2160.85**	79.73	**83.74**	**76.24**	**62.48**	**91.02%**

Table 4: Component ablation of ReCo on LLaVA-1.5-7B.

Variant	Pre	Post	Shift	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	TextVQA Acc. ↑
Retain 192 tokens ↓ 66.7%									
ReCo (Ours)	✓	✓	✓	60.17	**63.32**	**1783.85**	**87.69**	**69.46**	56.82
w/o Shift	✓	✓	–	59.42	63.29	1782.49	87.19	68.67	56.88
w/o Post	✓	–	✓	**60.45**	63.14	1776.87	86.69	68.91	56.69
w/o Pre	–	✓	✓	60.12	63.21	1779.38	86.84	69.01	56.52
Pre only	✓	–	–	60.03	63.15	1779.53	87.26	68.91	56.77
Post only	–	✓	–	59.84	63.14	1763.83	86.99	69.01	**57.15**
Shift only	–	–	✓	59.80	63.11	1768.38	85.24	68.62	56.22
Retain 128 tokens ↓ 77.8%									
ReCo (Ours)	✓	✓	✓	**59.79**	**62.97**	**1762.00**	**87.44**	**68.77**	**56.53**
w/o Shift	✓	✓	–	59.72	62.03	1713.98	86.64	68.67	56.06
w/o Post	✓	–	✓	59.67	62.03	1725.33	86.75	68.72	56.52
w/o Pre	–	✓	✓	59.47	62.44	1757.88	86.38	68.52	56.43
Pre only	✓	–	–	59.52	62.80	1736.93	86.77	68.02	55.98
Post only	–	✓	–	58.62	62.29	1738.49	86.76	68.62	56.52
Shift only	–	–	✓	58.85	62.63	1732.93	83.68	68.32	55.48
Retain 64 tokens ↓ 88.9%									
ReCo (Ours)	✓	✓	✓	**58.69**	**61.58**	1644.12	**85.92**	**68.67**	**55.64**
w/o Shift	✓	✓	–	58.51	61.34	1671.82	84.68	68.36	55.14
w/o Post	✓	–	✓	58.64	61.00	1680.04	83.49	68.47	54.60
w/o Pre	–	✓	✓	58.57	61.43	**1683.60**	83.96	68.02	55.06
Pre only	✓	–	–	58.55	61.49	1655.21	85.12	**68.67**	54.72
Post only	–	✓	–	58.34	61.34	1649.48	84.57	**68.67**	55.48
Shift only	–	–	✓	56.51	61.00	1671.50	78.05	68.29	52.90

single component leads to performance degradation (e.g., removing Shift drops MME by 48.02; removing Pre drops POPE by 1.06), indicating that the three views provide complementary information rather than redundant cues. Single-component variants exhibit distinct task preferences: Pre maintains competitive results on POPE, aiding the retention of object-level grounding evidence; Post performs favorably on TextVQA, reflecting the utility of language-aligned representations; Shift yields limited performance in isolation, yet its omission causes notable degradation, highlighting the necessity of relational transition dynamics as a complementary signal. Under the 64-token budget, ReCo outperforms all two-component variants on POPE by 1.24–2.43, suggesting that joint modeling facilitates the retention of diverse task-relevant evidence under substantial compression.

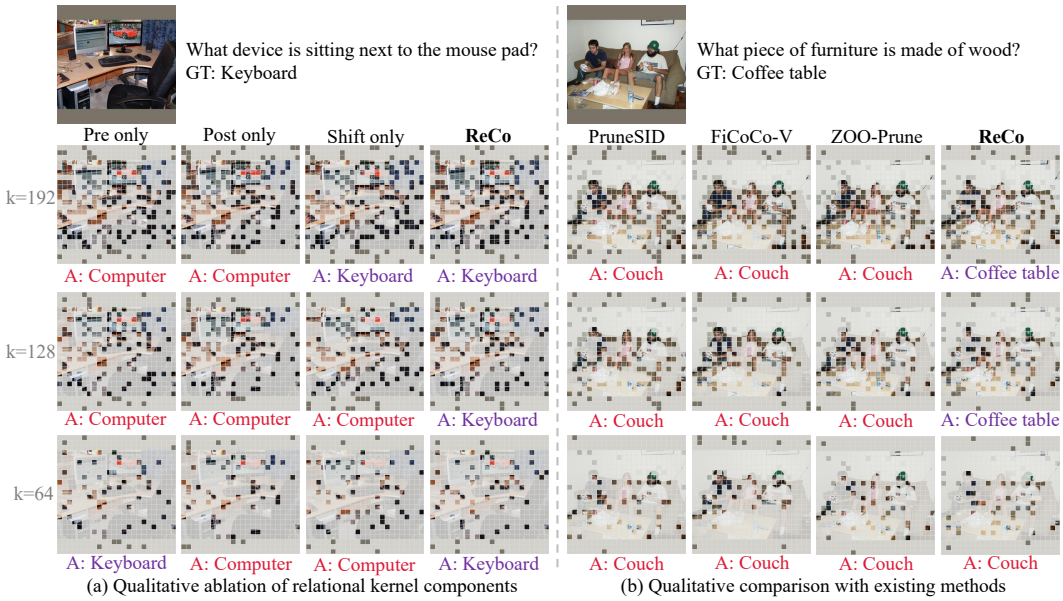


Figure 3: Qualitative results across budgets. (a) Ablation of relational components. (b) Comparison with existing methods. Purple and red denote correct and incorrect predictions, respectively.

4.4 Qualitative Analysis

Figure 3 qualitatively illustrates the complementarity among ReCo's relational components and its behavior relative to existing pruning methods. As shown in Figure 3(a), the Post-only variant persistently focuses on globally salient regions and predicts "Computer" across all budgets. Shift-only progressively loses the evidence required to identify the keyboard as the budget decreases, whereas Pre-only succeeds only at $k = 64$. In contrast, full ReCo integrates visual-space relations, language-aligned relations, and projection-induced relational shifts, reliably retaining the localized evidence needed to predict "Keyboard" across all three budgets. In the multi-object scene in Figure 3(b), PruneSID, FiCoCo-V, and ZOO-Prune concentrate retained tokens on the people and sofa while insufficiently covering the less prominent foreground table, resulting in the incorrect prediction "Couch" across all budgets. ReCo distributes tokens more evenly and correctly identifies the "Coffee table" at $k = 192$ and $k = 128$, consistent with the facility-location objective's property of penalizing redundant coverage on already represented regions. At $k = 64$, however, the constrained token capacity leads to the loss of critical local cues and causes ReCo to revert to "Couch", reflecting both the utility of relational coverage and its boundary under aggressive compression. Additional qualitative results are provided in Appendix A.6.

5 Conclusion

This paper presents ReCo, a training-free framework that prunes visual tokens by modeling relational transitions across cross-modal projection. By unifying visual-space, language-aligned, and transition-shift relations via scale alignment, ReCo optimizes a submodular facility-location objective to select a representative and complementary token coreset, effectively mitigating visual redundancy while preserving grounding cues. Evaluations on LLaVA-1.5, LLaVA-NeXT, and Qwen2.5-VL show that ReCo retains 95.48%, 96.00%, and 91.02% performance under 88.9%–90% token reduction, outperforming the runner-up on Qwen2.5-VL by 3.47 percentage across fixed-length and dynamic-resolution settings. Nonetheless, its task-agnostic selection may overlook localized details (e.g., fine text) under tight token budgets, and dense pairwise relation matrices incur quadratic complexity with sequence length. Future work will explore query guidance and sparse relational modeling to improve localized retention and enhance computational scalability.

AI USE STATEMENT

In this work, we used GLM-5.3-Flash, developed by Zhipu AI, to assist with manuscript writing and language polishing, including improvements to grammar, clarity, readability, and phrasing.

REFERENCES

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9392–9401. IEEE, 2025.

K. H. I. Arif, J. Yoon, D. S. Nikolopoulos, H. Vandierendonck, D. John, and B. Ji. Hired: Attention-guided token dropping for efficient inference of high-resolution vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 1773–1781. AAAI, 2025.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. 2023. doi: 10. 48550. *arXiv preprint ARXIV.2308.12966*, 1, 2023.

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 (ICLR)*, 2023.

Jianjian Cao, Peng Ye, Shengze Li, Chong Yu, Yansong Tang, Jiwen Lu, and Tao Chen. Madtp: Multimodal alignment-guided dynamic token pruning for accelerating vision-language transformer. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15710–15719. IEEE, 2024.

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. Springer, 2024.

Joonmyung Choi, Sanghyeok Lee, Jongha Kim, Sehyung Kim, Dohwan Ko, Jihyung Kil, and Hyunwoo J. Kim. Docprune: Efficient document question answering via background, question, and comprehension-aware token pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3543–3552, June 2026.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, volume 2024, pp. 35549–35562, 2024.

Zhengyao Fang, Pengyuan Lyu, Chengquan Zhang, Guangming Lu, Jun Yu, and Wenjie Pei. Prune redundancy, preserve essence: Vision token compression in vlms via synergistic importance-diversity. In *International Conference on Learning Representations (ICLR)*, pp. 51976–51999, 2026.

Seunghoon Han, Hyewon Lee, Soyoung Park, Jong-Ryul Lee, and Sungsu Lim. MR-pruner: Training-free multi-resolution visual token pruning for multi-modal large language models. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1084–1093. IEEE, 2026a.

Yuhang Han, Xuyang Liu, Zihan Zhang, Pengxiang Ding, Junjie Chen, Honggang Chen, Donglin Wang, Qingsen Yan, and Siteng Huang. Filter, correlate, compress: Training-free token reduction for mllm acceleration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 4601–4609, 2026b.

Xin Huang, Ashish Khetan, Rene Bidart, and Zohar Karnin. Pyramid-BERT: Reducing complexity via successive core-set based token selection. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 8798–8817. Association for Computational Linguistics, 2022.

Huawei Ji, Yuanhao Sun, Yuan Jin, Cheng Deng, Jiaxin Ding, Luoyi Fu, and Xinbing Wang. VisPCO: Visual token pruning configuration optimization via budget-aware Pareto-frontier learning for vision-language models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 9281–9301. Association for Computational Linguistics, 2026a.

Jie Ji, Gen Li, and Fatemeh Afghah. ID-pruner: Disentangling importance and diversity for training-free visual token pruning. In *IEEE International Conference on Image Processing (ICIP)*, pp. 1–6. IEEE, 2026b.

Vishal Kaushal, Rishabh Iyer, Suraj Kothawade, Rohan Mahadev, Khoshrav Doctor, and Ganesh Ramakrishnan. Learning from less data: A unified data subset selection and active learning framework for computer vision. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1289–1299. IEEE, 2019.

Youngeun Kim, Youjia Zhang, Huiling Liu, Aecheon Jung, Sunwoo Lee, and Sungeun Hong. Zoo-prune: training-free token pruning via zeroth-order gradient estimation in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39572–39582, 2026.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Ziniu Liu, Shuheng Zhou, Mingqing Liu, Hao Deng, and Huijia Zhu. CrisPrune: Combining contextual relevance and intrinsic saliency for efficient visual token pruning in MLLMs. In *Findings of the Association for Computational Linguistics (ACL)*, pp. 13546–13564. Association for Computational Linguistics, 2026.

Qiankun Ma, Ziyao Zhang, Haofei Wang, Zhen Song, Jie Chen, and Hairong Zheng. Apet: Approximation-error guided token compression for efficient vlms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26306–26316, 2026.

Surendra Pathak and Bo Han. Towards efficient large vision-language models: A comprehensive survey on inference strategies. *arXiv preprint arXiv:2603.27960*, 2026.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22857–22867. IEEE, 2025.

Yi Wang, Haofei Zhang, Qihan Huang, Anda Cao, Gongfan Fang, Wei Wang, Xuan Jin, Jie Song, Mingli Song, and Xinchao Wang. Rethinking token reduction for large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24727–24737, June 2026.

Kai Wei, Rishabh Iyer, and Jeff Bilmes. Submodularity in data subset selection and active learning. In *International Conference on Machine Learning (ICML)*, pp. 1954–1963. PMLR, 2015.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, and Dahua Lin. Pyramiddrop: Accelerating your large vision-language models via pyramid visual redundancy reduction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14593–14603. IEEE, 2025.

R. Xu, Y. Wang, Y. Luo, and B. Du. Rethinking visual token reduction in LVLMs under cross-modal misalignment. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 27323–27331. AAAI, 2026.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19792–19802. IEEE, 2025.

W. Ye, Q. Wu, W. Lin, and Y. Zhou. Fit and prune: Fast and training-free visual token pruning for multi-modal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 22128–22136. AAAI, 2025.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. MM-vet: Evaluating large multimodal models for integrated capabilities. In *International Conference on Machine Learning (ICML)*, pp. 57730–57754. PMLR, 2024.

Y. Yu, B. Chen, Y. Zhang, T. Xie, M. Jing, and L. Zuo. Instruction-guided cross-modal clustering for training-free visual token pruning in vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 12213–12221. AAAI, 2026.

Wang Zeng, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, Wanli Ouyang, and Xiaogang Wang. Not all tokens are equal: Human-centric visual analysis via token clustering transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11091–11101. IEEE, 2022.

Hao Zhang, Lyu Mengsi, Chenrui He, Yulong Ao, and Yonghua Lin. TrimTokenator: Towards adaptive visual token pruning for large multimodal models. In *Findings of the Association for Computational Linguistics (ACL)*, pp. 32633–32650. Association for Computational Linguistics, 2026.

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 *International Conference on Machine Learning (ICML)*, pp. 74840–74857. PMLR, 2025.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, et al. H2o: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in neural information processing systems*, 36:34661–34710, 2023.

Xiuyuan Zhu, Ke Lu, Hao Wu, and Jian Xue. AdaPrune: A two-stage filter-select methods for visual token pruning in specialized VLMs. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 12647–12651. IEEE, 2026.

A Appendix

A.1 Algorithm and Theoretical Analysis

The mathematical definitions of the pre-projection relation, post-projection relation, relational shift, affinity kernels, kernel alignment, and kernel fusion are given in Sections 3.2 and 3.3. This appendix provides the complete selection procedure and its theoretical properties.

Complete ReCo Algorithm. Algorithm 1 summarizes the complete ReCo pipeline. The relational representations and the fused kernel are constructed exactly as defined in the main text. The algorithm then applies greedy facility-location selection and restores the original raster-scan order of the selected tokens. The complete computational procedure of ReCo is summarized in Algorithm 1. The algorithm first constructs the relational transition kernel and then performs greedy facility-location selection. When multiple candidates yield the same marginal gain, the candidate with the smaller original token index is selected. The selected tokens are finally fed into the language model in their original sequence order.

Submodularity of Facility-Location Selection. Let $\mathcal{V} = \{1, \ldots, N\}$ and define the facility-location objective $\mathcal{F}(S)$ as in Eq. equation 12, with the convention $\mathcal{F}(\emptyset) = 0$.

Proposition. If $K_{ij}^{\text{fused}} \geq 0$ for all i, j, then $\mathcal{F}(S)$ is normalized, monotone, and submodular. For each row i, define $f_i(S) = \max_{j \in S} K_{ij}^{\text{fused}}$ and $f_i(\emptyset) = 0$. Since all affinities are nonnegative, enlarging the set cannot decrease $f_i(S)$; hence each f_i is monotone. For $A \subseteq B \subseteq \mathcal{V}$ and $e \notin B$, the marginal gain from adding e is

$$f_i(A \cup \{e\}) - f_i(A) = \max\big(K_{ie}^{\text{fused}} - f_i(A), 0\big)\,.$$

Because $f_i(A) \leq f_i(B)$, the marginal gain obtained by adding e to A is no smaller than that obtained by adding it to B. Therefore, each f_i satisfies the diminishing-returns property. Since $\mathcal{F}(S) = \sum_i f_i(S)$, it is also monotone and submodular. Consequently, under the cardinality constraint $|S| \leq K$, the greedy procedure in Algorithm 1 achieves the standard $(1 - 1/e)$ approximation guarantee with respect to the optimal cardinality-constrained subset. Regarding the affinity matrix,

Algorithm 1 ReCo: Relational Transition Coreset Token Pruning

Require: Visual features $X \in \mathbb{R}^{N \times d_v}$, projected features $Y \in \mathbb{R}^{N \times d_l}$, retention budget K, and $\varepsilon > 0$
Ensure: Selected token index set $S \subseteq \{1, \dots, N\}$ with $|S| = K$

// 1. Relational transition extraction

$\hat{x}_i \leftarrow x_i / \max(\|x_i\|_2, \varepsilon)$ and $\hat{y}_i \leftarrow y_i / \max(\|y_i\|_2, \varepsilon), \quad \forall i \in \{1, \dots, N\}$

$R^{\mathrm{pre}}_{ij} \leftarrow \hat{x}_i^\top \hat{x}_j$ and $R^{\mathrm{post}}_{ij} \leftarrow \hat{y}_i^\top \hat{y}_j, \quad \forall i, j \in \{1, \dots, N\}$

for $q \in \{\mathrm{pre}, \mathrm{post}\}$ **do**

  $\mu^q_i \leftarrow \dfrac{1}{\max(N-1, 1)} \sum_{j \neq i} R^q_{ij}, \quad \forall i$

  $\widetilde{R}^q_{ij} \leftarrow R^q_{ij} - \mu^q_i$ if $i \neq j$; otherwise $\widetilde{R}^q_{ij} \leftarrow 0$

end for

$\Delta_i \leftarrow \widetilde{R}^{\mathrm{post}}_{i,:} - \widetilde{R}^{\mathrm{pre}}_{i,:}, \quad \forall i$

$\widehat{\Delta}_i \leftarrow \Delta_i / \max(\|\Delta_i\|_2, \varepsilon), \quad \forall i$

// 2. Kernel construction, alignment, and fusion

$K^{\mathrm{pre}}_{ij} \leftarrow (1 + R^{\mathrm{pre}}_{ij})/2$

$K^{\mathrm{post}}_{ij} \leftarrow (1 + R^{\mathrm{post}}_{ij})/2$

$K^{\mathrm{shift}}_{ij} \leftarrow (1 + \widehat{\Delta}_i^\top \widehat{\Delta}_j)/2, \quad \forall i, j$

$H \leftarrow I_N - \dfrac{1}{N} \mathbf{1}_N \mathbf{1}_N^\top$

for $q \in \{\mathrm{pre}, \mathrm{post}, \mathrm{shift}\}$ **do**

  $K^q_c \leftarrow H K^q H$

  $A^q \leftarrow \dfrac{K^q_c}{\max(\|K^q_c\|_F / N, \varepsilon)}$

end for

$K^{\mathrm{fused}} \leftarrow \mathrm{ReLU}\big(A^{\mathrm{pre}} + A^{\mathrm{post}} + A^{\mathrm{shift}}\big)$

// 3. Facility-location greedy selection

$S \leftarrow \emptyset$ and $c_i \leftarrow 0$ for all $i \in \{1, \dots, N\}$

for $t = 1$ **to** K **do**

  $j^* \leftarrow \arg\max_{j \notin S} \sum_{i=1}^{N} \max\big(K^{\mathrm{fused}}_{ij} - c_i, 0\big)$

  $S \leftarrow S \cup \{j^*\}$

  $c_i \leftarrow \max\big(c_i, K^{\mathrm{fused}}_{ij^*}\big), \quad \forall i \in \{1, \dots, N\}$

end for

return $\mathrm{SortAscending}(S)$ ▷ Preserve the original raster-scan order

the guarantee requires only entrywise nonnegativity, not positive semidefiniteness. We therefore do not assume that elementwise ReLU preserves positive semidefiniteness, as this property is unnecessary for the facility-location objective.

Complexity and Efficiency Analysis. Let N denote the number of visual tokens, and let d_v and d_l denote the pre- and post-projection feature dimensions, respectively. Computing the pre- and post-projection relation matrices requires $O(N^2 d_v)$ and $O(N^2 d_l)$ operations. The relation-centering, kernel alignment, and kernel fusion steps require $O(N^2)$ operations.

Because each relational-shift descriptor has dimension N, explicitly constructing the pairwise shift kernel requires $O(N^3)$ operations in the naive implementation. At each greedy iteration, evaluating all candidate tokens against all N rows requires $O(N^2)$ operations. Therefore, selecting K tokens requires $O(KN^2)$ operations.

The overall naive selection complexity is

$$O\big(N^2(d_v + d_l) + N^3 + KN^2\big), \tag{14}$$

and the dominant auxiliary memory requirement is $O(N^2)$.

The timing measurements and end-to-end inference analysis are reported separately in Appendix A.5.

A.2 Experimental Setup

All experiments are conducted on 8 NVIDIA RTX A6000 GPUs, each with 48 GB of memory. We use the official model checkpoints without additional training or parameter updates. Evaluation is

Table 5: Naive computational and auxiliary-memory complexity of ReCo.

Component	Complexity
Pre/post relation matrices	$O(N^2(d_v + d_l))$
Relation centering and kernel fusion	$O(N^2)$
Relational-shift kernel	$O(N^3)$
Facility-location greedy selection	$O(KN^2)$
Auxiliary memory	$O(N^2)$

implemented with PyTorch and `lmms_eval`. Evaluation samples are distributed across GPUs using Accelerate, while each sample is processed on a single GPU with a batch size of 1. For LLaVA-1.5-7B, we retain $K \in \{192, 128, 64\}$ of the original 576 visual tokens. For AnyRes LLaVA-NeXT-7B, we retain $K \in \{640, 320, 160\}$ tokens from its resolution-dependent visual sequence. For Qwen2.5-VL-7B-Instruct, we retain 20% or 10% of the visual tokens generated for each input. Unless otherwise specified, ReCo uses all three relational descriptors (pre-projection, post-projection, and relational shift), followed by double-centering, RMS normalization, nonnegative kernel fusion, and greedy facility-location selection.

A.3 Datasets and Evaluation Metrics

Evaluation Benchmarks. We evaluate ReCo on seven adopted benchmarks covering diverse multimodal tasks:

- **GQA** focuses on compositional visual reasoning over real-world images, evaluating object attributes, spatial relationships, counting, and multi-step reasoning.
- **MMBench-EN Dev** is a multiple-choice benchmark evaluating object recognition, attribute identification, spatial perception, logical reasoning, and fine-grained visual understanding.
- **MME** evaluates LVLMs across perception and cognition suites. The perception suite covers object existence, counting, position, color, and text recognition, while the cognition suite examines commonsense reasoning, numerical calculation, translation, and code reasoning.
- **POPE** evaluates object hallucination through binary questions regarding object presence in an image, testing whether the model predicts nonexistent objects.
- **ScienceQA-IMG** contains multimodal multiple-choice questions from natural, social, and language sciences, assessing the model's ability to combine visual evidence with scientific knowledge.
- **VQAv2-TestDev** is a standard visual question answering benchmark covering objects, attributes, counts, actions, and scene contexts in natural images.
- **TextVQA-Val** evaluates reading and reasoning over scene text. As questions frequently depend on localized textual cues, it is sensitive to the retention of fine-grained visual features.

Metric Computation. We evaluate the seven benchmarks described above using their official task-specific protocols. GQA, MMBench-EN Dev, and ScienceQA-IMG are evaluated using accuracy; TextVQA-Val and VQAv2-TestDev use consensus-based VQA accuracy; POPE is evaluated using F1-score; and MME reports the combined perception-and-cognition score (P+C).

Accuracy-based metrics. GQA and ScienceQA-IMG use standard accuracy:

$$\text{Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}[\hat{y}_i = y_i], \tag{15}$$

where $\hat{y}_i$ and y_i denote the normalized prediction and ground-truth answer, respectively. For GQA, answers are converted to lowercase and punctuation is removed before comparison. For ScienceQA-IMG, the numerical ground-truth index is converted into the corresponding option letter, and a prediction is considered correct if it matches the target option.

MMBench-EN Dev is evaluated as a multiple-choice task. Free-form responses are first mapped to one of the candidate option labels. Because MMBench contains circular variants of each base question, a question is considered correct only if all of its variants are answered correctly:

$$\text{Accuracy}_{\text{MMBench}} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \prod_{v \in \mathcal{V}(q)} \mathbb{I}[\hat{c}_{q,v} = c_{q,v}], \tag{16}$$

where $\mathcal{Q}$ denotes the set of base questions and $\mathcal{V}(q)$ denotes the circular variants of question q.

VQA consensus accuracy. TextVQA-Val and VQAv2-TestDev are evaluated using the official consensus-based VQA metric. Predictions and reference answers are normalized using the EvalAI procedure, including case, punctuation, digit, and article normalization. Following the official evaluation protocol, machine accuracy is averaged over all leave-one-annotator-out subsets to remain consistent with the human agreement evaluation. Given a normalized prediction $\hat{a}_i$ and K human reference answers $\mathcal{A}_i = \{a_{i,1}, \ldots, a_{i,K}\}$, the score is

$$\text{Accuracy}_{\text{VQA}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{K} \sum_{k=1}^{K} \min\left(1, \frac{1}{3} \sum_{\ell \neq k} \mathbb{I}[a_{i,\ell} = \hat{a}_i]\right). \tag{17}$$

POPE F1-score. POPE evaluates object hallucination through binary "yes" or "no" questions. We report the F1-score computed from precision and recall:

$$\text{Precision} = \frac{TP}{TP + FP}, \qquad \text{Recall} = \frac{TP}{TP + FN}, \qquad \text{F1} = \frac{2\,\text{Precision}\,\text{Recall}}{\text{Precision} + \text{Recall}}. \tag{18}$$

When a denominator is zero, the corresponding metric is set to zero.

MME score. MME consists of paired yes/no questions organized into perception and cognition categories. Let $s_{c,j,1}, s_{c,j,2} \in \{0, 1\}$ denote the correctness of the two questions in pair j of category c. The combined MME score reported in our experiments is

$$\text{MME}_{\text{P+C}} = \sum_{c \in \mathcal{C}_{\text{P}} \cup \mathcal{C}_{\text{C}}} \frac{100}{|\mathcal{P}_c|} \sum_{j \in \mathcal{P}_c} \left[\frac{s_{c,j,1} + s_{c,j,2}}{2} + \mathbb{I}[s_{c,j,1} + s_{c,j,2} = 2]\right], \tag{19}$$

where $\mathcal{C}_{\text{P}}$ and $\mathcal{C}_{\text{C}}$ denote the perception and cognition categories, respectively, and $\mathcal{P}_c$ is the set of question pairs in category c. The MME score follows its original scale and is not normalized to $[0, 1]$.

Average performance retention. Because the seven benchmarks use different metric scales, we compute the average relative performance with respect to the corresponding unpruned model:

$$\text{Avg.} = \frac{100}{M} \sum_{m=1}^{M} \frac{s_m^{\text{pruned}}}{s_m^{\text{full}}}, \tag{20}$$

where s_m^{pruned} and s_m^{full} denote the pruned and unpruned scores on benchmark m, respectively. We use $M = 7$ when all benchmark results are available.

A.4 Additional Experimental Results

Qwen2.5-VL-3B-Instruct. To examine the effectiveness of ReCo across different model scales, we further evaluate it on Qwen2.5-VL-3B-Instruct. As shown in Table 6, ReCo achieves the highest average performance retention under both token budgets. It retains 94.99% of the full-model performance with 20% of the visual tokens and 89.05% with only 10% retained, outperforming the second-best method by 5.33 and 3.24 percentage points, respectively. At 20% retention, ReCo achieves the best performance on six of the seven benchmarks. Under the more aggressive 10% retention setting, it continues to achieve the best results on GQA, MMBench, ScienceQA, VQAv2, and TextVQA. These results demonstrate that ReCo remains effective on a smaller-scale LVLM under substantial visual-token reduction.

Table 6: Performance comparison on Qwen2.5-VL-3B-Instruct. Avg†. is the mean relative score over all seven metrics with respect to the model using 100% visual tokens. Bold and underlined values denote the best and second-best results, respectively.

Method	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	VQAv2 Acc. ↑	TextVQA Acc. ↑	Avg.† ↑
Retain 100% visual tokens								
Qwen2.5-VL-3B-Instruct	60.01	77.92	2143.08	86.68	81.36	81.98	77.52	100.00%
Retain 20% visual tokens ↓ 80.0%								
PruneSID (ICLR 2026)	<u>56.12</u>	70.62	1948.64	**85.26**	<u>80.02</u>	<u>74.41</u>	50.44	<u>89.66%</u>
FiCoCo-V (AAAI 2026)	53.72	<u>71.31</u>	<u>1960.02</u>	78.87	76.15	72.60	<u>55.12</u>	88.11%
ZOO-Prune (CVPR 2026)	54.33	67.78	1780.64	<u>84.68</u>	77.94	71.53	28.79	82.64%
ReCo (Ours)	**57.35**	**74.83**	**2065.12**	82.48	**81.46**	**77.13**	**67.89**	**94.99%**
Retain 10% visual tokens ↓ 90.0%								
PruneSID (ICLR 2026)	<u>53.16</u>	65.03	**1906.21**	<u>79.15</u>	<u>78.43</u>	<u>70.71</u>	<u>50.96</u>	<u>85.81%</u>
FiCoCo-V (AAAI 2026)	50.53	<u>65.55</u>	1793.68	72.60	74.17	67.70	45.31	81.14%
ZOO-Prune (CVPR 2026)	50.11	59.79	1592.48	**82.60**	75.11	65.35	20.70	75.51%
ReCo (Ours)	**54.36**	**71.05**	<u>1866.63</u>	77.79	**79.92**	**73.08**	**59.98**	**89.05%**

Table 7: Performance comparison on Qwen3-VL-8B-Instruct. Avg†. is the mean relative score over all seven metrics with respect to the model using 100% visual tokens. Bold and underlined values denote the best and second-best results, respectively.

Method	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	VQAv2 Acc. ↑	TextVQA Acc. ↑	Avg.† ↑
Retain 100% visual tokens								
Qwen3-VL-8B-Instruct	61.56	85.40	2361.53	88.28	94.45	83.33	80.05	100.00%
Retain 20% visual tokens ↓ 80.0%								
PruneSID (ICLR 2026)	**60.04**	77.92	<u>2087.30</u>	<u>86.20</u>	**87.16**	**78.26**	<u>63.72</u>	<u>91.51%</u>
FiCoCo-V (AAAI 2026)	53.88	75.43	1946.11	78.36	81.80	72.13	40.14	81.48%
ZOO-Prune (CVPR 2026)	<u>59.59</u>	<u>79.47</u>	2068.25	**86.33**	<u>86.81</u>	<u>78.20</u>	57.27	90.36%
ReCo (Ours)	57.79	**79.73**	**2143.12**	85.07	86.22	77.55	**71.82**	**92.63%**
Retain 10% visual tokens ↓ 90.0%								
PruneSID (ICLR 2026)	<u>54.71</u>	<u>72.16</u>	<u>1966.26</u>	<u>80.68</u>	<u>83.79</u>	**73.42**	<u>50.67</u>	<u>84.02%</u>
FiCoCo-V (AAAI 2026)	49.73	67.35	1781.59	71.47	79.82	66.60	28.83	73.79%
ZOO-Prune (CVPR 2026)	**55.57**	71.65	1918.01	**83.19**	**83.99**	<u>72.64</u>	39.36	82.13%
ReCo (Ours)	53.95	**75.00**	**1990.51**	<u>80.68</u>	83.04	72.31	**64.38**	**86.61%**

Qwen3-VL-8B-Instruct. To further evaluate the generalizability of ReCo, we conduct additional experiments on Qwen3-VL-8B-Instruct under dynamic-resolution inputs. As shown in Table 7, ReCo achieves the highest average performance retention under both token budgets. It retains 92.63% of the full-model performance with 20% of the visual tokens and 86.61% with only 10% retained, outperforming the second-best method by 1.12 and 2.59 percentage points, respectively. ReCo also achieves the best results on MMBench, MME, and TextVQA under both retention settings. These results demonstrate that the proposed relational-transition-based coreset selection generalizes effectively to the Qwen3-VL architecture.

Selection-objective ablation. Table 8 isolates the effect of the subset-selection objective by keeping the Pre/Post/Shift representation, kernel alignment, fusion, and token budgets unchanged. ReCo with facility-location selection obtains the highest score in all 18 dataset–budget comparisons. Relative to ReCo, row-sum Top-K reduces the equal-dataset macro relative score by 21.68%, 37.34%, and 40.64% at $K = 192$, 128, and 64, respectively. The reductions are particularly large on GQA,

Table 8: Selection-objective ablation of ReCo on LLaVA-1.5-7B. All variants use the same fused Pre/Post/Shift kernel, centered-RMS alignment, ReLU nonnegativity, and normalized relational-shift construction. Bold and underlined values denote the best and second-best results within each token budget, respectively.

Selector	GQA Acc. ↑	MMB Acc. ↑	MME P+C ↑	POPE F1 ↑	SQA Acc. ↑	TextVQA Acc. ↑
Retain 192 tokens ↓ 66.7%						
ReCo (Ours)	**60.17**	**63.32**	**1783.85**	**87.69**	**69.46**	**56.82**
Row-sum Top-K	49.67	51.37	1374.33	48.72	66.48	44.28
Kernel k-center	<u>58.05</u>	<u>55.67</u>	<u>1610.18</u>	<u>85.90</u>	<u>67.82</u>	<u>50.04</u>
Retain 128 tokens ↓ 77.8%						
ReCo (Ours)	**59.79**	**62.97**	**1762.00**	**87.44**	**68.77**	**56.53**
Row-sum Top-K	42.26	33.16	1108.22	16.76	65.20	42.82
Kernel k-center	<u>57.41</u>	<u>53.61</u>	<u>1557.20</u>	<u>85.13</u>	<u>67.18</u>	<u>49.04</u>
Retain 64 tokens ↓ 88.9%						
ReCo (Ours)	**58.69**	**61.58**	**1644.12**	**85.92**	**68.67**	**55.64**
Row-sum Top-K	39.35	27.92	970.82	11.93	64.70	42.62
Kernel k-center	<u>55.57</u>	<u>50.34</u>	<u>1483.50</u>	<u>82.66</u>	<u>66.53</u>	<u>47.68</u>

MMBench, MME, POPE, and TextVQA, while the decrease on ScienceQA is more moderate. The widening gap under smaller budgets indicates that independently selecting tokens with high fused-kernel row sums does not provide an adequate substitute for iterative marginal coverage when only a small subset can be retained.

Kernel k-center is consistently stronger than row-sum Top-K, but remains below facility location on every evaluated dataset and budget. Its equal-dataset macro relative gaps are 6.95%, 8.11%, and 9.09% at $K = 192$, 128, and 64, respectively. The differences are relatively small on POPE and ScienceQA, but are larger on MMBench, MME, and TextVQA. Thus, promoting farthest-point diversity alone preserves more task performance than independent row-sum ranking, but does not recover the average relational coverage optimized by facility location. Because all three variants use the same fused kernel, these results provide controlled evidence that the facility-location objective contributes beyond the relational representation itself.

A.5 Inference-Time Evaluation

We evaluate the inference-time cost of FiCoCo-V, PruneSID, ZOO-Prune, and ReCo using LLaVA-1.5-7B on GQA, MME, POPE, ScienceQA-IMG, TextVQA, and MMBench-EN-Dev. We consider retained visual-token budgets $K \in \{64, 128, 192\}$. All measurements use batch size 1 and NVIDIA RTX A6000 GPUs. Each method-budget configuration is evaluated on the complete evaluation splits used in our experiments, yielding 35,346 samples per configuration.

We measure GPU time with CUDA events. Image loading, PIL decoding, and CPU-side image preprocessing are performed before the timed region and are excluded. For method m, budget K, and dataset d with N_d samples, the dataset-level mean is

$$\bar{T}_{m,K,d} = \frac{1}{N_d} \sum_{i=1}^{N_d} T_{m,K,d,i}. \tag{21}$$

The aggregate figures use the equal-weight six-dataset macro average

$$\bar{T}^{\text{macro}}_{m,K} = \frac{1}{6} \sum_{d=1}^{6} \bar{T}_{m,K,d}, \tag{22}$$

which prevents the larger GQA and POPE evaluation sets from dominating the reported aggregate.

Timing metrics. **Pruning-stage latency** records the method-specific visual-token reduction stage. For ReCo and ZOO-Prune, it includes token scoring or kernel construction, token selection, optional

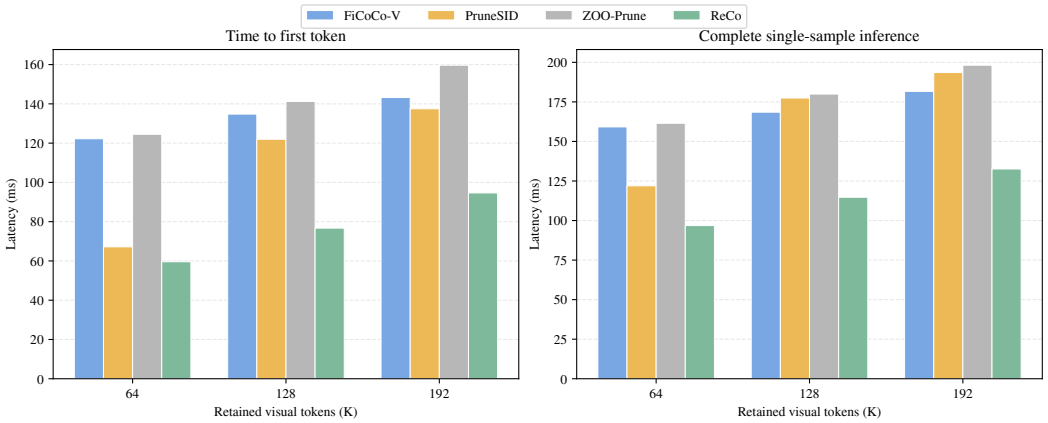


Figure 4: Macro-average inference latency of LLaVA-1.5-7B over six datasets at three retained visual-token budgets. TTFT is measured with a separate one-token generation pass, while complete inference is measured through EOS or the dataset-specific generation limit. Lower values indicate lower latency. Each dataset receives equal weight in the macro average.

feature transport, and sequence compaction after visual encoding. For PruneSID, it covers the official PruneSID CLIP selector forward. For FiCoCo-V, it covers the integrated CLIP encoding and merging path. **LLM prefill latency** is the GPU time of the first language-model forward that consumes the complete reduced multimodal prompt and constructs the initial key-value cache. **Time to first token (TTFT)** is measured from entry into `model.generate()` until the first output token is completed. It includes visual encoding, token reduction, multimodal input construction, LLM prefill, and first-token generation. TTFT uses a generation pass restricted to one new token. **Complete single-sample inference latency** is measured using a separate generation pass from entry into `model.generate()` until EOS or the dataset-specific `max_new_tokens`.

Overall comparison. Figure 4 reports the macro-average TTFT and complete inference latency. ReCo has the lowest mean latency at all three budgets. Its TTFT is 59.58, 76.71, and 94.64 ms for $K = 64$, 128, and 192, respectively. Relative to the fastest non-ReCo method at each budget, these values represent reductions of 11.3%, 37.1%, and 31.2%. ReCo's complete-inference latency is 96.80, 114.71, and 132.58 ms, corresponding to reductions of 20.6%, 31.9%, and 27.0% relative to the fastest non-ReCo method. The fastest non-ReCo method is selected separately for each metric and budget and is therefore not fixed.

Timing components. Figure 5 separates pruning-stage latency from LLM prefill latency. The pruning-stage measurement includes the computation required by each method to produce its retained visual-token sequence. ReCo has the lowest measured pruning-stage latency at all three budgets, requiring 9.64, 16.69, and 23.56 ms for $K = 64$, 128, and 192, respectively. For comparison, ZOO-Prune requires 73.16, 80.37, and 88.01 ms, corresponding to reductions of 86.8%, 79.2%, and 73.2% for ReCo. ReCo and ZOO-Prune also have similar LLM prefill latency at the same budget: their macro-average differences are approximately 1.16, 1.02, and 1.03 ms. These results indicate that ReCo's lower TTFT is primarily associated with a lower-cost visual-token selection procedure, while the subsequent language-model prefill cost remains comparable at the same retained-token budget.of visual encoding.

Per-dataset reduction. For each dataset d and budget K, Figure 6 reports the TTFT reduction relative to the fastest non-ReCo method,

$$R_{\text{fastest}}(d, K) = 100 \left(1 - \frac{\bar{T}_{\text{ReCo},K,d}}{\min_{m \neq \text{ReCo}} \bar{T}_{m,K,d}}\right) \%, \tag{23}$$

where the minimum is taken over FiCoCo-V, PruneSID, and ZOO-Prune. The selected comparison method may vary across datasets and budgets and is determined only by mean TTFT. We also report

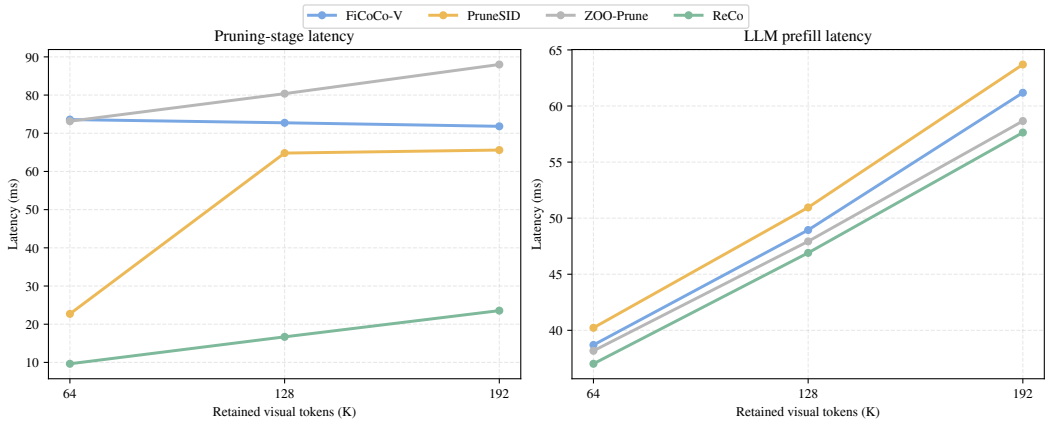


Figure 5: Timing-component analysis at three retained visual-token budgets. The left panel reports the method-specific latency required to determine and construct the retained visual-token sequence, including visual-feature computation when required by the selection procedure. The right panel reports the first LLM prefill forward after token reduction. Lower values indicate lower latency.

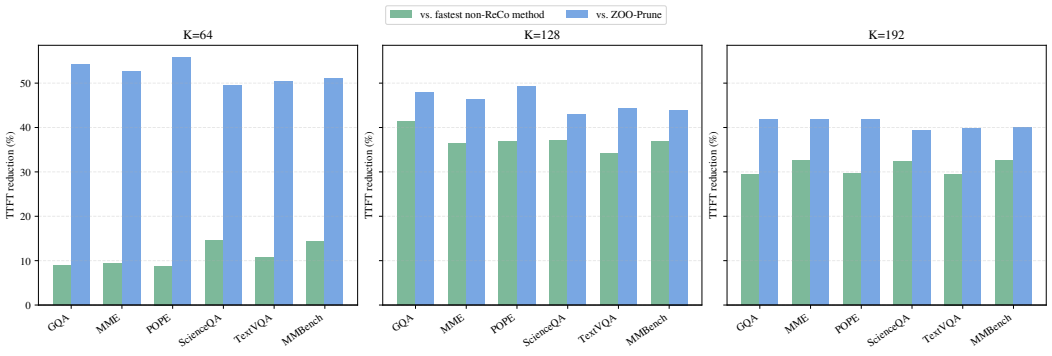


Figure 6: Per-dataset reduction in mean TTFT achieved by ReCo. "Fastest non-ReCo method" denotes the lowest-latency method among FiCoCo-V, PruneSID, and ZOO-Prune for each dataset and token budget; its identity is not fixed. "ZOO-Prune" denotes a fixed comparison under the shared LLaVA inference path. Positive values indicate lower mean TTFT for ReCo.

the reduction relative to the fixed ZOO-Prune baseline,

$$R_{\text{ZOO}}(d, K) = 100 \left(1 - \frac{\bar{T}_{\text{ReCo},K,d}}{\bar{T}_{\text{ZOO-Prune},K,d}} \right) \%. \tag{24}$$

ReCo reduces mean TTFT relative to the fastest non-ReCo method by 8.8–14.6% at $K = 64$, 34.1–41.4% at $K = 128$, and 29.4–32.6% at $K = 192$. The reduction is positive for every evaluated dataset and budget.

A.6 Additional Qualitative Results

Figure 7 presents additional qualitative comparisons among the individual relational components and the complete ReCo framework under different token budgets.

As shown in Figure 7, the single-component variants exhibit considerable sensitivity to both the input content and the token budget. No individual relational view consistently preserves the evidence required for color recognition, attribute identification, object-state reasoning, visual recognition, and spatial reasoning. For example, in the person-attribute example, all three single-component variants

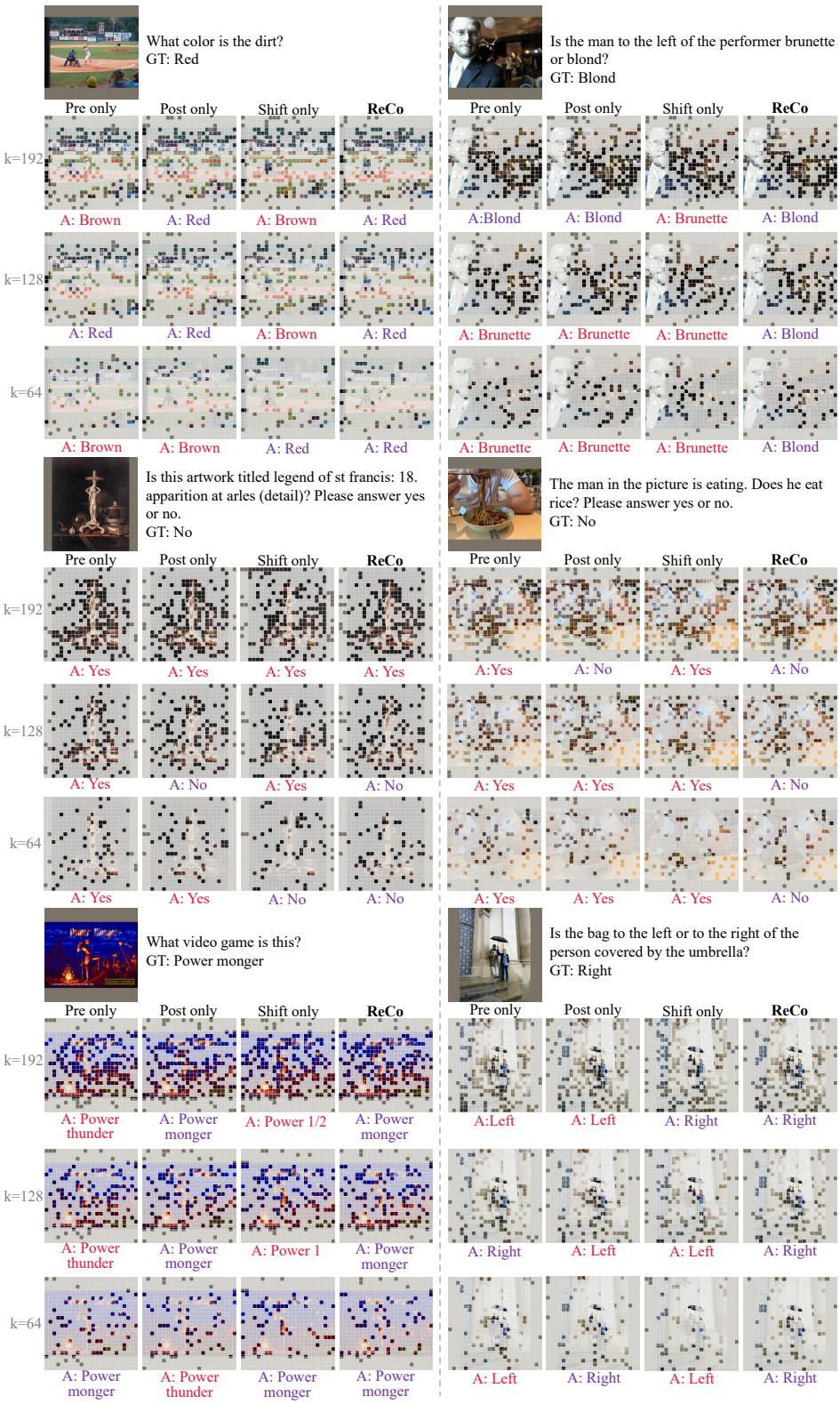


Figure 7: Additional qualitative comparisons of ReCo and its component variants under different visual-token budgets. The columns show the retained-token patterns and predictions obtained using Pre only, Post only, Shift only, and the complete ReCo framework, while the rows correspond to $K \in \{192, 128, 64\}$. Purple and red denote correct and incorrect predictions, respectively.

fail at the lowest budget, whereas ReCo consistently predicts "Blond" across all three budgets. Similarly, in the eating and spatial-relation examples, ReCo preserves the evidence required to produce the correct answers under every budget, while the individual components yield unstable predictions. By jointly integrating pre-projection relations, post-projection relations, and projection-induced relational shifts, ReCo produces the correct answer in 17 of the 18 displayed example-budget combinations. It also remains correct across all three budgets in five of the six examples, indicating that the three relational views provide complementary evidence under substantial token reduction. In the artwork-recognition example, ReCo fails at $K = 192$ but succeeds at $K = 128$ and $K = 64$, showing that prediction behavior is not necessarily monotonic at the individual-example level. Overall, these results complement the quantitative component ablation and illustrate the benefit of jointly modeling the three relational views.

Additional Comparisons with Existing Methods. Figure 8 presents additional qualitative comparisons between ReCo and existing visual-token pruning methods under three retention budgets.

As shown in Figure 8, ReCo produces correct answers in 16 of the 18 displayed example-budget combinations. It is correct on all six examples at $K = 192$ and on five of the six examples at both $K = 128$ and $K = 64$. In the clothing example, ReCo predicts "Tank top" at all three budgets, whereas the other methods produce "Shirt" or "Outfit" in several settings. In the helmet example, ReCo also predicts "Policeman" consistently across the three budgets. For the vegetable example, ReCo identifies "Beans" at all budgets, while the baseline methods predict "Peas" under the more restrictive budgets. ReCo also maintains the correct answer "No" for the question concerning the woman's emotional state at all three budgets. The figure also illustrates the limitations of aggressive token reduction. ReCo gives an incorrect answer for the calf example at $K = 128$ and for the belt example at $K = 64$. Nevertheless, in these examples ReCo preserves the relevant object, attribute, and spatial information in more budget settings than the compared methods. Overall, the qualitative results are consistent with the quantitative experiments and provide additional evidence that combining pre-projection relations, post-projection relations, and relational shifts is useful for visual-token selection.

A.7 Analysis of Projector-Induced Relational Transitions

We further analyze how the multimodal projector changes the pairwise relations among visual tokens. The statistics are computed using 500 unique images per dataset after image-level deduplication, except for GQA, which contains 398 eligible images. The analysis uses the same pre-projector and post-projector representations as ReCo and does not rely on answer labels or task-specific correctness information.

Rank and Neighborhood Migration. Figure 9 characterizes relational changes at the global-ranking and local-neighborhood levels. The Spearman correlations between pre-projector and post-projector relation ranks range from 0.642 to 0.723, with an equal-dataset macro average of 0.668. The concentration of probability mass around the diagonal indicates that the projector preserves part of the original relational ordering. However, the off-diagonal mass shows that this ordering is not fully preserved. For each token, the top-k neighborhood overlap is defined as the fraction of neighbors shared by its pre-projector and post-projector top-k sets. The mean overlap ranges from 0.717 to 0.739 for $k = 4$, from 0.733 to 0.751 for $k = 8$, and from 0.741 to 0.757 for $k = 16$. Thus, approximately one quarter of the local neighbors change after projection on average. Although the overlap increases with k, as expected for larger neighborhoods, it remains below one across all datasets and neighborhood sizes. The similar distributions across datasets suggest that this relational reorganization is associated with the shared vision-to-language projection rather than being confined to a particular benchmark.

Organization and Concentration of Relational Changes. Figure 10 examines the distribution of relational changes across tokens. In the magnitude-centrality plane, the all-token distributions are concentrated around the per-image mean change magnitude and moderate-to-high transition centrality. This general pattern appears across all datasets, although the precise density distributions vary. The mean unnormalized relation-change norm ranges from 2.804 on GQA to 3.039 on TextVQA, with a pooled mean of 2.872. The 10% of tokens with the largest change norms account for 21.70%–22.95% of the total squared change energy, with a pooled mean of 22.30%. This

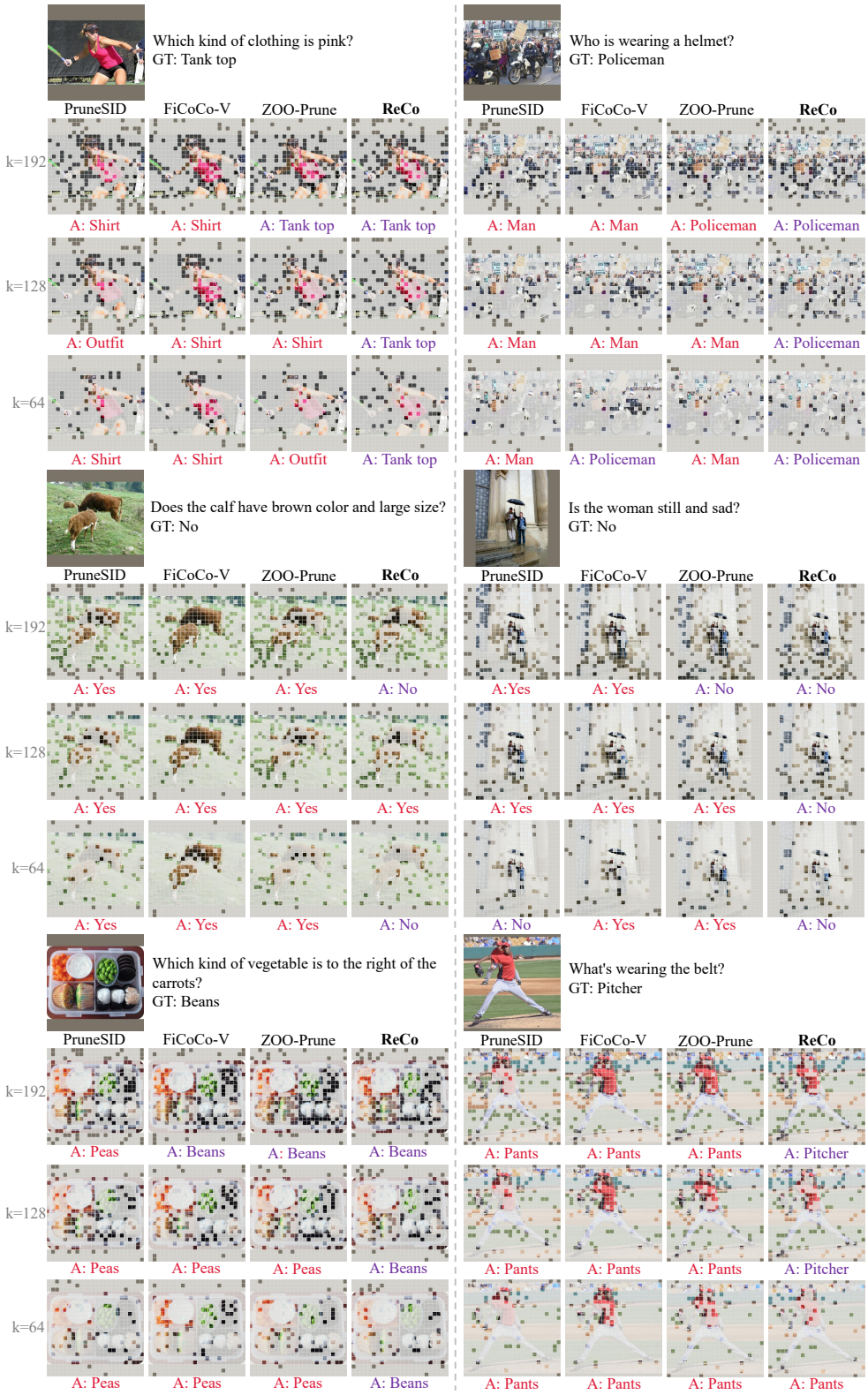


Figure 8: Additional qualitative comparisons between ReCo and existing visual-token pruning methods. The columns correspond to PruneSID, FiCoCo-V, ZOO-Prune, and ReCo, while the rows correspond to $K \in \{192, 128, 64\}$. Purple and red answer labels indicate correct and incorrect predictions, respectively.

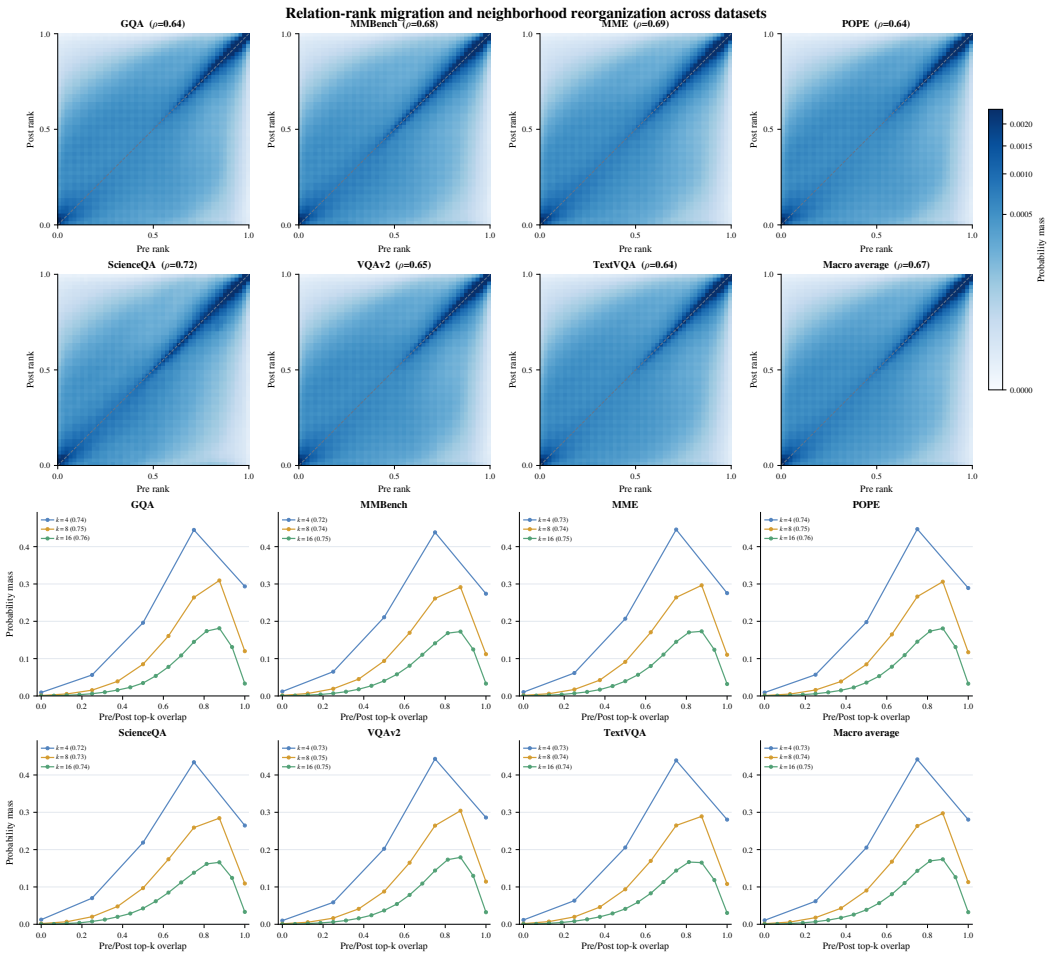


Figure 9: Relation-rank migration and neighborhood reorganization across seven datasets. The upper panels show the joint distributions of normalized pairwise relation ranks before and after projection. The dashed diagonal represents unchanged ranks, and ρ denotes the Spearman rank correlation. The lower panels show the distributions of token-wise top-k neighborhood overlap for $k \in \{4, 8, 16\}$, with the mean overlap shown in parentheses. The macro panels assign equal weight to each dataset.

share is higher than the 10% expected under a uniform allocation of per-token energy, indicating that relational changes are unevenly distributed across tokens. However, the top 10% of tokens do not account for the majority of the energy, suggesting moderate rather than extreme concentration. The selected-token contours substantially overlap with the dense regions of the all-token distributions. The selected-minus-all offsets are slightly negative: Δm ranges from -0.096 to -0.053, while Δc ranges from -0.024 to -0.005. Therefore, ReCo does not simply select tokens with the largest change magnitude or centrality. This observation is consistent with its facility-location objective, which selects tokens according to their joint coverage of the pre-projection, post-projection, and relational-shift kernels. Taken together, Figures 9 and 10 show that the multimodal projector modifies both global relation rankings and local token neighborhoods, and that these changes are non-uniformly distributed across tokens.

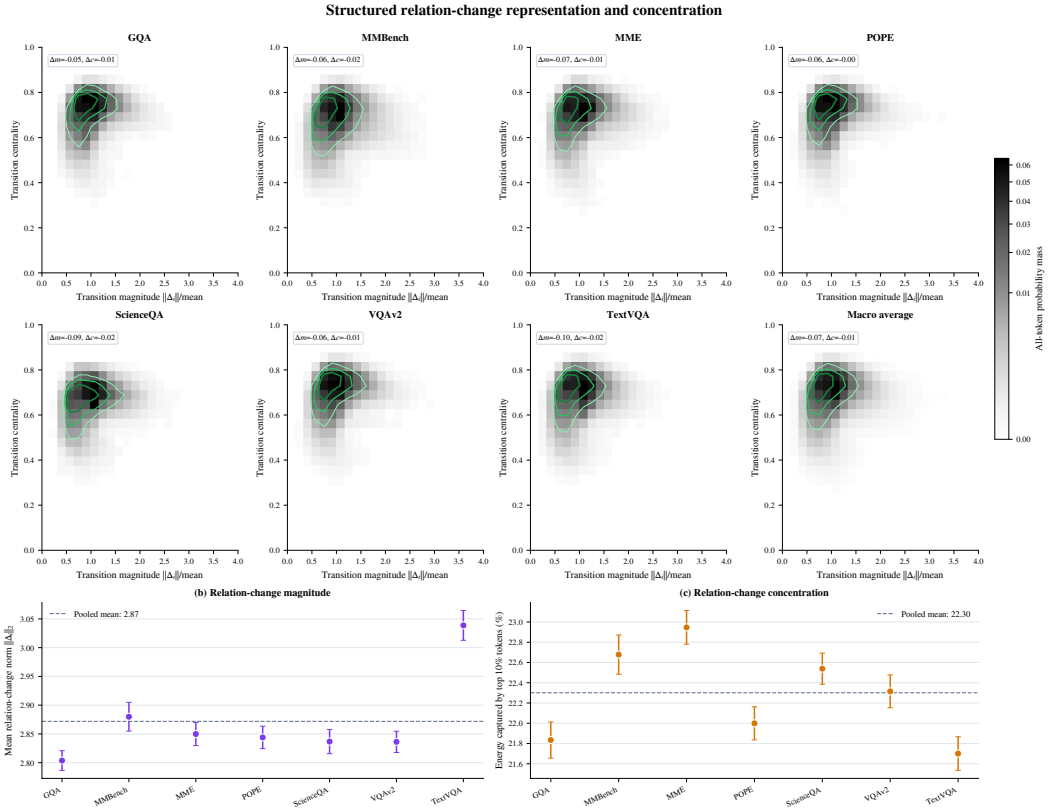


Figure 10: Distribution and concentration of projector-induced relational changes. In the upper panels, the gray density represents all visual tokens in terms of normalized relation-change magnitude $\|\Delta_i\|_2 / \operatorname{mean}_j \|\Delta_j\|_2$ and transition centrality. Green contours represent the density of tokens selected by ReCo at $K = 128$. The annotations give the selected-minus-all differences in mean normalized magnitude (Δm) and mean centrality (Δc). The lower panels report the per-image mean relation-change norm and the percentage of total squared relation-change energy captured by the 10% of tokens with the largest change norms. Points denote dataset means, error bars denote 95% confidence intervals over images, and dashed lines denote means pooled over all evaluated images.