FROM STATIC COMPRESSION TO QUERY-ADAPTIVE REPRESENTATION FOR EFFICIENT VISUAL DOCUMENT RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual document retrieval aims to efficiently identify fine-grained query-relevant evidence from long and visually rich documents. Multi-vector retrieval models are effective at localizing such evidence through fine-grained late interaction, but storing and matching hundreds of vectors per page incurs substantial memory and computation costs. Existing compression methods reduce this overhead through pruning or merging, yet the resulting page representations are typically fixed during offline indexing and reused for all future queries. This creates a fundamental mismatch between query-agnostic compression and query-dependent retrieval, since different queries may rely on different semantic aspects of the same page. We propose **Query-Conditioned Reconstruction (QCR)**, a training-free framework that preserves compact reconstructable variation during indexing and instantiates query-specific page representations at retrieval time. QCR augments static compressed representatives with low-rank residual memory, selectively activates groups with high query-relevant reconstruction potential, and reconstructs query-conditioned vectors without accessing the original full multi-vector representation. A budgeted MaxSim-coverage objective further selects a compact set of complementary vectors for late interaction. Experiments across multiple retrieval encoders and benchmarks show that QCR better preserves Full-Index retrieval quality under aggressive compression, while improving evidence retrieval and downstream long-document question answering.

1 INTRODUCTION

Visual document retrieval aims to identify query-relevant pages from visually rich document collections, such as scientific articles, financial reports, presentation slides, and enterprise records (Ding et al., 2026; Gao et al., 2026). Relevant evidence may appear in text, tables, figures, charts, or spatial layouts, and often occupies only a small portion of a page or document (Kim et al., 2026). Effective retrieval therefore requires preserving fine-grained visual and semantic cues while efficiently searching over large collections of document pages. Recent vision-language retrieval systems increasingly rely on fine-grained multi-vector representations for visual document retrieval. ColPali (Faysse et al., 2025) establishes a late-interaction framework that represents each page with multiple localized visual embeddings and matches them against query tokens, while recent multimodal retrieval models such as Jina Embeddings and Nemotron ColEmbed (Akram et al., 2026; Moreira et al., 2026) further demonstrate the effectiveness of multi-vector representations for fine-grained document retrieval. While this design provides strong fine-grained matching, it incurs substantial storage and computation costs because each page may contain hundreds or even thousands of vectors. As document collections (Macé et al., 2025) and document lengths grow (Zhu et al., 2026), this overhead becomes a major scalability bottleneck, motivating recent work on multi-vector compression through pruning (Liu et al., 2026; Zhang et al., 2025), merging (Ma et al., 2025), learned latent representations (Luo et al., 2024; Zhang et al., 2026; Yan et al., 2026a), and structure-aware aggregation (Zuo et al., 2026; Yan et al., 2026b).

Despite their different designs, most existing compression methods share a fundamental limitation: the compact page representation is determined during offline indexing, before future queries are observed, and is then reused for all subsequent query interactions. This creates an inher-

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.

ent mismatch between query-agnostic compression and query-dependent retrieval, since different queries may rely on different fine-grained regions or semantic aspects of the same page. Pruning may remove query-specific evidence, while merging can collapse multiple semantic directions into a small set of fixed representatives. Although late interaction is query-aware at the *MaxSim scoring* stage, it can only operate on the information retained in the compressed representation. As compression becomes more aggressive, this mismatch can reduce the score margin between relevant and near-miss pages, making fine-grained ranking around the Top-K cutoff increasingly difficult. This motivates us to preserve compact reconstructable variation during indexing and defer the final instantiation of the page representation until the query is observed.

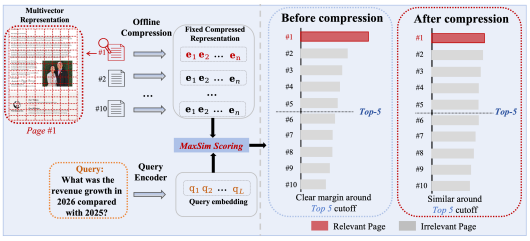


Figure 1: Static compression weakens query-dependent discriminability under aggressive compression.

We address this limitation with **Query-Conditioned Reconstruction (QCR)**, a training-free framework that separates the representation stored at indexing time from the representation used for retrieval. Rather than treating offline compression as the final representation step, QCR augments each compressed page with a lightweight low-rank residual memory that summarizes within-group variation suppressed by static compression. When a query arrives, QCR selectively activates query-relevant residual directions and reconstructs query-specific page vectors without accessing the original full multi-vector representation. The reconstruction is constrained by page-derived residual statistics, so adaptation is restricted to directions supported by the indexed page. In this way, QCR turns a fixed compressed index into a query-adaptive representation while keeping the stored memory compact. To control query-time cost, QCR reconstructs only a small number of relevant groups for each query token. A residual-aware activation score identifies groups whose static centers or preserved residual variation are likely to support the query, after which closed-form reconstruction generates query-conditioned vectors from the selected groups. Because independently retaining the strongest candidates may produce redundant matches to the same query semantics, we further introduce a budgeted MaxSim-coverage objective that selects at most B complementary vectors to form the final page representation. QCR is applied independently to the compact memory of every page, and the resulting query-specific representations are ranked directly using standard *MaxSim scoring*. The framework introduces no learnable parameters and can be applied on top of different grouping-based compression strategies. Experiments across multiple retrieval encoders and visual document benchmarks show that QCR better preserves Full-Index retrieval quality under aggressive compression while maintaining a compact index. Our contributions are summarized as follows:

- We propose **Query-Conditioned Reconstruction (QCR)**, a training-free framework that preserves compact residual variation during indexing and reconstructs query-specific representations at retrieval time.
- We develop residual-aware group activation and budgeted MaxSim-coverage to selectively reconstruct relevant groups and retain complementary query-specific vectors under a compact representation budget.
- We evaluate QCR across multiple retrieval encoders and benchmarks, showing consistent gains in compressed page retrieval and evidence recovery, which further translate to improved downstream long-document QA.

2 Query-Conditioned Reconstruction for Adaptive Compressed Page Retrieval

Given a query and a document corpus, our goal is to identify query-relevant pages using compact multi-vector representations. QCR separates indexing-time storage from retrieval-time representation construction. During offline indexing, each page is compressed into a compact residual memory that stores static representatives together with a low-rank summary of within-group variation. At retrieval

time, the query selectively activates this memory to instantiate a query-specific page representation for scoring.

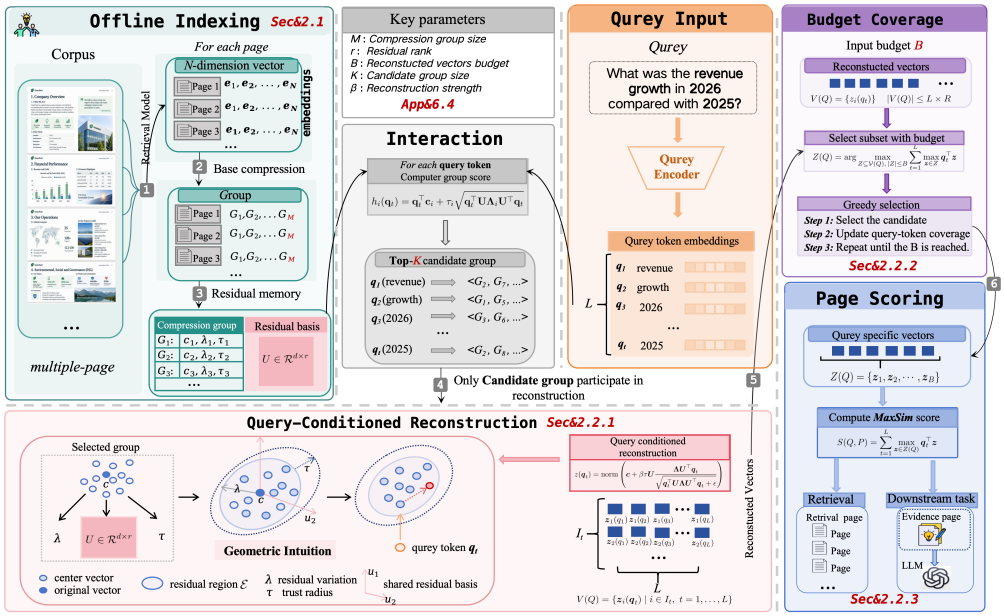


Figure 2: Overview of Query-Conditioned Reconstruction (QCR). Each page is first encoded into a compact residual memory during offline indexing. At retrieval time, query-relevant groups are selectively activated and reconstructed, followed by budgeted MaxSim coverage to obtain the final query-specific representation for page scoring.

As illustrated in Fig. 2, QCR contains three main components. First, *Compact Residual Memory* augments static compressed representatives with a low-rank summary of within-group variation that would otherwise be suppressed by static compression. Second, *Query-Conditioned Reconstruction* identifies groups with high query-relevant reconstruction potential and selectively reconstructs query-specific vectors from the stored residual memory. Finally, *Budgeted MaxSim Coverage* selects a compact set of complementary vectors that jointly cover different query-token semantics. Reconstruction is restricted to the Top-K groups for each query token, while the final representation is limited to $B = M$ vectors, matching the interaction budget of the corresponding static representation.

2.1 Compact Residual Memory

Consider a document corpus $\mathcal{D} = \{P_1, \ldots, P_J\}$, where each page P_j is represented by a set of visual embeddings $E_j = \{\boldsymbol{e}_{j,n}\}_{n=1}^{N_j}$ with $\boldsymbol{e}_{j,n} \in \mathbb{R}^d$. Each page is compressed independently. For clarity, we omit the page index and consider a single page $E = \{\boldsymbol{e}_n\}_{n=1}^{N}$. A grouping-based base compressor partitions the page embeddings into $M \ll N$ groups $\{G_i\}_{i=1}^{M}$, while QCR is agnostic to the specific grouping strategy. For each group, we compute its mean center and normalized representative:

$$\boldsymbol{c}_i = \frac{1}{|G_i|} \sum_{n \in G_i} \boldsymbol{e}_n, \qquad \boldsymbol{\mu}_i = \frac{\boldsymbol{c}_i}{\|\boldsymbol{c}_i\|_2}. \tag{1}$$

Rather than retaining only the fixed representative, QCR additionally preserves a compact summary of the within-group variation that would otherwise be suppressed by static compression. For each $\boldsymbol{e}_n \in G_i$, we define the residual $\boldsymbol{r}_n = \boldsymbol{e}_n - \boldsymbol{c}_i$. Collecting all page residuals into $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N]$, we retain the leading r left singular vectors as a shared low-rank basis $\boldsymbol{U} \in \mathbb{R}^{d \times r}$, and project each residual into this subspace as $\boldsymbol{a}_n = \boldsymbol{U}^\top \boldsymbol{r}_n$. For each group, the residual variation is summarized by diagonal second-order statistics:

$$\boldsymbol{\lambda}_i = \frac{1}{|G_i|} \sum_{n \in G_i} (\boldsymbol{a}_n \odot \boldsymbol{a}_n), \qquad \boldsymbol{\Lambda}_i = \mathrm{diag}(\boldsymbol{\lambda}_i). \tag{2}$$

Together with a group-specific trust radius τ_i, these statistics define a local feasible region

$$\mathcal{E}_i = \left\{ \boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a} : \|\boldsymbol{a}\|_2 \le \tau_i \right\}. \tag{3}$$

The trust radius τ is determined from a fixed quantile of the empirical whitened residual norms within each group; the detailed estimation procedure is provided in Appendix A.1. The resulting compact memory is $\mathcal{M} = (\boldsymbol{U}, \{\boldsymbol{c}_i, \boldsymbol{\lambda}_i, \tau_i\}_{i=1}^{M})$, which stores each group center together with a compact summary of its within-group variation. The feasible region $\mathcal{E}$ is defined before normalization, and reconstructed vectors are normalized for late-interaction scoring, allowing each group to be selectively adapted to the query.

2.2 Query-Conditioned Reconstruction with Budgeted Coverage

2.2.1 Query-Conditioned Reconstruction

Let $Q = \{\boldsymbol{q}_t\}_{t=1}^{L}$ denote the normalized query-token embeddings. Reconstructing every compressed group for every query token would introduce unnecessary computation. We therefore first estimate which groups have high query-relevant reconstruction potential and restrict explicit reconstruction to only a small subset. Because the final retrieval score is computed using normalized vectors, we adopt a cosine-aware activation criterion. Let $\boldsymbol{P}_i = \boldsymbol{I} - \boldsymbol{\mu}_i\boldsymbol{\mu}_i^\top$ denote the projection onto the tangent space orthogonal to the normalized group center $\boldsymbol{\mu}_i$. For query token $\boldsymbol{q}_t$, we define the activation score

$$\bar{h}_i(\boldsymbol{q}_t) = \boldsymbol{q}_t^\top \boldsymbol{\mu}_i + \frac{\tau_i}{\|\boldsymbol{c}_i\|_2}\sqrt{\boldsymbol{q}_t^\top \boldsymbol{P}_i \boldsymbol{U}\boldsymbol{\Lambda}_i\boldsymbol{U}^\top \boldsymbol{P}_i \boldsymbol{q}_t}. \tag{4}$$

The first term measures the similarity between the query token and the static group representative, while the second estimates the additional query alignment supported by the preserved residual variation around that center. For each query token, we retain only the Top-K groups according to $\bar{h}_i(\boldsymbol{q}_t)$. Unlike selection based solely on static similarity, this criterion can activate a group whose center is not among the closest matches but whose residual memory contains query-relevant variation. We emphasize that $\bar{h}_i$ is an efficient first-order cosine-aware surrogate rather than the exact maximum cosine similarity attainable within the normalized feasible region. For each activated group, we next construct a feasible reconstruction direction aligned with the query. Specifically, we define

$$\boldsymbol{v}_i(\boldsymbol{q}_t) = \tau_i \boldsymbol{U}\frac{\boldsymbol{\Lambda}_i\boldsymbol{U}^\top \boldsymbol{P}_i \boldsymbol{q}_t}{\sqrt{\boldsymbol{q}_t^\top \boldsymbol{P}_i \boldsymbol{U}\boldsymbol{\Lambda}_i\boldsymbol{U}^\top \boldsymbol{P}_i \boldsymbol{q}_t + \epsilon}}, \qquad \boldsymbol{z}_i(\beta, \boldsymbol{q}_t) = \operatorname{norm}\left(\boldsymbol{c}_i + \beta \boldsymbol{v}_i(\boldsymbol{q}_t)\right), \tag{5}$$

where $\beta \in [0, 1]$ controls how far the reconstruction moves along the feasible residual direction. Rather than treating β as a global hyperparameter, we select it analytically for each query–group pair by maximizing the cosine alignment $\boldsymbol{q}_t^\top \boldsymbol{z}_i(\beta, \boldsymbol{q}_t)$ over $\beta \in [0, 1]$. This one-dimensional objective admits a closed-form stationary point, and we choose the best feasible value among the stationary point and the two interval boundaries. The detailed solution is provided in Appendix A.1. QCR does not recover specific original vectors; instead, it instantiates query-specific vectors from the residual variation preserved around each group center. The query selects useful directions, while page-derived residual statistics constrain the feasible adaptation, without accessing the original full multi-vector representation.

2.2.2 Budgeted MaxSim Coverage

Applying reconstruction to the activated groups produces a set of query-conditioned candidates for each page. Let $\mathcal{I}_t$ denote the Top-K groups activated for query token $\boldsymbol{q}_t$. Together with the static representatives, the within-page candidate pool is:

$$\mathcal{V}(Q) = \{\boldsymbol{\mu}_i\}_{i=1}^{M} \cup \{\boldsymbol{z}_i(\boldsymbol{q}_t) \mid i \in \mathcal{I}_t,\, t = 1, \ldots, L\}. \tag{6}$$

The candidate pool contains at most $M + LK$ vectors. Retaining all reconstructed candidates would increase the subsequent late-interaction cost, while independently selecting the highest-scoring candidates may allocate multiple representation slots to the same query semantics and leave other query tokens insufficiently covered. We therefore select a compact set of complementary vectors by directly maximizing MaxSim coverage:

$$Z(Q) = \arg \max_{Z \subseteq \mathcal{V}(Q),\, |Z| \le M} \sum_{t=1}^{L} \max_{\boldsymbol{z} \in Z} \boldsymbol{q}_t^\top \boldsymbol{z}. \tag{7}$$

We set the final representation budget to $B = M$, matching the number of vectors in the corresponding static compressed representation. The objective is directly aligned with the final MaxSim scoring function and favors candidates that provide complementary coverage across query tokens. We optimize this objective greedily; the detailed selection procedure and complexity analysis are provided in Appendix A.2.

2.2.3 PAGE SCORING

QCR is applied independently to the compact memory of each page, producing a query-specific representation $Z_j(Q)$ for page P_j. Its relevance to query Q is computed using the standard ***MaxSim*** score:

$$S(Q, P_j) = \sum_{t=1}^{L} \max_{\boldsymbol{z} \in Z_j(Q)} \boldsymbol{q}_t^\top \boldsymbol{z}. \tag{8}$$

QCR is applied independently to every page in the document collection, and pages are directly ranked according to $S(Q, P_j)$ without requiring an additional reranking stage.

2.3 DISCUSSION

Q1: How dependent is QCR on the underlying base compressor? Our goal is not to reverse arbitrary compression loss, but to preserve a compact amount of within-group variation that can be selectively exploited once the query is observed. In our experiments, we intentionally adopt a simple per-page clustering scheme to isolate the contribution of query-conditioned reconstruction without introducing additional complexity from a sophisticated base compressor. More generally, QCR is compatible with different grouping-based compression strategies, although its effectiveness naturally depends on the variation preserved by the underlying grouping.

Q2: What additional storage and computation does QCR introduce? Compared with purely static compression, QCR introduces additional storage for the low-rank residual basis and group-level residual statistics, together with extra query-time computation for residual-aware activation, selective reconstruction, and budgeted MaxSim coverage. These costs are explicitly controlled in our design. All residual-memory components are included in the prescribed byte-level storage budget, and the number of stored groups is adjusted accordingly rather than adding residual memory on top of an equally sized static index. At retrieval time, reconstruction is restricted to the Top-K groups for each query token, while the final representation contains $B = M$ vectors, matching the interaction budget of the corresponding static representation. QCR therefore trades a moderate amount of additional online computation for greater query adaptivity, while remaining substantially more compact and efficient than direct interaction with the Full Index. Detailed storage accounting, latency breakdowns, and scalability analyses are provided in Appendix A.3 and A.6.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

Datasets. We evaluate compressed visual document retrieval on ViDoRe v1 (Faysse et al., 2025) and ViDoRe v2 (Macé et al., 2025), covering visually rich pages with diverse text, tables, figures, and layouts. We further evaluate the downstream impact of retrieval on MMLongBench-Doc (Ma et al., 2024) and LongDocURL (Deng et al., 2025). Using their annotated supporting pages, we report evidence retrieval and QA performance to examine whether retrieval gains under compression translate into better downstream answering.

Retrieval Backbones. We conduct experiments with multiple pretrained multi-vector retrieval models, including jina-3.8B (Akram et al., 2026), ColQwen3-4B (Faysse et al., 2025), and Nemotron ColEmbed-3B (Moreira et al., 2026).

Baselines. We compare against the uncompressed Full Index and representative training-free compression methods, including pruning-based SAP (Liu et al., 2026), and Visual-Cue-based pruning (Zhang et al., 2025); merging-based Semantic Clustering (Ma et al., 2025) and AnchorFold (Zuo et al., 2026); and the prune-then-merge method Sculpting (Yan et al., 2026b).

Implementation Details. We use simple per-page clustering as the base compressor, with each group represented by its normalized mean, to isolate the contribution of QCR from additional compressor complexity. All methods are evaluated under matched byte-level storage budgets, including the complete residual memory of QCR. The group number M is determined by the storage budget, the residual rank r by a fixed explained-energy criterion, and we set $B = M$. Reconstruction strength is selected analytically, leaving the per-token reconstruction budget K as the only main retrieval-time hyperparameter; it is calibrated once on a held-out set and fixed thereafter. All embeddings are ℓ_2-normalized and scored with standard MaxSim. Additional implementation and storage-accounting details are provided in the Appendix A.3, while the hyperparameter selection procedure is described in Appendix A.4.

Evaluation Metrics. For visual document retrieval, we report Recall@5 and NDCG@5 retention, defined as the compressed NDCG@5 divided by the Full-Index NDCG@5. We use ρ to denote the byte-level index-storage ratio relative to the Full Index (Appendix A.3). For downstream QA, we report QA accuracy and Evidence Recall@5 based on annotated supporting pages.

3.2 Main Results on Visual Document Retrieval

Figure 3 compares NDCG@5 retention under different storage ratio ρ on ViDoRe v1 and ViDoRe v2 across three multi-vector retrieval backbones.

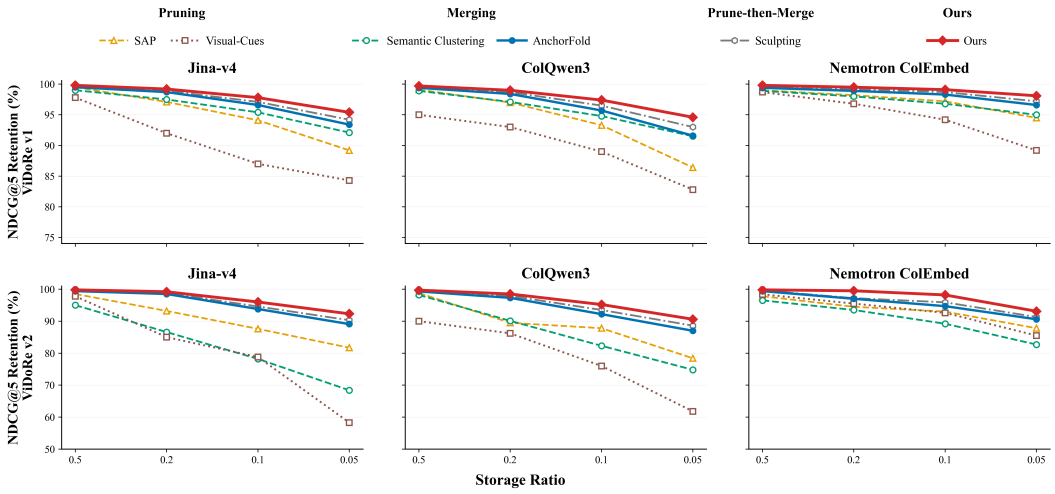


Figure 3: NDCG@5 retention on ViDoRe v1 (top) and ViDoRe v2 (bottom) across three multi-vector retrieval backbones and different storage ratios. Our method consistently achieves higher retention, with the advantage becoming more pronounced under aggressive compression.

Our method consistently achieves the highest NDCG@5 retention across all dataset backbone combinations. While most methods perform similarly under mild compression, the gap widens as the storage ratio decreases. At ρ of 0.1 and 0.05, pruning- and merging-based methods degrade substantially, whereas our method preserves a larger fraction of the Full-Index performance. The advantage over AnchorFold, the strongest static compression baseline in our experiments, also increases under stronger compression. This is consistent with our motivation: aggressive compression forces each static representative to summarize increasingly heterogeneous content, whereas query-conditioned reconstruction can selectively recover query-relevant variation from the residual memory. This trend is consistent across all three retrieval backbones and is particularly pronounced on ViDoRe v2. Overall, the results show that query-adaptive reconstruction is most beneficial under aggressive compression, where fixed representations become increasingly insufficient for preserving query-relevant information.

Table 1 further evaluates relevant-page recovery under aggressive compression. At a storage ratio of 0.05, our method achieves the highest Recall@5 across both benchmarks and all three retrieval backbones. Compared with Sculpting, the strongest competing compression method, QCR improves

Table 1: Recall@5 (%) across different datasets and retrieval backbones. Compressed methods are evaluated at a ρ of 0.05, while Full Index is reported as the uncompressed reference.

Dataset	Backbone	Uncompressed	Pruning		Merging		Prune–Merge	Adaptive
		Full Index	SAP	Visual-Cues	Sem. Clust.	AnchorFold	Sculpting	Ours
ViDoRe v1	jina-3.8B	92.6	76.4	74.8	79.0	81.3	82.4	**85.7**
	ColQwen3-4B	95.1	80.1	78.6	82.8	85.0	86.1	**88.9**
	ColEmbed-3B	96.0	81.7	80.2	84.4	86.2	87.1	**90.1**
ViDoRe v2	jina-3.8B	87.3	66.9	64.5	70.2	73.2	75.0	**79.1**
	ColQwen3-4B	90.5	70.5	68.1	74.0	77.0	78.8	**82.6**
	ColEmbed-3B	91.8	72.0	69.6	75.6	78.4	80.0	**84.0**

Recall@5 by around three points on ViDoRe v1 and about four points on ViDoRe v2. The gains are consistent across different backbones and align with the NDCG@5 trends in Fig. 3. The advantage becomes more pronounced on ViDoRe v2, where aggressive compression makes it harder for a small set of fixed representatives to retain all query-relevant evidence. In contrast, QCR preserves compact within-group variation and selectively activates it for the realized query, improving the likelihood that relevant pages remain within the top-ranked retrieval set. Additional results across a wider range of storage ratios are provided in Appendix A.5.

3.3 Evidence-Guided Downstream QA

We further evaluate whether the benefits of query-conditioned reconstruction extend to downstream question answering with retrieved evidence. To enable a unified analysis across MMLongBench-Doc and LongDocURL, we categorize queries according to their evidence requirements using the supporting-page and task annotations provided by the benchmarks. Specifically, *Single-Evidence* queries rely on a single localized evidence source, while *Multi-Evidence* queries require multiple complementary evidence units. *Cross-Page* queries require supporting evidence distributed across multiple pages, and *Compositional* queries additionally involve comparison, aggregation, numerical reasoning, or multi-step reasoning over the retrieved evidence. These categories capture different forms of evidence complexity rather than a strict difficulty hierarchy. We use the same frozen Qwen2.5-VL-7B-Instruct model (Wu et al., 2025) with an identical prompting template and evidence budget (Zhou et al., 2025). For each category, we report Evidence Recall@5 and downstream QA performance.

Table 2: Evidence retrieval and downstream QA performance across different evidence requirements at a storage ratio of 0.05. ER denotes Evidence Recall@5 and QA denotes downstream QA accuracy. Total ER and Total QA are macro-averaged over the four evidence categories. Full Index is reported as the uncompressed reference.

Method	Single-Evidence		Multi-Evidence		Cross-Page		Compositional		Total	
	ER	QA	ER	QA	ER	QA	ER	QA	ER	QA
Full Index	94.1	75.0	91.0	70.8	84.3	65.2	79.0	58.6	87.1	67.4
SAP	89.1	70.2	80.3	62.5	68.2	55.0	59.8	47.6	74.4	58.8
Visual-Cues	88.5	69.6	79.2	61.6	66.7	54.1	58.4	46.6	73.2	58.0
Semantic Clustering	90.6	71.7	82.9	64.4	72.1	57.7	64.5	50.1	77.5	61.0
AnchorFold	92.0	73.1	86.1	67.3	77.2	61.1	70.0	53.0	81.3	63.6
Sculpting	<u>92.4</u>	<u>73.4</u>	<u>87.3</u>	<u>68.2</u>	<u>79.3</u>	<u>61.9</u>	<u>72.1</u>	<u>54.0</u>	<u>82.8</u>	<u>64.4</u>
Ours	**93.0**	**74.0**	**89.4**	**69.4**	**82.6**	**64.0**	**76.9**	**56.7**	**85.5**	**66.0**

As shown in Table 2, the benefit of QCR becomes more pronounced as the required evidence grows more diverse and distributed. For Single-Evidence queries, most methods preserve localized evidence reasonably well, resulting in relatively small differences. The gap widens for Multi-Evidence and Cross-Page queries, where multiple complementary cues must be retained simultaneously. Compared with Sculpting, the strongest competing compression method, QCR improves Evidence Recall@5 from 79.3 to 82.6 on Cross-Page queries and from 72.1 to 76.9 on Compositional queries, with corresponding QA gains from 61.9 to 64.0 and from 54.0 to 56.7. Overall, QCR achieves a Total ER of 85.5 and Total QA of 66.0, outperforming Sculpting by 2.7 and 1.6 points, respectively. These

results suggest that fixed compressed representations remain effective when evidence is localized, but become increasingly restrictive as the required information spans multiple semantic aspects or pages. By selectively activating query-relevant variation preserved in the residual memory, QCR improves evidence coverage and translates these retrieval gains into stronger downstream QA performance.

3.4 Fine-Grained Analysis and Ablation Study

Appendix A.5 provides additional fine-grained analyses and ablation studies. The results show that QCR becomes increasingly beneficial under stronger compression, more complex queries, and higher page-level semantic heterogeneity. Component-wise ablations further confirm that residual reconstruction, residual-aware activation, and budgeted MaxSim coverage each provide complementary gains beyond those attributable to additional storage capacity. Overall, these findings support preserving compact within-group variation during indexing and adapting the page representation to the realized query at retrieval time.

3.5 Efficiency Analysis

As shown in Fig. 4, we evaluate the online efficiency–fidelity trade-off on ViDoRe v1 using ColQwen3-4B at a storage ratio of 0.05 and the optimal hyperparameter configuration. Static compression methods achieve lower query latency but incur a larger loss in retrieval fidelity under aggressive compression. QCR introduces a moderate amount of query-time computation for query-conditioned reconstruction, increasing latency from 8.0 ms for Sculpting to 10.2 ms. In return, it improves NDCG@5 retention from 93.0% to 94.6% and Recall@5 from 86.1% to 88.9%. QCR nevertheless remains approximately $4.4\times$ faster than direct interaction with the Full Index, which requires 44.8 ms per query. These results show that QCR occupies a favorable operating point between static compressed representations and the Full Index, trading a limited amount of additional online computation for substantially better retrieval fidelity. Detailed offline indexing costs, latency breakdowns, and document-length scalability results are provided in Appendix A.6.

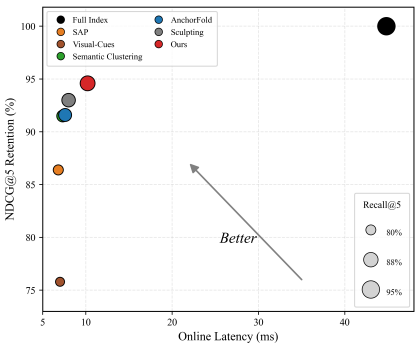


Figure 4: Query-time latency versus retrieval fidelity.

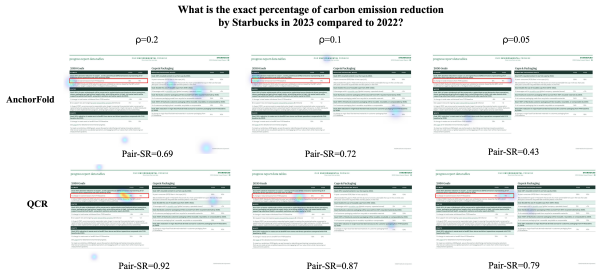


Figure 5: Qualitative comparison of AnchorFold and QCR on a ViDoRe v2 query–page pair across retention ratios. Heatmaps show positive MaxSim contributions, with relevant evidence highlighted in red. Pair-SR measures preservation of the original matching score.

3.6 Qualitative Case Study

Figure 5 compares how AnchorFold and QCR preserve query-relevant matching signals as the document-vector budget decreases. Across the three retention ratios, QCR consistently achieves a higher Pair-SR, indicating better preservation of the original query–page matching score. At $\gamma = 0.20$, QCR retains visible positive MaxSim contributions in both marked evidence regions, whereas AnchorFold preserves contributions mainly in the lower region. Under stronger compression, QCR continues to retain query-relevant responses at $\gamma = 0.10$ and 0.05, while the corresponding signals under AnchorFold become substantially weaker. These examples illustrate how query-conditioned reconstruction can preserve fine-grained matching signals that are attenuated by static compression. More challenging cases are provided in the Appendix A.7.

4 Related Work

Visual document retrieval aims to interpret information from documents whose semantics are jointly expressed through text, tables, figures, charts, and spatial layout (Dong et al., 2026; Ding et al., 2026; Gao et al., 2026). Early document understanding systems often rely on OCR to first convert page images into text and then apply text-based retrieval or language models to the recognized content (Qin et al., 2026; Zhang et al., 2026; Zhong et al., 2026). While effective for text-dominant documents, such pipelines can lose information that is difficult to express through plain text alone, including layout relationships, visual grouping, table structure, figure content, and other spatial cues (Wen et al., 2025; Ding et al., 2026). Moreover, OCR errors may propagate to subsequent retrieval and reasoning stages. Recent vision-language models Radford et al. (2021); Chen et al. (2024); Hu et al. (2025b) instead encode document pages directly from pixels, allowing textual content and visual structure to be represented in a shared semantic space (Hu et al., 2025a; Duan et al., 2025). This visual-first paradigm has become increasingly important for understanding visually rich documents containing heterogeneous and interleaved modalities.

Long and visually complex documents pose an additional challenge because the evidence required for a query is often sparse and distributed across multiple pages (Xu et al., 2025; Zhu et al., 2022; Wang et al., 2025). Directly processing the entire document with a multimodal large language model (Bai et al., 2025; An et al., 2026) is therefore computationally expensive and may introduce substantial irrelevant context. A common solution is to first retrieve a small set of query-relevant pages and then perform downstream reasoning over the retrieved evidence. Recent visual document retrieval frameworks, represented by ColPali (Faysse et al., 2025), encode each page into multiple localized visual embeddings and employ late interaction with query-token embeddings for fine-grained matching (Khattab & Zaharia, 2020; Yan et al., 2026c). Such multi-vector representations preserve localized visual and spatial evidence and provide an effective retrieval interface for downstream document question answering and long-document reasoning. Despite their strong fine-grained retrieval capability, multi-vector representations incur substantial storage and computation costs because each page may be represented by hundreds or even thousands of embeddings (Lee et al., 2025; Akram et al., 2026). A growing body of work therefore seeks to reduce the document-side representation budget while preserving the benefits of late interaction. Pruning-based methods remove redundant or low-importance embeddings according to attention, saliency, structural importance, token redundancy, or other document-side relevance signals (Ma et al., 2025; Liu et al., 2026). These methods preserve the original form of the retained vectors, but information associated with discarded embeddings becomes unavailable at retrieval time, which can be problematic under aggressive compression. Another line of work reduces the representation budget by merging multiple embeddings into a smaller set of representatives (Archish et al., 2026; Clavié et al., 2024). Typical strategies include spatial pooling, semantic clustering, hierarchical merging, and structure-aware aggregation (Cha et al., 2026; Luo et al., 2024). For example, AnchorFold (Zuo et al., 2026) retains representative anchor vectors and aggregates surrounding embeddings around the anchors. Compared with hard pruning, merging allows information from multiple original embeddings to contribute to a compressed representation, but a fixed representative must summarize the semantic variation within its group, potentially averaging out fine-grained query-relevant information as compression becomes more aggressive. Moreover, several approaches seek to combine pruning and aggregation, such as Sculpting (Yan et al., 2026b), which first prunes less informative visual patches and then merges similar remaining patches.

5 Conclusion

We presented Query-Conditioned Reconstruction (QCR), a training-free framework for query-adaptive compressed visual document retrieval. Instead of committing to a fixed page representation during indexing, QCR preserves compact low-rank residual variation and selectively instantiates query-specific vectors at retrieval time. Together with budgeted MaxSim coverage, this enables the compressed representation to retain complementary query-relevant signals under a fixed interaction budget. Experiments across multiple retrieval encoders and benchmarks show that QCR better preserves Full-Index retrieval quality under aggressive compression, with consistent gains in page retrieval, evidence recovery, and downstream long-document QA. These results highlight the benefit of deferring part of the representation decision until the query is observed, rather than relying solely on static compressed representations.

AI USE STATEMENT

In this work, we did not use generative AI tools for any task that requires disclosure under the ICLR 2027 AI Policy. Research ideation, method design, theoretical derivations, and the substantive implementation of QCR were all carried out by the human authors. The remaining required-disclosure categories (e.g. studies involving human subjects) are not applicable to this work. We did use generative AI tools for one recommended-disclosure task: light editing of the manuscript for grammar, phrasing, and typographical corrections. We have reviewed all AI-assisted edits and verified that they preserve the intended technical content. We also note that several benchmarks used for evaluation (Section 3.1) may internally invoke LLM-based verification such as LLM-as-a-judge as part of their officially released protocols; we run each benchmark under its standard protocol without modification, so any such LLM usage is inherited from the benchmark rather than introduced by our method. We take full responsibility for the final content of this work, including all text, claims, and artifacts.

REPRODUCIBILITY STATEMENT

Section 3.1 and Appendix A.4 describe the training pipeline and list every hyperparameter used in the main and ablation runs. Section 3.1 specify the evaluation protocol, including sampling settings and per-benchmark metrics. All datasets and benchmarks used in this work are publicly available and cited at first use. The theoretical claim behind the greedy selection is stated and proved in Appendix A.2.

REFERENCES

Mohammad Kalim Akram, Saba Sturua, Nastia Havriushenko, Quentin Herreros, Michael Günther, Maximilian Werk, and Han Xiao. jina-embeddings-v5-text: Task-targeted embedding distillation. *arXiv preprint arXiv:2602.15547*, 2026.

Xiang An, Yin Xie, Feilong Tang, Yunyao Yan, Huajie Tan, Didi Zhu, Changrui Chen, Xiuwei Zhao, Bin Qin, Kaicheng Yang, et al. Llava-onevision-2: Towards next-generation perceptual intelligence. *arXiv preprint arXiv:2605.25979*, 2026.

S Archish, Ankit Garg, Kirankumar Shiragur, and Neeraj Kayal. Incorporating token importance in multi-vector retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 32860–32866, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Sungguk Cha, DongWook Kim, Mintae Kim, Youngsub Han, Byoung-Ki Jeon, and Sangyeob Lee. Reinpool: Reinforcement learning pooling multi-vector embeddings for retrieval system. *arXiv preprint arXiv:2601.07125*, 2026.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24185–24198, 2024.

Benjamin Clavié, Antoine Chaffin, and Griffin Adams. Reducing the footprint of multi-vector retrieval with minimal performance impact via token pooling. *arXiv preprint arXiv:2409.14683*, 2024.

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, et al. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1135–1159, 2025.

Yihao Ding, Siwen Luo, Yue Dai, Yanbei Jiang, Zechuan Li, Qiang Sun, Geoffrey Martin, Wei Liu, and Yifan Peng. A survey on mllm-based visually rich document understanding: Methods, challenges, and emerging trends. *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 13319–13340, 2026.

Kuicai Dong, CHANG YUJING, Shijie Huang, Yasheng Wang, Ruiming Tang, and Yong Liu. Benchmarking retrieval-augmented multimomal generation for document question answering. *Advances in Neural Information Processing Systems*, 38, 2026.

Yuchen Duan, Zhe Chen, Yusong Hu, Weiyun Wang, Shenglong Ye, Botian Shi, Lewei Lu, Qibin Hou, Tong Lu, Hongsheng Li, et al. Docopilot: Improving multimodal models for document-level understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4026–4037. IEEE, 2025.

Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. Colpali: Efficient document retrieval with vision language models. In *International Conference on Learning Representations*, volume 2025, pp. 61424–61449, 2025.

Sensen Gao, Shanshan Zhao, Xu Jiang, Lunhao Duan, Yong Xien Chng, Qing-Guo Chen, Weihua Luo, Kaifu Zhang, Jia-Wang Bian, and Mingming Gong. Scaling beyond context: A survey of multimodal retrieval-augmented generation for document understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4458–4489, 2026.

Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5817–5834, 2025a.

Anwen Hu, Haiyang Xu, Liang Zhang, Jiabo Ye, Ming Yan, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. mplug-docowl2: High-resolution compressing for ocr-free multi-page document understanding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5817–5834, 2025b.

Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48, 2020.

Juyeon Kim, Geon Lee, Dongwon Choi, Taeuk Kim, and Kijung Shin. Hybrid-vector retrieval for visually rich documents: Combining single-vector efficiency and multi-vector accuracy. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 1073–1089, 2026.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. Nv-embed: Improved techniques for training llms as generalist embedding models. In *International Conference on Learning Representations*, volume 2025, pp. 79310–79333, 2025.

Zhuchenyang Liu, Ziyu Hu, Yao Zhang, and Yu Xiao. Structural anchor pruning: Training-free multi-vector compression for visual document retrieval. *arXiv preprint arXiv:2601.20107*, 2026.

Chuwei Luo, Yufan Shen, Zhaoqing Zhu, Qi Zheng, Zhi Yu, and Cong Yao. Layoutllm: Layout instruction tuning with large language models for document understanding. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15630–15640. IEEE, 2024.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, et al. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010, 2024.

Yubo Ma, Jinsong Li, Yuhang Zang, Xiaobao Wu, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Haodong Duan, Jiaqi Wang, Yixin Cao, et al. Towards storage-efficient visual document retrieval: An empirical study on reducing patch-level embeddings. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 19568–19580, 2025.

Quentin Macé, António Loison, and Manuel Faysse. Vidore benchmark v2: Raising the bar for visual retrieval. *arXiv preprint arXiv:2505.17166*, 2025.

Gabriel de Souza P Moreira, Ronay Ak, Mengyao Xu, Oliver Holworthy, Benedikt Schifferer, Zhiding Yu, Yauhen Babakhin, Radek Osmulski, Jiarui Cai, Ryan Chesler, et al. Nemotron colembed v2: Top-performing late interaction embedding models for visual document retrieval. *arXiv preprint arXiv:2602.03992*, 2026.

Hanxiang Qin, Alexander Martin, Rohan Jha, Chunsheng Zuo, Reno Kriz, and Benjamin Van Durme. Multi-vector index compression in any modality. *arXiv preprint arXiv:2602.21202*, 2026.

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.

Hengyi Wang, Haizhou Shi, Shiwei Tan, Weiyi Qin, Wenyuan Wang, Tunyu Zhang, Akshay Nambi, Tanuja Ganu, and Hao Wang. Multimodal needle in a haystack: Benchmarking long-context capability of multimodal large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3221–3241, 2025.

Zichen Wen, Yifeng Gao, Weijia Li, Conghui He, and Linfeng Zhang. Token pruning in multimodal large language models: Are we solving the right problem? In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 15537–15549, 2025.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025.

Mengyao Xu, Gabriel Moreira, Ronay Ak, Radek Osmulski, Yauhen Babakhin, Zhiding Yu, Benedikt Schifferer, and Even Oldridge. Llama nemoretriever colembed: Top-performing text-image retrieval model. *arXiv preprint arXiv:2507.05513*, 2025.

Yibo Yan, Mingdong Ou, Yi Cao, Jiahao Huo, Xin Zou, Shuliang Liu, James Kwok, and Xuming Hu. Visual late chunking: An empirical study of contextual chunking for efficient visual document retrieval. *arXiv preprint arXiv:2604.10167*, 2026a.

Yibo Yan, Mingdong Ou, Yi Cao, Xin Zou, Jiahao Huo, Shuliang Liu, James Kwok, and Xuming Hu. Sculpting the vector space: Towards efficient multi-vector visual document retrieval via prune-then-merge framework. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 24883–24925, 2026b.

Yibo Yan, Mingdong Ou, Yi Cao, Xin Zou, Shuliang Liu, Jiahao Huo, Yu Huang, James Kwok, and Xuming Hu. Beyond the grid: Layout-informed multi-vector retrieval with parsed visual document representations. *arXiv preprint arXiv:2603.01666*, 2026c.

Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond text-visual attention: Exploiting visual cues for effective token pruning in vlms. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 20857–20867. IEEE, 2025.

Qizhe Zhang, Mengzhen Liu, Lichen Li, Ming Lu, Yuan Zhang, Junwen Pan, Qi She, and Shanghang Zhang. Beyond attention or similarity: Maximizing conditional diversity for token pruning in mllms. *Advances in Neural Information Processing Systems*, 38:25438–25468, 2026.

Yufeng Zhong, Lei Chen, Zhixiong Zeng, Xuanle Zhao, Deyang Jiang, Liming Zheng, Jing Huang, Haibo Qiu, Peng Shi, Siqi Yang, et al. Reading or reasoning? format decoupled reinforcement learning for document ocr. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33164–33173, 2026.

Ao Zhou, Zebo Gu, Tenghao Sun, Jiawen Chen, Mingsheng Tu, Zifeng Cheng, Yafeng Yin, Zhiwei Jiang, and Qing Gu. Hierarchical vision-language reasoning for multimodal multiple-choice question answering. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 13784–13790, 2025.

Dawei Zhu, Rui Meng, Jiefeng Chen, Sujian Li, Tomas Pfister, and Jinsung Yoon. Doclens: A tool-augmented multi-agent framework for long visual document understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26804–26829, 2026.

Fengbin Zhu, Wenqiang Lei, Fuli Feng, Chao Wang, Haozhou Zhang, and Tat-Seng Chua. Towards complex document understanding by discrete reasoning. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 4857–4866, 2022.

Haoyu Zuo, Yibo Yan, Xin Zou, Shuliang Liu, Yi Cao, Mingdong Ou, and Xuming Hu. Anchorfold: A focus-then-fold framework via recursive attention propagation for efficient multi-vector visual document retrieval. *arXiv preprint arXiv:2608.08732*, 2026.

A Appendix

A.1 Exact Cosine Oracle for Normalized Reconstruction

The main formulation uses a closed-form cosine-aware score for efficient group activation and reconstruction. In this section, we define the corresponding exact optimization problem under the normalized retrieval geometry and use it as an oracle for analysis. Importantly, this oracle is not used during inference; it serves only to quantify how closely the efficient closed-form surrogate approximates the exact cosine objective.

Normalized feasible region. Recall that the pre-normalization residual region of group G_i is

$$\mathcal{E}_i = \left\{ \boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a} : \|\boldsymbol{a}\|_2 \le \tau_i \right\}. \tag{9}$$

Since late-interaction retrieval operates on normalized embeddings, the actual set of attainable retrieval vectors is the radial projection of $\mathcal{E}_i$ onto the unit sphere:

$$\overline{\mathcal{E}}_i = \left\{ \frac{\boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a}}{\left\| \boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a} \right\|_2} : \|\boldsymbol{a}\|_2 \le \tau_i \right\}. \tag{10}$$

Thus, while the pre-normalized reconstruction belongs to $\mathcal{E}_i$, the vector actually used by MaxSim belongs to $\overline{\mathcal{E}}_i$. We therefore interpret document-supported reconstruction in terms of this normalized feasible region rather than the translated ellipsoid itself.

Exact cosine oracle. For a normalized query token $\boldsymbol{q}_t$, we define the exact cosine potential of group G_i as the maximum similarity attainable by any vector in $\overline{\mathcal{E}}_i$:

$$h_i^{\mathrm{oracle}}(\boldsymbol{q}_t) = \max_{\|\boldsymbol{a}\|_2 \le \tau_i} \boldsymbol{q}_t^\top \frac{\boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a}}{\left\| \boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a} \right\|_2}. \tag{11}$$

Unlike the unnormalized ellipsoidal support function, Eq. 11 is defined directly in the same cosine geometry used by the final MaxSim scorer. It therefore provides an exact reference for the maximum query alignment supported by the stored residual region.

The corresponding oracle reconstruction is

$$\boldsymbol{a}_i^*(\boldsymbol{q}_t) = \arg\max_{\|\boldsymbol{a}\|_2 \le \tau_i} \boldsymbol{q}_t^\top \operatorname{norm}\left(\boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a} \right), \qquad \boldsymbol{z}_i^{\mathrm{oracle}}(\boldsymbol{q}_t) = \operatorname{norm}\left(\boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a}_i^*(\boldsymbol{q}_t) \right). \tag{12}$$

By construction, $\boldsymbol{z}_i^{\mathrm{oracle}}(\boldsymbol{q}_t) \in \overline{\mathcal{E}}_i$ and directly maximizes the cosine similarity used during retrieval.

Numerical evaluation of the oracle. Eq. 11 is a low-dimensional constrained optimization problem because $\boldsymbol{a} \in \mathbb{R}^r$ with $r \ll d$. We nevertheless avoid solving it during retrieval because doing so independently for every query token and compressed group would introduce substantial online optimization overhead. Instead, we evaluate the oracle only for diagnostic experiments in the supplementary analysis.

For a candidate value $\gamma \in [0, 1]$, testing whether a cosine similarity of at least γ is attainable can be written as the feasibility problem

$$\|\boldsymbol{a}\|_2 \le \tau_i, \qquad \boldsymbol{q}_t^\top \boldsymbol{x}(\boldsymbol{a}) \ge \gamma \|\boldsymbol{x}(\boldsymbol{a})\|_2, \qquad \boldsymbol{x}(\boldsymbol{a}) = \boldsymbol{c}_i + \boldsymbol{U}\boldsymbol{\Lambda}_i^{1/2}\boldsymbol{a}. \tag{13}$$

For fixed non-negative γ, these constraints form a second-order-cone feasibility problem. We therefore obtain $h_i^{\mathrm{oracle}}(\boldsymbol{q}_t)$ to numerical precision by bisection over γ, solving the corresponding low-dimensional feasibility problem at each step. This procedure provides a global reference for the normalized cosine objective while remaining practical for offline analysis.

Relation to the closed-form cosine-aware surrogate. Directly evaluating Eq. 11 for every group would be unnecessarily expensive at query time. The main method therefore uses the first-order cosine-aware approximation

$$\bar{h}_i(\boldsymbol{q}_t) = \boldsymbol{q}_t^\top \boldsymbol{\mu}_i + \frac{\tau_i}{\|\boldsymbol{c}_i\|_2}\sqrt{\boldsymbol{q}_t^\top \boldsymbol{P}_i \boldsymbol{U}\boldsymbol{\Lambda}_i\boldsymbol{U}^\top \boldsymbol{P}_i \boldsymbol{q}_t}, \qquad \boldsymbol{P}_i = \boldsymbol{I} - \boldsymbol{\mu}_i\boldsymbol{\mu}_i^\top, \tag{14}$$

which is obtained by linearizing normalized similarity around the static representative $\boldsymbol{\mu}_i$. The projector $\boldsymbol{P}_i$ removes the radial component of the perturbation, which does not affect the normalized direction to first order, and retains the residual directions that can change cosine similarity.

We use the exact oracle only to evaluate the quality of this approximation. In particular, we measure the overlap between the Top-K groups selected by $\bar{h}_i(\boldsymbol{q}_t)$ and those selected by $h_i^{\mathrm{oracle}}(\boldsymbol{q}_t)$:

$$\mathrm{Overlap@}\,K = \frac{\left|\mathcal{I}_t^{\mathrm{sur}} \cap \mathcal{I}_t^{\mathrm{oracle}}\right|}{K}, \tag{15}$$

where $\mathcal{I}_t^{\mathrm{sur}}$ and $\mathcal{I}_t^{\mathrm{oracle}}$ denote the Top-K groups according to the surrogate and exact cosine oracle, respectively. We additionally compare the resulting retrieval performance when group activation is performed using the two scores. High selection overlap and similar retrieval accuracy indicate that the closed-form score captures the group ordering induced by the exact normalized objective while avoiding iterative optimization at retrieval time.

Trust-Radius Estimation We provide additional details on the estimation of the group-specific trust radius τ_i used in query-conditioned reconstruction. Recall that for each original vector $\boldsymbol{e}_n \in G_i$, its residual relative to the group center is $\boldsymbol{r}_n = \boldsymbol{e}_n - \boldsymbol{c}_i$, and its coordinate in the shared residual subspace is $\boldsymbol{a}_n = \boldsymbol{U}^\top \boldsymbol{r}_n$.

Since different residual directions may exhibit different levels of variation, we normalize the residual coordinates using the group-specific second-order statistics:

$$\widetilde{\boldsymbol{a}}_n = (\boldsymbol{\Lambda}_i + \epsilon \boldsymbol{I})^{-1/2}\boldsymbol{U}^\top \boldsymbol{r}_n, \qquad n \in G_i, \tag{16}$$

where ϵ is a small constant for numerical stability. The norm $\|\widetilde{\boldsymbol{a}}_n\|_2$ measures the magnitude of each residual after accounting for the anisotropic variation of the group.

We then define the trust radius as a fixed quantile of these normalized residual magnitudes:

$$\tau_i = \mathrm{Quantile}_\alpha\left(\{\|\widetilde{\boldsymbol{a}}_n\|_2 \,:\, n \in G_i\}\right), \tag{17}$$

where α is a global quantile level shared across all groups. Thus, τ_i is estimated directly from the empirical residual distribution of group G_i, rather than treated as an independently tuned parameter.

This construction adapts the reconstruction range to the local variability of each group. Groups with broader residual distributions permit larger query-conditioned movements, whereas compact groups impose tighter constraints. As a result, the reconstructed vectors remain within semantic variations supported by the indexed page rather than moving arbitrarily toward the query.

A.2 Greedy Optimization for Budgeted MaxSim Coverage

We provide additional details on the greedy optimization used to construct the final query-specific representation. After query-relevant group activation and conditioned reconstruction, the candidate pool for a query $Q = \{\boldsymbol{q}_t\}_{t=1}^L$ is $\mathcal{V}(Q) = \{\boldsymbol{\mu}_i\}_{i=1}^M \cup \{\boldsymbol{z}_i(\boldsymbol{q}_t) \mid i \in \mathcal{I}_t,\ t = 1, \ldots, L\}$, where $\mathcal{I}_t$ contains the Top-K groups activated for query token $\boldsymbol{q}_t$. The pool therefore contains both static group representatives and query-conditioned candidates. Since retaining all candidates would increase the subsequent late-interaction cost, we select at most B vectors by maximizing the MaxSim coverage objective

$$F(Z) = \sum_{t=1}^{L} \max_{\boldsymbol{z} \in Z} \boldsymbol{q}_t^\top \boldsymbol{z}, \qquad Z \subseteq \mathcal{V}(Q), \quad |Z| \le B. \tag{18}$$

A simple alternative would be to rank candidates by their individual query similarity and retain the Top-B vectors. However, this strategy does not account for redundancy: several highly ranked candidates may all correspond to the same query token or semantic aspect, while other query

Algorithm 1 Greedy Budgeted MaxSim Coverage

Require: Query embeddings $Q = \{\boldsymbol{q}_t\}_{t=1}^{L}$, candidate pool $\mathcal{V}(Q)$, budget B
$Z \leftarrow \emptyset$
$s_t \leftarrow -1$ for all $t = 1, \ldots, L$
Precompute $A_{t,v} = \boldsymbol{q}_t^\top \boldsymbol{v}$ for all $\boldsymbol{v} \in \mathcal{V}(Q)$
for $b = 1, \ldots, B$ **do**
  $\boldsymbol{v}^* \leftarrow \arg\max_{\boldsymbol{v} \in \mathcal{V}(Q) \setminus Z} \sum_{t=1}^{L} [\max\{s_t, A_{t,v}\} - s_t]$
  if $\Delta(\boldsymbol{v}^* \mid Z) \leq 0$ **then**
    break
  end if
  $Z \leftarrow Z \cup \{\boldsymbol{v}^*\}$
  $s_t \leftarrow \max\{s_t, A_{t,v^*}\}$ for all t
end for
return Z

concepts remain uncovered. The objective in Eq. 18 instead evaluates candidates jointly and favors complementary vectors that improve the best available match for different query tokens.

Let $s_t(Z)$ denote the current coverage of query token $\boldsymbol{q}_t$. Since all query and candidate vectors are normalized, their inner products lie in $[-1, 1]$; we therefore initialize $s_t(\emptyset) = -1$. For a candidate $\boldsymbol{v} \notin Z$, its marginal coverage gain is

$$\Delta(\boldsymbol{v} \mid Z) = \sum_{t=1}^{L} \left[\max\left\{s_t(Z), \boldsymbol{q}_t^\top \boldsymbol{v}\right\} - s_t(Z)\right]. \tag{19}$$

Starting from $Z = \emptyset$, we repeatedly add the candidate with the largest marginal gain and update the per-token coverage scores. The procedure terminates after B vectors have been selected or when no remaining candidate provides additional coverage. Intuitively, once a query token is already well covered, another vector matching the same token contributes little additional gain, whereas a candidate that improves a poorly covered query token is favored. This diminishing-return property encourages semantic complementarity in the selected representation.

The greedy procedure also admits an efficient implementation. Let $C = |\mathcal{V}(Q)| \leq M + LK$ denote the number of candidates. We first compute the query–candidate similarity matrix once, which requires $O(LCd)$ operations. Subsequent greedy iterations operate only on this precomputed $L \times C$ matrix, requiring $O(BLC)$ scalar operations in total and avoiding repeated d-dimensional similarity computation. The selected representation contains at most B vectors, so the subsequent MaxSim interaction operates on a substantially smaller representation than the full candidate pool.

The coverage objective is a sum of per-token maximum-similarity functions and exhibits diminishing returns under the cardinality constraint. After shifting the objective by the constant lower bound induced by normalized cosine similarities, it is a monotone submodular set function. Consequently, the standard greedy procedure provides the usual $(1-1/e)$ approximation guarantee for the corresponding cardinality-constrained maximization problem. More importantly in our setting, the greedy objective is directly aligned with the downstream MaxSim scoring rule: the vectors retained during compression are selected according to the same notion of complementary query-token coverage used during retrieval.

A.3 Storage Accounting

We compare all compression methods under matched total index storage rather than matched retained-vector counts. Let a page contain N original embeddings in $\mathbb{R}^d$. The Full Index therefore stores Nd scalar values for this page. QCR stores M group centers $\{\boldsymbol{c}_i\}_{i=1}^{M}$, a shared residual basis $\boldsymbol{U} \in \mathbb{R}^{d \times r}$, group-specific residual statistics $\lambda_i \in \mathbb{R}^r$, and one trust radius τ_i for each group. Accordingly, the number of stored scalar values for one page is

$$S_{\mathrm{QCR}} = Md + dr + Mr + M, \tag{20}$$

whereas the Full Index requires $S_{\mathrm{Full}} = Nd$. When all components are stored using the same numerical precision, the corresponding per-page storage ratio is $\rho_{\mathrm{QCR}} = \frac{Md+dr+Mr+M}{Nd}$. The

normalized group representatives used during retrieval are computed directly from the stored centers and therefore do not require additional index storage. For dataset-level reporting, we compute the storage ratio from the total serialized retrieval-time index size across all pages,

$$\rho = \frac{\sum_j \mathrm{Bytes}(M_j)}{\sum_j \mathrm{Bytes}(E_j)}, \tag{21}$$

where $\mathrm{Bytes}(M_j)$ includes every tensor required by QCR at retrieval time. For each target storage ratio, we adjust the number of groups M such that the resulting total index size matches the prescribed storage budget as closely as possible. Importantly, the auxiliary residual memory is included in this budget rather than added on top of an equal number of retained vectors. The same accounting principle is applied to competing methods: all method-specific auxiliary index components required at retrieval time are included whenever applicable. Thus, comparisons at a given storage ratio reflect approximately equal total index memory rather than equal numbers of compressed representatives.

A.4 Hyperparameter Selection

QCR contains several quantities that control storage capacity and query-time computation. Rather than tuning all of them for retrieval performance, we determine most quantities directly from the target storage budget or the statistics of the indexed page. Specifically, M is storage-determined, r is selected by residual explained energy, B is tied to M, and the reconstruction strength is optimized analytically for each query–group pair. Consequently, K is the only main retrieval-time hyperparameter requiring validation-based calibration. We detail each choice below.

Table 3: Determination of the main quantities used by QCR.

Quantity	Role	Determination
M	Groups per page	Target storage budget ρ
r	Residual rank	Explained residual energy η
K	Groups reconstructed per token	Held-out one-SE calibration
B	Final representation budget	$B = M$
β	Reconstruction strength	Per-query analytic optimum

Consider a page containing N original vectors in $\mathbb{R}^d$. The Full Index stores $S_{\mathrm{Full}} = Nd$ scalar values. For a page compressed into M groups with residual rank r, QCR stores M group centers, a shared residual basis $\boldsymbol{U} \in \mathbb{R}^{d\times r}$, M residual-variation vectors $\lambda_i \in \mathbb{R}^r$, and M trust radii. Baed on Eq .20, the largest admissible number of groups has the closed form

$$M = \left\lfloor \frac{\rho N d - dr}{d + r + 1} \right\rfloor. \tag{22}$$

We additionally restrict $1 \le M \le N$. Thus, M is not selected according to retrieval accuracy: increasing the target storage ratio automatically permits more groups, whereas a more restrictive budget reduces M. In our adaptive-rank implementation, r depends on the residual spectrum produced by the grouping and therefore weakly depends on M. We resolve this coupling deterministically. For each candidate group count M, we construct the corresponding groups, compute the residual spectrum, determine $r(M)$ using the explained-energy rule described below, and evaluate the actual storage $S_{\mathrm{QCR}}(M, r(M))$. We then select the largest M satisfying

$$M^* = \max\left\{M : S_{\mathrm{QCR}}(M, r(M)) \le \rho N d\right\}. \tag{23}$$

This procedure is performed only during offline index construction and guarantees that the residual memory is included within, rather than added on top of, the prescribed storage budget.

Residual Rank from Explained Energy The residual rank controls how much within-group variation is retained by the shared low-rank basis. Instead of selecting r according to downstream retrieval accuracy, we determine it directly from the residual spectrum. Let $\boldsymbol{R} = [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_N]$ denote the residual matrix and let $\sigma_1 \ge \sigma_2 \ge \cdots$ be its singular values. Given a fixed explained-energy threshold η, we choose the smallest rank satisfying:

$$r = \min\left\{ r' : \frac{\sum_{k=1}^{r'} \sigma_k^2}{\sum_k \sigma_k^2} \ge \eta \right\}. \tag{24}$$

We use the same global threshold η for all pages, storage ratios, datasets, and retrieval encoders. This allows pages with simple residual structure to use a smaller basis, while pages exhibiting richer within-group variation retain additional residual directions. Importantly, η controls reconstruction fidelity in the residual space and is not tuned against retrieval metrics.

Storage-Aligned Final Representation Budget After query-conditioned reconstruction, the candidate pool contains both static representatives and reconstructed vectors. Although this pool may hold up to $M + LK$ candidates, retaining all of them would inflate the subsequent late-interaction cost. We therefore cap the final query-specific representation at $B = M$, ensuring $|Z(Q)| \leq M$. This aligns the final interaction budget with the number of representatives allowed by the underlying compressed page representation. Consequently, QCR changes which vectors are instantiated for the realized query rather than increasing the number of vectors passed to the final MaxSim scorer. It also eliminates B as an independently tuned hyperparameter and enables a controlled comparison with static compressed representations.

Query-Adaptive Reconstruction Strength Rather than using a globally tuned reconstruction strength, we determine the movement magnitude independently for each selected query–group pair. Let $\boldsymbol{v}_i(\boldsymbol{q}_t)$ denote the feasible residual reconstruction direction for group G_i and query token $\boldsymbol{q}_t$, scaled so that the full update remains within the group-specific residual region. We parameterize the reconstructed vector as

$$\boldsymbol{z}_i(\beta, \boldsymbol{q}_t) = \mathrm{norm}\big(\boldsymbol{c}_i + \beta \boldsymbol{v}_i(\boldsymbol{q}_t)\big), \qquad 0 \leq \beta \leq 1, \tag{25}$$

and select β by maximizing the cosine alignment with the current query token, i.e., $\beta_i^*(\boldsymbol{q}_t) = \arg\max_{\beta \in [0,1]} \boldsymbol{q}_t^\top \boldsymbol{z}_i(\beta, \boldsymbol{q}_t)$.

This one-dimensional problem admits a closed-form candidate solution. With $A = \boldsymbol{q}_t^\top \boldsymbol{c}_i$, $B_q = \boldsymbol{q}_t^\top \boldsymbol{v}_i$, $C = \|\boldsymbol{c}_i\|_2^2$, $D = \boldsymbol{c}_i^\top \boldsymbol{v}_i$, $E = \|\boldsymbol{v}_i\|_2^2$, the cosine objective and its interior stationary point are

$$f(\beta) = \frac{A + \beta B_q}{\sqrt{C + 2\beta D + \beta^2 E}}, \qquad \beta_0 = \frac{AD - B_q C}{B_q D - AE}. \tag{26}$$

We evaluate $f(\beta)$ at $\beta = 0$, $\beta = 1$, and $\beta = \beta_0$ whenever $\beta_0 \in (0, 1)$, and choose the candidate with the highest cosine similarity: $\beta_i^*(\boldsymbol{q}_t) = \arg\max_{\beta \in \{0,1,\beta_0\} \cap [0,1]} f(\beta)$. Thus, the reconstruction strength adapts automatically to each query–group pair without introducing an additional validation-tuned parameter. The constraint $0 \leq \beta \leq 1$ ensures that the update does not exceed the document-supported reconstruction range defined for the group.

A.5 Supplementary Experiments

Recall Tables 4–6 report Recall@5 (%) at storage ratios of 0.50, 0.20, and 0.10, showing that our method consistently achieves the best performance across datasets and backbones.

Table 4: Recall@5 (%) at a storage ratio of 0.50.

Dataset	Backbone	Full Index	SAP	Visual-Cues	Sem. Clust.	AnchorFold	Sculpting	Ours
ViDoRe v1	jina-3.8B	92.6	90.8	90.2	91.2	91.6	91.9	**92.1**
	ColQwen3-4B	95.1	93.8	93.3	94.0	94.4	94.6	**94.8**
	ColEmbed-3B	96.0	94.9	94.5	95.2	95.5	95.6	**95.8**
ViDoRe v2	jina-3.8B	87.3	84.8	84.0	85.3	85.9	86.2	**86.7**
	ColQwen3-4B	90.5	88.5	87.7	89.0	89.6	89.8	**90.2**
	ColEmbed-3B	91.8	90.0	89.3	90.4	90.9	91.1	**91.5**

Fine-Grained Analysis We first examine whether the benefit of query-conditioned reconstruction depends jointly on the compression level and the evidence complexity of the query. For each query category, we report the improvement in Evidence Recall@5 over the strongest static compression baseline under different storage ratios. This analysis disentangles two factors that make compressed retrieval difficult: the amount of information suppressed by compression and the diversity of evidence required by the query.

Table 5: Recall@5 (%) at a storage ratio of 0.20.

Dataset	Backbone	Full Index	SAP	Visual-Cues	Sem. Clust.	AnchorFold	Sculpting	Ours
ViDoRe v1	jina-3.8B	92.6	86.8	85.6	88.0	89.0	89.6	**90.5**
	ColQwen3-4B	95.1	89.5	88.4	90.5	91.7	92.2	**93.0**
	ColEmbed-3B	96.0	91.0	90.2	92.1	93.0	93.5	**94.1**
ViDoRe v2	jina-3.8B	87.3	79.0	77.8	80.5	82.0	82.8	**84.0**
	ColQwen3-4B	90.5	82.5	80.8	84.0	85.5	86.4	**87.8**
	ColEmbed-3B	91.8	84.5	83.0	86.0	87.4	88.2	**89.5**

Table 6: Recall@5 (%) at a storage ratio of 0.10.

Dataset	Backbone	Full Index	SAP	Visual-Cues	Sem. Clust.	AnchorFold	Sculpting	Ours
ViDoRe v1	jina-3.8B	92.6	82.0	80.6	83.8	85.1	86.0	**87.6**
	ColQwen3-4B	95.1	85.0	83.8	87.0	88.8	89.7	**91.2**
	ColEmbed-3B	96.0	86.7	85.5	88.8	90.1	91.0	**92.4**
ViDoRe v2	jina-3.8B	87.3	73.3	71.5	75.2	77.3	78.8	**81.5**
	ColQwen3-4B	90.5	76.2	74.3	78.8	81.0	82.5	**85.0**
	ColEmbed-3B	91.8	78.8	77.0	81.2	83.2	84.6	**87.0**

Table 7: Evidence Recall@5 improvement over the strongest static compression baseline under different storage ratios and query difficulties.

Query Type	0.50	0.20	0.10	0.05
Single-Evidence	+0.2	+0.3	+0.5	+0.6
Multi-Evidence	+0.4	+0.8	+1.3	+2.1
Cross-Page	+0.6	+1.4	+2.2	+3.3
Compositional	+0.8	+1.8	+3.0	**+4.8**

Table 8: NDCG@5 retention (%) under different levels of page semantic heterogeneity on ViDoRe v2 with ColQwen3 at a storage ratio of 0.10.

Method	Low	Medium	High
SAP	94.0	88.0	81.4
Visual-Cues	88.0	75.5	64.5
Semantic Clustering	92.0	83.0	71.9
AnchorFold	96.5	93.0	87.1
Sculpting	<u>97.0</u>	<u>94.3</u>	<u>89.2</u>
Ours	**97.4**	**95.8**	**92.4**

As shown in Table 7, the benefit of query-conditioned reconstruction increases with both compression strength and query complexity. Under mild compression, the improvements remain small, indicating that static representations retain most of the evidence required by the query. As the storage budget decreases, however, the gap grows progressively, particularly for queries requiring multiple complementary evidence sources. At a storage ratio of 0.05, QCR improves Evidence Recall@5 by only 0.6 points for Single-Evidence queries, but by 3.3 and 4.8 points for Cross-Page and Compositional queries, respectively. This trend suggests that query adaptation becomes increasingly valuable when aggressive compression must preserve multiple query-relevant semantic cues simultaneously. Our method is motivated by the observation that static merging becomes increasingly lossy when a compressed group contains semantically diverse vectors. We quantify page-level semantic heterogeneity using the average residual energy $H(P) = \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{e}_n - \boldsymbol{c}_{g(n)}\|_2^2$, where $\boldsymbol{c}_{g(n)}$ denotes the compressed center assigned to $\boldsymbol{e}_n$. Pages are divided into three equally sized groups according to $H(P)$.

Table 8 shows that the advantage of QCR increases with semantic heterogeneity. On low-heterogeneity pages, QCR exceeds Sculpting by only 0.4 NDCG@5 retention points, since a small set of static representatives already captures most of the page variation. The gap increases to 1.5 points for medium heterogeneity and 3.2 points for highly heterogeneous pages. This behavior is consistent with the role of residual memory: when groups contain more diverse semantic content, a single static center suppresses more potentially useful variation, whereas QCR can selectively activate the variation preserved by the residual memory according to the realized query.

Ablation Study We progressively add the main components of QCR to examine their individual contributions. Starting from a Static Centroid baseline with the same group assignments, we introduce query-conditioned reconstruction, residual-aware group selection, and budgeted MaxSim coverage in

sequence. Figure 6 shows that each component consistently improves performance. Adding residual

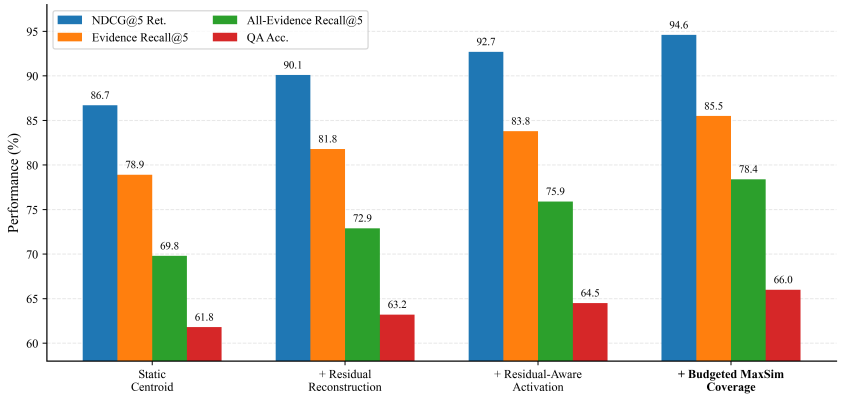


Figure 6: Progressive ablation of QCR using the same base grouping configuration. NDCG@5 retention is evaluated on ViDoRe v1 with ColQwen3, while evidence retrieval and QA follow the downstream evaluation protocol used in the main experiments.

reconstruction increases NDCG@5 retention from 86.7% to 90.1%, indicating that preserving reconstructable within-group variation is already more effective than relying on static centroids alone. Residual-aware activation further improves all metrics, raising Evidence Recall@5 from 81.8% to 83.8% and All-Evidence Recall@5 from 72.9% to 75.9%, which suggests that reconstruction is more effective when guided by residual-aware group relevance rather than static similarity alone. Finally, budgeted MaxSim coverage yields the best overall performance, improving NDCG@5 retention to 94.6%, All-Evidence Recall@5 to 78.4%, and QA accuracy to 66.0%. This indicates that selecting complementary query-specific vectors under a fixed budget is important for preserving complete evidence and supporting downstream reasoning. Overall, the results show that reconstruction, residual-aware activation, and coverage provide complementary benefits.

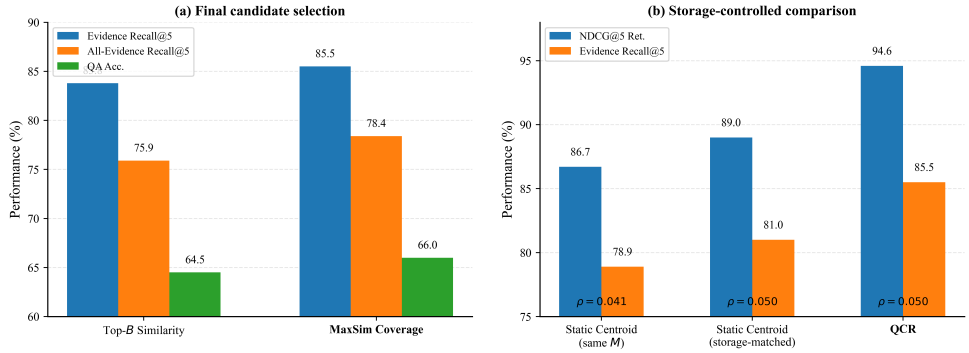


Figure 7: Ablation and storage-controlled analysis. **(a)** Comparison of final candidate-selection strategies under the same candidate pool and representation budget $B = M$. **(b)** Storage-controlled comparison between static centroid compression and QCR, where all residual-memory components are included in the reported storage.

Figure 7(a) isolates the effect of the final selection objective. Under the same candidate pool and representation budget, MaxSim coverage improves Evidence Recall@5 from 83.8% to 85.5%, All-Evidence Recall@5 from 75.9% to 78.4%, and QA accuracy from 64.5% to 66.0%, indicating that complementary query-token coverage is more effective than independently selecting the highest-scoring candidates. Figure 7(b) further controls for index capacity. Increasing the number of static centroids to match QCR's storage budget improves NDCG@5 retention from 86.7% to 89.0%, but remains substantially below QCR at 94.6%. This confirms that the gain cannot be explained solely by additional storage capacity; preserving reconstructable within-group variation provides a more effective use of the same index budget.

A.6 Efficiency and Scalability Analysis

Theoretical Efficiency and Overhead Analysis. QCR introduces additional storage and computation compared with purely static compression because it preserves residual information during indexing and performs query-conditioned adaptation at retrieval time. We analyze this overhead from three perspectives: index storage, offline index construction, and online retrieval. QCR is applied independently to the compact memory of each page; its efficiency therefore comes from low-rank residual modeling, selective reconstruction of only a few query-relevant groups, and a bounded final interaction budget.

Index storage. For a page containing N original embeddings of dimension d, the Full Index stores Nd scalar values. QCR stores M group centers, a shared rank-r residual basis, M group-specific residual statistics, and M trust radii, resulting in

$$S_{\mathrm{QCR}} = Md + dr + Mr + M, \qquad S_{\mathrm{Full}} = Nd. \tag{27}$$

The residual memory is explicitly included within the prescribed storage budget rather than added on top of an equal number of compressed representatives. For each target storage ratio, M is determined such that the complete QCR index satisfies the same byte-level budget as the competing methods. Since $r \ll d$ and $M \ll N$, the additional residual memory remains substantially smaller than storing the original full multi-vector representation.

Offline index construction. Beyond the base grouping operation, QCR computes within-group residuals, extracts a truncated low-rank basis, and estimates group-specific residual statistics and trust radii. These operations are performed only once during indexing and are independent of future queries. The resulting residual memory is reused across all subsequent queries, and no model training or parameter updates are required. QCR therefore incurs additional offline cost over centroid-only compression, but this one-time cost is amortized over repeated retrieval.

Online retrieval. Let L denote the number of query-token embeddings. Residual-aware activation evaluates the M groups of each page, after which explicit reconstruction is restricted to the Top-K groups for each query token. Thus, at most LK query–group pairs are reconstructed rather than all LM pairs. Together with the M static representatives, the resulting candidate pool contains at most $C \leq M + LK$ vectors. Budgeted MaxSim coverage then selects at most $B = M$ vectors, ensuring that the final interaction budget matches that of the underlying compressed representation. Query–candidate similarities are computed once and reused during greedy selection.

Ignoring lower-order operations, the dominant query-time cost per page is

$$\mathcal{O}(LM(d+r)) + \mathcal{O}(LKdr) + \mathcal{O}(LCd) + \mathcal{O}(MLC), \qquad C \leq M + LK, \tag{28}$$

corresponding to residual-aware group activation, selective reconstruction, query–candidate similarity computation, and budgeted coverage, respectively. In contrast, direct late interaction with the Full Index requires $\mathcal{O}(LNd)$ similarity computation. Since $M \ll N$, $K \ll M$, and $r \ll d$ in our operating regime, QCR replaces interaction with a large set of original page vectors by a controlled combination of low-rank computation and selective high-dimensional reconstruction. Moreover, once the query–candidate similarity matrix is available, the final MaxSim score can be obtained from the selected candidates without additional high-dimensional similarity computation.

Appendix Efficiency Experiment. We evaluate QCR from both indexing and query-time perspectives. All compressed methods are compared under the same total index-storage budget, where the auxiliary residual basis, group-wise residual statistics, and trust radii of QCR are explicitly included in the reported storage. Latency is measured on a single NVIDIA A800 GPU with batch size 1 and FP16 inference. We perform 20 warm-up runs followed by 200 measured runs and report the average latency. Offline latency includes document encoding and index construction, while online latency includes all operations required to produce the final retrieval ranking.

Table 9 shows that QCR introduces moderate additional online computation compared with static compression methods. Its query latency is 10.2 ms, compared with 8.0 ms for Sculpting, corresponding to a 27.5% increase. Nevertheless, QCR remains approximately $4.4\times$ faster than direct interaction with the Full Index. Under the same 0.05 storage budget, QCR improves NDCG@5 retention from 93.0% to 94.6% and Recall@5 from 86.1% to 88.9% over Sculpting. These results demonstrate that the additional query-time computation provides higher retrieval fidelity without increasing the prescribed index-storage budget.

Table 9: Efficiency comparison on ViDoRe v1 with ColQwen3-4B at a storage ratio of 0.05. Online latency is measured on documents containing approximately 200 pages.

Method	Storage	Offline Lat.	Online Lat.	NDCG@5 Ret. (%)	Recall@5 (%)
Full Index	1.00×	0.47 s	44.8 ms	100.0	95.1
SAP	0.05×	0.50 s	6.8 ms	86.4	80.1
Visual-Cues	0.05×	0.52 s	7.0 ms	75.8	78.6
Semantic Clustering	0.05×	0.56 s	7.3 ms	91.5	82.8
AnchorFold	0.05×	0.61 s	7.6 ms	91.6	85.0
Sculpting	0.05×	0.70 s	8.0 ms	<u>93.0</u>	<u>86.1</u>
Ours	0.05×	0.64 s	10.2 ms	**94.6**	**88.9**

To identify the source of the additional online cost, we separately measure residual-aware group activation, selective reconstruction, budgeted coverage, and final MaxSim scoring. As shown in Table 10a, query-conditioned operations account for 4.3 ms of the total 10.2 ms latency, while final MaxSim scoring requires 5.9 ms. Selective reconstruction itself contributes only 1.4 ms because it is restricted to the Top-K activated groups for each query token rather than all M stored groups. The greedy coverage procedure operates on the precomputed query–candidate similarity matrix and contributes 1.1 ms.

(a) Query-time latency breakdown of QCR on ViDoRe v1 with ColQwen3-4B.

Component	Latency (ms)
Residual-aware group activation	1.8
Selective reconstruction	1.4
Budgeted MaxSim coverage	1.1
Final MaxSim scoring	5.9
Total	10.2

(b) Query-time latency (ms) under different document lengths.

Method	50 Pages	100 Pages	200 Pages	500 Pages
Full Index	13.0	24.5	44.8	109.6
AnchorFold	2.9	4.7	7.6	17.5
Sculpting	3.1	5.0	8.0	18.3
Ours	3.9	6.2	10.2	23.7

Table 10: Query-time latency breakdown and document-length scaling on ViDoRe v1 with ColQwen3-4B.

Since QCR is applied independently to the compact memory of each page, we further examine how query latency scales with the number of pages in the input document. Table 10b shows that query latency increases approximately linearly with document length for all methods. At 200 pages, the reported latencies exactly correspond to those in Table 9: 44.8 ms for the Full Index, 7.6 ms for AnchorFold, 8.0 ms for Sculpting, and 10.2 ms for QCR. For a 500-page document, QCR requires 23.7 ms compared with 109.6 ms for the Full Index, while retaining query-conditioned adaptation.

Overall, QCR is not designed to minimize query latency at all costs. Static compression naturally incurs lower online overhead because its representations require no query-dependent adaptation. Instead, QCR targets an efficiency–fidelity trade-off in which a small portion of the storage budget is allocated to reconstructable residual variation and a controlled amount of online computation is used to instantiate query-specific representations. The resulting index remains substantially smaller and faster to search than the Full Index while retaining higher retrieval fidelity under aggressive compression.

A.7 Qualitative Analysis

We provide two additional qualitative examples that complement the case study presented in the main paper. While the main-text example focuses on preserving multiple query-relevant regions under aggressive compression, the following cases involve more fine-grained evidence localization in visually dense document layouts.

The qualitative example in the main paper shows that QCR better preserves complementary matching signals when a query requires information from multiple semantic regions of the same page. Figures 8 and 9 extend this analysis to two more challenging cases, where the relevant evidence is highly localized and surrounded by visually competing content.

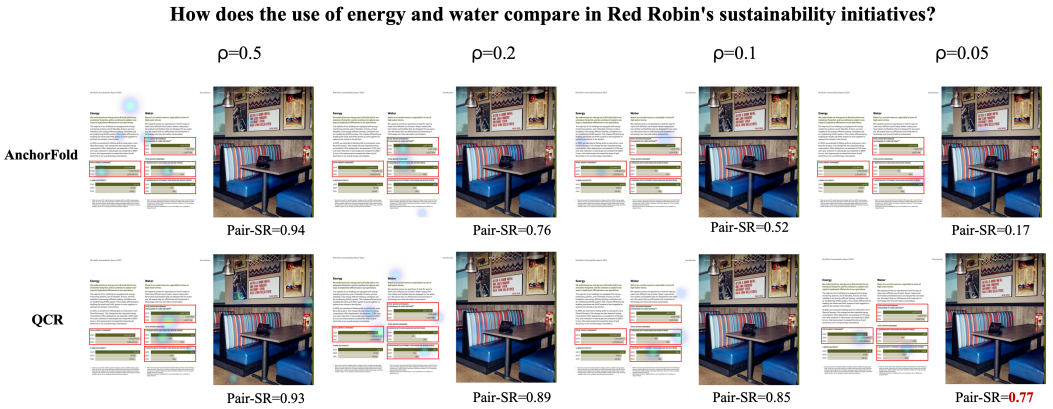


Figure 8: Qualitative comparison between AnchorFold and QCR for the query "How does the use of energy and water compare in Red Robin's sustainability initiatives? " under different storage ratios. The relevant evidence is embedded in a visually dense infographic containing dense **tabular layout** and textual elements.

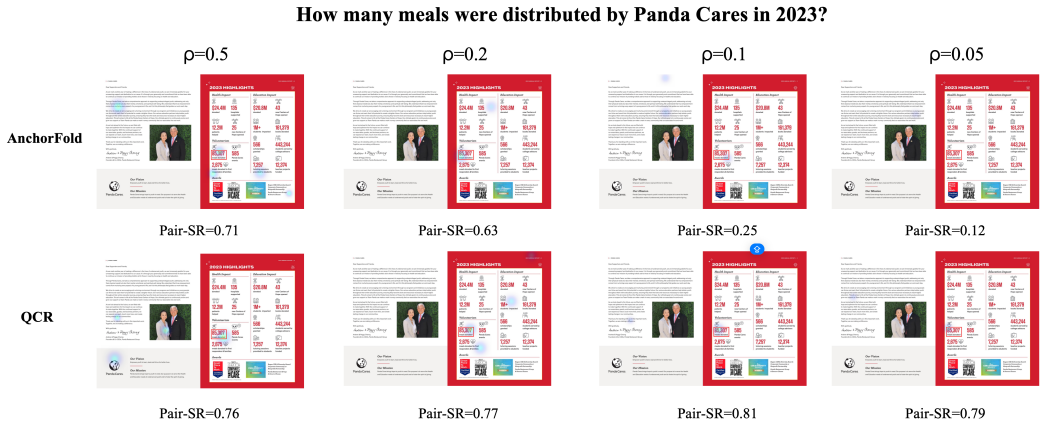


Figure 9: Qualitative comparison between AnchorFold and QCR for the query "How many meals were distributed by Panda Cares in 2023?" under different storage ratios. The relevant evidence is embedded in a visually dense infographic containing many competing **image** and textual elements.

As shown in Fig. 8, answering the query requires locating multiple precise numerical changes from a visually dense document and combining evidence from several local regions. Unlike coarse semantic matching, such queries depend on preserving multiple fine-grained regions simultaneously. When a page is represented by only a small number of fixed vectors, some of these localized signals can be attenuated during compression. As the storage ratio decreases, AnchorFold exhibits noticeably weaker responses over the distributed numerical evidence, whereas QCR continues to preserve strong matching signals across multiple relevant regions. This suggests that query-conditioned reconstruction can selectively activate local variation retained in the residual memory and better maintain complementary evidence under aggressive compression. As shown in Fig. 9, the second case presents an even more visually complex setting. The target value is embedded in a dense infographic containing many nearby statistics with similar visual structures, requiring the retriever not only to localize the relevant region but also to distinguish it from several visually and semantically competing numerical cues. Under such conditions, static compression is more likely to merge fine-grained distinctions into a limited set of fixed representatives, weakening the signal associated with the realized query. In contrast, QCR maintains more stable query-relevant responses across different storage ratios, indicating that its benefit extends beyond preserving localized evidence to emphasizing the semantic directions most relevant to the current query among visually similar alternatives. Together, these examples further demonstrate that query-adaptive representations provide

more robust fine-grained matching as evidence becomes more distributed and page layouts become more visually complex.