GLOBAL RELEVANCE IS NOT ENOUGH: REPRESENTATIVE CANDIDATE CONSTRUCTION AND RESIDUAL-BASED EVIDENCE PRESERVATION FOR EFFICIENT, RELIABLE MEDICAL MLLM INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Medical Multimodal Large Language Models (MLLMs) have shown great potential in clinical visual question answering and medical image interpretation, but their high inference cost limits efficient deployment in real-world clinical settings. Moreover, in safety-critical medical scenarios, pursuing efficiency alone is far from sufficient. We observe that existing acceleration methods, although able to preserve aggregate performance, can still induce substantial instance-level prediction shifts. To address this issue, we propose MedR2R (Representative-to-Residual Compression), a representative candidate construction and residual-based evidence preservation framework for efficient and reliable medical MLLM inference. Rather than determining token importance from global relevance alone, MedR2R adopts a two-stage coarse-to-fine selection strategy. First, after the vision encoder, it constructs a semantically representative candidate set under a relaxed intermediate token budget. Specifically, MedR2R models pairwise visual-token relationships using cosine similarity and combines attention-derived global relevance with marginal representation gain, encouraging the candidate set to represent the overall visual space rather than only highly attended regions. Second, after the visual tokens are contextualized by an intermediate language-model layer, MedR2R further refines the candidate set according to normalized residuals. Tokens that can be accurately approximated from a compact basis are treated as removable, whereas tokens with large normalized residuals are preserved as hard-to-replace visual evidence. Experiments across multiple medical VQA datasets and MLLMs demonstrate that, even under substantially reduced visual-token budgets, this coarse-to-fine two-stage selection strategy maintains competitive task performance while effectively mitigating harmful instance-level prediction shifts. These results further indicate that medical MLLM acceleration should not focus solely on aggregate performance, but should also account for the stability of individual predictions. Our code is available at: `https://anonymous.4open.science/r/MedR2R`.

1 INTRODUCTION

Medical Multimodal Large Language Models (MLLMs) have shown great potential in clinical visual question answering, medical image interpretation, and interactive diagnostic assistance (Li et al., 2023a). However, deploying medical MLLMs in real-world clinical environments faces numerous challenges (Li et al., 2026a). One major obstacle is inference efficiency: medical images are typically high-resolution and multi-view, and their visual encoders often generate a large number of patch-level visual tokens (Dosovitskiy et al., 2020; Chen et al., 2024a). When these visual tokens are fed into the language model together with textual instructions, they increase computational overhead.

Therefore, visual token reduction has become an important direction for accelerating MLLM inference. Existing methods typically reduce computational cost by pruning or merging low-information tokens, commonly selecting top-ranked tokens based on attention-derived global relevance (Chen et al., 2024a; Zhang et al., 2024; Xing et al., 2025). Although these methods can reduce compu-

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.

tation while maintaining overall performance, their selection criteria often rely heavily on global relevance. As a result, they tend to retain tokens in highly attended regions while discarding low-attention yet semantically important contextual information. This may weaken the model's ability to understand surrounding structures, spatial relationships, and subtle visual evidence.

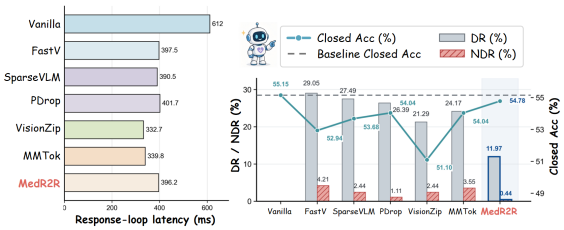


Figure 1: Performance–efficiency comparison. (Left) Measured response-loop latency on LLaVA-Med (Li et al., 2023a). (Right) Instance-level stability on VQA-RAD at 128 retained tokens: similar aggregate accuracy can conceal substantial changes in the individual answers produced after acceleration.

More importantly, maintaining aggregate performance does not necessarily imply stable predictions on individual medical cases. As illustrated in Fig. 1, under the same 128-token budget on VQA-RAD (Lau et al., 2018), several acceleration methods achieve closed-ended accuracy close to that of the vanilla model while still inducing instance-level prediction shifts. For example, the vanilla model achieves a closed-ended accuracy of 55.15%, whereas MMTok (Dong et al., 2026) attains a comparable accuracy of 54.04% but exhibits a Divergence Ratio (DR) of 24.17% and a Negative Divergence Ratio (NDR) of 3.55%. DR measures how frequently the accelerated model produces a prediction different from that of the vanilla model on the same input, while NDR measures the fraction of all evaluated cases in which a correct vanilla prediction becomes incorrect after acceleration (Sun et al., 2025). These results suggest that, in safety-critical medical applications, preserving accuracy alone is insufficient: efficient medical MLLMs must also account for instance-level prediction consistency and minimize harmful negative deviations that may introduce clinical risks.

In medical MLLMs, this limitation is critical. Medical reasoning often depends on small lesions, low-contrast abnormalities, and anatomical boundaries across the image (Nguyen et al., 2025; Fang & Liu, 2025; Wang et al., 2025; Li et al., 2026b). These cues may receive low attention yet remain essential for clinical questions. For example, subtle opacities, vessel boundaries, or peripheral anatomical regions can be diagnostically decisive (Luo et al., 2026). Discarding them may not affect average benchmark accuracy, but can alter individual predictions (Wen et al., 2025).

These observations indicate that reliability, rather than efficiency alone, is the defining challenge of medical MLLM acceleration: an accelerated model must preserve not only aggregate accuracy but also the case-level evidence underlying individual clinical predictions. Every pruned token is therefore a safety decision made without seeing its consequences. We therefore ask: *what makes a dropped visual token safe in medical MLLM inference?*

We study this question with existing acceleration methods on medical VQA benchmarks and identify three recurring limitations. **First**, global-relevance-driven selection concentrates the retained tokens on highly attended regions, so contextual structures, spatial relationships, and low-attention yet clinically decisive evidence may be discarded, reducing how broadly the complete visual scene is represented. **Second**, low global relevance does not necessarily imply safe removal (Chen et al., 2025; 2024b): tokens with large residuals may contain distinctive information, and discarding them can alter individual predictions even when aggregate accuracy remains stable. Moreover, token redundancy can change after contextualization by the language model, motivating further refinement beyond encoder-level selection. **Third**, aggregate metrics alone cannot reveal such changes: as shown in Fig. 1(b), similar overall accuracy can conceal frequent instance-level prediction shifts.

To this end, we propose **MedR2R** (Representative-to-Residual Compression), a two-stage representative candidate construction and residual-based evidence preservation framework for efficient and reliable medical MLLM inference (Fig. 2). Rather than treating token compression as a single importance ranking, MedR2R separates it into two sequential decisions at different representation levels. The encoder-side decision combines attention-derived global relevance with marginal representation gain to construct a representative candidate set under a relaxed intermediate token budget. The LLM-side decision then estimates normalized residuals for the candidate tokens after they are contextualized by an intermediate language-model layer. It compresses the candidate set by retaining the basis tokens together with tokens exhibiting large normalized residuals, while removing those that can be accurately approximated by the basis. This division assigns candidate recall to the en-

coder side, while residuals after contextualization guide final evidence retention. Our experiments jointly report task quality, measured efficiency, and instance-level stability, making the trade-offs among acceleration methods explicit rather than inferring reliability from aggregate accuracy alone. Our main contributions are summarized as follows:

- We formulate visual token compression as two sequential decisions at different representation levels: encoder-side candidate recall and LLM-side evidence retention. The first decision constructs an over-complete candidate pool under a relaxed budget; the second makes the final retention decision after contextualization.
- We use a marginal representation gain criterion based on pairwise visual token similarity for encoder-side candidate recall. Combined with attention-derived global relevance, it keeps the recalled pool from collapsing onto a few highly attended regions.
- We design a residual-based evidence retention mechanism at an intermediate language-model layer. By linearly approximating contextualized tokens with a compact basis, MedR2R characterizes irreplaceability using normalized residuals and retains basis and high-residual tokens, preserving hard-to-replace visual evidence within the candidate set.
- Experiments across multiple medical VQA datasets and MLLMs demonstrate that, compared with existing acceleration strategies, MedR2R reduces inference cost, preserves task performance, and mitigates instance-level prediction shifts.

2 Related Work

Medical Multimodal Large Language Models. Recent advances in MLLMs have enabled open-ended visual understanding by connecting visual encoders with large language models through projection or fusion modules (Liu et al., 2024; Li et al., 2023b; Bai et al., 2023; 2025b). Following this paradigm, medical MLLMs have been developed for clinical visual question answering, medical image interpretation, and interactive diagnostic assistance. Models such as LLaVA-Med (Li et al., 2023a) and MedVLM-R1 (Pan et al., 2025) adapt general-purpose MLLMs to the biomedical domain through medical image-text alignment and instruction tuning, showing promising performance across chest X-ray, CT, MRI, and pathology. However, medical MLLMs face high inference costs. Medical images are high-resolution, multi-view, and semantically dense (Zhang et al., 2025a), causing visual encoders to generate long patch-level token sequences. When fed into the language model with textual instructions, these tokens increase prefilling cost, memory consumption, and inference latency (Chen et al., 2024c). Therefore, effective and reliable visual token reduction is needed to accelerate medical vision-language inference.

Visual Token Selection in MLLMs. Many studies have focused on reducing visual token redundancy in MLLMs without requiring model training (Shang et al., 2025; Alvar et al., 2025; Wen et al., 2024; Kim et al., 2026). Most of these methods rely on attention scores to rank tokens by global relevance, such as text-to-vision attention in LLMs or CLS-token attention in vision transformers. They typically adopt a top-k strategy, retaining the top-ranked tokens, i.e., those with the highest attention scores. For example, FastV (Chen et al., 2024a) leverages early-layer text-to-vision attention to retain high-relevance tokens. SparseVLM (Zhang et al., 2024) uses important textual words as raters to guide token selection. PDrop (Xing et al., 2025) reduces visual redundancy and accelerates MLLM inference by dropping visual tokens. VisionZip (Yang et al., 2025) applies CLS-based attention in the vision transformer for token pruning. MMTok (Dong et al., 2026) formulates visual token selection by maximizing a submodular function over visual and textual information, for which an efficient algorithm can obtain a near-optimal solution with a theoretical guarantee. In contrast, our method adopts a cross-level coarse-to-fine compression strategy that separates candidate construction from final evidence retention. This design aims to preserve representative visual information while improving instance-level prediction stability.

3 Methods

As shown in Fig. 2, MedR2R reframes visual token compression as two sequential decisions made at different representation levels: encoder-side candidate recall, which determines which tokens enter the language model, and LLM-side evidence retention, which determines which contextualized representations remain for subsequent inference. The first decision constructs a broadly representative

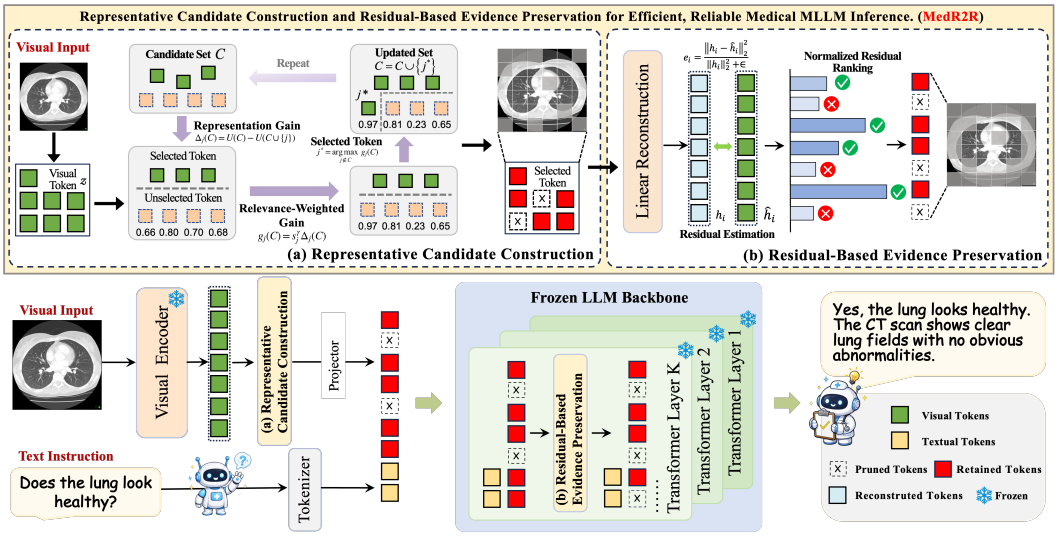


Figure 2: Overview of MedR2R. (a) The encoder-side candidate-recall module constructs a representative candidate set by jointly considering global token relevance and marginal representation gain. (b) The LLM-side evidence-retention module retains hard-to-replace tokens based on their normalized residuals, yielding visual tokens for efficient and reliable medical MLLM inference.

candidate set using global relevance and marginal representation gain. The second decision refines this set using residual-based evidence retention, reserving the final budget for a compact basis and tokens that the basis cannot readily approximate.

3.1 Two-Stage Inference Formulation

Given an image I and a textual instruction T, let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times d_v}$ denote the patch representations available at the vision-to-language interface. We use m for the intermediate candidate budget, k for the final budget, and ℓ_* for the language-model layer after which refinement occurs, with $1 \le k < m \le n$. An index set $\mathcal{C} \subseteq [n]$, where $[n] = \{1, \ldots, n\}$ and $|\mathcal{C}| = m$, identifies the first-stage candidates. Given a target token budget $k < n$, we set the intermediate candidate budget to $m = \min(n, 2k)$ by default. k specifies the desired final visual-token count, while m follows from a fixed expansion factor instead of being tuned independently for each input. The larger pool provides candidates for subsequent residual-based refinement.

MedR2R first selects $X_{\mathcal{C}}$ using marginal representation gain and global relevance. The model projects these candidates into the language-model embedding space and processes them together with T through the first ℓ_* layers. Their visual hidden states form $\mathbf{H} \in \mathbb{R}^{m \times d_\ell}$. Residual-based evidence preservation and token consolidation then produce $\widetilde{\mathbf{H}} \in \mathbb{R}^{k \times d_\ell}$, which the remaining layers process together with the retained textual hidden states. Compression thus produces merged hidden representations associated with retained token positions. The aim is to reduce computation while limiting changes to individual predictions. MedR2R performs selection from the input's features with frozen model parameters, it does not optimize an output-discrepancy loss or require a second forward pass at inference time.

3.2 Representative Candidate Construction

Encoder-side candidate recall. The first stage allocates a relaxed budget before committing to the final token count. A fixed global-relevance ranking cannot account for what is already represented by the selected tokens. MedR2R scores each new candidate by combining its global relevance with the additional representation it brings to the current pool, reserving the final retention decision for refinement inside the language model.

Global relevance and token relationships. For a vision encoder with a global token, let $a_i = \sum_{h=1}^{H_v} A_{g,i}^{(h)}$ be the attention received by patch i from global-token position g, summed over H_v heads at the feature-extraction layer. We rescale these values to obtain global-relevance weights s_i:

when $a_{\max} - a_{\min} > \epsilon$, $s_i = \max\{\epsilon, (a_i - a_{\min})/(a_{\max} - a_{\min})\}$; otherwise all weights are set to one. For backbones without a global token, the implementation uses a global-feature proxy: with $\mathbf{p}_i = \mathbf{x}_i / \max(\|\mathbf{x}_i\|_2, \epsilon)$ and $\bar{\mathbf{p}} = n^{-1} \sum_i \mathbf{p}_i$, it sets $a_i = (1 + \mathbf{p}_i^\top \bar{\mathbf{p}} / \max(\|\bar{\mathbf{p}}\|_2, \epsilon))/2$ before the same rescaling. We measure token affinity with nonnegative cosine similarity:

$$w_{ij} = \max\{0, \mathbf{p}_i^\top \mathbf{p}_j\}, \qquad c_i(\mathcal{C}) = \max_{j \in \mathcal{C}} w_{ij}, \qquad c_i(\emptyset) = 0. \tag{1}$$

Here $c_i(\mathcal{C})$ records how well the current candidates represent token i. Clipping negative similarities makes the zero initialization well defined and agrees with the implementation.

Selection through uncovered representation. We express the candidate-selection criterion in terms of the representation still uncovered:

$$\mathcal{U}(\mathcal{C}) = \sum_{i=1}^{n} \big[1 - c_i(\mathcal{C})\big]. \tag{2}$$

Adding a token is useful when it reduces this quantity. For an unselected index j, the global-relevance-weighted reduction is:

$$g_j(\mathcal{C}) = s_j^\gamma \big[\mathcal{U}(\mathcal{C}) - \mathcal{U}(\mathcal{C} \cup \{j\})\big] = s_j^\gamma \sum_{i=1}^{n} \big[w_{ij} - c_i(\mathcal{C})\big]_+, \tag{3}$$

where $[u]_+ = \max(u, 0)$ and $\gamma \geq 0$ controls the global-relevance weighting. Starting with $\mathcal{C} = \emptyset$, we repeatedly choose $j_* = \arg\max_{j \notin \mathcal{C}} g_j(\mathcal{C})$, add j_* to $\mathcal{C}$, and update $c_i \leftarrow \max(c_i, w_{ij_*})$ until $|\mathcal{C}| = m$. The scoring function jointly evaluates each candidate token's global relevance and the information already represented by the current candidate pool. The score is dynamically updated as the selected set changes: when tokens similar to a given token are already well represented by the current candidate set, that token provides only a small additional representation gain.

3.3 Residual-Based Evidence Preservation

LLM-side evidence retention. Let $\mathbf{h}_i$ be row i of $\mathbf{H}$, indexed within the ordered candidate sequence. MedR2R approximates these hidden states from a compact set of reference tokens and uses the normalized residuals to identify hard-to-replace evidence. This criterion operates on the representative candidate pool and determines which candidates survive the final budget.

Constructing a reference basis. Let $\mathcal{A} \subseteq [m]$ denote the basis indices, with $|\mathcal{A}| = r$ and $1 \leq r \leq k$. The implementation selects them deterministically by cosine farthest-point sampling. It initializes $\mathcal{A} = \{\arg\max_i \|\mathbf{h}_i\|_2\}$ and repeatedly adds

$$i_* = \arg\min_{i \notin \mathcal{A}} \max_{a \in \mathcal{A}} \cos(\mathbf{h}_i, \mathbf{h}_a) \tag{4}$$

until r indices have been selected. The resulting matrix $\mathbf{D} = \mathbf{H}_{\mathcal{A}} \in \mathbb{R}^{r \times d_\ell}$ is a reconstruction dictionary, its rows need not be linearly independent.

Measuring residual-based evidence retention. We fit all candidates to this dictionary using regularized least squares:

$$\mathbf{W}_* = \arg\min_{\mathbf{W} \in \mathbb{R}^{m \times r}} \|\mathbf{H} - \mathbf{W}\mathbf{D}\|_F^2 + \lambda \|\mathbf{W}\|_F^2, \qquad \widehat{\mathbf{H}} = \mathbf{W}_* \mathbf{D}, \tag{5}$$

where $\lambda > 0$. Equivalently,

$$(\mathbf{D}\mathbf{D}^\top + \lambda \mathbf{I}_r)\mathbf{W}_*^\top = \mathbf{D}\mathbf{H}^\top. \tag{6}$$

Thus, residual fitting requires solving an $r \times r$ system. The implementation performs the solve in FP32, with a pseudoinverse fallback for numerical failures.

We rank nonbasis candidates by their normalized squared residual:

$$e_i = \frac{\|\mathbf{h}_i - \widehat{\mathbf{h}}_i\|_2^2}{\|\mathbf{h}_i\|_2^2 + \epsilon}. \tag{7}$$

The denominator normalizes each residual by the token's feature energy, reducing the direct influence of feature magnitude on the ranking. A large e_i identifies a token poorly approximated by the current dictionary. This score measures linear non-redundancy, not diagnostic importance or a guarantee of answer correctness.

Algorithm 1 MedR2R Visual Evidence Preservation Pipeline

Require: Visual features $\mathbf{X} \in \mathbb{R}^{n \times d_v}$, instruction T; budgets m, k, affinities w_{ij}, global-relevance scores s_i; basis size r, layer ℓ_*, global-relevance exponent γ, regularizer λ, stabilizer ϵ.
Ensure: Compressed visual states $\widetilde{\mathbf{H}} \in \mathbb{R}^{k \times d_\ell}$ and retained indices $\mathcal{R}$

Initialize $\mathcal{C} \leftarrow \emptyset$ ▷ Start with an empty candidate set
Initialize coverage scores: $c_i \leftarrow 0$ for all $i \in [n]$ ▷ Track current coverage
// Representative candidate construction after the vision encoder
for $t = 1$ to m **do**
  for all $j \in [n] \setminus \mathcal{C}$ **do**
    Compute marginal representation gain: $g_j(\mathcal{C}) = s_j^\gamma \sum_{i=1}^{n} [w_{ij} - c_i]_+$
  end for
  Select next token: $j_\star \in \arg\max_{j \notin \mathcal{C}} g_j(\mathcal{C})$ ▷ Balance representation and global relevance
  Update candidate set: $\mathcal{C} \leftarrow \mathcal{C} \cup \{j_*\}$
  Update coverage scores: $c_i \leftarrow \max(c_i, w_{ij_*}), \forall i \in [n]$
end for
// Residual-based evidence preservation after layer ℓ_*
Feed ordered, projected $\mathbf{X}_\mathcal{C}$ with T through layer ℓ_* to obtain $\mathbf{H}$
Select basis indices $\mathcal{A}$ by farthest-point sampling; set $\mathbf{D} \leftarrow \mathbf{H}_\mathcal{A}$
Solve $(\mathbf{D}\mathbf{D}^\top + \lambda \mathbf{I}_r)\mathbf{W}_*^\top = \mathbf{D}\mathbf{H}^\top$ ▷ Regularized reconstruction
for all $i \in [m]$ **do**
  Linearly reconstruct $\mathbf{h}_i$ by $\mathbf{D}$: $\widehat{\mathbf{h}}_i = (\mathbf{W}_*)_{i,:}\mathbf{D}$
  Compute normalized residual: $e_i = \|\mathbf{h}_i - \widehat{\mathbf{h}}_i\|_2^2 / (\|\mathbf{h}_i\|_2^2 + \epsilon)$ ▷ Rank normalized residuals
end for
Select retained indices: $\mathcal{R} = \mathcal{A} \cup \mathrm{TopIdx}_{k-r}(\{(i, e_i) : i \in [m] \setminus \mathcal{A}\})$ ▷ Keep basis and high residuals
Assign each removed candidate to its most similar representative in $\mathcal{R}$
Average each group, including its representative, to obtain $\widetilde{\mathbf{H}}$
Restore representative order; retain textual states and update sequence metadata
return $\widetilde{\mathbf{H}}, \mathcal{R}$

Allocating the final budget. We retain all basis indices and use the remaining $k - r$ slots for the largest residuals:

$$\mathcal{R} = \mathcal{A} \cup \mathrm{TopIdx}_{k-r}\big(\{(i, e_i) : i \in [m] \setminus \mathcal{A}\}\big), \qquad |\mathcal{R}| = k, \tag{8}$$

where TopIdx returns token indices, not residual values. The basis provides reference representations, while the additional tokens retain directions that the basis approximates poorly.

3.4 Token Consolidation and Inference

After selection, MedR2R consolidates the discarded candidates into retained representatives. Each retained index is assigned to itself, each other candidate is assigned to its most similar retained hidden state:

$$\pi(i) = \begin{cases} i, & i \in \mathcal{R}, \\ \arg\max_{j \in \mathcal{R}} \cos(\mathbf{h}_i, \mathbf{h}_j), & i \notin \mathcal{R}. \end{cases} \tag{9}$$

Writing $\mathcal{G}_j = \{i : \pi(i) = j\}$, we form:

$$\widetilde{\mathbf{h}}_j = \frac{1}{|\mathcal{G}_j|} \sum_{i \in \mathcal{G}_j} \mathbf{h}_i, \qquad j \in \mathcal{R}. \tag{10}$$

The representative itself participates in the average, so every group is nonempty. This step carries information from the removed candidates into the compressed sequence, although averaging can also attenuate distinctive features. Its effect is evaluated separately from residual-based selection. The compressed visual states follow the original order of their representative indices. Textual states are kept, and the model adapter updates position information, attention masks, and populated cache entries consistently with the shorter sequence. Selection is applied during prefilling, generation then continues with the compressed state. Algorithm 1 summarizes the complete procedure.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and Evaluation Metrics. We evaluate MedR2R on five medical visual question answering benchmarks: SLAKE (Liu et al., 2021), VQA-RAD (Lau et al., 2018), VQA-Med (Ben Abacha et al., 2019), PathVQA (He et al., 2020), and PMC-VQA (Zhang et al., 2023). For performance, we report accuracy for closed-ended and multiple-choice questions, and token Recall and ROUGE-1 for open-ended questions (Lin, 2004; Xu et al., 2026). We additionally report two instance-level output-stability measures. Let a_i^0 and a_i^m be the answers from the vanilla model and acceleration method m, and let y_i be the reference answer. Before scoring, $\mathrm{Norm}(\cdot)$ lowercases the answer, replaces punctuation with spaces, removes the articles *a*, *an*, and *the*, and collapses whitespace. We define the DR as:

$$\mathrm{DR}(m) = \frac{1}{M}\sum_{i=1}^{M} \mathbf{1}\left[\mathrm{Norm}(a_i^m) \neq \mathrm{Norm}(a_i^0)\right], \tag{11}$$

which measures how often the normalized full response changes relative to the vanilla model. DR is deliberately reference-free and strict: on open-ended questions it measures normalized surface-form stability, not semantic equivalence. Thus, semantically equivalent paraphrases that remain different after normalization are counted as changes. For the NDR, $\mathrm{Correct}_i(a, y)$ is defined by question type: Closed yes/no answers are correct when the first normalized answer token matches the canonical label; multiple-choice answers are correct when the parsed option in $\{\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}\}$ matches the reference option, with a unique normalized option-text match used as fallback; and open-ended answers use normalized exact match to the dataset reference. We then compute:

$$\mathrm{NDR}(m) = \frac{1}{M}\sum_{i=1}^{M} \left\{\mathrm{Correct}_i(a_i^0, y_i) = 1 \wedge \mathrm{Correct}_i(a_i^m, y_i) = 0\right\}. \tag{12}$$

NDR therefore counts strict correct-to-incorrect regressions. For both metrics, M is the total number of evaluated questions in the benchmark, including closed-ended, open-ended, and multiple-choice questions; NDR is not conditioned on the subset answered correctly by the vanilla model. Normalized exact match is used only to make the open-ended binary transition reproducible, while Recall and Rouge remain the primary measures of open-ended answer quality.

Implementation Details. We use LLaVA-Med v1.5 (Li et al., 2023a), MedVLM-R1 (Pan et al., 2025), and Qwen3-VL-8B-Instruct (Bai et al., 2025a) as backbone MLLMs. MedR2R is applied in a training-free manner with all model parameters frozen and follows the official inference configuration of each backbone. MedR2R compresses visual tokens in two stages. Given a final budget k, the encoder-side stage constructs an intermediate candidate set $\mathcal{C}$ of size $m = \min(n, 2k)$ using global-relevance-weighted marginal representation gain, with nonnegative cosine similarity for token affinity and $\gamma = 1.0$. Global relevance is obtained from global-token attention when available, or from a feature-derived proxy otherwise. The LLM-side stage further reduces the m candidates to k tokens using residual-based selection after layer $\ell_* = 16$ for LLaVA-series models and $\ell_* = 14$ for MedVLM-R1 and Qwen3-VL. Our implementation is based on lmms-evals (Zhang et al., 2025b). Main-table comparisons match backbones, data splits, prompts, decoding settings, token budgets, and batch-1 A100 hardware. Appendix A gives test-set sizes and metric denominators.

4.2 MAIN RESULTS

Task performance and transfer. Tables 1 and 2 show that MedR2R preserves performance across token budgets and backbones. On LLaVA-Med at 128 tokens, it leads baselines in 11 of 13 metrics; at 64 tokens, it matches the full-token VQA-RAD (55.15) and PMC-VQA (29.15) accuracies. On MedVLM-R1, it leads baselines in all 13 columns at both budgets and, with 128 tokens, retains 63.24/63.97 VQA-RAD and 56.27/56.47 PathVQA accuracy relative to full tokens. These results are consistent with preserving complementary evidence rather than only top-ranked patches. Appendix B further reports Qwen3-VL-8B-Instruct results, supporting the generalizability of MedR2R.

Qualitative token coverage. Fig. 3 illustrates token allocation under compression: relative to attention-only selection, MedR2R combines broader candidate coverage with residual-based refinement. Appendix E extends the examples across fundus, MRI, CT, radiography, and pathology. These examples show how MedR2R preserves representative visual tokens across medical modalities.

Table 1: Performance of LLaVA-Med with MedR2R on medical VQA benchmarks.

Method	SLAKE			VQA-RAD			VQA-Med			PathVQA			PMC-VQA
	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.
Full-token reference, 576 Tokens (100%)													
Vanilla	43.75	16.63	5.57	55.15	31.21	11.74	57.81	15.14	5.11	61.66	7.63	2.61	29.15
LLM.int8()	37.50	15.11	4.82	47.79	27.19	10.36	54.69	14.76	5.10	57.74	6.52	2.62	27.30
Retain 192 Tokens (↓ 66.7%)													
FastV (ECCV24)	39.18	15.15	4.72	52.94	26.82	8.06	51.56	**14.74**	4.16	60.28	7.52	2.11	20.65
SparseVLM (ICML25)	39.66	14.35	4.18	53.31	28.64	8.83	51.56	14.27	3.90	**61.57**	7.29	2.08	20.90
PDrop (CVPR25)	39.90	14.22	4.19	53.68	28.94	9.14	54.69	14.32	4.20	59.01	6.41	1.98	21.55
VisionZip (CVPR25)	40.62	14.78	4.06	53.68	27.78	10.76	53.12	14.64	4.52	59.25	6.93	**2.71**	23.45
MMTok (ICLR26)	**40.87**	15.75	5.01	53.31	27.52	9.80	56.25	14.68	4.97	59.63	6.56	2.60	28.75
MedR2R (Ours)	**40.87**	**15.81**	**5.06**	**54.41**	**30.10**	**10.77**	**57.81**	14.33	**5.01**	59.63	**7.63**	2.28	**29.00**
Retain 128 Tokens (↓ 77.8%)													
FastV (ECCV24)	38.46	14.41	4.47	52.94	24.80	7.44	51.56	13.80	3.74	60.19	6.75	1.77	21.40
SparseVLM (ICML25)	39.90	14.43	4.60	53.68	24.01	7.60	51.56	14.21	3.94	**61.49**	6.35	1.72	22.15
PDrop (CVPR25)	39.90	14.40	4.39	54.04	27.42	7.93	51.56	13.78	4.12	59.07	6.50	1.93	23.05
VisionZip (CVPR25)	40.62	14.71	4.48	51.10	27.64	10.49	54.69	14.11	4.62	59.48	6.57	**2.64**	24.25
MMTok (ICLR26)	40.14	15.40	4.84	54.04	29.45	11.08	54.69	14.29	4.83	60.16	6.48	2.59	28.95
MedR2R (Ours)	**40.87**	**15.93**	**5.10**	**54.78**	**29.48**	**11.34**	**56.25**	**15.14**	**4.93**	59.72	**7.42**	2.09	**29.15**
Retain 64 Tokens (↓ 88.9%)													
FastV (ECCV24)	38.22	15.67	4.82	51.47	26.36	8.48	42.19	11.36	3.52	59.86	5.54	1.62	21.20
SparseVLM (ICML25)	39.18	14.75	4.76	54.04	24.69	8.52	50.00	9.50	2.80	**61.57**	4.33	1.37	21.35
PDrop (CVPR25)	39.66	14.53	4.80	51.47	27.80	9.97	51.56	12.90	4.26	59.19	6.43	1.93	22.65
VisionZip (CVPR25)	40.87	16.05	5.02	54.41	29.57	11.01	53.12	13.31	4.57	60.45	5.59	2.33	28.20
MMTok (ICLR26)	39.66	16.20	**5.47**	53.31	30.45	11.74	51.56	13.31	**4.68**	60.63	**6.96**	2.38	27.40
MedR2R (Ours)	**41.11**	**16.63**	5.07	**55.15**	**31.10**	**12.46**	**56.25**	**14.48**	4.56	59.66	6.83	**2.57**	**29.15**

Table 2: Performance of MedVLM-R1 with MedR2R on medical VQA benchmarks.

Method	SLAKE			VQA-RAD			VQA-Med			PathVQA			PMC-VQA
	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.
Full-token reference, 576 Tokens (100%)													
Vanilla	53.85	17.64	18.02	63.97	23.29	23.84	65.62	21.33	21.79	56.47	4.48	4.64	47.25
LLM.int8()	52.40	16.19	16.40	61.03	20.84	20.97	51.56	17.20	17.65	56.56	3.60	3.83	42.10
Retain 192 Tokens (↓ 66.7%)													
FastV (ECCV24)	44.23	15.52	15.59	60.29	20.70	21.47	59.38	19.27	19.76	55.47	4.25	4.45	43.75
SparseVLM (ICML25)	43.03	14.28	14.14	40.07	15.14	14.94	40.62	9.83	10.01	52.82	3.96	4.18	43.25
PDrop (CVPR25)	46.39	15.08	15.24	61.40	21.16	22.38	57.81	19.54	19.98	55.35	4.11	4.24	45.45
VisionZip (CVPR25)	47.60	13.86	14.29	61.03	21.00	21.76	54.69	19.72	20.25	55.59	4.33	4.56	45.00
MMTok (ICLR26)	47.12	16.81	17.33	58.82	21.16	22.21	60.94	19.94	20.20	54.73	4.37	4.55	46.15
MedR2R (Ours)	**53.37**	**17.51**	**17.91**	**62.50**	**21.99**	**23.04**	**64.06**	**21.23**	**21.61**	**56.00**	**4.47**	**4.63**	**46.20**
Retain 128 Tokens (↓ 77.8%)													
FastV (ECCV24)	44.47	15.29	15.34	59.56	19.77	20.57	57.81	18.27	18.83	54.41	3.77	3.97	42.20
SparseVLM (ICML25)	47.36	14.72	14.61	40.81	15.09	15.41	35.94	9.17	9.37	52.64	3.85	4.05	43.05
PDrop (CVPR25)	47.12	14.77	14.93	60.66	20.59	21.54	57.81	18.74	19.24	55.47	3.82	4.02	44.30
VisionZip (CVPR25)	47.84	13.78	14.16	59.19	20.79	21.72	50.00	18.97	19.58	55.00	4.39	4.56	44.55
MMTok (ICLR26)	46.39	15.34	15.72	62.50	21.24	22.27	59.38	20.47	21.03	54.76	4.32	4.52	44.60
MedR2R (Ours)	**52.88**	**16.22**	**16.61**	**63.24**	**21.91**	**22.76**	**62.50**	**21.15**	**21.72**	**56.27**	**4.44**	**4.64**	**45.45**

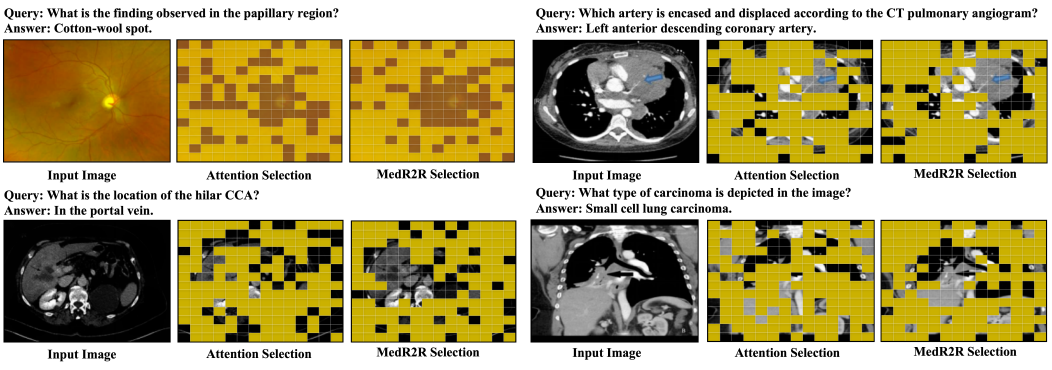


Figure 3: Qualitative comparison of retained visual tokens. Yellow cells indicate discarded patches and visible cells retained patches. Across fundus and CT examples, MedR2R better preserves feature coverage and question-relevant evidence than attention-only selection.

Runtime Efficiency. At 128 tokens, MedR2R speeds up LLaVA-Med/MedVLM-R1 by 1.54×/1.53× with similar memory (Table 3). Appendix D reports budget and overhead analyses

Table 3: Measured inference efficiency on LLaVA-Med v1.5 and MedVLM-R1. Each result is measured on the complete VQA-RAD test set. Evaluation time covers the model-response loop after initialization, and latency denotes the average inference time per sample.

LLaVA-Med v1.5					
Method	**Tokens**	**Eval. time↓**	**Rel. speed↑**	**Latency (ms)↓**	**Peak mem. (GiB)↓**
Vanilla	576	4:36	1.00×	612.0	15.11
FastV	128	2:59	1.54×	397.5	15.16
SparseVLM	128	2:56	1.57×	390.5	15.16
PDrop	128	3:01	1.52×	401.7	15.07
VisionZip	128	**2:30**	**1.84**×	**332.7**	15.06
MMTok	128	2:33	1.80×	339.8	16.12
MedR2R (Ours)	128	2:59	1.54×	396.2	15.10

MedVLM-R1					
Method	**Tokens**	**Eval. time↓**	**Rel. speed↑**	**Latency (ms)↓**	**Peak mem. (GiB)↓**
Vanilla	576	2:43	1.00×	361.4	5.63
FastV	128	**1:39**	**1.65**×	**218.5**	5.62
SparseVLM	128	1:59	1.37×	263.5	5.62
PDrop	128	1:40	1.62×	222.8	5.62
VisionZip	128	1:40	1.62×	222.8	6.02
MMTok	128	1:42	1.59×	227.0	5.62
MedR2R (Ours)	128	1:47	1.53×	236.9	5.66

Table 4: Instance-level stability of LLaVA-Med across five medical VQA benchmarks. Avg. denotes the macro-average over the five datasets.

Method	**Tokens**	**SLAKE**		**VQA-RAD**		**VQA-Med**		**PathVQA**		**PMC-VQA**		**Avg.**	
		DR↓	**NDR↓**	**DR↓**	**NDR↓**	**DR↓**	**NDR↓**	**DR↓**	**NDR↓**	**DR↓**	**NDR↓**	**DR↓**	**NDR↓**
FastV	128	30.07	3.20	29.05	4.21	28.60	5.00	31.05	3.68	45.65	11.10	32.88	5.44
SparseVLM	128	29.31	1.98	27.49	2.44	27.60	2.40	29.54	2.84	45.25	10.45	31.84	4.02
PDrop	128	28.18	1.51	26.39	1.11	26.60	0.80	28.01	2.65	43.70	9.70	30.58	3.15
VisionZip	128	21.58	1.70	21.29	2.44	21.40	1.20	22.47	3.42	42.55	8.65	25.86	3.48
MMTok	128	23.75	1.51	24.17	3.55	24.20	1.80	24.97	2.77	17.90	2.20	23.00	2.36
MedR2R (Ours)	128	**12.16**	**1.13**	**11.97**	**0.44**	**11.80**	**0.60**	**14.48**	**1.97**	**17.45**	**1.85**	**13.57**	**1.20**

Table 5: Ablation study of MedR2R components under the same token budget across five datasets.

Variant	**Global relevance**	**Candidate recall**	**Evidence retention**	**Consolidation**	**Closed Acc. (%) ↑**	**Open Recall (%) ↑**	**Rouge (%) ↑**	**DR (%) ↓**	**NDR (%) ↓**	**Latency (ms) ↓**
Global-relevance selection	√	–	–	–	46.51	15.18	5.06	32.17	5.78	380.4
Encoder-side candidate recall	√	√	–	–	47.12	16.02	5.41	25.38	3.63	388.5
LLM-side evidence retention	√	–	√	–	47.28	16.24	5.48	22.12	2.88	394.1
MedR2R w/o consolidation	√	√	√	–	47.81	16.85	5.75	16.23	1.94	395.5
Full MedR2R	√	√	√	√	**48.15**	**16.99**	**5.87**	**13.57**	**1.20**	396.2

and evaluates the compatibility of MedR2R with INT8 quantization in terms of latency and memory usage. Although VisionZip and MMTok achieve lower latency, MedR2R better preserves prediction consistency. On LLaVA-Med, it combines a $1.54\times$ inference speedup with the lowest DR and NDR among the compared compressed methods.

Prediction stability. Table 4 shows that MedR2R improves instance-level stability across all five datasets, achieving the lowest DR and NDR with macro-average rates of 13.57 and 1.20. The reductions are substantial across benchmarks: DR decreases from 21.58 to 12.16 on SLAKE, from 21.40 to 11.80 on VQA-Med, while NDR is consistently reduced. These results indicate that MedR2R more effectively preserves individual predictions under visual-token compression.

Ablation Study. Table 5 verifies the complementary roles of the sequential decisions on LLaVA-Med v1.5 with $k = 128$. The representative component tests encoder-side candidate recall, while the residual component tests LLM-side evidence retention; combining them lowers DR/NDR from 32.17/5.78 to 16.23/1.94. Feature consolidation further reduces DR/NDR to 13.57/1.20 and achieves the best accuracy, recall, and Rouge with negligible added latency. Analyses of feature coverage, residual profiles, and matched token-removal sensitivity are provided in Appendix C.

5 Conclusion

We presented MedR2R, a training-free framework for efficient and stable medical MLLM inference. MedR2R separates visual-token compression into encoder-side representative candidate construction and LLM-side residual-based evidence retention, followed by feature consolidation. This cross-level design preserves representative and hard-to-replace visual information without relying on a single final-budget importance score. Experiments across five medical VQA benchmarks and multiple backbones show that MedR2R maintains competitive task performance while improving instance-level prediction stability under visual-token reduction. These findings highlight the importance of considering prediction stability alongside computational efficiency in medical MLLM acceleration. Future work will explore adaptive visual budgeting and clinical generation tasks.

AI USE STATEMENT

We used LLMs solely for language polishing, such as improving wording and checking for typographical and grammatical errors. We did not use generative AI tools for any research-related tasks, including hypothesis generation, methodology design, experimental implementation, data analysis, or result interpretation. All AI-assisted modifications were reviewed by the authors, and we take full responsibility for the final content of this work.

ETHICS STATEMENT

To the best of our knowledge, this work has no potential ethical issues to disclose.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of this work, all implementation details have been clearly described, and we have also released the code.

REFERENCES

Saeed Ranjbar Alvar, Gursimran Singh, Mohammad Akbari, and Yong Zhang. Divprune: Diversity-based visual token pruning for large multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 9392–9401, 2025.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

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*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025b.

Asma Ben Abacha, Sadid A Hasan, Vivek V Datla, Dina Demner-Fushman, and Henning Müller. Vqa-med: Overview of the medical visual question answering task at imageclef 2019. In *Proceedings of CLEF (Conference and Labs of the Evaluation Forum) 2019 Working Notes*. 9-12 September 2019, 2019.

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, 2024a.

Wenting Chen, Linlin Shen, Jingyang Lin, Jiebo Luo, Xiang Li, and Yixuan Yuan. Fine-grained image-text alignment in medical imaging enables explainable cyclic image-report generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9494–9509, 2024b.

Yuan Chen, Zichen Wen, Yuzhou Wu, Xuyang Liu, Shuang Chen, Junpeng Ma, Weijia Li, Conghui He, and Linfeng Zhang. Ipcv: Information-preserving compression for mllm visual encoders. *arXiv preprint arXiv:2512.18747*, 2025.

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, 2024c.

Sixun Dong, Juhua Hu, Mian Zhang, Ming Yin, Yanjie Fu, and Qi Qian. Mmtok: Multimodal coverage maximization for efficient inference of vlms. In *International Conference on Learning Representations*, volume 2026, pp. 48075–48099, 2026.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Tianchen Fang and Guiru Liu. Regionmed-clip: A region-aware multimodal contrastive learning pre-trained model for medical image understanding. *arXiv preprint arXiv:2508.05244*, 2025.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*, 2020.

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.

Jason J Lau, Soumya Gayen, Asma Ben Abacha, and Dina Demner-Fushman. A dataset of clinically generated visual questions and answers about radiology images. *Scientific data*, 5(1):180251, 2018.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36: 28541–28564, 2023a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023b.

Xiang Li, Huihui Fang, Mingsi Liu, Jinghao Wang, Lixin Duan, Yuqi Fang, and Yanwu Xu. Beyond visual prompt: Unlocking parameter-efficient online adapter for continual test-time adaptation in medical image segmentation. *Information Fusion*, pp. 104256, 2026a.

Xiang Li, Huihui Fang, Yuning Wang, Cen Chen, Qi Liu, Jinke Wang, Yi Shi, Lixin Duan, Weihua Yang, Yue Liu, et al. Cross-modality anatomy knowledge distillation for retinal oct layer segmentation with limited labels. *IEEE Transactions on Circuits and Systems for Video Technology*, 2026b.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Bo Liu, Li-Ming Zhan, Li Xu, Lin Ma, Yan Yang, and Xiao-Ming Wu. Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering. In *2021 IEEE 18th international symposium on biomedical imaging (ISBI)*, pp. 1650–1654. IEEE, 2021.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 26296–26306, 2024.

Ting Luo, Jinxian Zhang, Tao Chen, Zhouyan He, Yanda Meng, Mengting Liu, Jiong Zhang, and Dan Zhang. Artifact-suppressed 3d retinal microvascular segmentation via multi-scale topology regulation. *Medical Image Analysis*, pp. 103988, 2026.

Dung Nguyen, Minh Khoi Ho, Huy Ta, Thanh Tam Nguyen, Qi Chen, Kumar Rav, Quy Duong Dang, Satwik Ramchandre, Son Lam Phung, Zhibin Liao, et al. Localizing before answering: A benchmark for grounded medical visual question answering. In *Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)*. International Joint Conferences on Artificial Intelligence Organization, 2025.

Jiazhen Pan, Che Liu, Junde Wu, Fenglin Liu, Jiayuan Zhu, Hongwei Bran Li, Chen Chen, Cheng Ouyang, and Daniel Rueckert. Medvlm-r1: Incentivizing medical reasoning capability of vision-language models (vlms) via reinforcement learning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 337–347. Springer, 2025.

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

Yizheng Sun, Hao Li, Chang Xu, Hongpeng Zhou, Chenghua Lin, Riza Theresa Batista-Navarro, and Jingyuan Sun. Does acceleration cause hidden instability in vision language models? uncovering instance-level divergence through a large-scale empirical study. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8453–8467, 2025.

Chuhan Wang, Zhenghao Chen, Jean YH Yang, and Jinman Kim. Improving lesion segmentation in medical images by global and regional feature compensation. *Pattern Recognition*, pp. 112461, 2025.

Yuxin Wen, Qingqing Cao, Qichen Fu, Sachin Mehta, and Mahyar Najibi. Efficient vision-language models by summarizing visual tokens into compact registers. *arXiv preprint arXiv:2410.14072*, 2024.

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.

Long Xing, Qidong Huang, Xiaoyi Dong, Jiajie Lu, Pan Zhang, Yuhang Zang, Yuhang Cao, Conghui He, Jiaqi Wang, Feng Wu, et al. Conical visual concentration for efficient large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14593–14603, 2025.

Dunyuan Xu, Xikai Yang, Juzheng Miao, Yaoqian Li, Jinpeng Li, and Pheng-Ann Heng. Medevo: Test-time self-evolution for medical multimodal large language models. *arXiv preprint arXiv:2603.07443*, 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19792–19802, 2025.

Jiong Zhang, Qihang Xie, Lei Mou, Dan Zhang, Da Chen, Caifeng Shan, Yitian Zhao, Ruisheng Su, and Mengguo Guo. Dsca: A digital subtraction angiography sequence dataset and spatio-temporal model for cerebral artery segmentation. *IEEE Transactions on Medical Imaging*, 2025a.

Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 881–916, 2025b.

Xiaoman Zhang, Chaoyi Wu, Ziheng Zhao, Weixiong Lin, Ya Zhang, Yanfeng Wang, and Weidi Xie. Pmc-vqa: Visual instruction tuning for medical visual question answering. *arXiv preprint arXiv:2305.10415*, 2023.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*, 2024.

APPENDIX

This appendix supplements MedR2R with evaluation details, additional backbone results, analyses of representative candidate construction and residual-based evidence preservation, token-budget and efficiency studies, cross-modality visualizations, and theoretical analysis.

A EVALUATION PROTOCOL AND IMPLEMENTATION DETAILS

A.1 BENCHMARKS AND METRIC DENOMINATORS

Table A.1 summarizes the five medical VQA benchmarks. SLAKE contains 1,061 English test questions, including 416 closed-ended and 645 open-ended questions. VQA-RAD contains 451 questions, with 272 closed-ended and 179 open-ended questions, and VQA-Med contains 500 questions, with 64 closed-ended and 436 open-ended questions. PathVQA contains 6,761 closed- and open-ended test questions, and PMC-VQA contains 2,000 multiple-choice questions. Following the main evaluation protocol, accuracy is computed over closed-ended or multiple-choice questions, while token Recall and ROUGE-1, denoted *Rouge* in the tables, are computed over open-ended questions. DR compares normalized full responses with the corresponding vanilla outputs. NDR counts correct-to-incorrect transitions using the question-type-specific correctness rules in the main text.

Table A.1: Evaluation subsets and quality metrics for the five medical VQA benchmarks.

Dataset	Evaluation subset	Question types	Quality metrics
SLAKE (Liu et al., 2021)	English test ($M = 1{,}061$)	Closed / open	Acc., Recall, Rouge
VQA-RAD (Lau et al., 2018)	Test ($M = 451$)	Closed / open	Acc., Recall, Rouge
VQA-Med (Ben Abacha et al., 2019)	Test ($M = 500$)	Closed / open	Acc., Recall, Rouge
PathVQA (He et al., 2020)	Test ($M = 6{,}761$)	Closed / open	Acc., Recall, Rouge
PMC-VQA (Zhang et al., 2023)	Test ($M = 2{,}000$)	Multiple choice	Acc.

A.2 INFERENCE SETTINGS

MedR2R uses representative candidate construction for encoder-side candidate recall and residual-based evidence preservation for LLM-side evidence retention, followed by token consolidation. Given n visual tokens and a final budget k, the default candidate budget is $m = \min(n, 2k)$. We set the global-relevance exponent to $\gamma = 1$, the basis size to $r = \min(10, k)$, the ridge regularizer to $\lambda = 10^{-5}$, and the numerical stabilizer to $\epsilon = 10^{-6}$. The first stage combines global relevance with marginal representation gain, as described in Section 3.2. Global relevance is obtained from global-token attention when available and from the feature-derived proxy otherwise. The ordered candidates are projected into the language-model embedding space and contextualized together with the textual instruction. Residual-based evidence preservation is applied after layer $\ell_* = 16$ for LLaVA-Med v1.5 and $\ell_* = 14$ for MedVLM-R1 and Qwen3-VL. The compact basis is constructed by deterministic cosine farthest-point sampling, and the regularized reconstruction system is solved in FP32 with a pseudoinverse fallback for numerical failures.

The retained set contains the basis tokens and the $k - r$ nonbasis candidates with the largest normalized residuals. Token consolidation then averages each retained representative with its assigned candidates. The compressed visual states follow the order of their representative positions, and the model adapter updates position information, attention masks, and populated cache entries accordingly. All model parameters remain frozen. Selection and consolidation take place during prefilling, after which generation continues with the compressed state.

B ADDITIONAL BACKBONE EVALUATION

Table B.1 evaluates Qwen3-VL-8B-Instruct on the same five benchmarks using final budgets of 192 and 128 visual tokens. FastV, SparseVLM, PDrop, VisionZip, MMTok, and MedR2R are compared at matched final budgets. The 576-token vanilla model and LLM.int8() respectively provide full-token and quantized references.

Table B.1: Qwen3-VL-8B-Instruct on the five benchmarks. All values are percentages, higher is better. Bold denotes the best value within each compressed-budget block (including ties), shading identifies MedR2R. Full-token and quantized references are shown separately.

Method	SLAKE			VQA-RAD			VQA-Med			PathVQA			PMC-VQA
	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.	Recall	Rouge	Acc.
Full-token reference, 576 tokens (100%)													
Vanilla	57.21	14.67	14.64	74.26	32.00	33.65	82.81	28.75	29.26	65.59	6.00	6.22	55.60
LLM.int8()	56.49	14.37	14.35	73.90	30.89	32.30	79.69	28.79	29.47	64.73	6.00	6.20	54.80
Retain 192 Tokens (↓ 66.7%)													
FastV (ECCV24)	56.01	13.12	13.15	66.54	29.88	30.67	75.00	26.57	27.18	62.67	5.57	5.55	53.40
SparseVLM (ICML25)	56.97	13.15	13.40	70.59	31.20	32.66	79.69	27.68	28.19	63.34	5.49	5.71	53.45
PDrop (CVPR25)	**57.21**	13.43	13.56	69.85	30.31	31.77	78.13	28.44	28.71	63.93	5.61	5.85	54.65
VisionZip (CVPR25)	**57.21**	13.28	13.37	72.06	31.26	32.73	68.75	27.89	27.89	63.20	5.68	5.73	54.45
MMTok (ICLR26)	56.49	14.26	14.31	68.38	30.85	32.21	**81.25**	28.13	28.86	62.64	5.64	5.78	54.25
MedR2R (Ours)	**57.21**	**14.36**	**14.35**	**72.79**	**31.31**	**32.94**	79.69	**28.59**	**29.24**	**64.85**	**5.69**	**5.92**	**54.95**
Retain 128 Tokens (↓ 77.8%)													
FastV (ECCV24)	56.25	13.48	13.20	67.28	28.03	28.99	73.44	25.78	26.34	60.69	5.20	5.25	52.05
SparseVLM (ICML25)	56.97	13.15	13.23	69.12	29.18	29.82	75.00	27.10	27.38	62.22	5.31	5.50	53.25
PDrop (CVPR25)	57.21	13.43	13.59	70.96	29.50	30.97	73.44	27.85	28.46	61.99	5.49	5.68	53.50
VisionZip (CVPR25)	57.21	13.93	**13.97**	71.32	29.57	31.12	73.44	28.15	28.17	61.04	5.53	5.71	54.05
MMTok (ICLR26)	56.49	**14.21**	13.92	70.59	31.09	31.84	78.13	27.31	27.71	63.76	**5.59**	5.71	52.10
MedR2R (Ours)	**57.45**	13.36	13.27	**72.06**	**31.62**	**33.28**	**79.69**	**28.29**	**29.01**	**63.96**	5.53	**5.73**	**54.95**

MedR2R achieves the best result in 12 of the 13 metric columns at 192 tokens, including ties, and in 10 columns at 128 tokens. At 128 tokens, it attains VQA-RAD accuracy of 72.06, Recall of 31.62, and Rouge of 33.28, compared with 74.26, 32.00, and 33.65 for the full-token model. On PMC-VQA, it reaches 54.95 accuracy at both compressed budgets, compared with the full-token result of 55.60.

C Analysis of the Two-Stage Selection Mechanism

We examine the two-stage selection mechanism on VQA-RAD through encoder-side feature coverage, LLM-side residual profiles, and downstream sensitivity to matched token removal. These analyses complement the component ablations in Table 5.

C.1 Feature Coverage of Representative Candidate Construction

We evaluate global-relevance-only top-128 selection and representative candidate construction as standalone encoder-side selectors, each choosing 128 of $n = 576$ visual tokens. Using $c_i(\mathcal{C})$ and $\mathcal{U}(\mathcal{C})$ from Section 3.2, mean feature coverage for each question is:

$$\frac{100}{n}\sum_{i=1}^{n} c_i(\mathcal{C}) = 100\left(1 - \frac{\mathcal{U}(\mathcal{C})}{n}\right).$$

Lower-tail coverage is the fifth percentile of $100c_i(\mathcal{C})$ over the original visual tokens. Both statistics are computed per question and then averaged over the evaluation set.

As shown in Fig. C.1, representative candidate construction increases average mean coverage from 64.85 to 89.30, a gain of 24.45 percentage points, and average fifth-percentile coverage from 37.74 to 66.07. Improvements in both statistics indicate broader representation of the encoder features, including tokens weakly covered by global-relevance-only selection.

C.2 Normalized Residuals and LLM-Side Evidence Retention

With $m = 256$, $k = 128$, and $r = 10$, we characterize the contextualized candidates retained by residual-based selection. Figure C.2(a) compares the per-question mean normalized residuals of retained and discarded candidates, with basis tokens included in the retained set. Their overall means are 0.536 and 0.348, respectively. The final allocation combines the compact reference basis with candidates whose contextualized features are less accurately reconstructed from it.

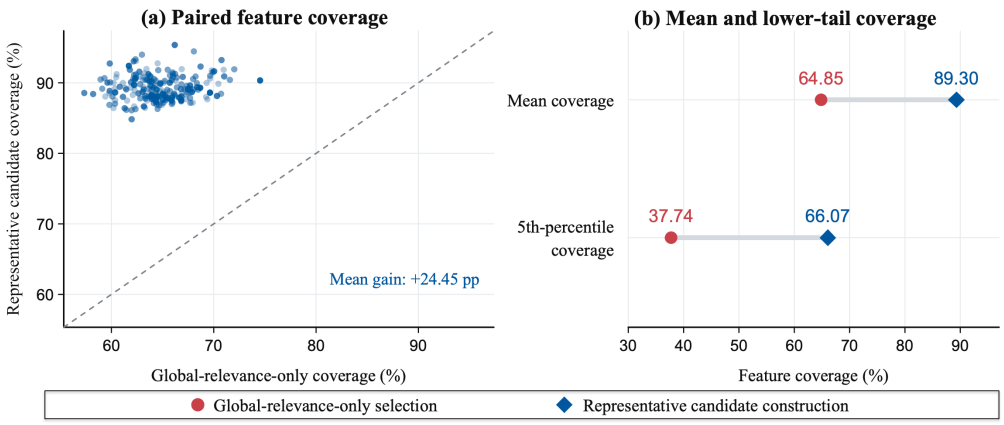


Figure C.1: Feature coverage on VQA-RAD at a matched 128-token selection budget. (a) Paired mean coverage for each question; the dashed line indicates equality. (b) Dataset averages of mean coverage and per-question fifth-percentile coverage.

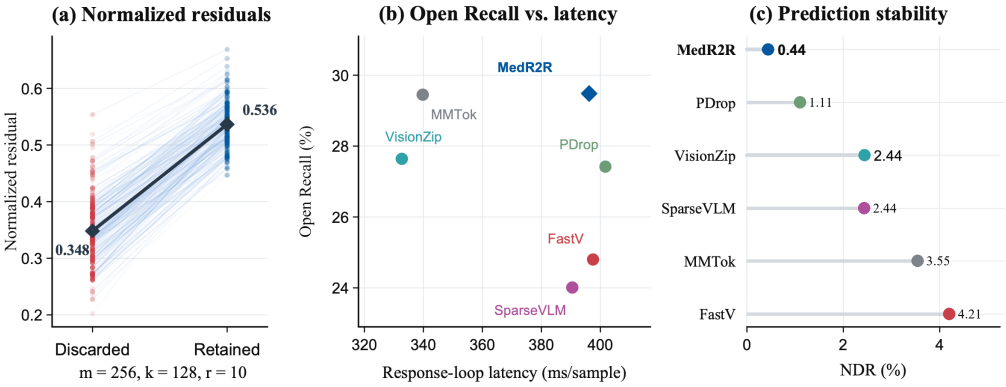


Figure C.2: Residual profiles and complete-method results on VQA-RAD at 128 tokens. (a) Paired mean normalized residuals of discarded and retained candidates; dark markers indicate overall means, and retained means include basis tokens. (b) Open Recall versus response-loop latency for the complete methods. (c) NDR for the same methods.

Panels (b)–(c) summarize task quality, latency, and prediction stability for the complete methods. MedR2R achieves the highest open Recall (29.48) and lowest NDR (0.44%) among the compared compressed methods, with a response-loop latency of 396.2 ms and a $1.54\times$ speedup over the full-token model.

C.3 Controlled Token-Removal Analysis

We further examine whether normalized residuals are associated with downstream answer sensitivity under a controlled token-removal setting. For each question, all policies operate on the same $m = 256$ candidate sequence after LLM layer 16, preserve the $r = 10$ basis tokens, and remove exactly 128 nonbasis candidates. The resulting $k = 128$ visual tokens are passed through the remaining layers without feature consolidation, and their decoded outputs are compared with those of the corresponding uncompressed 256-candidate reference. We compare removal based on high and low normalized residuals with policies based on encoder global attention, contextualized feature norm, and random removal.

We report two measures relative to the paired 256-candidate reference. The normalized answer-change rate follows the same full-response normalization and comparison used for DR. The task-aware quality-decrease rate records correct-to-incorrect transitions for closed-ended questions and decreases in token Recall for open-ended questions. Together, these measures characterize output sensitivity and downstream quality degradation under different token-removal criteria.

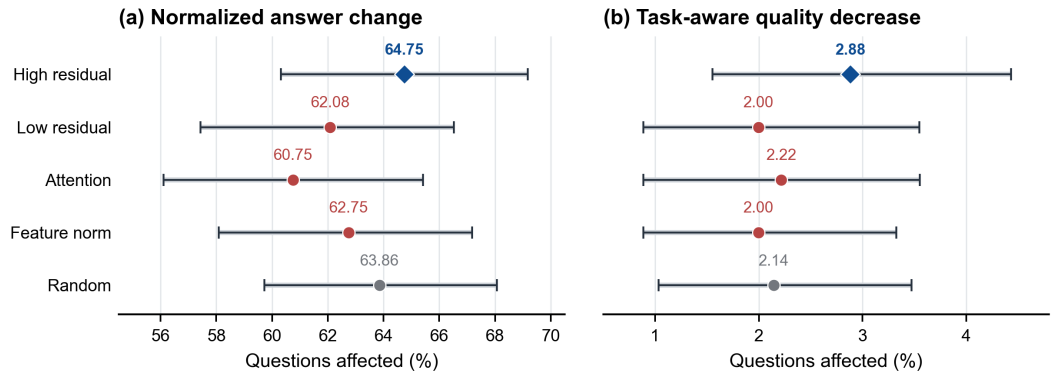


Figure C.3: Controlled token-removal analysis on VQA-RAD. All policies operate on the same 256-candidate sequence after LLM layer 16, preserve the basis tokens, and remove exactly 128 nonbasis candidates without consolidation. (a) Normalized full-answer changes. (b) Task-aware answer-quality decreases. Random denotes the mean over three fixed seeds; error bars show 95% paired bootstrap intervals over questions.

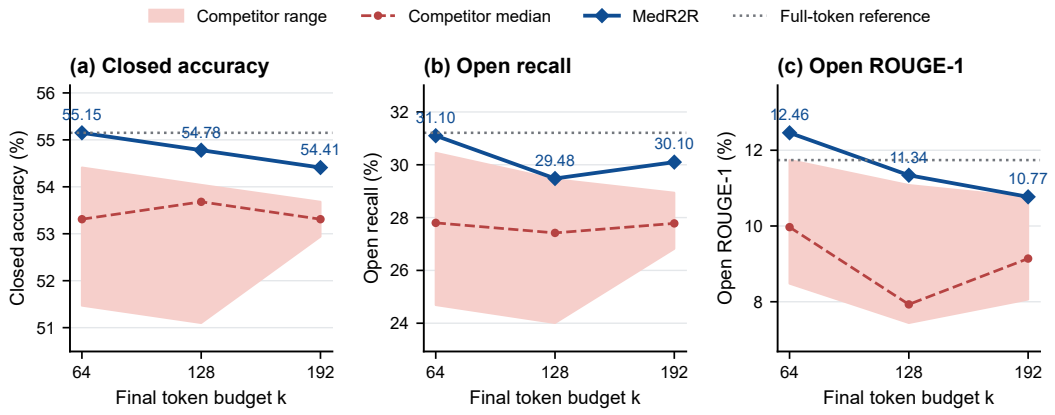


Figure D.1: Final-budget performance of LLaVA-Med v1.5 on VQA-RAD. (a) Closed accuracy, (b) open Recall, and (c) open Rouge. The shaded band spans the five compared compression methods, and the dashed curve shows their median.

As shown in Fig. C.3, high-residual deletion yields the highest observed rates of normalized answer changes (64.75%) and task-aware quality decreases (2.88%). The corresponding rates are 62.08%/2.00% for low-residual deletion, 60.75%/2.22% for attention, 62.75%/2.00% for feature norm, and 63.86%/2.14% for random deletion. The higher observed sensitivity under high-residual deletion is consistent with prioritizing poorly reconstructed candidates during LLM-side evidence retention.

D Token Budgets and Inference Efficiency

D.1 Final-Budget Performance under the Main Evaluation Protocol

Figure D.1 summarizes MedR2R results on VQA-RAD under final visual-token budgets of 64, 128, and 192. At each budget, the shaded band spans the minimum and maximum values across FastV, SparseVLM, PDrop, VisionZip, and MMTok, and the dashed curve shows their median. The full-token model provides a shared performance reference. MedR2R leads the compared compressed methods in all three VQA-RAD quality metrics at each displayed budget. At 64 tokens, its closed accuracy of 55.15 matches the full-token reference, while Recall and Rouge reach 31.10 and 12.46. At 128 tokens, the corresponding values are 54.78, 29.48, and 11.34; at 192 tokens, they are 54.41, 30.10, and 10.77. These results demonstrate competitive performance across the tested budgets.

Table D.1: Candidate- and final-budget sensitivity analysis of MedR2R on VQA-RAD under varying candidate-pool sizes and retained-token budgets. Time reports prefill plus the first output step on a single NVIDIA A100.

ρ	k	m	Acc.↑	Recall↑	Rouge↑	DR↓	NDR↓	Time (ms)↓
(a) Candidate pool at fixed $k = 128$								
1.0	128	128	51.84	26.26	9.50	16.63	2.22	56.48
1.5	128	192	53.68	28.49	10.61	14.19	1.33	66.37
2.0	128	256	54.78	29.48	11.34	11.97	0.44	77.75
3.0	128	384	54.04	28.49	10.06	13.53	1.11	95.65
4.5	128	576	53.31	26.82	8.94	14.63	1.77	68.62
(b) Final budget with $\rho = 2$								
2.0	64	128	55.15	31.10	12.46	15.30	1.55	55.08
2.0	128	256	54.78	29.48	11.34	11.97	0.44	77.75
2.0	192	384	54.41	30.10	10.77	13.08	0.89	97.37

D.2 Candidate-Pool Size and Selection Overhead

Table D.1 conducts a sensitivity study by varying the candidate budget m and the final budget k. We define the expansion factor ρ through $m = \min(n, \lceil \rho k \rceil)$. Panel (a) varies m with $k = 128$ fixed, while Panel (b) varies k with $\rho = 2$ fixed. Quality and stability are evaluated on VQA-RAD with a maximum generation length of 32 new tokens. Prefill-oriented latency is measured separately on a single NVIDIA A100. Each configuration is evaluated over three passes, with the first ten requests of each pass used for warm-up. Generation is limited to one new token, so the measured interval covers prefilling and the first output step. The reported time is averaged over the post-warm-up requests across the three passes.

Matched final-token budgets do not imply matched computation. MedR2R retains $m = 2k$ candidates until layer ℓ_*, so its realized compute is higher than methods that prune directly to k tokens before the LLM. We therefore report measured response-loop latency alongside token budgets as the system-level efficiency measure. At $k = 128$, MedR2R takes 396.2 ms per sample, compared with 332.7 ms for VisionZip and 339.8 ms for MMTok, while reducing NDR from 2.44/3.55 to 0.44.

Candidate-pool trade-offs. At fixed $k = 128$, the default $m = 256$ achieves the highest accuracy (54.78), Recall (29.48), and Rouge (11.34), together with the lowest DR (11.97) and NDR (0.44), among the tested candidate pools. Increasing m from 128 to 256 improves the quality and stability values in the table, while increasing prefill-oriented latency from 56.48 to 77.75 ms. Across the two-stage configurations with $m = 192$, 256, and 384, measured latency rises from 66.37 to 77.75 and 95.65 ms.

The boundary controls help characterize the cost of the two decisions. At $m = k$, the encoder-side output directly supplies the final tokens, bypassing residual-based selection and consolidation. At $m = n$, all encoder-side tokens enter the LLM, bypassing first-stage pruning. The latter control takes 68.62 ms, compared with 95.65 ms for $m = 384$, because it skips greedy candidate construction.

Final-budget trade-offs. At $\rho = 2$, the 64-token configuration provides the highest accuracy, Recall, and Rouge in this sensitivity scan, together with the lowest prefill-oriented latency of 55.08 ms. The 128-token configuration achieves the lowest DR and NDR, highlighting the complementary information provided by quality and stability metrics when selecting an operating budget. The $(m, k) = (256, 128)$ entry is shared by both panels.

D.3 Compatibility with 8-Bit Quantization

To further reduce memory overhead, we combine MedR2R with 8-bit weight quantization. MedR2R reduces the number of visual tokens processed by the language model, while INT8 quantization lowers the memory footprint of model parameters. The two techniques therefore target complementary sources of inference cost. Table D.2 evaluates four configurations: the full-token FP16 model, LLM.int8(), MedR2R in FP16, and MedR2R combined with LLM.int8(). All configurations use LLaVA-Med v1.5 on VQA-RAD with batch size one and greedy decoding under the same gener-

ation settings. The vision encoder remains in FP16, while the quantized configurations use 8-bit linear layers on the GPU. In the combined setting, MedR2R uses $m = 256$ and $k = 128$.

Table D.2: Compatibility of MedR2R and LLM.int8() on VQA-RAD using a single NVIDIA A100. Peak memory is the per-pass maximum PyTorch allocated memory, including resident model weights.

Method	Tokens	Acc.↑	Recall↑	Rouge↑	Inference latency (ms)↓	Peak memory (GiB)↓
Vanilla (FP16)	576	55.15	31.21	11.74	612.0	15.11
LLM.int8()	576	47.79	27.19	10.36	318.7	11.51
MedR2R (FP16)	128	54.78	29.48	11.34	396.2	15.10
MedR2R + INT8	128	53.68	28.23	10.24	295.1	11.16

Combining MedR2R with INT8 reduces latency from 396.2 to 295.1 ms and peak memory from 15.10 to 11.16 GiB, while largely preserving task performance. Compared with the full-token FP16 model, the combined setting achieves a $2.07\times$ speedup. As quantization may also introduce instance-level prediction shifts, future work will explore stability-aware joint token compression and quantization.

E Qualitative Token Allocation Across Modalities

Figures E.1–E.3 compare three token-selection configurations across medical-image modalities at a matched final token budget. Each row shows the input image, global-relevance-only selection, representative candidate construction used as a standalone selector, and the complete MedR2R pipeline. Red, green, and blue outlines indicate directly retained tokens for the corresponding configurations. In the MedR2R column, blue diagonals indicate candidates whose features are averaged into retained representatives during token consolidation.

The yellow dashed box identifies the same display window in each column, and the inset enlarges that window. The model processes the complete source image before token selection. Questions and reference answers are taken from the datasets. The examples visualize the distinct roles of the two decisions: representative candidate construction recalls a diverse pool of encoder-side features, and residual-based evidence preservation retains the basis together with hard-to-approximate contextualized candidates. Consolidation then aggregates the remaining candidates into the retained representatives. Local token counts intuitively reflect differences in token retention and consolidation within the regions of interest across methods, serving as a visual complement to the preceding quantitative analyses.

F Theoretical Analysis

This section analyzes the three operations in the MedR2R pipeline: representative candidate construction using global-relevance-weighted marginal representation gain, residual-based evidence preservation within the contextualized candidate pool, and token consolidation through group averaging. We then describe a sufficient condition linking representation fidelity to prediction stability.

F.1 Marginal Representation Gain and Global-Relevance Weighting

Define the represented-feature objective:

$$F(\mathcal{C}) = \sum_{i=1}^{n} c_i(\mathcal{C}) = n - \mathcal{U}(\mathcal{C}).$$

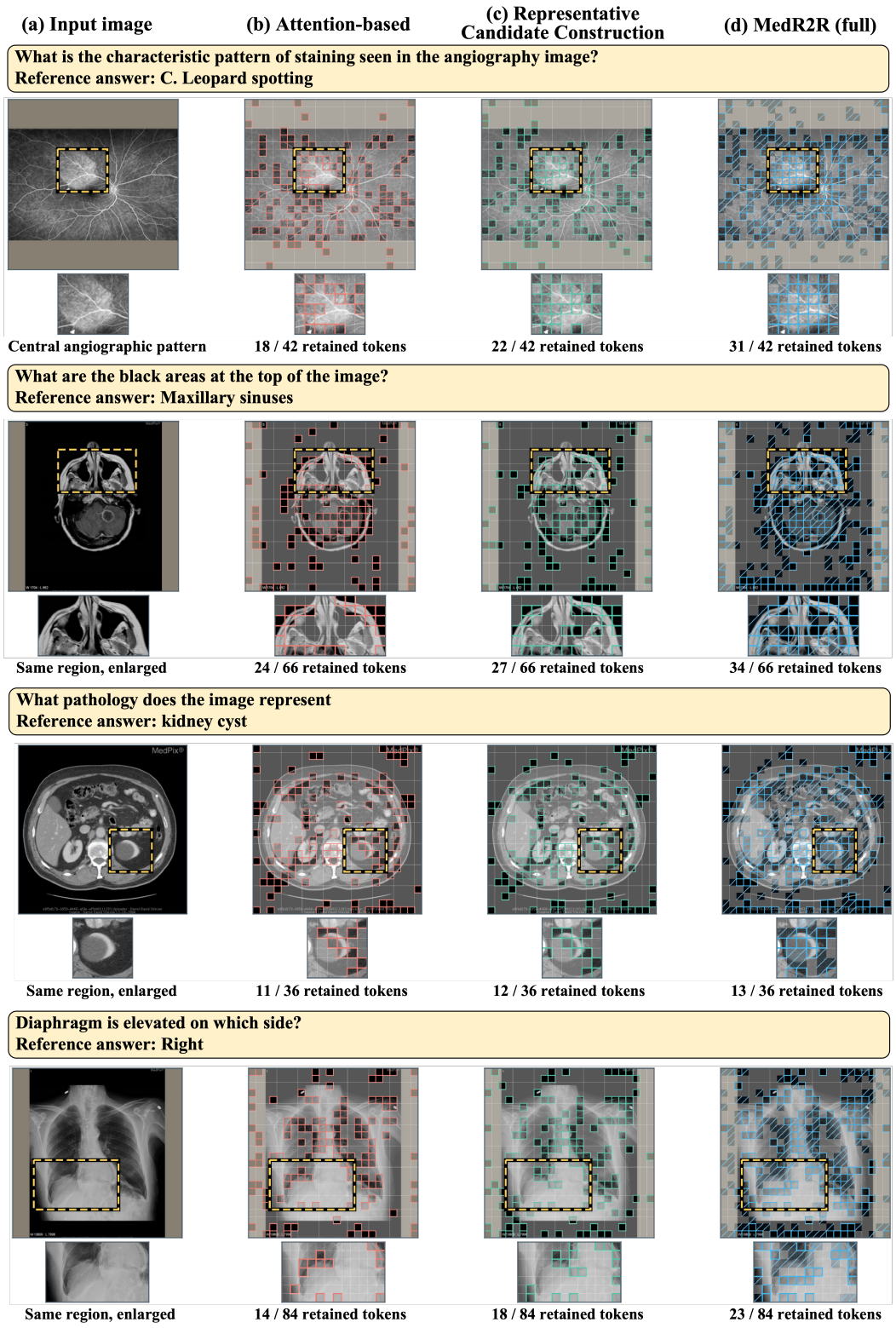


Figure E.1: Qualitative token allocation across modalities (I). Rows show retinal angiography, head MRI, abdominal CT, and chest radiography. Columns compare the input, global-relevance-only selection, representative candidate construction, and full MedR2R at a matched final token budget.

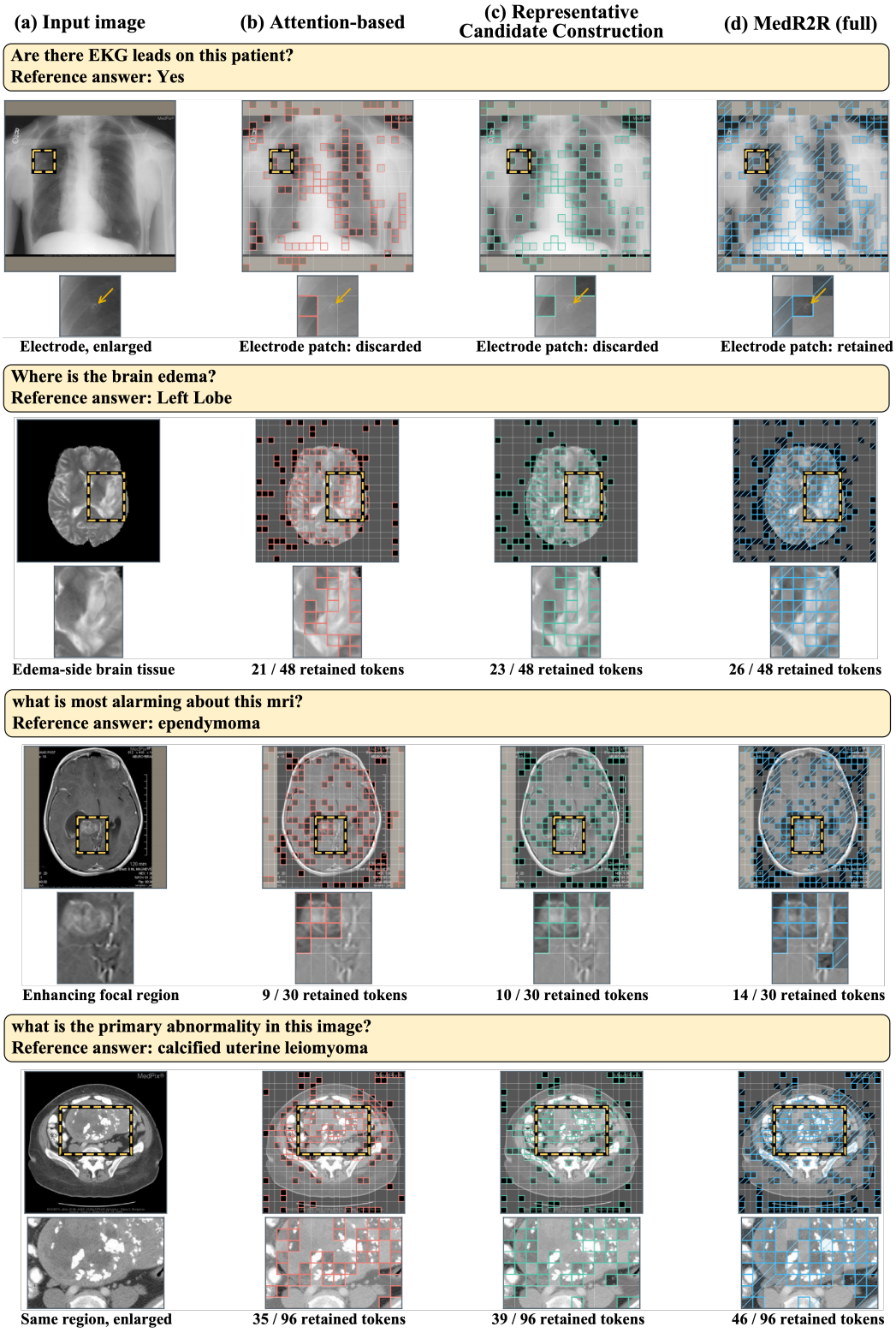


Figure E.2: Qualitative token allocation across modalities (II). Examples show an electrocardiography lead on chest radiography, cerebral edema and ependymoma on MRI, and a calcified uterine lesion on CT. The full pipeline combines representative candidate construction, residual-based evidence preservation, and consolidation. Outlines indicate directly retained tokens; blue diagonals indicate consolidated candidates.

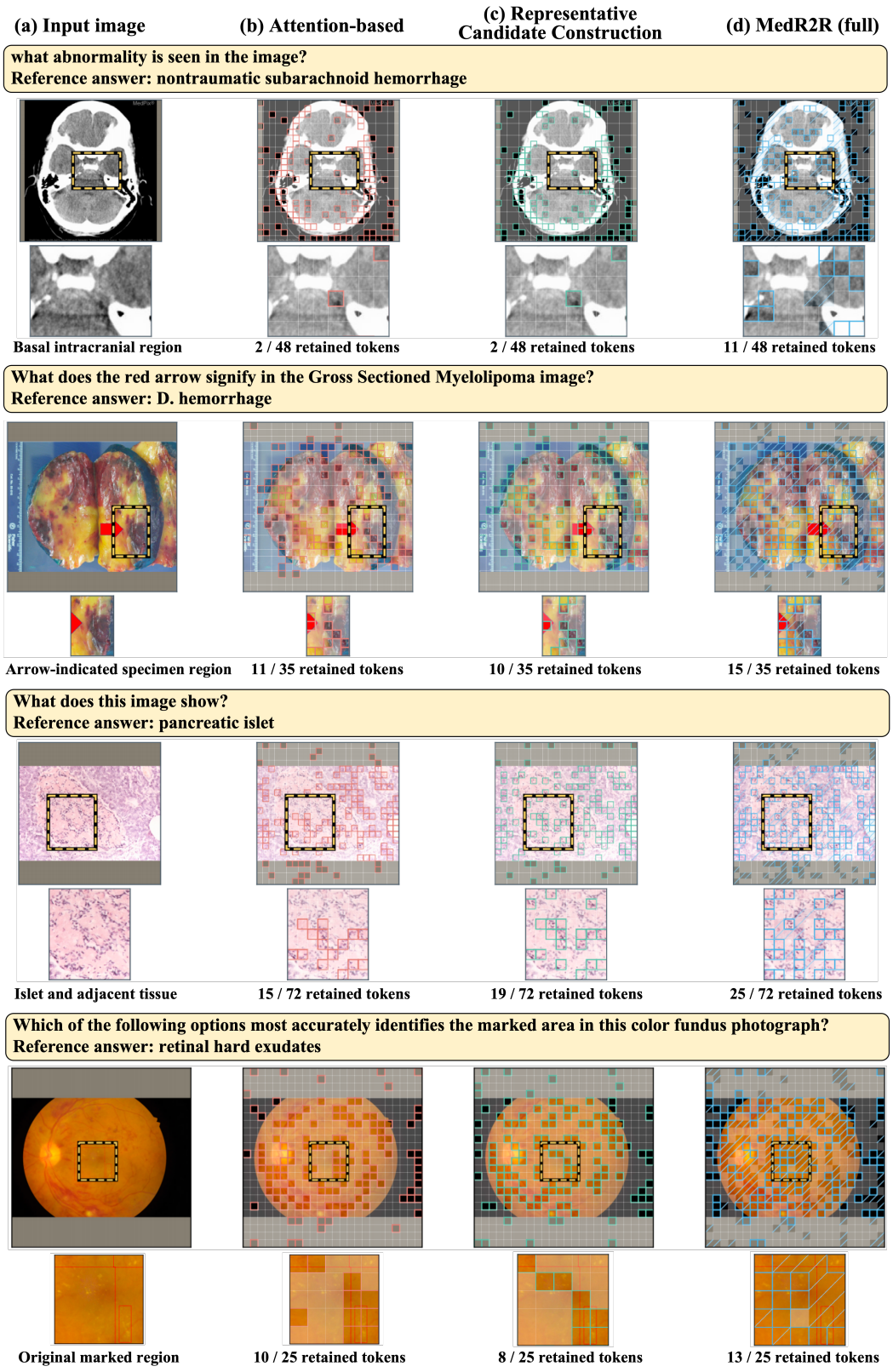


Figure E.3: Qualitative token allocation across modalities (III). Rows show head CT, gross pathology, histopathology, and color fundus photography.

Since $w_{ij} \in [0,1]$ and $c_i(\emptyset) = 0$, we have $F(\emptyset) = 0$, and F is monotone. For $\mathcal{C} \subseteq \mathcal{B}$ and $j \notin \mathcal{B}$, the marginal representation gain satisfies:

$$\begin{aligned}\Delta_j(\mathcal{C}) &= F(\mathcal{C} \cup \{j\}) - F(\mathcal{C}) \\ &= \mathcal{U}(\mathcal{C}) - \mathcal{U}(\mathcal{C} \cup \{j\}) \\ &= \sum_i [w_{ij} - c_i(\mathcal{C})]_+ \\ &\geq \sum_i [w_{ij} - c_i(\mathcal{B})]_+ = \Delta_j(\mathcal{B}).\end{aligned}$$

The inequality follows from $c_i(\mathcal{C}) \leq c_i(\mathcal{B})$. Thus, F is monotone submodular: a candidate provides diminishing additional representation as the selected pool grows. For unit-normalized features, the identity $\|\mathbf{p}_i - \mathbf{p}_j\|_2^2 = 2(1 - \mathbf{p}_i^\top \mathbf{p}_j)$ connects cosine similarity to proximity in feature space.

Global-relevance-weighted selection. MedR2R selects candidates using $g_j(\mathcal{C}) = q_j \Delta_j(\mathcal{C})$, where $q_j = s_j^\gamma > 0$ is fixed during candidate construction. These fixed weights preserve diminishing gains for each candidate: for $\mathcal{C} \subseteq \mathcal{B}$ and $j \notin \mathcal{B}$,

$$g_j(\mathcal{C}) = q_j \Delta_j(\mathcal{C}) \geq q_j \Delta_j(\mathcal{B}) = g_j(\mathcal{B}) \geq 0.$$

Global relevance therefore supplies a candidate-specific preference, while marginal representation gain accounts for the features already represented by the current pool. Recomputing their product after each selection adapts the ranking to the additional representation offered by the remaining candidates. When the relevance weights are uniform, the selection rule reduces to greedy maximization of F. The paired coverage results in Appendix C.1 evaluate the representation achieved by the candidate-construction rule on VQA-RAD.

F.2 Residual-Based Retention with a Fixed Dictionary

For fixed contextualized candidates $\mathbf{H}$ and dictionary $\mathbf{D} = \mathbf{H}_\mathcal{A}$, the regularized least-squares objective in Section 3.3 is strictly convex in its coefficients when $\lambda > 0$. Its solution is:

$$\mathbf{W}_* = \mathbf{H}\mathbf{D}^\top (\mathbf{D}\mathbf{D}^\top + \lambda \mathbf{I}_r)^{-1}, \qquad \widehat{\mathbf{H}} = \mathbf{W}_* \mathbf{D}.$$

The regularizer makes $\mathbf{D}\mathbf{D}^\top + \lambda \mathbf{I}_r$ positive definite, including when dictionary rows are linearly dependent. The implementation obtains the coefficients by solving the corresponding $r \times r$ linear system.

To characterize retention before consolidation, consider a fixed-dictionary reconstruction surrogate that preserves $\mathbf{h}_i$ exactly for retained indices and uses $\widehat{\mathbf{h}}_i$ for the remaining candidates. Its normalized error is:

$$J(\mathcal{R}) = \sum_{i \notin \mathcal{R}} \frac{\|\mathbf{h}_i - \widehat{\mathbf{h}}_i\|_2^2}{\|\mathbf{h}_i\|_2^2 + \epsilon} = \sum_{i \notin \mathcal{R}} e_i, \qquad \mathcal{A} \subseteq \mathcal{R}, \quad |\mathcal{R}| = k.$$

Retaining the $k - r$ largest nonbasis residuals minimizes J. To see this, let i be a retained nonbasis token and j a discarded candidate with $e_j > e_i$. Replacing i with j changes the objective by $e_i - e_j < 0$. Repeating these exchanges gives the retention rule in Eq. (8).

This result establishes the optimality of residual-based retention for the additive normalized-error surrogate within a fixed contextualized candidate pool and dictionary. The relaxed encoder-side budget supplies the candidates over which this final retention decision is made. Token consolidation then aggregates the remaining candidates as described next.

F.3 Token Consolidation under Fixed Group Assignments

For the cosine-based assignments π in Eq. (9), each group $\mathcal{G}_j$ includes its retained representative and is nonempty. The consolidated state $\widetilde{\mathbf{h}}_j$ in Eq. (10) minimizes the within-group squared distortion. Specifically, for any vector $\mathbf{z}$,

$$\sum_{i \in \mathcal{G}_j} \|\mathbf{h}_i - \mathbf{z}\|_2^2 = \sum_{i \in \mathcal{G}_j} \|\mathbf{h}_i - \widetilde{\mathbf{h}}_j\|_2^2 + |\mathcal{G}_j| \, \|\mathbf{z} - \widetilde{\mathbf{h}}_j\|_2^2.$$

The final term is minimized at $\mathbf{z} = \widetilde{\mathbf{h}}_j$. Setting $\mathbf{z} = \mathbf{h}_j$ additionally shows that group averaging achieves no greater fixed-group distortion than using the original representative as the group proxy. This property characterizes consolidation after the cosine assignments have been formed. Table 5 evaluates its contribution to task performance and prediction stability.

F.4 A Sufficient Condition for Prediction Stability

Representation fidelity supports prediction consistency when the reference decoding decision has a sufficient margin. For a shared decoding prefix, let $\mathbf{z}_t^0$ and $\mathbf{z}_t^c$ be the full-token and compressed next-token logits. For analysis, introduce an aligned, full-length surrogate state $\bar{\mathbf{Z}}_t$ and the full-model state $\mathbf{Z}_t^0$, each including the relevant visual, textual, and cache information. Suppose the logit map is locally L-Lipschitz on these aligned states, and let $\eta_t \geq 0$ bound the additional logit difference between surrogate computation and actual compressed inference. Then

$$\|\mathbf{z}_t^0 - \mathbf{z}_t^c\|_\infty \leq L\|\mathbf{Z}_t^0 - \bar{\mathbf{Z}}_t\|_F + \eta_t \equiv \delta_t.$$

The term η_t accounts for differences introduced by shortened attention, position changes, and consolidation. Let μ_t be the full-token winner's logit margin over the runner-up. If $\mu_t > 2\delta_t$, the winning token is preserved, because each logit changes by at most δ_t. When this condition holds at every step, including termination, induction on the shared prefix shows that greedy decoding produces the same answer.

This sufficient condition relates representation perturbations and decoding margins to output consistency. The preceding analyses characterize candidate representation, fixed-dictionary residual retention, and groupwise consolidation, while the task metrics and paired DR/NDR measurements in the main text quantify the observed quality and stability of compressed inference.