KePIR: Guiding Embedding Learning with Key Positional Information for Universal Multimodal Retrieval

Anonymous ACL submission

Abstract

Multimodal retrieval has emerged as a widely studied research direction. Most existing methods focus on matching global semantic information while overlooking fine-grained information units in the query that are critical to determining the retrieval results. We propose **KePIR**, which guides embedding learning with **Ke**y **P**ositional **I**nformation for Universal Multimodal **R**etrieval. At the token level, KePIR explicitly identifies textual key tokens and visual key tokens that are critical for semantic matching. Furthermore, we introduce a dual-decoder masked reconstruction mechanism to conditionally reconstruct key textual tokens and key visual-token representations, thereby guiding the multimodal encoder to preserve key fine-grained cues in the learned retrieval embeddings. Experimental results on the Universal Multimodal Retrieval (UMR) benchmark M-BEIR show that, with the same Qwen3-VL-2B-Instruct backbone, KePIR achieves better overall performance than both LamRA-Ret, a multi-stage training method, and CORAL, a method that also introduces a dual-decoder structure. KePIR also brings consistent improvements on most retrieval subtasks, validating the effectiveness of our method.

1 Introduction

Semantic retrieval is a widely studied task that aims to retrieve information relevant to a user's query from large-scale data collections (Wagner et al., 2001; Guo et al., 2022; Magnani et al., 2019; Li et al., 2017). It has been broadly applied to search engines (Khattab and Zaharia, 2020; Arivazhagan et al., 2023), retrieval-augmented generation (RAG) (Lewis et al., 2020; Gao et al., 2023), and knowledge-based question answering (Agrawal et al., 2015; Kafle and Kanan, 2016).

As multimodal data continue to grow, multimodal retrieval has emerged as a promising yet challenging research direction (Zhao et al., 2023; Kuang et al., 2025; Baldrati et al., 2023). It aims to retrieve query-relevant visual or textual evidence from heterogeneous sources, such as images, image–text web pages, and documents (Liu et al., 2021; Ventura et al., 2024; Zhang et al., 2024). Early works used captioning models to convert visual or multimodal inputs into text (Wu et al., 2023; Kuang et al., 2025; Radford et al., 2021), enabling standard text retrievers to be applied to multimodal retrieval. However, such conversion-based approaches may lose critical information due to the limitations of captioning models (Che et al., 2023; Li et al., 2024; Jia et al., 2021).

Recent research has gradually shifted toward representation-based retrieval paradigms, where text and multimodal content are encoded into a unified embedding space for retrieval (Radford et al., 2021; Zhang et al., 2021; Kong et al., 2025). However, existing retrieval methods often focus on modeling global semantic similarity, making it difficult to capture the fine-grained and discriminative information that is crucial for a given query (Chow et al., 2025). In complex multimodal retrieval tasks, retrieval decisions are often driven by a few key fine-grained semantic cues rather than global document semantics alone (Lin et al., 2023).

In existing multimodal retrieval models, such fine-grained semantics are usually implicitly encoded in local tokens. In practice, attention weights are commonly used to measure the relevance between tokens and the query, with high-attention tokens often regarded as internally important by the model (Zhao et al., 2025). However, these important tokens are often not effectively leveraged within the model and can be easily overwhelmed by the large number of low-weight tokens, thereby weakening their influence on the final representation and retrieval decision (Tang et al., 2025). Moreover, structural attention sinks (Gu et al., 2025) indicate that semantically irrelevant tokens may sometimes receive disproportionately

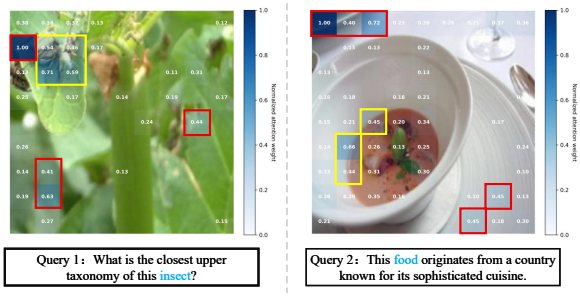


Figure 1: Normalized token-level attention visualization for two image–query pairs. Only tokens with attention weights above 0.1 are annotated, and tokens exceeding 0.4 are regarded as high-attention tokens.

high attention in multimodal models, for instance due to visual attention sinks (Kang et al., 2025a) or anomalous activation dimensions (Sun et al., 2024), which further interferes with the effective modeling of critical information.

As illustrated in Figure 1, the model's attention distribution over the image is highly imbalanced: only a small proportion of regions correspond to meaningful semantic units, such as the spider and food in the yellow boxes. Meanwhile, some high-attention regions, marked by red boxes, do not contain elements relevant to the textual semantics. These observations motivate our approach: explicitly identifying semantically relevant tokens in the input and using them to refine the model's representation for retrieval.

Therefore, we propose KePIR, a multimodal embedding learning method that guides representation learning with key positional information. Specifically, we first prompt a multimodal large language model to extract textual keywords from the input. We then locate key visual token indices according to the cross-modal attention between textual keywords and visual tokens, and further filter visual sink tokens using sink dimension values. Based on the extracted key positional information, we combine contrastive learning with masked reconstruction guided by key positional information, encouraging the multimodal encoder to preserve discriminative fine-grained cues in the embeddings.

To summarize, KePIR is a universal multimodal retrieval method that guides embedding learning of a vision-language model (VLM) backbone with key positional information. Our main contributions are as follows: (i) We design a key positional information extraction pipeline that extracts textual keywords with a multimodal large language model, selects key visual tokens via cross-modal attention, and filters visual sink tokens using sink dimension values. (ii) We introduce a dual-decoder masked reconstruction mechanism that uses key positional information to determine reconstruction targets and jointly optimizes it with contrastive learning. (iii) Experiments on the M-BEIR benchmark show that KePIR achieves better overall performance than LamRA-Ret and CORAL under the same Qwen3-VL-2B-Instruct backbone, with consistent gains on most retrieval subtasks.

2 Related Work

2.1 Universal Multimodal Retrieval Models

Early multimodal retrieval methods relied on dual-tower architectures (Zhai et al., 2023; Sun et al., 2023), but such architectures struggle to handle complex instructions and diverse query-candidate modality combinations. UniIR (Wei et al., 2023) alleviates this issue by appending task instructions to queries and adapting pretrained vision-language models CLIP (Radford et al., 2021) and BLIP (Li et al., 2022), enabling unified retrieval across text, image, and image-text inputs. As vision-language models (VLMs) show stronger abilities in instruction understanding and unified multimodal input processing, recent studies further adapt them as backbones for universal multimodal retrieval. VLM2Vec (Jiang et al., 2024) trains VLMs to produce unified embeddings for diverse retrieval tasks, while LamRA (Liu et al., 2025) improves VLM-based retrieval and reranking through multi-stage training. Despite their improved universality, these methods mainly rely on global representation alignment, diluting key fine-grained information that is critical for retrieval.

2.2 Masked Modeling for Embeddings

Masked modeling originates from self-supervised methods such as BERT (Devlin et al., 2019), where models are trained to predict masked input content, thereby improving encoder representations in modeling contextual dependencies and semantic information. Since retrieval embeddings must compress inputs while preserving matching-related information, masked reconstruction has also been widely adopted for retrieval representation learning. RetroMAE (Xiao et al., 2022) introduces masked auto-encoding into dense text retrieval, UniMAE (Qiao et al., 2025) extends this idea to decoder-only LLMs, and CORAL (Chow et al.,

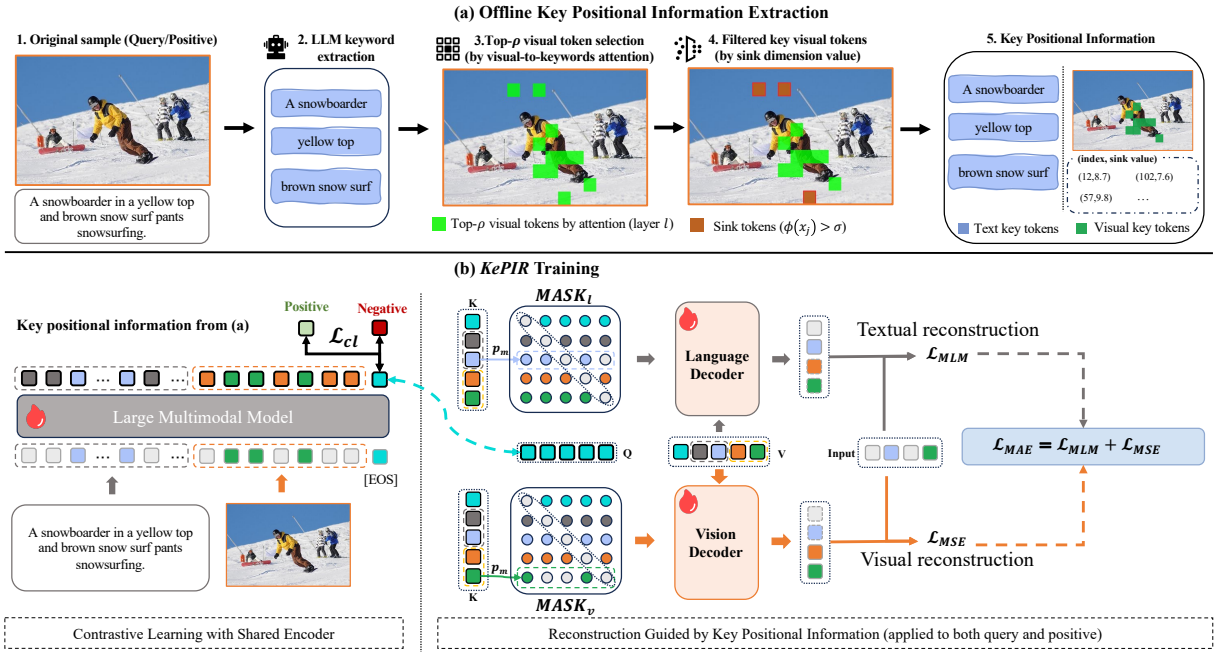


Figure 2: **The overall architecture of KePIR.** (a) Offline key positional information extraction identifies textual key tokens with an LLM, selects key visual tokens by cross-modal attention, and filters visual sink tokens using sink dimension value. (b) During training, KePIR combines contrastive learning with masked reconstruction guided by key positional information, independently applied to both the query and positive candidate through dual decoders.

2025), which incorporates textual and visual reconstruction into multimodal retrieval with joint contrastive learning.

However, existing methods usually rely on randomly selected reconstruction targets, which may provide supervision signals weakly related to retrieval objectives (Long et al., 2022). Such random targets can force the model to reconstruct frequent or uninformative tokens, rather than the fine-grained cues that determine retrieval relevance. In contrast, KePIR explicitly identifies key textual and visual information related to semantic matching and uses key positional information to guide masked reconstruction. In addition, CORAL introduces cross-sample conditional reconstruction into multimodal retrieval representation learning, while universal multimodal retrieval includes scenarios where queries and candidates have inconsistent modality combinations. This makes cross-sample reconstruction susceptible to non-unique cross-modal correspondence and noisy correspondence (Song and Soleymani, 2019; Chun et al., 2021; Huang et al., 2021). Accordingly, KePIR uses the key positional information of queries and candidates separately to impose independent reconstruction constraints, so that each reconstruction target is aligned with its own input representation.

2.3 Attention Sink

In transformer models, the phenomenon of attention sink was initially observed in language models (Xiao et al., 2023), where certain meaningless tokens (e.g., <BOS>) are assigned disproportionately high attention weights. More recently, similar behavior has been identified in VLMs, where certain irrelevant image tokens consistently attract high attention, despite being unrelated to the text (Sun et al., 2024). Motivated by this phenomenon, we introduce the attention sink mechanism to explicitly filter out irrelevant visual tokens during key visual token selection, thereby reducing noise interference in the visual modality.

3 Method

In this section, we describe key positional information extraction and the KePIR training objective that combines contrastive learning with masked reconstruction guided by key positional information.

3.1 Problem Formulation

We consider universal multimodal retrieval, where a query q retrieves relevant items from a candidate set $\mathcal{C} = \{c_i\}_{i=1}^{N}$. Queries and candidates may follow different modality combinations, including text-only, image-only, and image–text inputs. We

represent them as

$$q = \{q^T, q^V\}, \quad c_i = \{c_i^T, c_i^V\}, \tag{1}$$

where $(\cdot)^T$ denotes the textual component and $(\cdot)^V$ denotes the visual component. Either component may be empty.

Given a shared embedding function $f_\theta(\cdot)$ built on the multimodal encoder Φ_{enc}, we obtain

$$\mathbf{e}_q = f_\theta(q), \quad \mathbf{e}_{c_i} = f_\theta(c_i). \tag{2}$$

We rank candidates by cosine similarity:

$$s(q, c_i) = \cos(\mathbf{e}_q, \mathbf{e}_{c_i}). \tag{3}$$

The goal is to learn f_θ such that relevant query–candidate pairs are closer than irrelevant ones in the shared embedding space.

3.2 Key Positional Information Extraction

Before training, we extract key positional information relevant to retrieval from each training sample offline, based on its available modalities.

LLM-based Textual Keyword Extraction. For a training sample $x = \{x^T, x^V\}$, where x^T or x^V may be absent depending on the retrieval task, we extract textual keywords for samples with textual input using a multimodal large language model, as illustrated in step (a.2) of Figure 2.

Let $\mathcal{I}_{\text{txt}}$ denote the index set of textual tokens in x^T. If the sample also contains a visual component x^V, its visual token index set is denoted as $\mathcal{I}_{\text{vis}}$; otherwise, $\mathcal{I}_{\text{vis}}$ is treated as an empty set. Accordingly, the overall token index set is defined as

$$\mathcal{I} = \mathcal{I}_{\text{txt}} \cup \mathcal{I}_{\text{vis}}. \tag{4}$$

We feed x into the multimodal large language model to extract key phrases from its textual component x^T. The extracted key phrases are then aligned with the tokenized textual sequence indexed by $\mathcal{I}_{\text{txt}}$ according to their positions in x^T, yielding the textual key-token index set $\mathcal{K}_{\text{txt}} \subseteq \mathcal{I}_{\text{txt}}$. The prompt design and implementation details for keyword extraction are provided in Appendix A.

Attention-Based Visual Key Token Selection. For samples containing visual inputs, we select candidate visual key tokens according to the attention weights from visual to text tokens, as shown in step (a.3) of Figure 2. For image-only samples, we append the fixed auxiliary instruction "please find all key areas in this image" only during offline key visual-token extraction, and treat the token positions of "key areas" as the temporary $\mathcal{K}_{\text{txt}}$. We then use the attention map from a selected layer l and aggregate the attention weights over all heads in this layer.

Let $\alpha_{i,j}^{l,h}$ denote the attention weight from visual token $j \in \mathcal{I}_{\text{vis}}$ to textual key token $i \in \mathcal{K}_{\text{txt}}$ at layer l and head h. We compute the keyword-guided importance score of each visual token by averaging over all textual key positions and all heads:

$$a_j = \frac{1}{|\mathcal{K}_{\text{txt}}| H} \sum_{i \in \mathcal{K}_{\text{txt}}} \sum_{h=1}^{H} \alpha_{i,j}^{l,h}, \quad j \in \mathcal{I}_{\text{vis}}, \tag{5}$$

where H denotes the number of attention heads in layer l. We then rank all visual tokens in descending order according to a_j and retain the top-ρ proportion of ranked tokens as preliminary key visual-token candidates. Let $K_\rho = \lceil \rho \cdot |\mathcal{I}_{\text{vis}}| \rceil$, where $\rho \in (0, 1]$ denotes the proportion of visual tokens retained after sorting by attention weight. Let π be a permutation of $\mathcal{I}_{\text{vis}}$ such that $a_{\pi_1} \geq a_{\pi_2} \geq \cdots \geq a_{\pi_{|\mathcal{I}_{\text{vis}}|}}$. The candidate set is defined as:

$$\mathcal{K}'_{\text{vis}} = \{\pi_m \mid 1 \leq m \leq K_\rho\}. \tag{6}$$

Visual Key Token Filtering by Sink Dimension Value. After obtaining the preliminary key visual-token candidates, we filter key visual tokens by their sink dimension values, as shown in step (a.4) of Figure 2. This step removes visual tokens that receive high attention weights but are likely not semantically meaningful regions.

Following Kang et al. (2025b), we compute the sink dimension value of visual token j at layer l as:

$$\phi(\mathbf{x}_j^l) = \max_{\tilde{d} \in D_{\text{sink}}} \left| \frac{\mathbf{x}_j^l[\tilde{d}]}{\sqrt{\frac{1}{D} \sum_{d=1}^{D} \left(\mathbf{x}_j^l[d]\right)^2}} \right|, \tag{7}$$

where $\mathbf{x}_j^l \in \mathbb{R}^D$ is the hidden state of visual token j, D denotes the hidden dimension, and D_{sink} denotes the predefined set of sink dimensions. We filter candidates using threshold σ:

$$\mathcal{K}_{\text{vis}} = \left\{ j \in \mathcal{K}'_{\text{vis}} \;\middle|\; \phi(\mathbf{x}_j^l) < \sigma \right\}. \tag{8}$$

3.3 KePIR Training

Contrastive Learning Loss $\mathcal{L}_{\text{cl}}$. We optimize retrieval with InfoNCE loss (Wu et al., 2021), pulling each query toward its positive candidate while pushing it away from other candidates in the

same mini-batch. Given a mini-batch of N paired query-candidate samples $\{(q_i, c_i)\}_{i=1}^{N}$, we define $\psi(q, c) = \exp(s(q, c)/\tau)$, where τ is the temperature coefficient and $s(q, c) = \cos(\mathbf{e}_q, \mathbf{e}_c)$. The contrastive loss is:

$$\mathcal{L}_{\text{cl}} = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{\psi(q_i, c_i)}{\sum_{j=1}^{N}\psi(q_i, c_j)}. \tag{9}$$

For each query q_i, c_i is its positive candidate, while $\{c_j \mid j \neq i\}$ are used as in-batch negatives.

Encoder Representation. We use an instruction template to format x and encode it with Φ_{enc}, obtaining hidden states $\mathbf{H} = \Phi_{\text{enc}}(x) \in \mathbb{R}^{L\times D}$, where L is the sequence length. We take the hidden state at the last valid token, denoted as <eos>, as $\mathbf{z} = \mathbf{H}_{\text{<eos>}} \in \mathbb{R}^{D}$, and use its normalized vector as the embedding $\mathbf{e} = f_\theta(x) = \text{norm}(\mathbf{z})$. The instruction template is provided in Appendix B.

Masked Reconstruction Guided by Key Positional Information. As illustrated in Figure 2(b), during KePIR training we introduce a language decoder $\Phi_{\text{dec}}^{\text{txt}}$ and a vision decoder $\Phi_{\text{dec}}^{\text{vis}}$ to reconstruct the textual and visual modalities, respectively. In the reconstruction process, we construct two attention mask matrices from key positional information, MASK_l and MASK_v, and apply them to the attention computation of the two decoders.

For each attention-matrix row corresponding to a key textual-token or key visual-token index in $\mathcal{K}_{\text{txt}}$ or $\mathcal{K}_{\text{vis}}$, we mask a fixed proportion p_m of its valid attention entries, where $p_m = 0.5$ in our experiments. Following RetroMAE (Xiao et al., 2022), we also mask the main diagonal entries of the attention matrix to prevent each position from directly attending to itself. Let $\gamma_{r,c} \sim \text{Bernoulli}(p_m)$ indicate whether a valid attention entry is masked. Formally, $\text{MASK}_l \in \mathbb{R}^{L\times L}$ and $\text{MASK}_v \in \mathbb{R}^{L\times L}$ are defined as

$$\text{MASK}_l[r, c] = \begin{cases} -\infty, & c \notin \mathcal{I}, \\ -\infty, & r = c, \\ -\infty, & \substack{r\in\mathcal{K}_{\text{txt}}, \\ \gamma_{r,c}=1}, \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

$$\text{MASK}_v[r, c] = \begin{cases} -\infty, & c \notin \mathcal{I}, \\ -\infty, & r = c, \\ -\infty, & \substack{r\in\mathcal{K}_{\text{vis}}, \\ \gamma_{r,c}=1}, \\ 0, & \text{otherwise}. \end{cases} \tag{11}$$

We then process the input with Φ_{enc} to obtain the hidden state sequence

$$\mathbf{H} = [\mathbf{H}_{\text{txt}}, \mathbf{H}_{\text{vis}}, \mathbf{H}_{\text{<eos>}}] \in \mathbb{R}^{L\times D}.$$

The decoder inputs are constructed from $\mathbf{H}$. Specifically, we set $\mathbf{q} = [\mathbf{H}_{\text{<eos>}}, \mathbf{H}_{\text{<eos>}}, \ldots, \mathbf{H}_{\text{<eos>}}]$ and let $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_L]$ denote the learnable decoder position embeddings. The query, key, and value projections are computed as $\mathbf{Q} = (\mathbf{q} + \mathbf{P})\mathbf{W}^Q$, $\mathbf{K} = \mathbf{H}\mathbf{W}^K$, and $\mathbf{V} = \mathbf{H}\mathbf{W}^V$.

With $\text{M} \in \{\text{MASK}_l, \text{MASK}_v\}$, the masked attention output inside each decoder is

$$\mathbf{A} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{D}} + \text{M}\right)\mathbf{V}. \tag{12}$$

The attention output is further processed by $\Phi_{\text{dec}}^{\text{txt}}$ to produce $\hat{\mathbf{H}}^{\text{txt}}$, or by $\Phi_{\text{dec}}^{\text{vis}}$ to produce $\hat{\mathbf{H}}^{\text{vis}}$.

Reconstruction Objectives. For textual reconstruction, we predict the original token at each textual key-token index. Let y_i denote the ground-truth token at position i. The masked language modeling loss is

$$\mathcal{L}_{\text{MLM}}(x) = -\frac{1}{|\mathcal{K}_{\text{txt}}|}\sum_{i\in\mathcal{K}_{\text{txt}}}\log P(y_i \mid \hat{\mathbf{H}}_i^{\text{txt}}). \tag{13}$$

For visual reconstruction, we reconstruct the encoder hidden states at key visual-token indices:

$$\mathcal{L}_{\text{MSE}}(x) = \frac{1}{|\mathcal{K}_{\text{vis}}|}\sum_{j\in\mathcal{K}_{\text{vis}}}\left\|\hat{\mathbf{H}}_j^{\text{vis}} - \mathbf{H}_j\right\|_2^2. \tag{14}$$

If a modality is absent, its corresponding reconstruction term is omitted. The masked reconstruction loss for x is

$$\mathcal{L}_{\text{MAE}}(x) = \mathcal{L}_{\text{MLM}}(x) + \mathcal{L}_{\text{MSE}}(x). \tag{15}$$

Training Objective. For a mini-batch of paired query-candidate samples $\{(q_i, c_i)\}_{i=1}^{N}$, we apply masked reconstruction to the query and positive candidate independently, using each side's own key positional information. The reconstruction loss is

$$\mathcal{L}_{\text{rec}} = \frac{1}{N}\sum_{i=1}^{N}\left[\mathcal{L}_{\text{MAE}}(q_i) + \mathcal{L}_{\text{MAE}}(c_i)\right]. \tag{16}$$

The final training objective combines contrastive learning and reconstruction:

$$\mathcal{L} = \mathcal{L}_{\text{cl}} + \lambda\mathcal{L}_{\text{rec}}, \tag{17}$$

where λ controls the contribution of the reconstruction objective.

Table 1: Main results on M-BEIR. The first row specifies retrieval tasks, where q^T and q^I denote text and image queries, and c^T and c^I denote text and image candidates. We compare zero-shot baselines, dual-encoder retrieval models, and Qwen3-VL-2B-Instruct-based methods. Following UniIR (Wei et al., 2023), Fashion200K and FashionIQ are evaluated using Recall@10, while all other tasks are evaluated using Recall@5.

Methods	$q^T \to c^I$			$q^T \to c^T$	$q^T \to (c^I, c^T)$		$q^I \to c^T$			$q^I \to c^I$	$(q^I, q^T) \to c^T$		$(q^I, q^T) \to c^I$		$(q^I, q^T) \to (c^I, c^T)$		Avg.
	VN R@5	COCO R@5	F200K R@10	WebQA R@5	EDIS R@5	WebQA R@5	VN R@5	COCO R@5	F200K R@10	NIGHTS R@5	OVEN R@5	InfoS R@5	FIQ R@10	CIRR R@5	OVEN R@5	InfoS R@5	
Zero-shot																	
CLIP-L	**43.30**	61.10	6.60	36.20	43.30	45.10	41.30	79.00	7.70	26.10	24.20	20.50	7.00	13.20	38.80	26.40	32.49
SigLIP	30.10	75.70	**36.50**	39.80	27.00	43.50	30.80	88.20	**34.20**	28.90	29.70	25.10	14.40	22.70	41.70	27.40	37.23
BLIP	16.40	74.40	15.90	44.90	26.80	20.30	17.20	83.20	19.90	27.40	16.10	10.20	2.30	10.60	27.40	16.60	26.85
BLIP2	16.70	63.80	14.00	38.60	26.90	24.50	15.00	80.00	14.20	25.40	12.20	5.50	4.40	11.80	27.30	15.80	24.76
Qwen3-VL-2B-Instruct	1.70	27.90	1.90	13.10	5.60	1.90	1.20	13.30	1.20	6.50	5.70	2.60	0.10	4.30	14.30	10.90	7.01
Backbone With Dual Encoder																	
UniIR-BLIP$_{FF}$	23.40	79.70	26.10	80.00	50.90	79.80	22.80	89.90	28.90	**33.00**	41.00	22.40	29.20	52.20	55.80	33.00	46.76
UniIR-CLIP$_{SF}$	42.60	**81.10**	18.00	84.70	**59.40**	78.70	**43.10**	**92.30**	18.30	32.00	45.50	27.90	24.40	44.60	67.60	48.90	50.57
Backbone With Qwen3-VL-2B-Instruct																	
LamRA-Ret	26.57	75.66	17.93	87.01	55.97	79.25	27.73	91.42	18.94	31.18	50.79	49.15	30.00	48.66	68.98	57.39	51.04
CORAL	25.46	77.44	18.32	87.86	50.82	78.89	27.88	90.60	18.55	31.18	47.97	45.62	29.89	49.52	67.55	55.91	50.22
VLM2Vec	26.50	78.24	17.56	86.72	54.14	78.29	27.56	91.46	18.86	30.71	49.50	49.88	**30.63**	49.40	68.63	56.60	50.92
Qwen3-VL-Embedding	23.49	78.35	14.95	81.47	44.65	63.44	22.42	87.54	15.12	31.46	42.75	37.74	23.31	48.42	58.75	46.35	45.01
KePIR (Ours)	28.09	79.39	19.43	**88.51**	53.86	**79.81**	27.96	91.52	19.73	31.51	**51.36**	**50.20**	30.40	**52.49**	**69.83**	**58.04**	**52.01**

4 Experiments

4.1 Experimental Setup

Datasets & Baselines. We trained and evaluated our proposed method on the M-BEIR dataset (Wei et al., 2023) and compared its performance against a series of baseline methods. The M-BEIR dataset encompasses eight distinct retrieval tasks in 10 different retrieval datasets, comprising a total of 1.1 million training samples.

We compare KePIR with representative baselines under the M-BEIR, including zero-shot vision-language models, dual-encoder retrieval models, and recent multimodal embedding methods. For CLIP-L, SigLIP, BLIP, BLIP2, UniIR-BLIP$_{FF}$, and UniIR-CLIP$_{SF}$, we report the results from UniIR (Wei et al., 2023). For the Qwen3-VL-2B-Instruct-based setting, we reproduce three representative training methods following their original papers: VLM2Vec, which adapts VLMs for retrieval with contrastive learning (Jiang et al., 2024); LamRA-Ret, which also uses contrastive learning but further introduces a multi-stage training pipeline (Liu et al., 2025); and CORAL, which incorporates textual and visual reconstruction with dual decoders as auxiliary objectives during contrastive fine-tuning (Chow et al., 2025). In addition, we directly evaluate the official Qwen3-VL-Embedding-2B model, a dedicated multimodal embedding model that maps diverse multimodal inputs into a unified representation space (Li et al., 2026). For controlled comparison, KePIR also adopts Qwen3-VL-2B-Instruct as the backbone.

Implementation Details. We adopt Qwen3-VL-2B-Instruct as the backbone model. The batch size per GPU is 16, and we use GradCache (Luyu Gao, 2021) with a chunk batch size of 4 for memory efficient contrastive training. The learning rate is 1×10^{-4}, and the temperature parameter is $\tau = 0.02$. The model is trained for one epoch. For visual key token selection, we set the retained visual token proportion to $\rho = 0.5$. In the visual sink tokens filtering step, we set the sink dimension value threshold to $\sigma = 34$ and use $D_{\text{sink}} = \{1793, 1999\}$ as the sink dimensions, with details provided in Appendix C. The dual decoder masked reconstruction loss weight is set to $\lambda = 0.1$. No hard negatives are used in any of our experiments. We adopt LoRA (Hu et al., 2022) fine-tuning with $r = 128$ and $\alpha = 256$ for the language model component. All experiments are conducted on 16 A100 80G GPUs.

4.2 Main Results

Table 1 reports the performance on the M-BEIR benchmark. Overall, KePIR achieves the highest average score among all compared methods. Compared with the strongest dual-encoder baseline UniIR-CLIP$_{SF}$, KePIR improves the average score from 50.57 to 52.01. This demonstrates the effectiveness of KePIR for UMR.

Among methods based on Qwen3-VL-2B-Instruct, KePIR achieves the best average performance and obtains the best results on most subtasks. Compared with LamRA-Ret and VLM2Vec, KePIR improves the average score by 0.97 and

Table 2: Effect of varying the key visual-token retention ratio ρ and auxiliary reconstruction weight λ on M-BEIR. Results are averaged by retrieval setting, where T, I, and IT denote text, image, and image-text inputs, respectively. Values in parentheses denote relative changes over the baseline trained only with contrastive learning. Detailed performance on each individual dataset is provided in Appendix D.1.

Method	ρ	λ	**Retrieval setting average**								**Avg.**
			$T\to I$	$T\to T$	$T\to IT$	$I\to T$	$I\to I$	$IT\to T$	$IT\to I$	$IT\to IT$	
w/ $\mathcal{L}_{\text{cl}}$ Only	–	–	40.77	86.72	66.22	45.96	30.71	49.69	40.02	62.62	50.92
KePIR	0.5	0.05	41.45	87.54	65.98	45.83	31.04	48.81	40.80	62.97	51.10 (+0.35%)
	0.5	0.1	42.30	88.51	**66.84**	**46.40**	**31.51**	**50.78**	41.45	**63.94**	**52.01** (+2.14%)
	0.5	0.2	42.33	88.84	65.57	46.03	31.18	50.27	41.34	63.20	51.62 (+1.37%)
	0.1	0.1	42.33	88.35	65.65	45.91	30.61	46.72	40.27	62.79	50.91 (-0.02%)
	0.3	0.1	42.29	87.94	65.89	45.96	30.61	49.37	**41.47**	62.65	51.38 (+0.90%)
	0.7	0.1	**42.68**	**89.21**	65.68	45.93	31.27	47.37	**41.55**	63.07	51.35 (+0.85%)

1.09 points. KePIR also outperforms the official Qwen3-VL-Embedding model on M-BEIR, while we note that Qwen3-VL-Embedding is designed as a general-purpose multimodal embedding model for broader retrieval scenarios rather than being specialized for this benchmark. These results show that, beyond contrastive learning and multi-stage training, key positional information provides useful supervision for preserving key retrieval cues in embeddings.

It is also worth noting that KePIR outperforms CORAL by 1.79 points in average score, although both methods introduce textual and visual reconstruction as auxiliary objectives. The difference lies in the masking and reconstruction stage. CORAL relies on cross-sample conditional reconstruction, where the representation from one side is used to reconstruct the paired target. In universal multimodal retrieval, however, queries and candidates may have different modality compositions, such as text-to-image, image-to-text, and image-text-to-image retrieval. This makes cross-sample reconstruction more vulnerable to non-unique or noisy cross-modal correspondences. In contrast, KePIR constructs attention masks from key positional information and applies reconstruction independently to the query and positive candidate, allowing each side to preserve its own key textual and visual information during embedding learning.

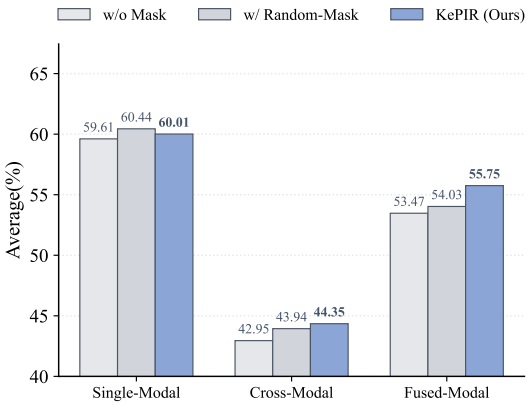


Figure 3: Comparison of different masking strategies across three modality-based task categories on M-BEIR. Average retrieval performance is evaluated for the baseline without masking (w/o Mask), random masking (w/ Random-Mask), and our proposed masking based on key positional information (KePIR). Detailed metrics are in Appendix D.2.

4.3 Ablation Study & Analysis

Balancing Auxiliary Reconstruction and Retention of Visual Tokens. We investigate the delicate balance between the auxiliary reconstruction objective and the retention of visual key tokens, as detailed in Table 2. The model achieves its optimal performance with an average score of 52.01 when balancing the primary contrastive loss with a moderate auxiliary reconstruction weight ($\lambda = 0.1$) and retaining half of the visual tokens ($\rho = 0.5$). When varying the reconstruction weight, we observe that insufficient supervision ($\lambda = 0.05$) yields only marginal improvements over the contrastive-only baseline, whereas an excessive auxiliary reconstruction constraint ($\lambda = 0.2$) begins to overshadow the primary retrieval objective, leading to a performance drop. Similarly, the model is highly sensitive to the retention ratio of key visual tokens. Aggressively discarding visual tokens ($\rho = 0.1$) loses critical semantic matching clues, yielding results roughly equivalent to the baseline. Conversely, retaining too large a proportion of visual tokens ($\rho = 0.7$) reintroduces noisy, irrelevant visual tokens that dilute the semantic representation of key

information, validating the necessity of precisely balancing key positional information extraction.

Scaling Backbone Model Sizes. We investigate the scalability of our approach across different backbone model sizes, as detailed in Figure 4. Notably, as the model scales from 2B to 4B parameters, KePIR consistently outperforms the contrastive-only baseline, and the performance margin between them widens significantly—leaping from a 0.32 to a 1.66 absolute improvement. This amplified gain demonstrates that our auxiliary reconstruction mechanism based on key positional information can effectively leverage the stronger cross-modal understanding capabilities of larger models.

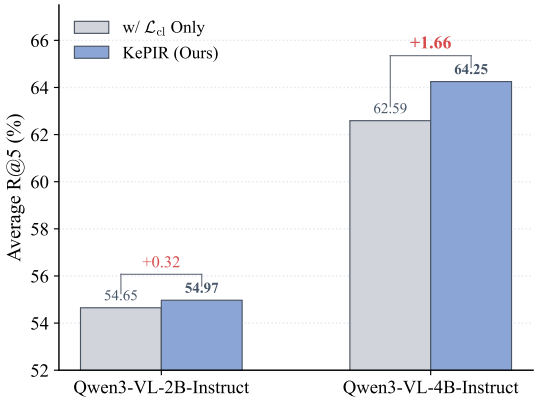


Figure 4: Retrieval performance of KePIR and the contrastive-only baseline across different backbone model sizes. Evaluated on the image-text to image-text retrieval task using OVEN and InfoSeek. Detailed metrics are in Appendix D.4.

Comparison of Masking Strategies. To evaluate the effectiveness of the key-position masking strategy, we compare it against two baselines: (1) **w/o Mask**, which applies no masking to the attention matrix during reconstruction, and (2) **w/ Random-Mask**, which randomly masks 50% of the elements in the attention matrix during reconstruction.

As shown in Figure 3, the masking mechanism exhibits a regularization effect in Single-Modal tasks, where both random masking (60.44%) and KePIR (60.01%) outperform the unmasked baseline (59.61%). In Cross-Modal tasks, where queries and candidates are from different modalities, masking also improves retrieval performance over the unmasked baseline (42.95%): random masking reaches 43.94%, while KePIR further improves the result to 44.35%.

In Fused-Modal tasks, which involve more complex feature interactions, the unmasked baseline achieves 53.47%. Random masking improves this to 54.03%, while KePIR reaches the highest performance of 55.75%. These results indicate that when processing complex modality combinations, KePIR effectively captures key semantic information, providing more accurate supervision signals for the retrieval task.

Table 3: Comparison of performance with and without visual sink token filtering (w/o Filter) across three modality-based task categories on M-BEIR. Detailed metrics are in Appendix D.3.

Method	Single	Cross	Fused	Avg.
w/o Filter	59.91	44.03	54.45	51.22
KePIR (Ours)	**60.01**	**44.35**	**55.75**	**52.01**

Effectiveness of Visual Sink Token Filtering. As shown in Table 3, the sink dimension value filtering strategy improves the average retrieval performance from 51.22% to 52.01%. Notably, the performance gain is positively correlated with modality complexity: single-modality tasks show a marginal improvement of 0.10% and cross-modality tasks see a 0.32% gain, whereas fused-modality tasks achieve a significant absolute increase of 1.30%, jumping from 54.45% to 55.75%. This demonstrates that filtering by sink dimension values can effectively eliminate visual tokens that exhibit high attention but lack semantic relevance, thereby reducing the interference of these irrelevant visual tokens on the final embedding.

5 Conclusion

In this paper, we propose KePIR to enhance the effectiveness of universal multimodal retrieval. The main idea of KePIR is to explicitly capture fine-grained positional information to guide the embedding learning process. Specifically, KePIR extracts textual keywords via an LLM, selects key visual tokens through cross-modal attention, and filters visual noise using sink dimension values. Regarding the optimization of KePIR, besides applying a contrastive loss for global semantic alignment, we introduce a dual-decoder masked reconstruction mechanism guided by the extracted key positional information. We conduct experiments on the M-BEIR benchmark to demonstrate the advantages of KePIR compared to existing retrieval approaches.

Limitations

Although KePIR shows advantages over existing retrieval methods, it still has several limitations. The key visual-token extraction process relies on the attention distributions and internal representations of large multimodal models. When applied to backbones with weaker multimodal understanding capabilities or substantially different architectures, the identified key tokens may become less reliable, which can affect the effectiveness of masked reconstruction. In addition, the dual-decoder reconstruction objective introduces extra computation and memory overhead during training. These decoders are used only as auxiliary supervision modules and are removed after training, so KePIR keeps the same inference and retrieval cost as the underlying embedding model. However, the additional reconstruction branches still increase the resource requirements of the training stage.

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2015. Vqa: Visual question answering. *International Journal of Computer Vision*, 123.

Manoj Ghuhan Arivazhagan, Lan Liu, Peng Qi, Xinchi Chen, William Yang Wang, and Zhiheng Huang. 2023. Hybrid hierarchical retrieval for open-domain question answering. In *Findings of the Association for Computational Linguistics: ACL 2023*.

Alberto Baldrati, Lorenzo Agnolucci, Marco Bertini, and A. Bimbo. 2023. Zero-shot composed image retrieval with textual inversion. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Chang Che, Qunwei Lin, Xinyu Zhao, Jiaxin Huang, and Liqiang Yu. 2023. Enhancing multimodal understanding with clip-based image-to-text transformation. In *Proceedings of the 2023 6th International Conference on Big Data Technologies*.

Wei Chow, Yuan Gao, Linfeng Li, Xian Wang, Qi Xu, Hang Song, Lingdong Kong, Ran Zhou, Yi Zeng, Yidong Cai, Botian Jiang, Shilin Xu, Jiajun Zhang, Minghui Qiu, Xiangtai Li, Tianshu Yang, Siliang Tang, and Juncheng Li. 2025. Merit: Multilingual semantic retrieval with interleaved multi-condition query. *arXiv preprint arXiv:2506.03144*.

Sanghyuk Chun, Seong Joon Oh, Rafael Sampaio De Rezende, Yannis Kalantidis, and Diane Larlus. 2021. Probabilistic embeddings for cross-modal retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. 2025. When attention sink emerges in language models: An empirical view. In *International Conference on Learning Representations*, volume 2025.

Jiafeng Guo, Yinqiong Cai, Yixing Fan, Fei Sun, Ruqing Zhang, and Xueqi Cheng. 2022. Semantic models for the first-stage retrieval: A comprehensive review. *ACM Transactions on Information Systems (TOIS)*, 40(4).

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2).

Zhenyu Huang, Guocheng Niu, Xiao Liu, Wenbiao Ding, Xinyan Xiao, Hua Wu, and Xi Peng. 2021. Learning with noisy correspondence for cross-modal matching. In *Advances in Neural Information Processing Systems*, volume 34. Curran Associates, Inc.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2024. Vlm2vec: Training vision-language models for massive multimodal embedding tasks. *arXiv preprint arXiv:2410.05160*.

Kushal Kafle and Christopher Kanan. 2016. Visual question answering: Datasets, algorithms, and future challenges. *Comput. Vis. Image Underst.*, 163.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. 2025a. See what you are told: Visual attention sink in large multimodal models. *arXiv preprint arXiv:2503.03321*.

Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. 2025b. See what you are told: Visual attention sink in large multimodal models. In *International Conference on Learning Representations*, volume 2025.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. ReferItGame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. 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*.

Fanheng Kong, Jingyuan Zhang, Yahui Liu, Hongzhi Zhang, Shi Feng, Xiaocui Yang, Daling Wang, Yu Tian, Qi Wang, Fuzheng Zhang, and Guorui Zhou. 2025. Modality curation: Building universal embeddings for advanced multimodal information retrieval. *arXiv preprint arXiv:2505.19650*.

Jiayi Kuang, Ying Shen, Jingyou Xie, Haohao Luo, Zhe Xu, Ronghao Li, Yinghui Li, Xianfeng Cheng, Xika Lin, and Yu Han. 2025. Natural language understanding and inference with mllm in visual question answering: A survey. *ACM Computing Surveys*, 57(8).

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33. Curran Associates, Inc.

Hui Li, Tsz Nam Chan, Man Lung Yiu, and Nikos Mamoulis. 2017. Fexipro: Fast and exact inner product retrieval in recommender systems. *Proceedings of the 2017 ACM International Conference on Management of Data*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, and others. 2026. Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*.

Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. 2024. Monkey: Image resolution and text label are important things for large multi-modal models. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Weizhe Lin, Jinghong Chen, Jingbiao Mei, Alexandru Coca, and Bill Byrne. 2023. Fine-grained late-interaction multi-modal retrieval for retrieval augmented visual question answering. *Advances in Neural Information Processing Systems*, 36.

Yikun Liu, Yajie Zhang, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiangchao Yao, Yanfeng Wang, and Weidi Xie. 2025. Lamra: Large multimodal model as your advanced retrieval assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*.

Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. 2021. Image retrieval on real-life images with pre-trained vision-and-language models. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Dingkun Long, Yanzhao Zhang, Guangwei Xu, and Pengjun Xie. 2022. Retrieval oriented masking pre-training language model for dense passage retrieval. *arXiv preprint arXiv:2210.15133*.

Jiawei Han Jamie Callan Luyu Gao, Yunyi Zhang. 2021. Scaling deep contrastive learning batch size under memory limited setup. In *Proceedings of the 6th Workshop on Representation Learning for NLP*.

Alessandro Magnani, Feng Liu, Min Xie, and Somnath Banerjee. 2019. Neural product retrieval at walmart.com. *Companion Proceedings of The 2019 World Wide Web Conference*.

Dan Qiao, Yuan Gao, Zheming Yang, Di Yang, Ziheng Wu, Pengcheng Lu, Minghui Qiu, Juntao Li, and Min Zhang. 2025. Decoder-only LLMs can be masked auto-encoders. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vienna, Austria. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*.

Yale Song and Mohammad Soleymani. 2019. Polysemous visual-semantic embedding for cross-modal retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Mingjie Sun, Xinlei Chen, J. Zico Kolter, and Zhuang Liu. 2024. Massive activations in large language models. *arXiv preprint arXiv:2402.17762*.

Quan Sun, Yuxin Fang, Ledell Yu Wu, Xinlong Wang, and Yue Cao. 2023. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*.

Zecheng Tang, Baibei Ji, Juntao Li, Lijun Wu, Haijia Gui, and Min Zhang. 2025. Revisiting long-context modeling from context denoising perspective. *arXiv preprint arXiv:2510.05862*.

Lucas Ventura, Antoine Yang, Cordelia Schmid, and Gül Varol. 2024. Covr: Learning composed video retrieval from web video captions. *ArXiv*, abs/2308.14746.

Anthony D Wagner, E Juliana Paré-Blagoev, Jill Clark, and Russell A Poldrack. 2001. Recovering meaning: left prefrontal cortex guides controlled semantic retrieval. *Neuron*, 31(2).

Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. 2023. Uniir: Training and benchmarking universal multimodal information retrievers. *arXiv preprint arXiv:2311.17136*.

Chuhan Wu, Fangzhao Wu, and Yongfeng Huang. 2021. Rethinking infonce: How many negative samples do you need? *arXiv preprint arXiv:2105.13003*.

Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. 2023. Cap4video: What can auxiliary captions do for text-video retrieval? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2023. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*.

Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. 2022. RetroMAE: Pre-training retrieval-oriented language models via masked auto-encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Kai Zhang, Yi Luan, Hexiang Hu, Kenton Lee, Siyuan Qiao, Wenhu Chen, Yu Su, and Ming-Wei Chang. 2024. Magiclens: Self-supervised image retrieval with open-ended instructions. *ArXiv*, abs/2403.19651.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Ruochen Zhao, Hailin Chen, Weishi Wang, Fangkai Jiao, Xuan Long Do, Chengwei Qin, Bosheng Ding, Xiaobao Guo, Minzhi Li, Xingxuan Li, and others. 2023. Retrieving multimodal information for augmented generation: A survey. *arXiv preprint arXiv:2303.10868*.

Shiyu Zhao, Zhenting Wang, Felix Juefei-Xu, Xide Xia, Miao Liu, Xiaofang Wang, Mingfu Liang, Ning Zhang, Dimitris N Metaxas, and Licheng Yu. 2025. Accelerating multimodal large language models by searching optimal vision token reduction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*.

A Instructions for Textual Keyword Extraction from Samples

To identify keywords from multimodal samples, we use Qwen3-VL-32B-Instruct to preprocess the textual components. Depending on whether the input contains an image, we employ two prompt templates.

A.1 Textual Keyword Extraction with Image Context

For image-text samples, the model uses the image as contextual evidence to identify keywords in the accompanying text that are visually grounded and useful for retrieval. The prompt for this setting is shown in Figure 5.

A.2 Textual Keyword Extraction without Image Context

For text-only samples, the model extracts keywords directly from the textual input without relying on visual context. The prompt is designed to preserve concise and discriminative phrases that appear verbatim in the original text, as shown in Figure 6.

B Input Template for Retrieval Embedding

Following Qwen3-VL-Embedding (Li et al., 2026), we format each query or candidate as a Qwen3-VL conversational input, as shown in Figure 7. The instruction is placed in the system message, and the multimodal instance is placed in the user message. We append <|endoftext|> in the assistant turn and use its last hidden state as the dense representation. In Section 3.3, this token is denoted as <eos>.

C Visual Attention Sinks Analysis

We analyze visual attention sinks for Qwen3-VL-2B-Instruct and Qwen3-VL-4B-Instruct on RefCOCOg (Kazemzadeh et al., 2014), a referring expression grounding dataset in which each image is paired with a natural-language expression and a bounding box of the referred object. The bounding-box annotations allow us to evaluate whether high-attention visual tokens correspond to the region described by the text.

The analysis consists of three steps. First, we identify sink dimensions by counting the largest normalized activation dimensions of the <BOS> token on RefCOCOg. Second, we select the attention layer whose top-attended visual tokens best cover

Prompt for Textual Keyword Extraction with Image Context

User: You are a strict image-grounded text keyword extractor. Your task is to extract keywords or phrases from the user-provided text that can be directly grounded to visible content in the image.
Be conservative: if the text does not contain any explicit object, part, attribute, text, position, or relation that can directly correspond to visible image content, return an empty list.

Rules:

- Each keyword must be a continuous substring copied verbatim from the user-provided text.
- Do not rewrite, summarize, translate, complete, infer, or use synonyms.
- Do not extract words that do not physically appear in the text.
- Extract **0 to 5 keywords or phrases**.
- Each keyword should be concise, preferably **1 to 4 words**.
- **Keywords must not overlap**:
  - One keyword must not be a substring of another.
  - Two keywords must not cover the same span of text.
  - If two candidates are semantically redundant, keep only the more specific and visually groundable one.
- Extract only text spans that can directly correspond to visible image content, such as:
  - Visible objects: shirt, pants, table, phone, dog, building, etc.
  - Visible parts: sleeve, collar, screen, wheel, handle, etc.
  - Visible attributes: white, red, damaged, striped, round, standing, etc.
  - Visible text or logos.
  - Visible spatial relations: left, above, next to, in hand, etc.
- Do not extract words that only describe the retrieval target but do not directly correspond to a visible region in the current image.
  Example: for an image showing white clothing plus the query "find gray and larger pants in the same style", if "gray", "larger", and "pants" are not directly visible in the image, return no keywords for those terms.
- Do not extract task intent words such as `find`, `search`, `retrieve`, `recommend`, `similar`, `same style`, `related item`, or `relevant content`.
- If the text is only an instruction, retrieval intent, comparison request, or contains no text span that can be directly grounded to a visible image region, return an empty keyword list.

Output Requirements:

- Return **ONLY one fenced JSON block**.
- The output must start with "```json" and end with "```".
- Do not include explanations, comments, or any text outside the fenced JSON block.
- Use exactly the following format:

```
{
  "keywords": ["Keyword1", "Keyword2", "Keyword3"]
}
```

For an empty result, return exactly:

```
{
  "keywords": []
}
```

User-provided text:

```
{{text}}
```

Figure 5: Prompt for textual keyword extraction with image context.

Prompt for Textual Keyword Extraction Without Image Context

User: You are a strict text keyword extractor. Your task is to extract only truly important keywords or phrases that represent the core meaning of the user-provided text and help precise search retrieval.
Be conservative: if the text does not contain sufficiently important, specific, and retrieval-useful keywords, return an empty list. Do not extract weak keywords just to fill the output.

Rules:

- Each keyword must be a continuous substring copied verbatim from the user-provided text.
- Do not rewrite, summarize, translate, complete, infer, or use synonyms.
- Do not extract words that do not physically appear in the text.
- Extract **0 to 5 keywords or phrases**.
- Each keyword should be concise, preferably **1 to 4 words**.
- Prefer extracting:
  - Core entities, subjects, objects, events, or concepts.
  - Important qualifiers or attributes that significantly narrow retrieval scope.
  - Proper nouns, technical terms, or specific noun phrases.
- Do not extract low-value or generic expressions, such as:
  - Pronouns: `this`, `that`, `it`, `these`, `those`.
  - Empty task words: `find`, `search`, `look for`, `recommend`, `related`, `content`, `information`.
  - Standalone generic adjectives: `good`, `big`, `small`, `new`, `large`, unless they are part of an important specific phrase.
- **Keywords must not overlap**:
  - One keyword must not be a substring of another.
  - Two keywords must not cover the same span of text.
  - If two candidates are semantically redundant, keep only the more specific and more retrieval-useful one.
- If only generic, non-discriminative, or retrieval-useless words can be extracted, return an empty keyword list.

Output Requirements:

- Return **ONLY one fenced JSON block**.
- The output must start with "```json" and end with "```".
- Do not include explanations, comments, or any text outside the fenced JSON block.
- Use exactly the following format:

```
{
  "keywords": ["Keyword1", "Keyword2", "Keyword3"]
}
```

For an empty result, return exactly:

```
{
  "keywords": []
}
```

User-provided text:

```
{{text}}
```

Figure 6: Prompt used for textual keyword extraction without image context.

Input template for retrieval embedding

```
<|im_start|>system
{Instruction}
<|im_end|>
<|im_start|>user
{Instance}
<|im_end|>
<|im_start|>assistant
<|endoftext|>
```

Figure 7: Input Template for Retrieval Embedding

the ground-truth bounding box. Third, with the selected layer and sink dimensions fixed, we search for the sink filtering threshold σ.

Sink Dimensions Identification. We identify sink dimensions by analyzing the normalized activation pattern of the sequence-initial token in Qwen3-VL on RefCOCOg. For each RefCOCOg sample, we construct the standard image–text chat input using the image and its referring expression. At each transformer layer l, we take the hidden state of the first input token and compute a per-dimension sink score:

$$\eta_d^l = \left| \frac{\mathbf{x}_0^l[d]}{\sqrt{\frac{1}{D}\sum_{d'=1}^{D}(\mathbf{x}_0^l[d'])^2}} \right|, \tag{18}$$

where $\mathbf{x}_0^l$ denotes the hidden state of the first input token at layer l, and D is the hidden dimension.

For each sample and each layer, we select the top-

Table 4: Identified sink dimensions for Qwen3-VL-Instruct backbones on RefCOCOg.

Model	Sink Dimensions
Qwen3-VL-2B-Instruct	{1793, 1999}
Qwen3-VL-4B-Instruct	{12, 19}

Table 5: Selected transformer layers for visual key positional information extraction on RefCOCOg.

Model	Selected Transformer Layer
Qwen3-VL-2B-Instruct	14
Qwen3-VL-4B-Instruct	21

k dimensions with the largest η_d^l values, with $k = 2$ in our implementation. We then aggregate their occurrence counts over all RefCOCOg samples and all transformer layers. The dimensions that appear most frequently among these top-k sets are treated as sink dimensions and used as D_{sink} in the subsequent visual token filtering step.

As shown in Table 4, each Qwen3-VL backbone exhibits a small set of recurring high-activation dimensions. We identify these dimensions separately for each backbone and use the corresponding set as D_{sink} when computing visual token sink scores.

Transformer Layer Selection for Visual Key Positional Information Extraction. To determine which transformer layer provides the most reliable attention signal for extracting visual key positional information, we evaluate the alignment between attention-ranked visual tokens and ground-truth referred regions on RefCOCOg. For each sample, we construct an image–text input using the image and its referring expression. The ground-truth bounding box is mapped onto the visual-token grid of the processed image. A visual token is regarded as a bounding-box token if the overlap between its token cell and the ground-truth box is at least 0.1 of the token-cell area.

For each layer l, we compute the attention score of each visual token by aggregating the attention weights from visual to text tokens, where the textual positions correspond to the referring expression. Let $\mathcal{Q}_s$ denote the token positions of the referring expression in sample s, and let $\mathcal{V}_s$ denote its visual-token indices. The attention score of visual token $j \in \mathcal{V}_s$ at layer l is

$$a_{s,j}^l = \frac{1}{|\mathcal{Q}_s| H} \sum_{i \in \mathcal{Q}_s} \sum_{h=1}^{H} \alpha_{i,j}^{l,h}. \tag{19}$$

We rank all visual tokens in $\mathcal{V}_s$ by $a_{s,j}^l$ in descending order. For each sample s, the number of selected visual tokens is

$$K_s = \min\left(|\mathcal{V}_s|, \lceil m|\mathcal{B}_s| \rceil\right), \tag{20}$$

where $\mathcal{B}_s$ denotes the set of bounding-box tokens, and $m = 2$ in our experiments. Let π_s^l be the descending-order permutation of $\mathcal{V}_s$ induced by $a_{s,j}^l$. The selected top-attended visual-token set is

$$\mathcal{S}_s^l = \left\{ \pi_{s,r}^l \mid 1 \le r \le K_s \right\}. \tag{21}$$

We evaluate $\mathcal{S}_s^l$ using bounding-box precision and recall:

$$P_s^l = \frac{|\mathcal{S}_s^l \cap \mathcal{B}_s|}{|\mathcal{S}_s^l|}, \quad R_s^l = \frac{|\mathcal{S}_s^l \cap \mathcal{B}_s|}{|\mathcal{B}_s|}. \tag{22}$$

The final transformer layer is selected by the mean bounding-box recall:

$$l^* = \arg\max_l \frac{1}{|\mathcal{D}_{\text{eff}}|} \sum_{s \in \mathcal{D}_{\text{eff}}} R_s^l, \tag{23}$$

where $\mathcal{D}_{\text{eff}}$ denotes the set of samples whose ground-truth box covers at least one visual token.

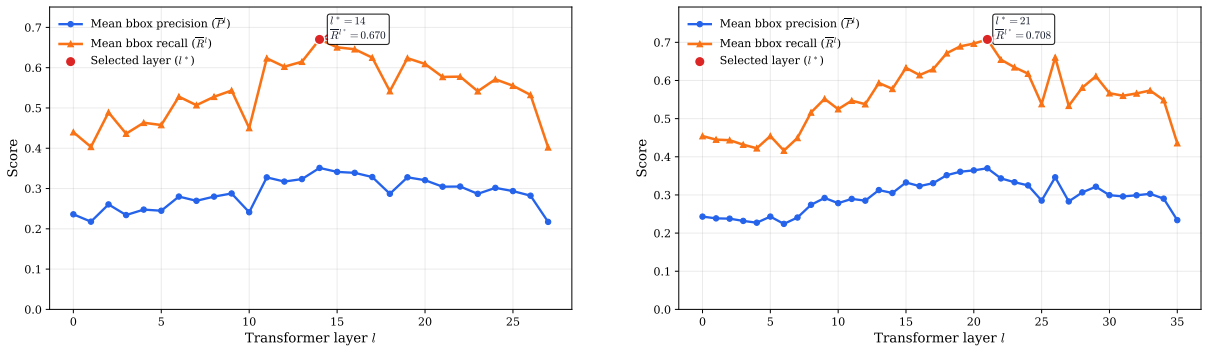
As shown in Figure 8, $\overline{R}^l$ varies across transformer layers, and the peak layer is selected as l^*. We perform this analysis separately for each Qwen3-VL-Instruct backbone, and the selected layers are reported in Table 5.

Sink Dimension Value Threshold Selection. After identifying the sink dimensions and selecting the transformer layer for visual key positional information extraction, we determine the filtering threshold for sink dimension values on RefCOCOg. For each Qwen3-VL-Instruct backbone, we use its selected transformer layer and corresponding sink dimensions. We first rank visual tokens by the attention score $a_{s,j}^{l^*}$ and select the top K_s tokens as in the layer-selection step. We then filter these attention-ranked visual tokens according to their sink dimension values.

For a visual token j in sample s, its sink dimension value is computed as

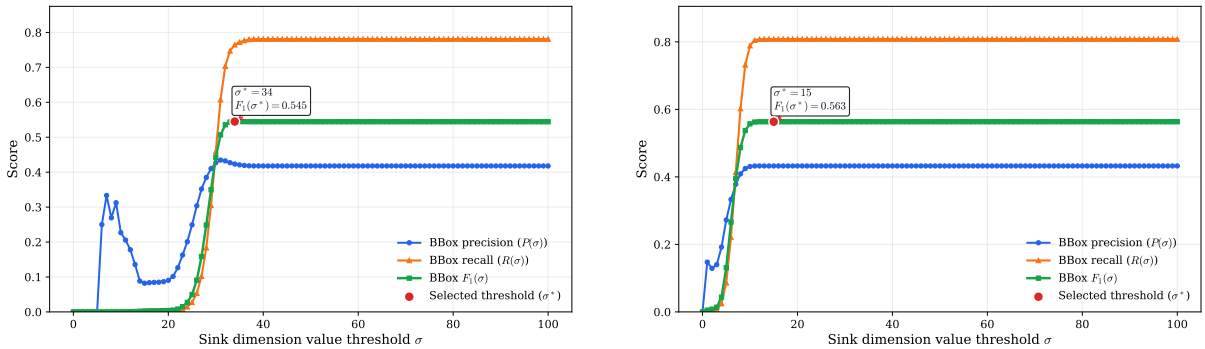
$$\phi_{s,j}^{l^*} = \max_{\tilde{d} \in D_{\text{sink}}} \left| \frac{\mathbf{x}_{s,j}^{l^*}[\tilde{d}]}{\sqrt{\frac{1}{D} \sum_{d=1}^{D} (\mathbf{x}_{s,j}^{l^*}[d])^2}} \right|, \tag{24}$$

where $\mathbf{x}_{s,j}^{l^*}$ is the hidden state of visual token j at the selected transformer layer l^*, and D_{sink} denotes



(a) Qwen3-VL-2B-Instruct (b) Qwen3-VL-4B-Instruct

Figure 8: Transformer layer selection for visual key positional information extraction on RefCOCOg. The selected layer corresponds to the highest mean bounding-box recall among top-attended visual tokens.



(a) Qwen3-VL-2B-Instruct (b) Qwen3-VL-4B-Instruct

Figure 9: Sink dimension value threshold selection on RefCOCOg. The selected threshold σ^* maximizes bounding-box F_1 after filtering top-attended visual tokens by sink dimension values.

the sink dimensions identified for the corresponding backbone.

Given a candidate threshold σ, we retain only the selected visual tokens whose sink dimension values do not exceed σ:

$$\mathcal{S}_s^{l^*}(\sigma) = \left\{ j \in \mathcal{S}_s^{l^*} \mid \phi_{s,j}^{l^*} \leq \sigma \right\}. \tag{25}$$

We evaluate the filtered token set using bounding-box precision, recall, and F1:

$$P(\sigma) = \frac{\sum_{s \in \mathcal{D}_{\text{eff}}} |\mathcal{S}_s^{l^*}(\sigma) \cap \mathcal{B}_s|}{\sum_{s \in \mathcal{D}_{\text{eff}}} |\mathcal{S}_s^{l^*}(\sigma)|}, \tag{26}$$

$$R(\sigma) = \frac{\sum_{s \in \mathcal{D}_{\text{eff}}} |\mathcal{S}_s^{l^*}(\sigma) \cap \mathcal{B}_s|}{\sum_{s \in \mathcal{D}_{\text{eff}}} |\mathcal{B}_s|}, \tag{27}$$

$$F_1(\sigma) = \frac{2P(\sigma)R(\sigma)}{P(\sigma) + R(\sigma)}. \tag{28}$$

We sweep σ from 0 to 100 with a step size of 1 and select the threshold that maximizes $F_1(\sigma)$:

$$\sigma^* = \arg\max_{\sigma} F_1(\sigma). \tag{29}$$

Table 6: Selected sink dimension value thresholds on RefCOCOg.

Model	Selected Threshold σ^*
Qwen3-VL-2B-Instruct	34
Qwen3-VL-4B-Instruct	15

As shown in Figure 9, the optimal threshold balances removing high-sink visual tokens and retaining enough referred-region tokens. We perform this threshold search separately for each Qwen3-VL-Instruct backbone, and the selected thresholds are reported in Table 6.

D Detailed Experimental Results

This section reports detailed results for the ablation studies and scaling analysis in the main paper. We provide per-dataset M-BEIR scores for the key hyperparameter study, masking strategy comparison, visual sink token filtering analysis, and backbone scaling experiment.

Table 7: Detailed per-dataset results for varying the key visual-token retention ratio ρ and reconstruction loss weight λ on M-BEIR. The baseline uses only the contrastive loss $\mathcal{L}_{\text{cl}}$, while all KePIR variants add the auxiliary reconstruction objective with different hyperparameter settings.

Method	ρ	λ	$q^T \to c^I$			$q^T \to c^T$	$q^T \to (c^I, c^T)$		$q^I \to c^T$			$q^I \to c^I$	$(q^I, q^T) \to c^T$		$(q^I, q^T) \to c^I$		$(q^I, q^T) \to (c^I, c^T)$		Avg.
			VN R@5	COCO R@5	F200K R@10	WebQA R@5	EDIS R@5	WebQA R@5	VN R@5	COCO R@5	F200K R@10	NIGHTS R@5	OVEN R@5	InfoS R@5	FIQ R@10	CIRR R@5	OVEN R@5	InfoS R@5	
w/ $\mathcal{L}_{\text{cl}}$ Only	–	–	26.50	78.24	17.56	86.72	54.14	78.29	27.56	91.46	18.86	30.71	49.50	49.88	30.63	49.40	68.63	56.60	50.92
KePIR	0.5	0.05	26.91	78.25	19.20	87.54	52.58	79.37	27.83	91.14	18.53	31.04	48.30	49.31	30.02	51.58	67.57	58.36	51.10
	0.5	0.1	28.09	79.39	19.43	88.51	53.86	79.81	27.96	91.52	19.73	31.51	51.36	50.20	30.40	52.49	69.83	58.04	52.01
	0.5	0.2	27.77	79.79	19.43	88.84	52.05	79.09	27.68	91.44	18.98	31.18	50.11	50.43	30.23	52.45	68.92	57.47	51.62
	0.1	0.1	28.09	79.31	19.60	88.35	52.45	78.85	27.49	91.56	18.67	30.61	48.38	45.05	29.67	50.86	67.74	57.84	50.91
	0.3	0.1	27.87	79.87	19.14	87.94	51.77	80.01	27.60	91.52	18.76	30.61	49.08	49.65	30.68	52.25	67.76	57.54	51.38
	0.7	0.1	28.42	79.96	19.66	89.21	52.30	79.05	27.69	91.44	18.67	31.27	49.04	45.70	29.97	53.12	67.82	58.31	51.35

Table 8: Detailed backbone scaling results on OVEN and InfoSeek for the $IT \to IT$ retrieval setting. Each metric reports the contrastive-only baseline and KePIR.

Backbone	OVEN						InfoSeek						Avg. R@5	
	R@1		R@5		R@10		R@1		R@5		R@10			
	$\mathcal{L}_{\text{cl}}$ Only	KePIR	$\mathcal{L}_{\text{cl}}$ Only	KePIR	$\mathcal{L}_{\text{cl}}$ Only	KePIR	$\mathcal{L}_{\text{cl}}$ Only	KePIR	$\mathcal{L}_{\text{cl}}$ Only	KePIR	$\mathcal{L}_{\text{cl}}$ Only	KePIR	$\mathcal{L}_{\text{cl}}$ Only	KePIR
Qwen3-VL-2B-Instruct	41.48	**42.34**	62.41	**62.99**	69.79	**70.51**	24.23	**24.38**	46.89	**46.95**	56.59	**56.88**	54.65	**54.97**
Qwen3-VL-4B-Instruct	45.68	**46.92**	66.84	**68.52**	73.74	**75.23**	35.86	**36.61**	58.33	**59.98**	66.53	**68.16**	62.59	**64.25**

Table 9: Modality-category grouping used in Figure 3.

Class	Setting	Datasets
Single-Modal	$T \to T$	WebQA
	$I \to I$	NIGHTS
Cross-Modal	$T \to I$	VN, COCO, F200K
	$I \to T$	VN, COCO, F200K
Fused-Modal	$T \to IT$	EDIS, WebQA
	$IT \to T$	OVEN, InfoSeek
	$IT \to I$	FIQ, CIRR
	$IT \to IT$	OVEN, InfoSeek

D.1 Detailed Results for Key Visual-Token Retention Ratio and Reconstruction Weight

Table 7 provides the full per-dataset results corresponding to the averaged results in Table 2. The best overall performance is achieved when the key visual-token retention ratio is $\rho = 0.5$ and the reconstruction loss weight is $\lambda = 0.1$, suggesting that moderate visual token retention and auxiliary reconstruction supervision provide the best balance for retrieval representation learning.

D.2 Detailed Results for Masking Strategies

Table 10 reports the per-dataset results for different masking strategies. Compared with random masking and no masking, KePIR achieves the best average performance, showing that masking guided by key positional information provides more effective reconstruction supervision. Table 9 lists the modality grouping used for the category-level analysis in Figure 3.

D.3 Detailed Results for Sink Dimension Value Filtering

Table 11 reports the detailed per-dataset results with and without visual sink token filtering. The results show that filtering visual tokens with high sink dimension values improves the overall retrieval performance, indicating that the proposed filtering strategy helps reduce the interference of visually irrelevant high-attention tokens.

D.4 Detailed Results for Backbone Scaling

Table 8 provides the detailed results for the backbone scaling experiment on OVEN and InfoSeek under the $IT \to IT$ retrieval setting. KePIR improves over the contrastive-only baseline for both Qwen3-VL-2B-Instruct and Qwen3-VL-4B-Instruct, and the improvement is larger for the 4B backbone, suggesting that the proposed key positional information guided reconstruction can better leverage stronger backbone models.

E License and Use of Scientific Artifacts

We use public benchmarks, pretrained models, and baseline artifacts only for research purposes and in accordance with their intended use, licenses, or terms of use. We do not redistribute any original datasets, checkpoints, or third-party artifacts.

Table 10: Detailed per-dataset results for different masking strategies on M-BEIR, corresponding to the category-level comparison in Figure 3.

Method	$q^T \to c^I$			$q^T \to c^T$	$q^T \to (c^I, c^T)$		$q^I \to c^T$			$q^I \to c^I$	$(q^I, q^T) \to c^T$		$(q^I, q^T) \to c^I$		$(q^I, q^T) \to (c^I, c^T)$		Avg.
	VN R@5	COCO R@5	F200K R@10	WebQA R@5	EDIS R@5	WebQA R@5	VN R@5	COCO R@5	F200K R@10	NIGHTS R@5	OVEN R@5	InfoS R@5	FIQ R@10	CIRR R@5	OVEN R@5	InfoS R@5	
w/o Mask	27.11	76.50	16.58	89.16	52.58	79.25	27.72	91.58	18.22	30.05	46.64	45.02	29.99	49.26	67.37	57.66	50.29
w/ Random-Mask	27.57	79.43	19.43	89.08	52.67	78.97	27.61	91.20	18.41	31.79	47.60	45.71	29.92	52.45	67.21	57.74	51.05
KePIR	28.09	79.39	19.43	88.51	53.86	79.81	27.96	91.52	19.73	31.51	51.36	50.20	30.40	52.49	69.83	58.04	52.01

Table 11: Detailed per-dataset results with and without visual sink token filtering on M-BEIR, corresponding to the category-level comparison in Table 3.

Method	$q^T \to c^I$			$q^T \to c^T$	$q^T \to (c^I, c^T)$		$q^I \to c^T$			$q^I \to c^I$	$(q^I, q^T) \to c^T$		$(q^I, q^T) \to c^I$		$(q^I, q^T) \to (c^I, c^T)$		Avg.
	VN R@5	COCO R@5	F200K R@10	WebQA R@5	EDIS R@5	WebQA R@5	VN R@5	COCO R@5	F200K R@10	NIGHTS R@5	OVEN R@5	InfoS R@5	FIQ R@10	CIRR R@5	OVEN R@5	InfoS R@5	
w/o Filter	27.67	79.50	19.20	88.88	52.42	79.65	28.08	91.60	18.14	30.94	48.87	46.14	30.43	52.54	68.22	57.31	51.22
KePIR	28.09	79.39	19.43	88.51	53.86	79.81	27.96	91.52	19.73	31.51	51.36	50.20	30.40	52.49	69.83	58.04	52.01