SHARPEN: BRINGING THE REQUESTED CHANGE INTO FOCUS FOR COMPOSED IMAGE RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Composed image retrieval searches for an image that preserves relevant content from a reference image while showing a change specified by a modification text. In the training-free describe-then-retrieve paradigm, a multimodal large language model (MLLM) writes a target description, and Contrastive Language-Image Pretraining (CLIP) ranks database images by similarity to it. The description can favor unchanged context, placing reference-like images that miss the change above the target. A description-based similarity score also cannot compare a candidate directly with the reference to check what it preserves. To bring the requested change into focus, we propose SHARPEN, which adjusts scores across the database and within the selected candidate set. Across the database, the global adjustment strengthens the requested change in the query embedding and suppresses popular images that resemble many composed queries before selecting candidates. Within the candidate set, the local adjustment reuses this MLLM to judge the change without the reference, and the complete request with it. Both the change judgment and the reference-conditioned judgment contribute to the final order. With a 4B open-weight MLLM and no training, SHARPEN achieves state-of-the-art training-free results on three widely used benchmarks.

1 INTRODUCTION

Composed image retrieval (CIR) searches for an image using a reference image and a modification text describing the requested change: given a cat in a suitcase and "has two children instead of a cat," the target should show children in a suitcase (Liu et al., 2021; Wu et al., 2021). This form of search lets users specify what should change without describing every detail of the desired image. A successful result must satisfy both reference preservation and modification compliance: it should retain the relevant context while showing the requested change.

Zero-shot CIR (ZS-CIR) uses no annotated reference–modification–target triplets (Saito et al., 2023; Baldrati et al., 2023). In the training-free describe-then-retrieve paradigm, a multimodal large language model (MLLM) writes a target description, and Contrastive Language-Image Pretraining (CLIP) ranks database images by similarity to that description (Radford et al., 2021; Karthik et al., 2024; Yang et al., 2024b; Tang et al., 2025b). The description makes the desired scene searchable with a standard text-to-image retriever.

A target description combines the requested change with context retained from the reference. Similarity to that description can reward retained context even when an image misses the change: for the cat-in-a-suitcase query introduced above, Figure 1a shows five reference-like cats ranked first and the first target ranked 85th. The target must rise above these reference-like images. Description similarity also cannot compare a database image with the reference to check what it preserves. Children in an unrelated setting would fail to preserve the relevant reference context. We want the ranking to favor images that show the requested change and retain relevant reference content.

To bring the requested change into focus, SHARPEN applies global and local score adjustments. The global adjustment changes scores across the database before selecting the top-ranked images as a candidate set, which has two parts: change amplification gives the modification text a distinct contribution to the query embedding, and popularity suppression lowers the scores of popular images that resemble many other composed queries. The local adjustment uses the MLLM that writes the target description to judge each candidate for the change without the reference and for the complete

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



Figure 1: **Describe-then-retrieve and SHARPEN on CIRCO.** (a) Top five results for one composed query, with the rank of the first target. (b) Fraction of composed queries with a target in the top k.

request with it. These judgments combine with the retrieval scores to produce the final ranking. Figure 1b shows the benefit on CIRCO: SHARPEN raises the fraction of queries with a target ranked first from 18% to 49% and with a target in the top 50 from 84% to 90%.

We evaluate SHARPEN on three widely used CIR benchmarks: CIRCO, CIRR, and FashionIQ. We use a 4B open-weight MLLM to generate target descriptions and assess candidates. Across all three benchmarks, SHARPEN improves on describe-then-retrieve using the same MLLM and CLIP encoder and achieves state-of-the-art training-free results on each benchmark's primary metric.

Our contributions are:

- We formulate a global adjustment of database scores and a local adjustment within the candidate set. A chain-rule decomposition separates change evidence from reference preservation given the change, guiding the two candidate judgments.
- We introduce SHARPEN, a training-free pipeline that strengthens the requested change in retrieval and reuses the MLLM that writes the target description to assess selected candidates without and with the reference.
- Ablations on three benchmarks show that removing any component of SHARPEN lowers the primary metrics.

2 RELATED WORK

Zero-shot composed image retrieval. Much ZS-CIR work focuses on expressing the desired target before retrieval from the image database. Textual-inversion methods train a mapping network on unlabeled images, image–text pairs, or text alone to represent the reference with pseudo-word tokens that can be combined with a modification text: Pic2Word (Saito et al., 2023), SEARLE (Baldrati et al., 2023), Context-I2W (Tang et al., 2024), LinCIR (Gu et al., 2024), and PrediCIR (Tang et al., 2025a) develop this approach without annotated CIR triplets. Our tables list them as training-based ZS-CIR. Training-free methods use pretrained models directly. CLIP (Radford et al., 2021) supplies the contrastively trained image and text encoders used in the describe-then-retrieve paradigm. CIReVL (Karthik et al., 2024) composes target descriptions through language, LDRE (Yang et al., 2024b) ensembles target descriptions, OSrCIR (Tang et al., 2025b) uses reflective reasoning, CoTMR (Sun et al., 2025) adds multi-scale reasoning, and IP-CIR (Li et al., 2025) incorporates an imagined image proxy. SHARPEN starts from describe-then-retrieve but retains the modification text and reference image for assessing retrieved candidates.

Refining description-based retrieval. Because a target must preserve context and satisfy a change, several methods give the modification text a distinct role in retrieval. SEIZE (Yang et al., 2024a) introduces a semantic editing increment, STiTch (Li et al., 2026) uses semantic transitions and transportation, G-Mixer (Lim et al., 2026) combines image and text features through geodesic mixing, and SDR-CIR (Sun et al., 2026) includes similarity to the modification text while adjusting the reference contribution. In SHARPEN, the encoded modification text is added to the description

embedding as a direction in the same space, while contrastive scores have a density-ratio interpretation (van den Oord et al., 2018; Poole et al., 2019). Query-bank normalization discounts images that are similar to many queries (Bogolin et al., 2022). Our popularity suppression adopts the nearest-neighbor normalization rule (Chowdhury et al., 2024) for composed-query retrieval.

MLLM scores for candidate assessment. Separate encoders support efficient retrieval (Reimers & Gurevych, 2019; Karpukhin et al., 2020), while cross- and poly-encoders allow more interaction between a query and candidate during scoring (Nogueira & Cho, 2019; Humeau et al., 2020). Embedding dimension also limits which top-k subsets single-vector retrieval can express (Weller et al., 2026). Language-model ranking has used query likelihood (Sachan et al., 2022; Zhuang et al., 2023), relevance-label probabilities (Zhuang et al., 2024), and generated permutations (Sun et al., 2023). VQAScore (Lin et al., 2024) assesses image–text alignment through the probability of a Yes answer, although model judgments can exhibit systematic biases (Zheng et al., 2023; Wang et al., 2024). Among CIR systems, G-Mixer (Lim et al., 2026) reranks candidates using MLLM-derived semantics, while WISER (Wang et al., 2026) uses a verifier and additional refinement. Our proposed assessment uses binary Yes scores obtained without and with the reference, and their contrast diagnoses what the reference adds. It reuses the MLLM that writes the target description.

3 METHOD

3.1 OVERVIEW

A composed query $x = (I_r, T)$ pairs a reference image I_r with a modification text T describing the requested change, and the image database is $\mathcal{D} = \{I_j\}_{j=1}^{M}$. In the describe-then-retrieve paradigm (Karthik et al., 2024; Tang et al., 2025b), an MLLM Φ writes a target description $d = \Phi(I_r, T)$, and CLIP (Radford et al., 2021) with text and image encoders E_T and E_I embeds the description and each database image. With $\mathrm{N}(u) = u/\|u\|$ and cached image embeddings $g_j = \mathrm{N}(E_I(I_j))$, the initial query embedding and score are

$$q_0 = \mathrm{N}(E_T(d)), \qquad S_0(x, j) = \langle q_0, g_j \rangle, \tag{1}$$

and sorting $S_0(x, j)$ over the database gives the describe-then-retrieve ranking.

For a reference cat in a suitcase and the modification text "has two children instead of a cat," this ranking can place cats in suitcases ahead of the requested children in a suitcase (Figure 2a). The desired ranking instead favors images that show the requested change and keep the relevant reference content. We denote such a desired distribution over database images by $p^\star(j \mid x)$; its values are unknown, but any score $S(x, j)$ induces a distribution with the same ranking

$$p_S(j \mid x) = \frac{\exp S(x, j)}{\sum_{\ell=1}^{M} \exp S(x, \ell)},$$

and our aim is to move p_{S_0} toward $p^\star$.

To achieve that goal and bring the requested change into focus, we adjust both the query used to search the database and the order of the selected candidates. Figure 2 illustrates the idea. In panel (a), the global adjustment moves q_0 toward the requested change and lowers the scores of popular images that resemble many composed queries. Targets that the initial query might miss can then rise into the top-N candidate set $\mathcal{C}$. In panel (b), a target and a reference-like candidate can still start with similar retrieval weights. The local adjustment uses the same MLLM to assess each candidate without and with the reference, then changes their relative weights so the candidate that meets both requirements rises. The resulting retrieval scores s_j and assessment scores v_j combine in the final scores F_j.

3.2 GLOBAL ADJUSTMENT: CHANGE-FOCUSED CANDIDATE RETRIEVAL

Candidate selection depends on scores assigned across the whole database, and the global adjustment changes those scores. The generated target description states the requested change alongside content retained from the reference. Its embedding q_0 can therefore give high scores to reference-like images that miss the change. The modification text identifies what to strengthen in the query embedding; an image-specific suppression then lowers the scores of popular images.

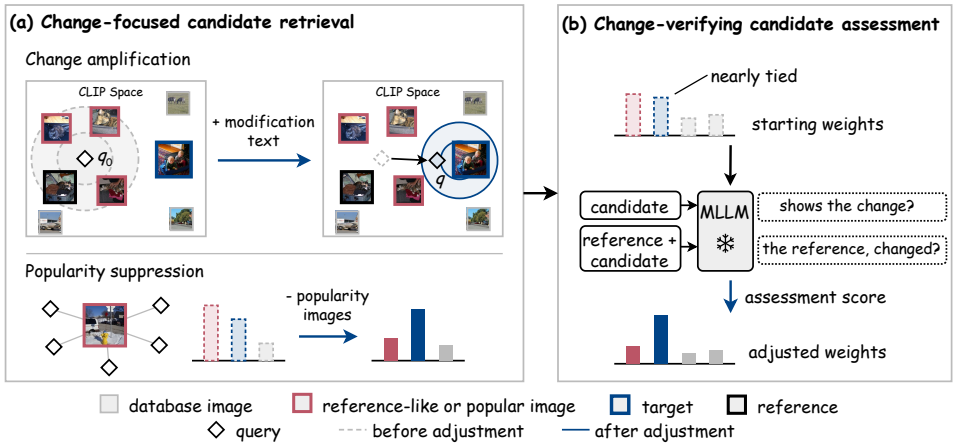

Figure 2: **Method overview.** (a) Global adjustment moves the query toward the requested change and suppresses popular images before candidate selection. (b) Local adjustment uses MLLM judgments without and with the reference to reorder the candidates.

This failure can be understood by separating the initial query point into what it shares with nearby queries and what distinguishes this composed query. Let $\bar{q}$ represent that conceptual local mean:

$$q_0 = \bar{q} + \big(q_0 - \bar{q}\big). \tag{2}$$

For inner-product scores, $p_{S_0}(j \mid x) \propto \exp\langle \bar{q}, g_j \rangle \exp\langle q_0 - \bar{q}, g_j \rangle$. The shared factor can favor images that match common context even when they miss the requested change. The query-specific factor distinguishes this composed query from its neighbors; the modification text identifies the particular change to strengthen within that distinction.

Amplifying the requested change. The modification text states the requested change directly. Change amplification encodes that text separately and adds it to q_0, giving the change a distinct contribution to every database score:

$$f = \mathrm{N}(E_T(T)), \qquad q = \mathrm{N}(q_0 + \gamma f), \tag{3}$$

where f is the change embedding, q is the amplified query embedding, and $\gamma > 0$ is the amplification strength. Up to a normalization shared across database images, $\langle q, g_j \rangle \propto \langle q_0, g_j \rangle + \gamma \langle f, g_j \rangle$. Images aligned with the modification text gain a distinct score contribution, while q_0 carries the reference context from the target description. This movement changes scores across the database and costs one additional CLIP text encoding and a vector addition per composed query.

Suppressing popular images. Moving q_0 alone may still rank an image highly because it resembles the target descriptions of many composed queries; we call such an image popular. To measure its popularity, we compare each database embedding g_j with a query bank $\mathcal{B}$ of description embeddings. Let $\mathrm{NN}_k(g_j; \mathcal{B})$ contain its k nearest query-bank embeddings. Their mean $\bar{b}_j$ and the popularity score h_j are

$$\bar{b}_j = \frac{1}{k} \sum_{b \in \mathrm{NN}_k(g_j; \mathcal{B})} b, \qquad h_j = \langle \bar{b}_j, g_j \rangle. \tag{4}$$

Popularity suppression subtracts this score from the image's cosine score:

$$s_j = \langle q, g_j \rangle - \gamma_h h_j, \tag{5}$$

where $\gamma_h > 0$ is the suppression strength. A popular image loses part of the score it would receive regardless of this request. The mean $\bar{b}_j$ is computed for image j; $\bar{q}$ is a conceptual mean of nearby query embeddings. The image-specific mean defines no single adjusted query embedding for the database; database image embeddings stay fixed. The operation uses similarities between cached database image and query-bank embeddings. We select the top-N database images under s_j as the candidate set $\mathcal{C}$, which the local adjustment assesses next.

3.3 LOCAL ADJUSTMENT: CHANGE-VERIFYING CANDIDATE ASSESSMENT

Retrieval can place candidates that show the requested change close together even when they differ in how well they preserve the reference content, and the local adjustment separates them within the candidate set. Each retrieval score s_j combines similarity to q with a penalty for popularity. Neither component examines the candidate beside the reference, so this difference remains unresolved. We ask the MLLM that wrote the target description to make that comparison, using the top-N candidate set $\mathcal{C}$ to keep the cost manageable.

The retrieval scores give these candidates their starting weights, which the MLLM's judgment then changes. Let $p_G(j \mid x)$ denote the retrieval contribution normalized over $\mathcal{C}$, with its scale set at fusion, and let v_j be the assessment score for candidate I_j. The local adjustment gives each candidate weight

$$p_L(j \mid x) = \frac{p_G(j \mid x) \exp v_j}{\sum_{\ell \in \mathcal{C}} p_G(\ell \mid x) \exp v_\ell}, \qquad j \in \mathcal{C}. \tag{6}$$

With $v_j = 0$ for every candidate, $p_L = p_G$. A candidate with a larger v_j gains relative weight. To decide what v_j should reward, we return to the two requirements of the request.

For one candidate I_j, let Δ mean that it shows the requested change and K that it preserves the relevant reference content. Both must hold for the image to meet the request. Here $p(K, \Delta \mid j, x)$ describes those requirements for one image, while p_G and p_L assign weights across images. The chain rule separates the joint requirement in log space:

$$\log p(K, \Delta \mid j, x) = \underbrace{\log p(\Delta \mid j, x)}_{\text{change evidence}} + \underbrace{\log \frac{p(K, \Delta \mid j, x)}{p(\Delta \mid j, x)}}_{\text{reference preservation}}. \tag{7}$$

This decomposition follows a simple inspection: first check whether the candidate shows the requested change, then whether the changed scene keeps the relevant reference content. The second term is the conditional $\log p(K \mid \Delta, j, x)$. Guided by these two requirements, we ask the MLLM one question without the reference for the change and another with the reference for the complete request. The candidate-only score and the reference-conditioned score provide the contributions to v_j; their contrast shows how the judgment changes when the reference is present.

Assessing evidence of the change. We begin with the change alone. The candidate-only prompt $P_a(I_j, T)$ shows the MLLM image I_j and modification text T, then asks whether the image shows that change. We score this question from the first-token probabilities of Yes and No. For any assessment prompt P, we normalize over these two responses, so that $\widehat{p}_\Phi(\text{Yes} \mid P) + \widehat{p}_\Phi(\text{No} \mid P) = 1$, and take the log-odds of Yes:

$$\begin{aligned} L(P) &= \log \frac{\widehat{p}_\Phi(\text{Yes} \mid P)}{\widehat{p}_\Phi(\text{No} \mid P)} = \log p_\Phi(\text{Yes} \mid P) - \log p_\Phi(\text{No} \mid P), \\ a_j &= L(P_a) = \log \frac{\widehat{p}_\Phi(\Delta \mid j, x)}{1 - \widehat{p}_\Phi(\Delta \mid j, x)}. \end{aligned} \tag{8}$$

Here $\widehat{p}_\Phi(\Delta \mid j, x) = \widehat{p}_\Phi(\text{Yes} \mid P_a)$ is the binary-normalized Yes probability for the change, and a_j is its log-odds. The normalization leaves the ratio unchanged, so $L(P)$ equals the difference of the two raw log-probabilities. A larger a_j indicates stronger model support for the change.

Assessing the complete request. We then give the MLLM the reference image, the candidate, and the modification text, in that order. The reference-conditioned prompt $P_r(I_r, I_j, T)$ asks whether the candidate shows the reference after the requested change. We denote the binary-normalized Yes probability for this prompt by $\widehat{p}_\Phi(K, \Delta \mid j, x)$; it records the MLLM's judgment of the complete request. Its log-odds is the reference-conditioned score:

$$r_j = L(P_r) = \log \frac{\widehat{p}_\Phi(K, \Delta \mid j, x)}{1 - \widehat{p}_\Phi(K, \Delta \mid j, x)}. \tag{9}$$

Their log-odds difference is

$$i_j = r_j - a_j. \tag{10}$$

This contrast records whether seeing the reference strengthens or weakens the model's support for a candidate. Since $r_j = a_j + i_j$, the measured pair (a_j, r_j) carries the same information as (a_j, i_j), and the fusion below weights the two measured scores. Each candidate requires two scoring forward passes, for $2N$ total per composed query.

3.4 Combining the Two Adjustments

The final candidate order combines the retrieval score s_j from the global adjustment with change evidence a_j and the reference-conditioned score r_j from the local adjustment. These scores have different numerical scales, so each is standardized over $\mathcal{C}$ before weighting.

For a score sequence $t \in \{s, a, r\}$, let $\bar{t}$ and σ_t denote its mean and standard deviation within the candidate set. Define $\hat{z}(t_j) = (t_j - \bar{t})/\sigma_t$, taking $\hat{z}(t_j) = 0$ when $\sigma_t = 0$. The assessment score v_j and final score F_j are

$$v_j = w_a\, \hat{z}(a_j) + w_r\, \hat{z}(r_j)\,, \qquad F_j = w_s\, \hat{z}(s_j) + v_j, \qquad w = (w_s, w_a, w_r) \geq 0. \tag{11}$$

The weights control how strongly retrieval, change evidence, and the complete request contribute to the final ranking. With $p_G(j \mid x) \propto \exp(w_s \hat{z}(s_j))$ normalized over $\mathcal{C}$, substituting into Equation 6 gives $p_L(j \mid x) \propto \exp(F_j)$. Ranking by F_j thus implements the local adjustment. For the full-database ranking, we place candidates in F_j order, followed by the remaining images in s_j order.

4 Experiments

We evaluate how well SHARPEN performs against prior methods on three benchmarks, whether the global and local adjustments each contribute to its gains, how sensitive it is to its hyperparameters, and its inference cost.

4.1 Experimental setup

Benchmarks and metrics. We evaluate on CIRCO (Baldrati et al., 2023), CIRR (Liu et al., 2021), and FashionIQ (Wu et al., 2021), described in Appendix C.1. Their primary metrics are CIRCO mAP@5, CIRR full-database Recall@1, and FashionIQ averaged Recall@10.

Implementation details. Qwen3-VL-4B-Instruct (Qwen Team, 2025b) generates target descriptions and assesses candidates; Appendix D.3 reports results with four other MLLMs. OpenCLIP ViT-g/14 (Cherti et al., 2023) supplies image and text embeddings. The transductive query bank uses the description embeddings of the evaluated composed queries, with each composed query's own description excluded from the bank when that query is scored; it uses no target labels. Candidate assessment uses the two prompts of Section 3.3 with the log-odds scores of Equation 8, combined as in Equation 11; Appendix D.1 reports two probability-based alternatives. All three benchmarks share one setting: $\gamma = 0.5$, $\gamma_h = 0.75$, $k = 16$, $N = 50$, and $w = (w_s, w_a, w_r) = (0.33, 0.60, 0.42)$. The parameters were tuned on a small set of validation splits. All experiments ran on one NVIDIA RTX 5080 GPU. Appendices A, B, and D.4 give the prompts, the inference algorithm, and a case study.

Baselines. The controlled describe-then-retrieve baseline, denoted BASELINE, uses the same MLLM, description prompt, and CLIP encoder as SHARPEN without the global and local adjustments. Results for other methods are those reported by their authors; Appendix C.2 describes these methods. In all tables, bold marks the best result; Tables 1 and 2 also underline the second best, rank training-free methods only, and mark ties alike.

4.2 Main results

Tables 1 and 2 show that, with a 4B open-weight MLLM, SHARPEN achieves the best primary metric among the compared training-free methods on all three benchmarks, ahead of systems built on GPT-4o and a 72B MLLM. On CIRCO and CIRR, its primary metrics also exceed those of both training-based methods by at least 14.71 and 16.96 points, respectively. Compared with BASELINE, which uses the same MLLM and encoder, SHARPEN raises the three primary metrics by 22.01, 25.47, and 14.51 points. With the models held fixed, these gains come from how the scores are

Table 1: CIRCO and CIRR results (%). R@k denotes Recall@k. †WISER also uses a 7B candidate-assessment model and an image editor.

		CIRCO (mAP@k)				CIRR (R@k)			CIRR (R_{subset}@k)		
Method	Model	5	10	25	50	1	5	10	1	2	3
Training-based ZS-CIR											
LinCIR (Gu et al., 2024)	–	19.71	21.01	23.13	24.18	35.25	64.72	76.05	63.35	82.22	91.98
PrediCIR (Tang et al., 2025a)	–	23.70	24.60	25.40	26.00	37.00	66.10	77.90	–	–	–
Training-free ZS-CIR, proprietary model											
CIReVL (Karthik et al., 2024)	GPT-3.5	26.77	27.59	29.96	31.03	34.65	64.29	75.06	67.95	84.87	93.21
SEIZE (Yang et al., 2024a)	GPT-3.5	32.46	33.77	36.46	37.55	38.87	69.42	79.42	74.15	89.23	95.71
LDRE (Yang et al., 2024b)	GPT-3.5	31.12	32.24	34.95	36.03	36.15	66.39	77.25	68.82	85.66	93.76
OSrCIR (Tang et al., 2025b)	GPT-4o	30.47	31.14	35.03	36.59	37.26	67.25	77.33	69.22	85.28	93.55
G-Mixer (Lim et al., 2026)	GPT-4o	31.79	32.54	35.49	36.87	39.18	69.83	79.35	72.36	87.25	93.85
WISER (Wang et al., 2026)	GPT-4o†	36.53	38.14	40.46	41.26	49.54	77.40	85.76	78.10	89.06	92.68
Training-free ZS-CIR, open-weight MLLM											
CoTMR (Sun et al., 2025)	Qwen2-VL-72B	32.23	32.72	35.60	36.83	36.36	67.52	77.82	71.19	86.34	93.87
STiTch (Li et al., 2026)	Qwen2-VL-7B	34.40	35.56	38.07	40.02	39.23	69.95	79.56	73.56	**89.50**	**95.86**
BASELINE	Qwen3-VL-4B	16.40	17.79	19.71	20.65	28.49	56.97	68.33	56.11	75.10	85.82
SHARPEN	Qwen3-VL-4B	**38.41**	**39.25**	**41.09**	**41.38**	**53.96**	**79.91**	**86.82**	**79.17**	88.45	91.56

Table 2: FashionIQ results (%) per category. †As in Table 1.

		Dress		Shirt		Top&Tee		Average	
Method	Model	R@10	R@50	R@10	R@50	R@10	R@50	R@10	R@50
Training-based ZS-CIR									
LinCIR (Gu et al., 2024)	–	38.08	60.88	46.76	65.11	50.48	71.09	45.11	65.69
PrediCIR (Tang et al., 2025a)	–	39.70	62.40	48.20	67.40	53.70	73.60	47.20	67.80
Training-free ZS-CIR, proprietary model									
CIReVL (Karthik et al., 2024)	GPT-3.5	27.07	49.53	33.71	51.42	35.80	56.14	32.19	52.36
SEIZE (Yang et al., 2024a)	GPT-3.5	39.61	**61.02**	43.60	**65.42**	45.94	**71.12**	43.05	**65.85**
LDRE (Yang et al., 2024b)	GPT-3.5	26.11	51.12	35.94	58.58	35.42	56.67	32.49	55.46
OSrCIR (Tang et al., 2025b)	GPT-4o	33.02	54.78	38.65	54.71	41.04	61.83	37.57	57.11
G-Mixer (Lim et al., 2026)	GPT-4o	34.71	58.85	39.65	59.61	44.77	67.47	39.71	61.98
WISER (Wang et al., 2026)	GPT-4o†	38.57	58.35	**45.98**	61.58	49.21	66.96	44.59	62.30
Training-free ZS-CIR, open-weight MLLM									
CoTMR (Sun et al., 2025)	Qwen2-VL-72B	34.51	57.36	38.32	62.24	41.90	64.30	38.25	61.32
STiTch (Li et al., 2026)	Qwen2-VL-7B	35.04	56.74	39.48	56.59	42.86	64.95	39.12	59.43
BASELINE	Qwen3-VL-4B	25.78	47.25	32.24	49.71	34.47	55.94	30.83	50.96
SHARPEN	Qwen3-VL-4B	**40.01**	55.73	**45.98**	58.10	**50.03**	64.30	**45.34**	59.38

adjusted. We further evaluate SHARPEN with four additional MLLMs in Appendix D.3. Across all five MLLMs, its gains over BASELINE persist without retuning and range from 12.49 to 26.07 points. With Qwen3-VL-8B, SHARPEN exceeds every compared training-free method on each benchmark.

On CIRCO, SHARPEN has the highest mAP at every cutoff and exceeds WISER by 1.88 points at mAP@5. On CIRR, it leads all three full-database recall metrics. Its Recall@1 exceeds WISER by 4.42 points and every other compared method by at least 14.73 points. It also has the best R_{subset}@1, which ranks the target among visually similar images.

Among training-free methods on FashionIQ, SHARPEN has the best average Recall@10 and leads or ties Recall@10 in every category.

4.3 GLOBAL AND LOCAL ADJUSTMENTS

We first test whether the global adjustment improves candidate retrieval by adding change amplification and popularity suppression to BASELINE, separately and together. Table 3 shows that each retrieval step alone improves all three primary metrics. Together, they add 6.49, 7.39, and 8.13

Table 3: Component ablation on primary metrics (%), with hyperparameters and fusion weights fixed across rows. The row with no checkmarks is BASELINE; the row with all four is SHARPEN.

Global adjustment		Local adjustment		CIRCO	CIRR	FashionIQ
Change amplification	Popularity suppression	Candidate-only	Reference-conditioned	mAP@5	R@1	R@10
				16.40	28.49	30.83
✓				19.43	33.34	34.41
	✓			18.76	30.33	34.60
✓	✓			22.89	35.88	38.96
		✓		33.70	48.67	37.77
			✓	26.95	42.67	38.76
		✓	✓	36.91	51.47	40.68
	✓	✓	✓	38.04	51.73	42.69
✓		✓	✓	35.88	53.91	43.53
✓	✓		✓	30.74	46.69	44.68
✓	✓	✓		35.94	51.11	42.67
✓	✓	✓	✓	**38.41**	**53.96**	**45.34**

Table 4: Fraction of composed queries with a target in the top 50 (%).

Configuration	CIRCO	CIRR	FashionIQ
BASELINE	84.09	87.95	50.96
+ change amplification	88.64	**93.26**	55.61
+ popularity suppression	84.09	88.16	54.61
+ both	**90.45**	93.21	**59.38**

points, exceeding the sum of their separate gains by 0.7 to 1.1 points. Both operations therefore improve the retrieval ranking before candidate assessment.

To check whether the retrieval gains bring more targets into the candidate set, we measure target inclusion, the fraction of composed queries with a target in the top 50. Table 4 shows that using both retrieval steps increases target inclusion by 6.36, 5.26, and 8.42 points on CIRCO, CIRR, and FashionIQ, respectively. Candidate assessment therefore receives more sets containing a target.

Adding candidate assessment to the full global adjustment isolates its effect on candidate order. Table 3 shows that this addition raises CIRCO mAP@5, CIRR Recall@1, and FashionIQ Recall@10 by 15.52, 18.08, and 6.38 points, respectively. Applied to the BASELINE candidate set with the same $N = 50$, the resulting ranking trails SHARPEN by 1.50, 2.49, and 4.66 points. The global adjustment therefore also improves the final ranking.

Removing one component at a time from the full system tests each operation's contribution: every removal in Table 3 lowers all three primary metrics. With the global adjustment fixed, removing either local question leaves only the other, showing whether the candidate-only and reference-conditioned questions add different information. Used alone, the candidate-only question is stronger on CIRCO and CIRR, while the reference-conditioned question is stronger on FashionIQ. Combining them exceeds the stronger single question on all three benchmarks, by 2.47 points on CIRCO and 2.85 on CIRR. The global and local adjustments therefore contribute complementary gains, and the local adjustment needs both judgments. Appendix D.2 further shows that the candidate-only question scores higher when it names the modification text than when it names the target description.

4.4 PARAMETER SENSITIVITY

We vary one parameter at a time around the shared setting to assess sensitivity to tuning. Figure 3 shows that CIRCO and FashionIQ vary by less than 1.5 points over $\gamma \in [0.3, 0.8]$, while CIRR improves further at larger γ. Moderate γ_h values work best on all three benchmarks, and increasing k beyond 8 changes the primary metrics only slightly. Setting the retrieval weight w_s to zero lowers

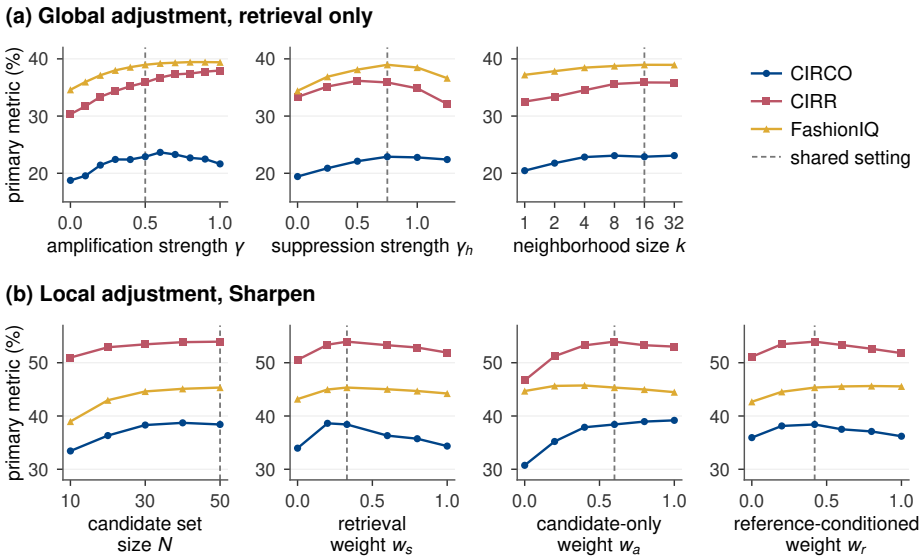

Figure 3: Parameter sensitivity on primary metrics.

Table 5: Inference cost per composed query versus candidate set size N.

	BASELINE	SHARPEN				
		$N = 10$	$N = 20$	$N = 30$	$N = 40$	$N = 50$
MLLM generations	1	1	1	1	1	1
MLLM scoring passes	0	20	40	60	80	100
Seconds per composed query	0.05	0.67	1.30	1.93	2.56	3.19

SHARPEN's primary metrics by 2.17 to 4.45 points, showing that the retrieval score still contributes after candidate assessment. For each weight varied individually, the shared setting's primary metric stays within 0.8 points of the best tested value on every benchmark. Gains from increasing N taper: at $N = 30$, SHARPEN stays within 0.8 points of its $N = 50$ result on every benchmark.

4.5 INFERENCE COST

A larger candidate set requires the MLLM to assess more images, so we measure inference cost as N varies, with one target-description generation per composed query. At $N = 30$, where SHARPEN stays within 0.8 points of its $N = 50$ result (Section 4.4), inference takes 1.93 seconds per composed query on one consumer GPU. Table 5 shows that time grows linearly with the $2N$ MLLM scoring passes, reaching 3.19 seconds at $N = 50$. The global adjustment makes no MLLM call and adds only one CLIP text encoding and vector operations on cached embeddings. Candidate assessment therefore accounts for nearly all of the added cost, and its number of passes does not grow with database size.

5 CONCLUSION

We presented SHARPEN, a training-free CIR method that brings the requested change into focus. Similarity to the target description can favor reference-like images that miss the change. To address this, SHARPEN adjusts scores across the image database and within the candidate set. Change amplification strengthens the requested change, and popularity suppression lowers the scores of popular images before candidate selection. Within the candidate set, the MLLM that writes the target description judges each candidate for the change without the reference and for the complete request with it. With a 4B open-weight MLLM, SHARPEN achieves the best primary metric among the compared training-free methods on CIRCO, CIRR, and FashionIQ.

AI USE STATEMENT

AI assistance was used to polish the language and presentation of the manuscript.

REPRODUCIBILITY STATEMENT

The manuscript provides the scoring equations, prompts, hyperparameters, and model and hardware configurations required to reproduce our experiments. The implementation is available in an anonymous GitHub repository: https://anonymous.4open.science/r/SHARPEN-B240/.

REFERENCES

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

Simion-Vlad Bogolin, Ioana Croitoru, Hailin Jin, Yang Liu, and Samuel Albanie. Cross modal retrieval with querybank normalisation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Neil Chowdhury, Franklin Wang, Sumedh Shenoy, Douwe Kiela, Sarah Schwettmann, and Tristan Thrush. Nearest neighbor normalization improves multimodal retrieval. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024.

Gemma Team. Gemma 4 technical report. *arXiv preprint arXiv:2607.02770*, 2026.

Geonmo Gu, Sanghyuk Chun, Wonjae Kim, Yoohoon Kang, and Sangdoo Yun. Language-only training of zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13225–13234, 2024.

Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*, 2020.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.

Shyamgopal Karthik, Karsten Roth, Massimiliano Mancini, and Zeynep Akata. Vision-by-language for training-free compositional image retrieval. In *International Conference on Learning Representations (ICLR)*, 2024.

Miaoge Li, Dongsheng Wang, Zening Sun, Jinsen Zhang, Wenhan Luo, and Jingcai Guo. STiTch: Semantic transition and transportation in collaboration for training-free zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12374–12384, 2026.

You Li, Fan Ma, and Yi Yang. Imagine and seek: Improving composed image retrieval with an imagined proxy. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3984–3993, 2025.

Jiyoung Lim, Heejae Yang, and Jee-Hyong Lee. G-MIXER: Geodesic mixup-based implicit semantic expansion and explicit semantic re-ranking for zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 33891–33900, 2026.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. In *European Conference on Computer Vision (ECCV)*, 2024.

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

Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with BERT. *arXiv preprint arXiv:1901.04085*, 2019.

Ben Poole, Sherjil Ozair, Aaron van den Oord, Alexander A. Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, 2019.

Qwen Team. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025a.

Qwen Team. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025b.

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. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

Devendra Singh Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. Improving passage retrieval with zero-shot question generation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

Kuniaki Saito, Kihyuk Sohn, Xiang Zhang, Chun-Liang Li, Chen-Yu Lee, Kate Saenko, and Tomas Pfister. Pic2word: Mapping pictures to words for zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

Yi Sun, Jinyu Xu, Qing Xie, Jiachen Li, Yanchun Ma, and Yongjian Liu. SDR-CIR: Semantic debias retrieval framework for training-free zero-shot composed image retrieval. In *Proceedings of the ACM Web Conference (WWW)*, pp. 2149–2159, 2026. doi: 10.1145/3774904.3792339.

Zelong Sun, Dong Jing, and Zhiwu Lu. CoTMR: Chain-of-thought multi-scale reasoning for training-free zero-shot composed image retrieval. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22675–22684, 2025.

Yuanmin Tang, Jing Yu, Keke Gai, Jiamin Zhuang, Gang Xiong, Yue Hu, and Qi Wu. Context-i2w: Mapping images to context-dependent words for accurate zero-shot composed image retrieval. In *AAAI Conference on Artificial Intelligence (AAAI)*, pp. 5180–5188, 2024.

Yuanmin Tang, Jing Yu, Keke Gai, Jiamin Zhuang, Gang Xiong, Gaopeng Gou, and Qi Wu. Missing target-relevant information prediction with world model for accurate zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24785–24795, 2025a.

Yuanmin Tang, Jue Zhang, Xiaoting Qin, Jing Yu, Gaopeng Gou, Gang Xiong, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Qi Wu. Reason-before-retrieve: One-stage reflective chain-of-thoughts for training-free zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14400–14410, 2025b.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

Tianyue Wang, Leigang Qu, Tianyu Yang, Xiangzhao Hao, Yifan Xu, Haiyun Guo, and Jinqiao Wang. WISER: Wider search, deeper thinking, and adaptive fusion for training-free zero-shot composed image retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16865–16875, 2026.

Orion Weller, Michael Boratko, Iftekhar Naim, and Jinhyuk Lee. On the theoretical limitations of embedding-based retrieval. In *International Conference on Learning Representations (ICLR)*, 2026.

Hui Wu, Yupeng Gao, Xiaoxiao Guo, Ziad Al-Halah, Steven Rennie, Kristen Grauman, and Rogerio Feris. Fashion iq: A new dataset towards retrieving images by natural language feedback. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Zhenyu Yang, Shengsheng Qian, Dizhan Xue, Jiahong Wu, Fan Yang, Weiming Dong, and Changsheng Xu. Semantic editing increment benefits zero-shot composed image retrieval. In *ACM International Conference on Multimedia (ACM MM)*, pp. 1245–1254, 2024a.

Zhenyu Yang, Dizhan Xue, Shengsheng Qian, Weiming Dong, and Changsheng Xu. Ldre: Llm-based divergent reasoning and ensemble for zero-shot composed image retrieval. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 80–90, 2024b.

Tianyu Yu, Zefan Wang, Chongyi Wang, et al. MiniCPM-V 4.5: Cooking efficient MLLMs via architecture, data, and training recipe. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11704–11715, 2026.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan, Xuanhui Wang, and Michael Bendersky. Beyond yes and no: Improving zero-shot LLM rankers via scoring fine-grained relevance labels. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pp. 358–370, 2024.

Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. Open-source large language models are strong zero-shot query likelihood models for document ranking. In *Findings of the Association for Computational Linguistics: EMNLP*, 2023.

A PROMPTS

The MLLM receives the reference image and modification text with the following instruction:

Target-description prompt

You are given a reference image and a manipulation text. Write one sentence describing the target image: the reference image after the manipulation has been applied. Output only the sentence.

The target-description prompt calls the modification text T "manipulation text." We preserve this wording to document the prompt associated with the reported results. Description generation uses greedy decoding without in-context examples or dataset-specific instructions. The same generated description is used by BASELINE and SHARPEN.

The prompt $P_a(I_j, T)$ receives candidate image I_j and modification text T:

Candidate-only prompt $P_a(I_j, T)$

Here is a requested change: "T"
Does the image shown satisfy this change? Answer with only Yes or No.

The prompt $P_r(I_r, I_j, T)$ receives the reference image first, the candidate image second, and modification text T:

Reference-conditioned prompt $P_r(I_r, I_j, T)$

Here is a requested change to the first image: "T"
Does the second image show the first image after this change? Answer with only Yes or No.

The candidate-only prompt checks the requested change, while the reference-conditioned prompt checks the complete request. Both are scored using first-token Yes/No log-odds.

B INFERENCE ALGORITHM

Algorithm 1 collects the retrieval, assessment, and fusion steps for one composed query. Database image embeddings are cached, and both assessment questions use the MLLM that writes the target description.

Algorithm 1 SHARPEN inference for a composed query

Require: Reference I_r, modification text T, database image embeddings $\{g_j\}$, query bank $\mathcal{B}$, MLLM Φ, CLIP text encoder E_T

$d \leftarrow \Phi(I_r, T)$ ▷ Generate the target description
$q_0 \leftarrow \mathrm{N}(E_T(d))$; $f \leftarrow \mathrm{N}(E_T(T))$
$q \leftarrow \mathrm{N}(q_0 + \gamma f)$ ▷ Change amplification
Obtain popularity scores h_j from $\mathcal{B}$ for every database image
$s_j \leftarrow \langle q, g_j \rangle - \gamma_h h_j$ ▷ Popularity suppression
$\mathcal{C} \leftarrow$ top-N database images by s_j
for $j \in \mathcal{C}$ **do**
    $a_j \leftarrow L(P_a(I_j, T))$ ▷ Candidate-only score
    $r_j \leftarrow L(P_r(I_r, I_j, T))$ ▷ Reference-conditioned score
end for
Standardize s_j, a_j, r_j within $\mathcal{C}$
$v_j \leftarrow w_a \hat{z}(a_j) + w_r \hat{z}(r_j)$; $F_j \leftarrow w_s \hat{z}(s_j) + v_j$
return $\mathcal{C}$ sorted by F_j, followed by remaining images sorted by s_j

C DATASETS AND COMPARED METHODS

C.1 DATASETS

We evaluate on two open-domain benchmarks, CIRCO and CIRR, and one domain-specific benchmark, FashionIQ. SHARPEN uses no annotated triplets from these benchmarks.

CIRCO. CIRCO (Baldrati et al., 2023) is built from the unlabeled set of COCO 2017 (Lin et al., 2014) and is the first CIR dataset to annotate multiple ground truths per composed query. Its 1,020 composed queries have 4.53 ground truths on average, and its image database contains 123,403 images. Because a composed query can have several targets, CIRCO uses mAP@k for $k \in \{5, 10, 25, 50\}$, which accounts for the ranks of all ground truths; mAP@5 is the primary metric.

CIRR. CIRR (Liu et al., 2021) is the first CIR dataset of natural, open-domain images, drawn from NLVR2 (Suhr et al., 2019). It contains 36,554 triplets over 21,552 images. The images are grouped into subsets of six semantically and visually similar images, and each triplet is drawn from one subset. Recall@k is the fraction of queries with the target among the top k images in the full database; $R_{\text{subset}}@k$ applies the same measure within each six-image subset. We report Recall@$\{1, 5, 10\}$ and $R_{\text{subset}}@\{1, 2, 3\}$, with Recall@1 as the primary metric.

FashionIQ. FashionIQ (Wu et al., 2021) contains 77,684 product images in three categories: Dress, Shirt, and Top&Tee. Each target image is paired with a reference image from the same category, and annotators wrote two modification texts describing how the target differs from its reference. FashionIQ uses Recall@10 and Recall@50 for each category and their averages over the three categories; the average Recall@10 is the primary metric.

C.2 COMPARED METHODS

Tables 1 and 2 compare SHARPEN with ten ZS-CIR methods in three groups. We use the results reported by their authors; the Model column lists the LLM or MLLM where applicable. Section 4.1 describes the controlled BASELINE.

Training-based ZS-CIR. These textual-inversion methods train a mapping network without annotated CIR triplets. The network maps the reference image to a pseudo-word token, which is combined with the modification text for text-to-image retrieval. LinCIR (Gu et al., 2024) trains the mapping network using text alone through self-masking projection. PrediCIR (Tang et al., 2025a) trains a world model on image–caption pairs to predict the target visual content missing from the reference image before mapping that content to a pseudo-word token.

Training-free ZS-CIR with a proprietary model. These methods use pretrained models directly. CIReVL (Karthik et al., 2024) is a two-stage method: a pretrained captioner describes the reference image, an LLM recomposes the caption according to the modification text, and CLIP ranks database images by similarity to the resulting target description. LDRE (Yang et al., 2024b) generates multiple target descriptions through divergent LLM reasoning over captions of the reference image and aggregates them with an ensemble. SEIZE (Yang et al., 2024a) has an LLM compose diverse captions of the reference image with the modification text and adds a semantic editing increment contributed by the modification text to the absolute similarity in its retrieval score. OSrCIR (Tang et al., 2025b) is a one-stage method: an MLLM processes the reference image and modification text together and generates the target description with a reflective chain-of-thought prompt. G-Mixer (Lim et al., 2026) builds composed query features by geodesic mixup of image and text features over a range of mixing ratios, collects a diverse candidate set, and re-ranks it with explicit semantics derived from an MLLM. WISER (Wang et al., 2026) follows a retrieve–verify–refine procedure: it retrieves with both edited captions and edited images, uses a verifier to assess retrieval confidence, fuses the two retrieval paths for confident results, and refines uncertain ones.

Training-free ZS-CIR with an open-weight MLLM. CoTMR (Sun et al., 2025) uses an MLLM to reason step by step over the composed query with a chain-of-thought prompt. It predicts at the

object scale which elements should be present in or absent from the target and combines the CLIP similarities of these outputs through multi-grained scoring. STiTch (Li et al., 2026) refines MLLM-generated target descriptions with a transition vector in the embedding space, models descriptions and images as discrete distributions, and scores them with a bidirectional transport distance.

D ADDITIONAL EXPERIMENTS

D.1 LOG-ODDS VERSUS PROBABILITY SCORES

Table 6: Primary metrics (%) with log-odds and probability scores for the two judgments.

	Shared weights			Weights chosen per form		
Score form	CIRCO mAP@5	CIRR R@1	FashionIQ R@10	CIRCO mAP@5	CIRR R@1	FashionIQ R@10
Probability $\widehat{p}_\Phi(\text{Yes} \mid P)$	33.05	49.29	44.66	35.88	52.14	45.08
Log-probability $\log \widehat{p}_\Phi(\text{Yes} \mid P)$	<u>36.10</u>	<u>52.28</u>	<u>45.12</u>	<u>37.54</u>	<u>52.57</u>	<u>45.58</u>
Log-odds $L(P)$ (SHARPEN)	**38.41**	**53.96**	**45.34**	**39.41**	**54.08**	**45.91**

We compare the log-odds score in Equation 8 with two probability-based alternatives. For both a_j and r_j, we use the binary-normalized Yes probability or its logarithm, computed from the same model outputs, and fuse the resulting scores as in Equation 11. Table 6 shows that, with the shared weights of Section 4.1, log-odds leads on all three benchmarks. It exceeds probability by 5.36 points in CIRCO mAP@5 and 4.67 points in CIRR R@1. Log-odds also leads when weights are chosen separately for each form and benchmark to maximize the primary metric, an optimistic setting for every form. Probability flattens confident rejections and acceptances, its logarithm flattens acceptances, and log-odds retains both; their results follow this order on every benchmark.

D.2 WORDING OF THE CANDIDATE-ONLY QUESTION

The introduction and Section 3.2 note that the target description combines the requested change with retained reference content, so similarity to it can favor reference-like images. The same concern applies to the local adjustment's candidate-only question. The candidate-only score a_j plays the change-evidence role in Equation 7, so its question names the modification text T. A question about the target description d would judge the change together with retained content. We test this choice by replacing T with d in the candidate-only question. Table 7 reports the results using the following prompt with candidate image I_j and the same first-token log-odds scoring:

Description-question prompt

Here is a description of a target image:
"d"
Does the image shown match this target description? Answer with only Yes or No.

Both wordings use the full global adjustment and no reference-conditioned question, so the retrieval-only and change-question rows match the corresponding rows of Table 3.

The change question exceeds the description question on all three primary metrics, by 8.87 points on CIRCO, 17.43 on CIRR, and 3.76 on FashionIQ. Relative to retrieval alone, the description question raises CIRCO mAP@5 by 4.18 points but lowers CIRR Recall@1 and FashionIQ Recall@10 by 2.20 and 0.05 points. Both questions have similar correlations with the retrieval score s_j. Computed within each top-50 candidate set and averaged over composed queries, the Spearman correlations on CIRCO, CIRR, and FashionIQ are 0.19, 0.24, and 0.22 for the description question and 0.11, 0.25, and 0.18 for the change question. The two questions correlate with each other at 0.35, 0.38, and 0.52, so they order candidates differently. Naming the change gives the candidate-only question its change-evidence role and yields the stronger ranking. Together with change amplification, this keeps the requested change explicit in both adjustments: in the query embedding during retrieval and in the MLLM judgment during assessment.

Table 7: Primary metrics (%) for two wordings of the candidate-only question with the global adjustment fixed.

Configuration	CIRCO mAP@5	CIRR R@1	FashionIQ R@10
Retrieval only	22.89	35.88	38.96
+ description question	27.07	33.68	38.91
+ change question	**35.94**	**51.11**	**42.67**

D.3 MODEL SENSITIVITY

Table 8: Primary metrics (%) for BASELINE and SHARPEN with different MLLMs.

		CIRCO mAP@5		CIRR R@1		FashionIQ R@10	
MLLM	Precision	BASELINE	SHARPEN	BASELINE	SHARPEN	BASELINE	SHARPEN
Qwen3-VL-4B (default)	FP8	16.40	38.41	28.49	53.96	30.83	45.34
Qwen3-VL-8B	FP8	18.99	43.71	28.80	54.87	32.16	46.52
Qwen2.5-VL-3B	BF16	9.94	29.34	21.12	46.52	15.34	32.13
MiniCPM-V 4.5	INT4	22.11	37.85	29.20	54.56	29.03	41.52
Gemma 4 E4B	FP8	16.36	30.88	31.00	50.18	28.28	42.79

The MLLM generates both target descriptions and candidate judgments, so its choice could affect the gains. We repeat the comparison with four alternatives to the default Qwen3-VL-4B: Qwen3-VL-8B (Qwen Team, 2025b), Qwen2.5-VL-3B (Qwen Team, 2025a), MiniCPM-V 4.5 (Yu et al., 2026), and Gemma 4 E4B (Gemma Team, 2026). Table 8 reports the results. Each MLLM generates its own target descriptions and assesses its own candidates, while the prompts, CLIP encoder, and all hyperparameters, including fusion weights, stay at the shared setting of Section 4.1. MiniCPM-V 4.5 uses its official INT4 AWQ checkpoint, and Gemma 4 E4B uses a third-party FP8 checkpoint.

With every MLLM, SHARPEN improves over BASELINE on all three benchmarks by 12.49 to 26.07 points without retuning. Choosing fusion weights separately for each MLLM and benchmark changes the primary metric by at most 0.72 points. On CIRCO and CIRR, SHARPEN with any of the five MLLMs, including the older and smaller Qwen2.5-VL-3B, beats BASELINE with any of them. The adjustments therefore contribute gains across model choices. A stronger MLLM still helps: Qwen3-VL-8B improves SHARPEN on all three benchmarks over Qwen3-VL-4B. Its results in Table 8 also exceed every compared training-free method in Tables 1 and 2 on each benchmark. On CIRR, SHARPEN exceeds them with every MLLM except Qwen2.5-VL-3B.

D.4 CASE STUDY

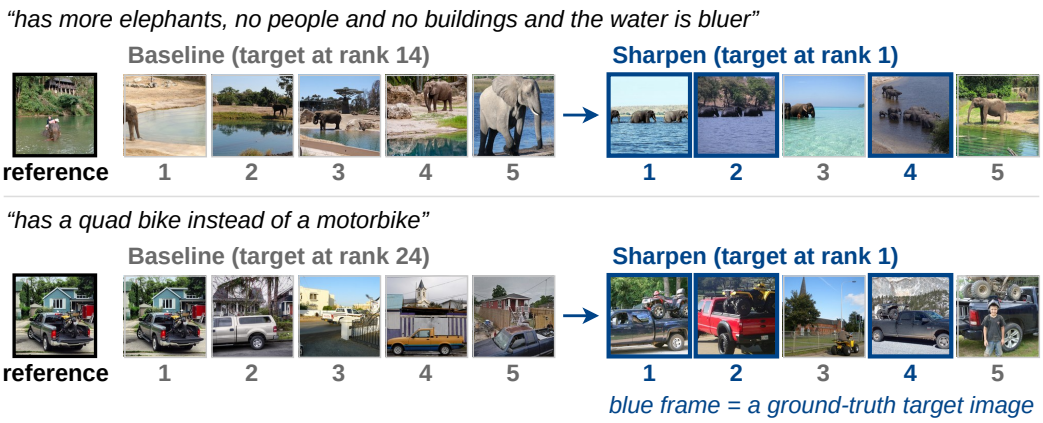

Figure 4: Top five results of BASELINE and SHARPEN on two CIRCO composed queries. Blue frames mark ground-truth targets.

Figure 4 compares the top five results for two CIRCO composed queries. For the query with an elephant in a river, BASELINE returns single elephants and ranks its first target 14th. For the query with a motorbike on a pickup truck, it ranks the reference itself first and its first target 24th. In both queries, SHARPEN ranks targets 1st, 2nd, and 4th, and each of its top three results shows the requested change. These examples illustrate how bringing the change into focus can move suitable targets ahead of reference-like images.