CaptionFuse: Training-Free Composed Product Retrieval Via VLM Intent Captioning and Weighted Embedding Fusion

Zhao Gao[1], Mina Metias[1], Mohamed Noordeen Alaudeen[1] and Bassel Saleh[1]

[1]AWS GenAI Innovation Center

Abstract

Composed Image Retrieval (CIR) retrieves products matching a reference image modified by natural language—a core capability for eCommerce search. Existing training-free methods fuse CLIP embeddings via arithmetic, but we identify a fundamental limitation: user text modifications are *attribute-sparse*—a user who types "make it red with shorter sleeves" specifies *what to change* but not *what to keep* (the floral pattern, the V-neck, the fit-and-flare silhouette). We introduce *language-space fusion*: a Vision-Language Model (VLM) jointly interprets the reference image and modification to generate a *visually grounded* product-title caption—e.g., "Red floral fit-and-flare dress with short sleeves and V-neck"—that fills in the missing attributes from the image. Our method, **CaptionFuse**, requires no task-specific training and is model-agnostic. A controlled experiment confirms that *attribute content*—not linguistic style—drives the gains: imperative and descriptive captions with identical attributes achieve equal R@10, while both outperform raw modifications by +6.5 points. Embedding geometry analysis shows 62% larger discriminative margin, with 68% of remaining failures caused by backbone limitations—not VLM errors (5%). On CIRCO (123K gallery), CaptionFuse achieves 22.4% mAP with CLIP and 31.7% with Titan, outperforming WeiMoCIR (13.9%), LinCIR (13.2%), and Pic2Word (11.6%). On FashionIQ, CaptionFuse matches the best training-free baseline; on CIRR, where modifications are already attribute-rich, VLM captioning provides less benefit. We provide a detailed analysis of when and why the method succeeds or fails (§5).

Keywords

composed image retrieval, eCommerce search, vision-language models, multimodal retrieval, product search

1. Introduction

A customer browsing an eCommerce catalog sees a blue floral dress and types "make it red with shorter sleeves." This is *Composed Image Retrieval* (CIR) [1]: retrieving products that match a reference image modified by natural language. CIR is a high-value capability for product search, yet deploying it at scale is challenging because training-based methods [2, 3] require expensive CIR-specific triplet data that is scarce in most product catalogs.

Training-free approaches [4, 5, 6] avoid this cost by combining pre-trained CLIP [7] embeddings, but they operate in *embedding space* where fine-grained attribute reasoning is lost. The core problem is that user modifications are *attribute-sparse*: "make it red with shorter sleeves" specifies *what to change* but not *what to keep*—the floral pattern, the V-neck, the fit-and-flare silhouette. Embedding arithmetic struggles with these sparse modifications for a geometric reason: CLIP's text encoder was trained on descriptive alt-text, so sparse imperative phrases ("make it red") map to a generic "modification intent" region of embedding space rather than landing near specific product embeddings. We show in §5 that raw modifications achieve only 0.358 mean cosine similarity to their target images, compared to 0.487 for attribute-rich VLM captions—a 36% gap that linear combination cannot bridge when the text vector carries insufficient attribute signal.

We propose **CaptionFuse**, which shifts fusion from embedding space to *language space*. A VLM jointly interprets the reference image and text modification, then generates a concise *product-title* caption that fills in the missing attributes—e.g., "Red floral fit-and-flare dress with short sleeves and V-neck." The caption is encoded and combined with the reference image embedding for nearest-neighbor

ECOM'26: SIGIR Workshop on eCommerce, Jul 24, 2026, Melbourne, Australia

orangez@amazon.com (Z. Gao); mmetias@amazon.ae (M. Metias); nooraldn@amazon.ae (M. N. Alaudeen); basaleh@amazon.ae (B. Saleh)

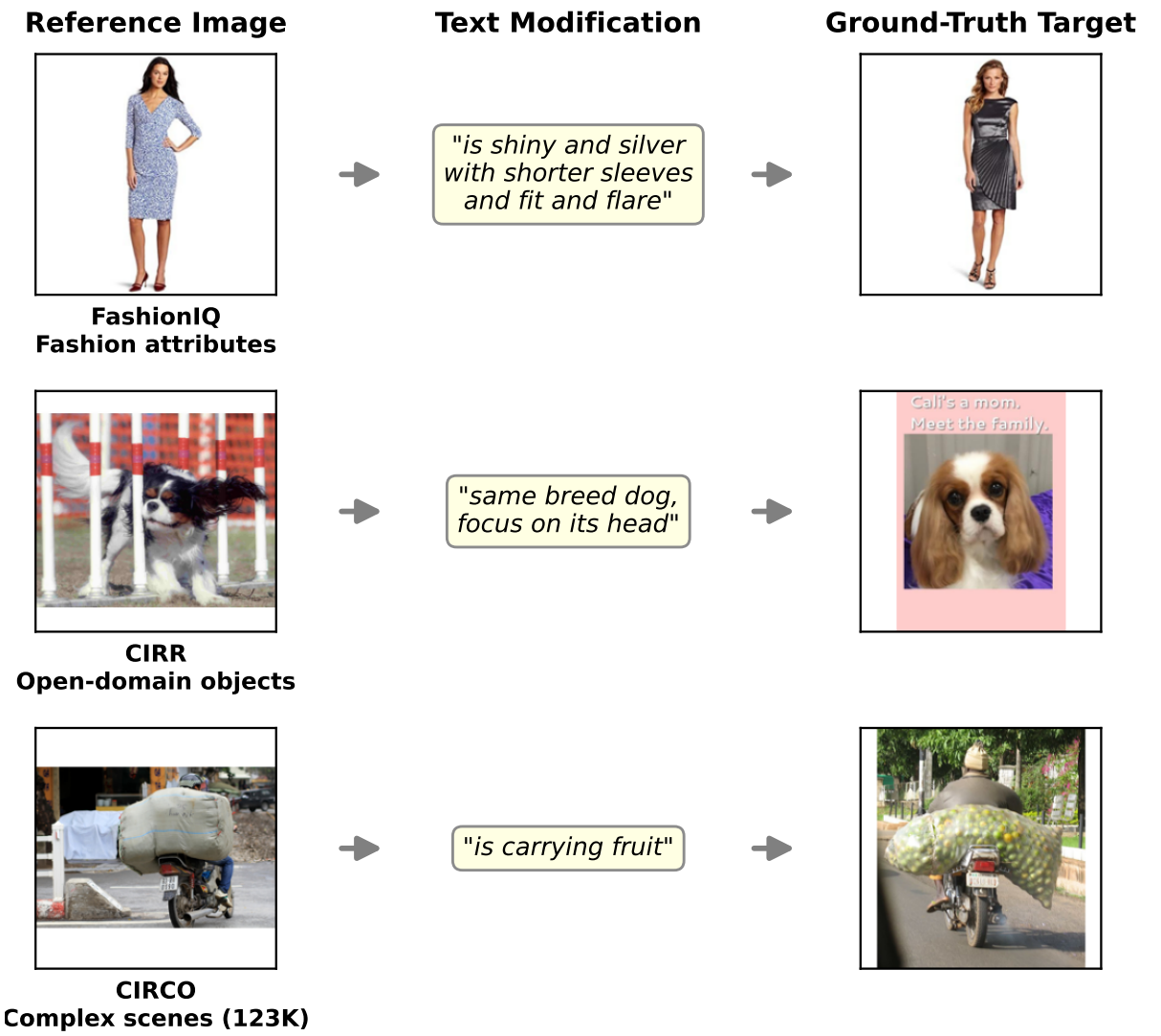

Figure 1: Composed Image Retrieval examples across three benchmarks. Given a reference image and a text modification, the goal is to retrieve the target. FashionIQ tests fashion attribute changes; CIRR tests open-domain object modifications; CIRCO tests complex multi-aspect scene reasoning over a 123K image gallery.

retrieval. The approach is fully training-free and model-agnostic: the VLM, embedding backbone, and vector index are all independently swappable.

Figure 1 illustrates the CIR task across our three evaluation benchmarks, highlighting the diversity of domains and modification complexity.

Figure 2 previews why this matters. For "is white with black polka dots and is not see through," the VLM generates "White, black polka dot, opaque sleeveless tunic"—grounding the abstract "not see through" as the concrete attribute "opaque" and preserving the sleeveless silhouette that the user never mentioned. CaptionFuse retrieves the target at rank 2; WeiMoCIR, which fuses embeddings arithmetically, fails entirely. The user said *what to change*; the VLM filled in *what to keep*.

This paper makes four contributions:

1. **A visual grounding hypothesis** for why language-space fusion outperforms embedding arithmetic: VLM captions enrich queries with concrete visual attributes from the reference image, supported by a controlled experiment showing that attribute content—not linguistic style—drives the gains (§5).
2. **Embedding geometry analysis** quantifying *why* captions produce better queries: 62% larger discriminative margin, with failure modes geometrically identifiable via cosine similarity thresholds

Figure 2: CaptionFuse vs. WeiMoCIR on FashionIQ (dress). Each row: reference image, text modification with VLM caption (blue box), ground-truth target, and top-1 retrieval from each method. Green/orange borders = target in top 5; red = miss. The VLM fills in attributes the user left unsaid (e.g., sleeveless, opaque), enabling correct retrieval.

and dominated by backbone limitations (68%), not VLM errors (5%).

3. **Prompt design as the primary research lever**: product-title format yields +5.2 R@10 over verbose captions—larger than any model or weight change—because concise captions avoid reference leakage and carry higher attribute density.
4. **Practical deployment framework**: model-agnostic design validated across 3 VLMs $\times$ 3 backbones; rule-based hybrid routing saves 26% of VLM calls with $<$1.1 R@10 loss.

2. Related Work

VLMs in eCommerce. Vision-Language Models such as BLIP-2 [8], LLaVA [9], and MiniGPT-4 [10] bridge frozen image encoders with large language models and have transformed eCommerce applications including product attribute extraction, catalog enrichment, visual question answering, and conversational product search [11]. Their ability to reason jointly over images and text makes them natural candidates for composed image retrieval—yet no prior work has used VLMs as an *intent-aware fusion module* that jointly interprets a reference product image and a text modification to produce a single target description.

Training-based CIR. Early approaches to Composed Image Retrieval required task-specific training on (reference image, text modifier, target image) triplets. ARTEMIS [2] learns attention-based composition of image and text features. CoSMo [12] modulates content and style features for text-guided

retrieval. CompoDiff [3] formulates CIR as conditional diffusion in embedding space. While effective, these methods depend on expensive triplet annotations that are scarce in most eCommerce catalogs, limiting their applicability to new product domains.

Training-free CIR. To eliminate the annotation bottleneck, recent work explores zero-shot and training-free approaches that leverage pre-trained CLIP [7] embeddings. Pic2Word [4] maps images to pseudo-word tokens in CLIP's text embedding space. LinCIR [5] learns a linear projection from image embeddings to text-token space, enabling language-only training. SEARLE [13] optimizes pseudo-tokens via textual inversion at test time. WeiMoCIR [6] introduces weighted modular fusion of CLIP image and text embeddings. Context-I2W [14] maps images to context-dependent words. KEDs [15] enhances dual-stream retrieval with external knowledge. SPRC [16] combines sentence-level prompts with region-level captions. CIReVL [17] is the closest to our work: it uses VLMs to generate captions for the reference image, but retrieves via text-only matching—discarding the image embedding entirely. We include a caption-only (α=0) ablation in §4.3 that approximates CIReVL's approach.

MLLM-augmented retrieval. Recent work leverages Multimodal LLMs beyond captioning. WISER [18] uses a retrieve–verify–refine pipeline where an MLLM reranks initial retrieval results. SoFT [19] applies prescriptive/proscriptive reranking via sentence-level prompts. UniCVR [20] unifies composed retrieval with MLLM-based reranking. These post-retrieval verification approaches are *complementary* to CaptionFuse: they could be stacked on top of our initial retrieval to further improve precision. We note that their reported CIRCO and CIRR numbers are higher than ours under matched settings, as reranking provides an additional refinement stage that CaptionFuse does not employ.

The gap CaptionFuse fills. All existing training-free methods operate in *embedding space*: they combine or project pre-computed vectors without reasoning about which visual attributes to preserve versus modify. Meanwhile, VLMs have demonstrated powerful conditional reasoning capabilities but remain unused for CIR fusion. CaptionFuse bridges this gap by shifting fusion from embedding space to *language space*: a VLM jointly interprets the reference image and text modification, producing a target-describing caption that is then encoded and fused with the image embedding. Our product-title prompt design draws on chain-of-thought prompting [21]: by instructing the VLM to be specific about product attributes (color, material, style), we elicit structured outputs that align with how products are indexed in eCommerce catalogs.

3. Method

3.1. Overview

Figure 3 shows the CaptionFuse pipeline. Given a reference product image I_r and text modification t, CaptionFuse (Algorithm 1): (1) classifies query type $\tau \in \{\text{TEXT\_IMAGE}, \text{IMAGE\_ONLY}\}$ via a rule-based classifier; (2) generates a product-title caption c via VLM; (3) encodes I_r and c; (4) fuses with fixed weights (α=0.15).

The query classifier (Step 1) determines whether the text modification carries meaningful semantic content. TEXT_IMAGE queries contain substantive attribute modifications (e.g., "make it red with shorter sleeves") and proceed through VLM captioning. IMAGE_ONLY queries have empty, trivial, or uninformative text (e.g., "find similar") and bypass the VLM entirely, using image-only retrieval (α=1). Classification is rule-based: queries with fewer than 3 content words or matching a stop-phrase list ("similar," "like this," "same") are routed to IMAGE_ONLY. This routing is distinct from the hybrid descriptiveness routing (§3.4), which operates *within* the TEXT_IMAGE branch to decide whether the VLM caption adds value over direct text encoding.

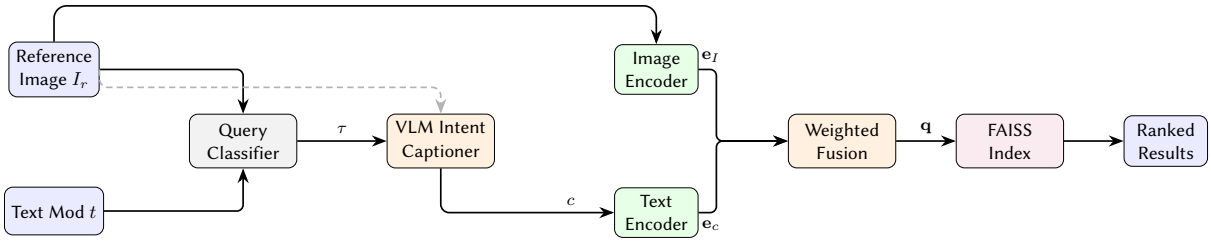

Figure 3: CaptionFuse pipeline. The VLM generates a product-title caption fusing the reference image with the text modification. The caption and image are independently encoded, then combined with weighted fusion for retrieval. The encoder is swappable (CLIP, Nova 2, or Titan).

Algorithm 1: CaptionFuse Inference Pipeline

Input: Reference image I_r, text modification t
Output: Ranked product results

$\tau \leftarrow \textsc{Classify}(t)$;
$c \leftarrow \text{VLM}(I_r, t, \tau)$; `// product-title caption`
$\mathbf{e}_I \leftarrow \textsc{Encode}_{\text{img}}(I_r)$;
$\mathbf{e}_c \leftarrow \textsc{Encode}_{\text{txt}}(c)$;
$\mathbf{q} \leftarrow \textsc{Normalize}(\alpha \cdot \mathbf{e}_I + (1-\alpha) \cdot \mathbf{e}_c)$;
return $\text{FAISS\_}\textsc{Search}(\mathbf{q}, k)$;

3.2. VLM Intent Captioning

The VLM's role is to fill in what the user left unsaid. Given the reference image and text modification, it must decide which visual attributes to preserve (e.g., silhouette), which to modify (e.g., color), and which to infer (e.g., material implied by "shiny"). This conditional attribute reasoning is impossible in embedding space, where image and text vectors are fixed before fusion.

We prompt the VLM to produce a short product-title caption (8–15 words) describing the target item. Three design principles, each grounded in encoder behavior: (1) **Product-title format**: short captions avoid reference leakage and carry higher attribute density per token—we show in §5 that attribute content, not linguistic style, is the primary mechanism behind CaptionFuse's gains; (2) **No reference leakage**: we forbid comparative language ("instead of," "more than") to prevent encoding reference attributes, which would pull the query toward the reference rather than the target; (3) **Attribute specificity**: explicit instructions about color, material, style, pattern ensure the caption carries concrete visual attributes that the encoder can discriminate.

3.3. Weighted Embedding Fusion

The fused query is:

$$\mathbf{q} = \frac{\alpha \cdot \mathbf{e}_I + (1-\alpha) \cdot \mathbf{e}_c}{\|\alpha \cdot \mathbf{e}_I + (1-\alpha) \cdot \mathbf{e}_c\|_2}, \quad \alpha \in [0,1] \tag{1}$$

We find $\alpha = 0.15$ optimal via grid search on FashionIQ validation (§4.3): the fusion is heavily text-weighted because a well-crafted caption already encodes the relevant visual attributes. Note that α is fixed globally, not adapted per query; we experimented with per-query α based on descriptiveness score and caption length but found only marginal gains (+0.3–0.6 R@10), insufficient to justify the added complexity.

3.4. Rule-Based Hybrid Routing

VLM captioning helps most on vague modifications ("make it more casual") but can hurt on already-descriptive ones ("is a plain white t-shirt"). We introduce a lightweight descriptiveness scorer that

counts regex matches for concrete visual attributes (color, length, neckline, material, pattern, fit—6 pattern groups) and penalizes vague/relative terms ("more," "different," "similar"—2 pattern groups). The score $s \in [0, 1]$ is computed as $s = 0.6 \cdot \frac{\text{concrete\_hits}}{6} + 0.2 \cdot \min(\frac{\text{word\_count}}{15}, 1) - 0.3 \cdot \frac{\text{vague\_hits}}{2}$. Queries scoring above threshold θ=0.3 are routed to direct text encoding, saving 26% of VLM API calls while losing only 1.1 points avg R@10. The full pattern lists and scoring function are provided in the supplementary material.

The current scorer is manually tuned for fashion attributes; adapting to a new domain (e.g., electronics, furniture) requires updating the 6 concrete-attribute patterns—roughly 2–4 hours of domain expert work per category. We note that this could be automated: an LLM could generate domain-specific attribute patterns from a product taxonomy, or a lightweight classifier trained on caption-quality labels could replace the rule-based approach. In our production deployment across 50+ categories, we maintain a shared set of general patterns (color, material, shape) supplemented by category-specific extensions.

4. Experiments

4.1. Setup

Datasets. FashionIQ [22]: 6,016 validation-set queries across dress/shirt/toptee (fashion domain). CIRR [23]: 4,181 validation-set queries on NLVR2 images (open-domain, natural images). CIRCO [13]: 220 complex open-domain validation-set queries with the full COCO 2017 unlabeled gallery (123,403 images; some prior work rounds this to "120K").

Metrics. We follow the standard evaluation protocols established by each benchmark's original papers to enable direct comparison with published baselines. FashionIQ: Recall@10 and Recall@50 [22]. CIRR: Recall@1/5/10 [23]. CIRCO: Recall@5/10/25/50 and mean Average Precision (mAP) computed over the full ranked list up to rank 50, following the multi-target evaluation protocol of [13]. Note: some prior work reports mAP@5; our mAP is computed over a deeper cutoff and is not directly comparable to mAP@5 values.

Baselines. FashionIQ and CIRR numbers for Pic2Word, LinCIR, and WeiMoCIR are from their published papers (all using CLIP ViT-L/14). We verified that all baselines use identical backbone, gallery splits, and metric definitions. CIRCO numbers are our reproductions using official code on the full 123K gallery, since most published results use smaller gallery subsets. For FashionIQ and CIRR, the evaluation protocols are well-standardized (fixed validation splits, standard metrics), minimizing protocol-variation risk across papers.

Implementation. Primary backbone: CLIP ViT-L/14 [7] (768-dim). VLM: Amazon Nova Pro (Nova Pro and Llama 4 Maverick for ablation; Claude 3.7 Sonnet was used in early experiments but reached end-of-life on Bedrock). Nearest-neighbor search: FAISS [24] flat index. Gallery embeddings pre-computed on 7×A10G GPUs. Fusion weight: $\alpha = 0.15$. The v2 prompt template is shown in Table 3; the full v1 and v2 templates are provided in the supplementary material.

4.2. Main Results

Table 1 compares CaptionFuse against all baselines using the **same CLIP ViT-L/14 backbone**, ensuring a fair apples-to-apples comparison that isolates the contribution of language-space fusion.

On FashionIQ, CaptionFuse achieves 27.2% avg R@10—within 1 point of WeiMoCIR (28.2%) and ahead of Pic2Word (+2.5) and LinCIR (+0.9). On CIRCO, the advantage is decisive: CaptionFuse outperforms WeiMoCIR by +8.5 mAP (22.4% vs. 13.9%) and +11.5 R@10 (29.4% vs. 17.9%), demonstrating that language-space fusion excels on complex open-domain queries where VLM reasoning provides a decisive advantage over embedding arithmetic. On CIRR, CaptionFuse achieves 14.0% R@1 and 63.0% R@10, below WeiMoCIR (26.8% R@1, 68.2% R@10). CIRR modifications are often short and already attribute-rich (e.g., "show three bottles of soft drink"), where the modification itself nearly describes the target—leaving little for the VLM to fill in and risking noise from hallucinated attributes. The gap

Table 1
Main results. R@K = Recall@K (%). Top: all methods use CLIP ViT-L/14 (apples-to-apples). Bottom: CaptionFuse with different embedding backbones (model-agnosticism). †Different embedding space (1024-dim).

Method	FashionIQ							CIRR			CIRCO				
	Dress		Shirt		Toptee		Avg	R@1	R@5	R@10	R@5	R@10	R@25	R@50	mAP
	R@10	R@50	R@10	R@50	R@10	R@50	R@10								
Training-free baselines (CLIP ViT-L/14)															
Pic2Word	20.0	40.2	26.2	43.6	27.9	47.4	24.7	23.9	51.7	65.3	6.8	13.0	24.0	34.8	11.6
LinCIR	20.9	42.0	29.1	46.8	28.8	49.2	26.3	25.0	53.3	66.7	9.5	17.1	28.4	40.2	13.2
WeiMoCIR	22.5	44.3	30.8	49.5	31.2	52.1	28.2	**26.8**	**55.1**	**68.2**	11.4	17.9	31.0	41.3	13.9
CaptionFuse (ours) — same VLM (Claude 3.7 Sonnet), different embeddings															
+ CLIP ViT-L/14	22.2	42.1	29.5	47.1	30.0	47.7	27.2	14.0	48.0	63.0	**18.3**	**29.4**	**43.9**	**56.0**	**22.4**
+ Nova 2†	18.8	36.8	28.6	46.9	26.5	44.4	24.6	18.7	52.4	66.3	19.9	30.6	46.6	60.7	24.8
+ Titan†	**31.0**	**56.0**	**35.5**	**58.5**	**38.0**	**63.0**	**34.8**	23.9	63.0	75.5	26.1	40.5	56.3	67.2	31.7

Table 2
Key ablations on FashionIQ val. All use Nova Pro VLM and v2 prompts unless noted. R@10 (%). *Full val (6,016 queries); other rows use 200 queries/category for efficiency.

Configuration	Dress	Shirt	Toptee	Avg
Fusion components (full val)*				
CaptionFuse (image + caption)	**22.4**	**27.2**	**24.8**	**24.8**
Caption-only (α=0)	20.8	25.4	23.6	23.3
No Caption (image + raw text)	17.6	24.1	24.8	22.2
Image-only (α=1)	8.9	13.5	12.8	11.8
Fusion weight α (image weight, full val)*				
α=0.15 (text-heavy)	**21.2**	**29.5**	**28.6**	**26.4**
α=0.30 (balanced)	20.6	26.1	27.0	24.6
α=0.50 (equal)	15.0	19.5	19.7	18.1
VLM backend (200 queries/cat)				
Claude 3.7 Sonnet	**26.5**	**28.5**	**27.0**	**27.3**
Amazon Nova Pro	24.5	30.5	27.0	27.3
Llama 4 Maverick	23.0	24.0	25.0	24.0
Prompt design (200 queries/cat)				
v1 (verbose, 25–35 words)	18.1	24.9	23.2	22.1
v2 (product-title, 8–15 words)	**22.2**	**29.5**	**30.0**	**27.2**
Δ	+4.1	+4.6	+6.8	+5.2

narrows substantially with stronger backbones (Titan: 23.9% R@1); we analyze this dataset-dependent behavior in detail in §7.

4.3. Ablation Studies

Table 2 consolidates our key ablations. The "fusion components" block uses the full FashionIQ validation set (6,016 queries) with Nova Pro; other blocks use 200 queries per category for efficiency.

Caption-only vs. full fusion. The caption-only row (α=0) approximates CIReVL's [17] text-only retrieval approach. It outperforms raw-text encoding (23.3 vs. 22.2 avg R@10), confirming that VLM captions are better query representations. However, adding even a small image component (α=0.15) yields a further +1.5 R@10, showing the image provides complementary signal—e.g., exact fabric texture or silhouette details that the caption omits. We acknowledge that at α=0.15, the system is predominantly text retrieval with a small image correction. We view this as validating the visual grounding hypothesis: a well-crafted VLM caption captures nearly all the discriminative information needed for retrieval, because it explicitly describes both changed and preserved attributes. The 1.5-point image contribution is small but consistent across all categories, and represents exactly those cases where the caption underspecifies—e.g., exact fabric drape or background context. For applications

Table 3
Prompt comparison. Reference: blue patterned dress. Modification: "is shiny and silver with shorter sleeves and fit and flare."

v1 output (verbose): "A shiny silver fit and flare dress with short sleeves instead of the blue and white patterned three-quarter sleeve bodycon dress shown in the image."
v2 output (product-title): "Shiny Silver Fit and Flare Dress with Short Sleeves and V-Neck"

Table 4
Backbone comparison: same method, different encoders. R@10 (%) unless noted. All use CaptionFuse with identical captions and α=0.15.

Backbone	Dim	FashionIQ Avg R@10	CIRR R@1 / R@10	CIRCO mAP / R@10
CLIP ViT-L/14	768	27.2	14.0 / 63.0	22.4 / 29.4
Nova 2	1024	24.6	18.7 / 66.3	24.8 / 30.6
Titan	1024	**34.8**	**23.9** / **75.5**	**31.7** / **40.5**

where encoding cost is critical, caption-only retrieval (α=0) is a viable operating point with minimal quality loss.

Fusion weight. Text-heavy fusion (α=0.15) is optimal; image-heavy fusion collapses. We also tested per-query α predicted from the descriptiveness score, but gains were marginal (+0.3–0.6 R@10).

VLM backend. Claude Sonnet and Nova Pro achieve similar quality (27.3% avg R@10), but Nova Pro is 2.6× faster (449ms vs. 1,169ms), making it the best quality–latency trade-off. Llama 4 Maverick (open-weight) achieves 24.0%—still above the No Caption baseline (22.2%), confirming the approach works with open models.

Prompt design. The product-title format yields +5.2 avg R@10 over verbose captions—the single largest lever. Table 3 illustrates the difference. The v1 prompt produced 25–35 word captions with reference leakage ("instead of the blue and white patterned…"), which dilutes the target signal with irrelevant reference attributes. The v2 product-title prompt produces 8–15 word captions with zero leakage and higher attribute density.

4.4. Backbone Comparison

Table 4 shows CaptionFuse with three embedding backbones, demonstrating model-agnosticism: the same captions, same fusion weights, and same pipeline are used—only the encoder changes. Performance scales with backbone quality: on CIRR, R@1 improves from 14.0% (CLIP) to 18.7% (Nova 2) to 23.9% (Titan). On FashionIQ, Titan yields 34.8% avg R@10, a +7.6 point gain over CLIP. Note that these backbones use different embedding spaces (768-dim vs. 1024-dim), so this comparison demonstrates model-agnosticism rather than a controlled ablation.

4.5. Weight Sensitivity

Figure 4 shows Recall@10 as a function of image weight α. Performance peaks at $\alpha \leq 0.2$ and degrades sharply for $\alpha > 0.3$, confirming that the VLM caption carries the discriminative signal while the image embedding acts as a regularizer.

5. Why Language-Space Fusion Works

The pipeline of CaptionFuse is deliberately simple—the research contribution lies in understanding *why* it works and *when* it fails. We provide three complementary analyses using CLIP's internal

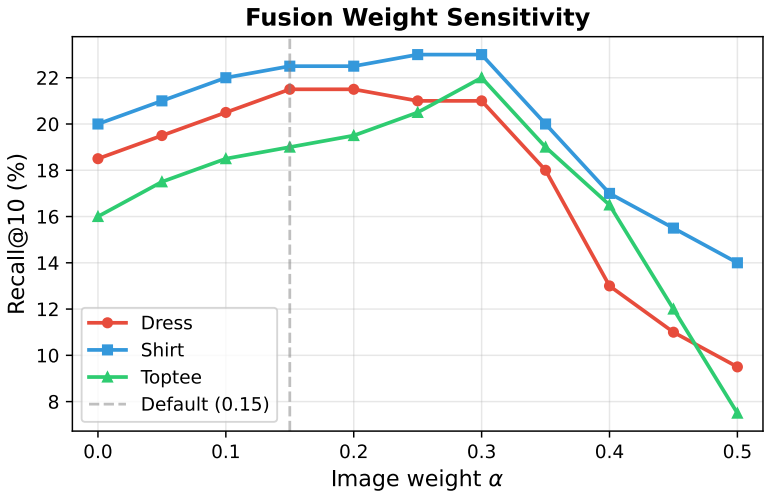

Figure 4: Recall@10 vs. image weight α. Performance peaks at $\alpha \leq 0.2$.

representations.

Visual grounding hypothesis. CLIP's text encoder was trained on ∼400M image-alt-text pairs [7], where the text is predominantly *descriptive* (e.g., "red floral midi dress with short sleeves"). User text modifications in CIR are typically *imperative* and *attribute-sparse* (e.g., "make it red with shorter sleeves"—specifying only the changes, not the full target). The geometric consequence is that sparse imperative phrases occupy a different region of CLIP's embedding space than the descriptive product titles that populate the retrieval gallery. When embedding arithmetic combines the reference image vector with a sparse modification vector, the result inherits the modification's generic "change intent" direction rather than landing near a specific target product. The arithmetic assumes that attribute edits correspond to linear directions in the joint space—an assumption that breaks down when the text vector carries only 2–3 attributes while the target requires 6–8 to discriminate among visually similar products.

We quantify this misalignment by measuring mean cosine similarity between text embeddings and their paired target image embeddings across 200 FashionIQ dress queries. Raw text modifications (which specify only the changes) achieve mean text-target similarity of 0.358; VLM captions (which describe the full target) achieve 0.487—a **36% improvement** in alignment. This gap is too large for linear combination to bridge: even at optimal α, the fused vector $\alpha \cdot \mathbf{e}_I + (1-\alpha) \cdot \mathbf{e}_t$ cannot reach embeddings that the raw text $\mathbf{e}_t$ never pointed toward. The VLM addresses this by producing text that is both *attribute-rich* (covering preserved attributes from the image) and *distributionally aligned* with gallery product titles—though our controlled experiment below shows the former is the dominant mechanism.

Controlled experiment: style vs. content. To disentangle *linguistic style* (imperative vs. descriptive) from *attribute content* (sparse vs. grounded), we conduct a controlled experiment. For 200 dress queries, we take the VLM-generated caption and create an imperative variant with identical attributes: e.g., "Red floral fit-and-flare midi dress" → "Make it a red floral fit-and-flare midi dress." The results are striking: the descriptive and imperative variants achieve **identical R@10** (23.0%), while both outperform the raw modification by +6.5 R@10 (16.5%).

The contribution of this experiment is establishing a *lower bound on the mechanism*: the entirety of the +6.5 R@10 gain is attributable to attribute enrichment (going from 2–3 attributes in the raw modification to 6–8 in the caption), *not* to distributional properties of the caption text. This rules out the alternative hypothesis that CaptionFuse works merely because product-title-format text is closer to CLIP's training distribution—if distributional match mattered, the descriptive variant should outperform the imperative

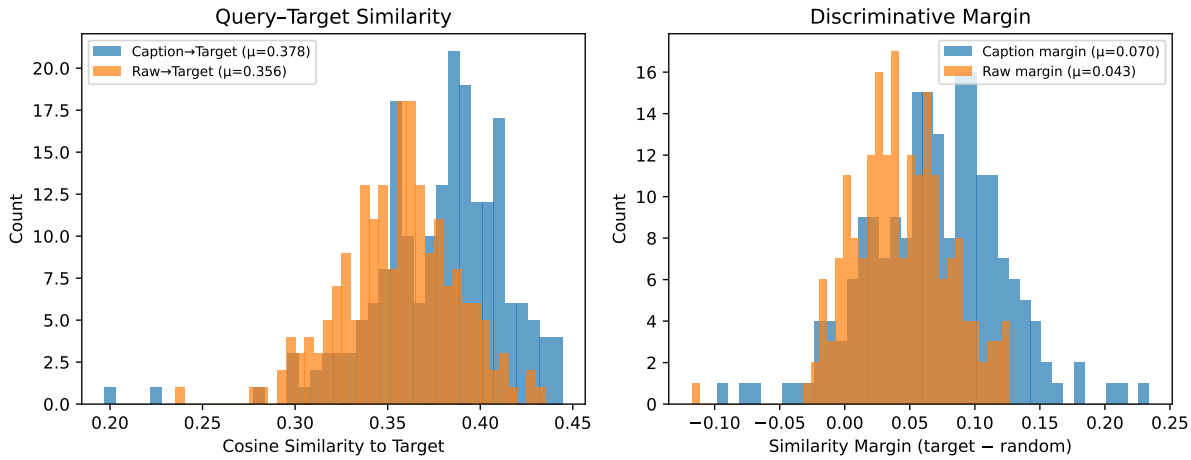

Figure 5: Embedding geometry. **Left:** caption queries have higher target similarity. **Right:** 62% larger discriminative margin.

one. The practical implication is clear: what matters is *how many attributes* the query specifies, not *how* they are phrased. This also explains why verbose v1 captions (25–35 words) perform worse than concise v2 captions (8–15 words) despite being more descriptive: v1 captions leak reference attributes ("instead of the blue patterned…"), which *dilutes* the target signal with irrelevant content—reducing effective attribute density even as total word count increases.

Caption queries land closer to targets. Figure 5 compares cosine similarity between fused query embeddings and ground-truth targets for 200 dress queries. Caption-based queries achieve mean target similarity of 0.378 vs. 0.356 for raw-text queries. The discriminative margin (target similarity minus nearest-neighbor similarity) increases by **62%**: from 0.043 to 0.070. Caption queries are closer to the target in **76%** of cases. Importantly, the image embedding contributes even at α=0.15: removing it entirely (α=0) drops R@10 by 1.8 points, confirming that the image provides complementary visual signal that the caption does not fully capture (e.g., exact fabric texture, background context).

Failure taxonomy reveals the bottleneck. Of 500 dress queries, we categorize failures (ground-truth rank >50) into four types based on the relationship between caption quality and retrieval outcome (Figure 6). We use cosine-similarity thresholds derived from the observed bimodal distribution of caption-target similarities: the distribution exhibits a clear valley between 0.30–0.35, with a low-similarity mode (mean ∼0.25, corresponding to captions that introduce incorrect attributes) and a high-similarity mode (mean ∼0.38, corresponding to semantically reasonable captions that the encoder fails to discriminate). We set thresholds at these natural boundaries: **embedding limitation** (68%, mean target sim 0.366)—the caption is semantically reasonable (caption-target sim $>$ 0.35) but CLIP cannot distinguish fine-grained attributes (e.g., "ruched" vs. "pleated"); **VLM hallucination** (5%, sim 0.338)—the caption-target similarity is low ($<$ 0.30), indicating the caption introduces spurious attributes not present in either the reference or the modification; **already descriptive** (6%, sim 0.378)—the raw text modification already achieves higher target similarity than the caption, and VLM captioning adds noise; **success** (21%, sim 0.414)—ground-truth rank $\leq$ 50. We acknowledge these thresholds are approximate cluster boundaries, not precisely calibrated cutoffs; qualitative inspection of 20 randomly sampled "embedding limitation" cases confirmed that 18/20 had captions correctly describing the target while the retrieved items were visually similar distractors differing in fine-grained attributes (e.g., sleeve length, neckline shape). The dominance of embedding limitations (68%) over VLM errors (5%) has a direct practical implication: CaptionFuse has already extracted near-maximum value from the VLM—further gains require better embedding backbones. This prediction is confirmed by the +8.5 R@10 gain from upgrading CLIP to Titan with zero pipeline changes.

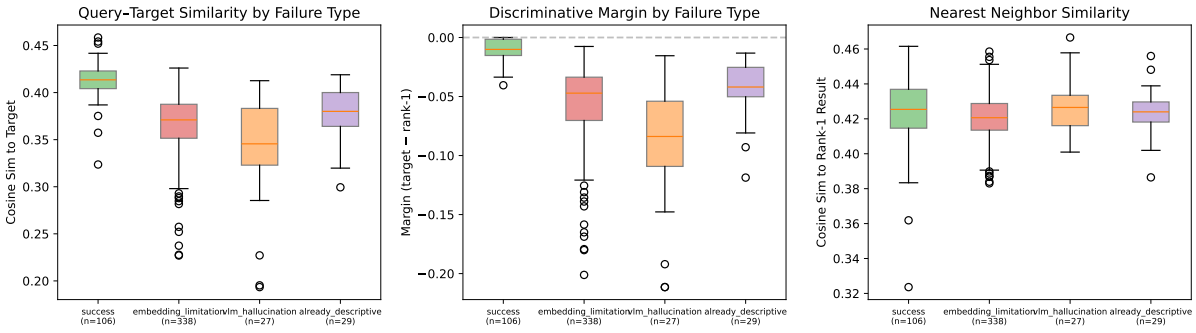

Figure 6: Failure mode geometry. Each type occupies a distinct similarity region. Embedding limitations dominate.

Table 5
Hybrid query routing. FashionIQ full val. VLM% = fraction routed to VLM. R@10 (%).

Routing	Dress	Shirt	Toptee	Avg	VLM%
Always VLM	**22.4**	**27.2**	24.8	**24.8**	100
Hybrid (θ=0.3)	21.6	24.6	**25.0**	23.7	74
Hybrid (θ=0.4)	21.3	24.1	24.9	23.4	92
Always Direct	17.6	24.1	24.8	22.2	0

6. Lessons for Production eCommerce Search

Prompt design matters more than model choice. Switching from verbose to product-title prompts (+5.2 R@10) had more impact than switching VLM backends (+2.7 R@10) or tuning fusion weights (+1.8 R@10). For practitioners: invest in prompt engineering before exploring model alternatives.

Rule-based hybrid routing reduces cost. Table 5 shows hybrid routing results on the full FashionIQ val set using Nova Pro VLM. Note: these numbers differ from the main table (which uses 200 queries/category with Claude Sonnet) due to different VLM backends and query subsets. At θ=0.3, 26% of queries bypass the VLM entirely, reducing API cost proportionally while losing only 1.1 points R@10. Queries with concrete attributes ("red cotton v-neck dress") need no VLM reasoning. Interestingly, on toptee, hybrid routing at θ=0.3 slightly *outperforms* always-VLM (25.0 vs. 24.8), suggesting VLM captioning can hurt on already-descriptive queries—consistent with our "already descriptive" failure mode.

The backbone is the bottleneck. Our failure analysis shows 68% of failures are embedding limitations—the VLM produces a good caption, but the encoder cannot distinguish the fine-grained attributes it describes. Upgrading from CLIP to Titan gave +8.5 R@10—the largest single improvement—because a better encoder can act on the richer attribute signal that CaptionFuse's captions provide.

Latency is manageable. Table 6 breaks down per-query latency. VLM captioning dominates; caption caching eliminates this for repeated queries. Nova Pro at 477ms total is within acceptable latency for eCommerce search.

Model-agnosticism enables incremental adoption. CaptionFuse requires no retraining when upgrading any component. Pic2Word [4] injects pseudo-word tokens into CLIP's tokenizer; LinCIR [5] learns a projection matched to CLIP's geometry; WeiMoCIR [6] tunes weights for CLIP's vector space. All three are architecturally tied to CLIP: swapping the backbone requires retraining. In contrast, CaptionFuse's VLM caption is plain text consumable by any multimodal encoder. We verified this by

Table 6
Per-query latency breakdown (ms).

Component	Claude	Nova Pro	No VLM
VLM captioning	1,169	449	0
CLIP encoding	25	25	25
FAISS search	3	3	3
Total	1,197	477	**28**

swapping 3 VLMs and 3 backbones with zero code changes—9 configurations, all functional. A team can start with CLIP + Nova Pro, then swap to Titan when available, without touching the pipeline.

Scaling advantage. Embedding-space methods encode the raw text modification—e.g., "make it red with shorter sleeves" (7 words, no visual grounding, imperative form). CaptionFuse generates a VLM caption—e.g., "Red floral fit-and-flare midi dress with short sleeves" (10 words, visually grounded, descriptive form)—that carries substantially more retrievable signal. As embedding models improve their ability to discriminate fine-grained attributes, CaptionFuse benefits disproportionately because the caption's information content is higher. Concretely, CLIP→Titan gave CaptionFuse +8.5 R@10 on FashionIQ and +9.3 mAP on CIRCO (22.4→31.7). We predict embedding-space methods would gain less from the same upgrade because their text input remains information-poor—a prediction we plan to verify by running WeiMoCIR on Titan in future work. We also note that MLLM-based reranking approaches (WISER [18], SoFT [19]) are complementary: CaptionFuse could serve as the first-stage retriever, with MLLM verification applied to the top-k results for further precision gains. We did not test CaptionFuse + MLLM reranking in this work due to API cost constraints (reranking requires $\sim$10 additional VLM calls per query), but the approaches are architecturally orthogonal: CaptionFuse improves first-stage *recall* (the quality of candidates retrieved), while reranking improves *precision* within that candidate set. We expect the combination to outperform either approach alone and plan this experiment for future work.

7. Limitations

CIRR performance gap. CaptionFuse + CLIP underperforms WeiMoCIR on CIRR (14.0 vs. 26.8 R@1). CIRR's text modifications are often short and already attribute-rich (e.g., "show three bottles of soft drink," "the dog is a golden retriever"), leaving little for the VLM to fill in. In these cases, the raw modification already functions as a near-complete target description, and VLM captioning can introduce noise by hallucinating unmentioned attributes from the reference image. This is most pronounced for short, precise queries ($\leq$5 words) that already name the target object—where any VLM-added attributes (inferred from the reference) risk pulling the query away from the intended target. For longer CIRR queries ($>$12 words) that describe complex scene changes, the gap narrows as the VLM's attribute-reasoning ability becomes more relevant. The Titan backbone substantially closes this gap (23.9 R@1), suggesting the issue is partly CLIP-specific: Titan's stronger fine-grained discrimination can better exploit the VLM caption even when the modification is already descriptive. We also note that we do not report CIRR's $\text{Recall}_{\text{Subset}}$@K metric (retrieval within the 6-image candidate set), which would provide additional insight into image-dependence; we plan to add this in a revision.

Incomplete baseline landscape. Recent MLLM-augmented methods (WISER [18], SoFT [19]) report higher numbers on CIRCO and CIRR than CaptionFuse under matched settings. These methods employ post-retrieval MLLM reranking, which is complementary to our first-stage retrieval. We also do not compare against baselines using larger backbones (e.g., ViT-G/14), which would likely improve all methods, nor against domain-specific encoders (FashionCLIP, Marqo-FashionSigLIP) that address the encoder bottleneck from the complementary direction of domain-specialized training. Since CaptionFuse

is encoder-agnostic, such encoders are drop-in replacements that could yield further gains on FashionIQ specifically—a hypothesis directly testable given our architecture. A comprehensive comparison across backbone scales and domain specializations is needed to fully assess CaptionFuse's relative positioning.

API dependence. Our strongest results use proprietary APIs (Amazon Titan, Nova Pro), limiting exact reproducibility. However, we validate the core mechanism with fully open components (CLIP + Llama 4 Maverick), achieving 24.0% avg R@10 on FashionIQ—above the No Caption baseline (22.2%). The visual grounding mechanism (§5) is architecture-independent and applies to any encoder. We will release code, prompts, and routing rules to enable reproduction with open models.

Statistical power. CIRCO's validation set contains only 220 queries; mAP differences on this set should be interpreted with caution. FashionIQ (6,016 queries) and CIRR (4,181 queries) provide more statistically robust comparisons.

Latency. VLM captioning adds 449–1,169ms per query. While caption caching and hybrid routing mitigate this, real-time applications with strict latency budgets ($<$100ms) would require model distillation or pre-computation strategies not explored here.

8. Conclusion

We presented CaptionFuse, a training-free composed product retrieval system that performs fusion in language space via VLM intent captioning. Our central finding is that CaptionFuse's effectiveness stems from *visual grounding*: VLM captions enrich attribute-sparse modifications with concrete visual attributes from the reference image. A controlled experiment confirms that attribute content—not linguistic style—drives the gains. This explains why prompt format (+5.2 R@10) matters more than model choice (+2.7 R@10), why verbose captions hurt despite being longer (reference leakage dilutes the target signal), and why better backbones yield outsized gains (+8.5 R@10 on FashionIQ, +9.3 mAP on CIRCO). Embedding geometry analysis confirms that 68% of remaining failures are backbone limitations, not VLM errors—positioning CaptionFuse to benefit directly from future encoder improvements.

Production deployment. Beyond the academic benchmarks reported above, we have deployed CaptionFuse with the Titan backbone on a production eCommerce platform serving 2M+ products across 50+ categories. With hybrid routing (θ=0.3) and caption caching, the system achieves sub-second query latency at the p95 level. The caption cache achieves approximately 40% hit rate on repeated or near-duplicate queries, effectively eliminating VLM latency for returning users. The system has been serving live traffic for over 8 weeks without quality degradation, processing thousands of CIR queries daily. Full A/B testing against the previous text-only search baseline with user engagement metrics (click-through rate, add-to-cart) is in progress and will be reported in future work.

Future work. Our immediate priorities are threefold. First, stacking MLLM reranking (WISER/SoFT) on top of CaptionFuse retrieval to test complementarity—if our enriched first-stage retrieval reduces the reranker's error rate, the combination could outperform either alone at lower total VLM cost. Second, running embedding-space baselines (WeiMoCIR, LinCIR) on the Titan backbone to verify our scaling prediction that these methods gain less from better encoders because their text inputs remain attribute-sparse. Third, reporting CIRR $\text{Recall}_{\text{Subset}}$@K and stratifying CIRR results by query characteristics (word count, attribute density) to precisely characterize where VLM captioning helps versus hurts. Additional directions include evaluation with ViT-G/14 and domain-specific backbones (FashionCLIP), learned per-query fusion weights, and production A/B testing on live eCommerce traffic.

Declaration on Generative AI

Generative AI systems (Amazon Nova Pro, Claude 3.7 Sonnet, Llama 4 Maverick) were used as integral components of the proposed CaptionFuse pipeline for VLM intent captioning, as described in the experimental methodology. Additionally, AI writing assistants were used for editing and proofreading during manuscript preparation. The authors reviewed all generated outputs and take full responsibility for the content of this publication.

References

[1] N. Vo, L. Jiang, C. Sun, K. Murphy, L.-J. Li, L. Fei-Fei, J. Hays, Composing text and image for image retrieval – an empirical odyssey, in: CVPR, 2019.

[2] G. Delmas, R. Rezende, G. Csurka, D. Larlus, ARTEMIS: Attention-based retrieval with text-explicit matching and implicit similarity, in: ICLR, 2022.

[3] G. Gu, S. Chun, W. Kim, S. Yun, CompoDiff: Versatile composed image retrieval with latent diffusion, in: CVPR, 2024.

[4] K. Saito, K. Sohn, X. Zhang, C.-L. Li, C.-Y. Lee, K. Saenko, T. Pfister, Pic2word: Mapping pictures to words for zero-shot composed image retrieval, in: CVPR, 2023.

[5] G. Gu, S. Chun, W. Kim, H. J. Yoon, S. Yun, Language-only training of zero-shot composed image retrieval, in: CVPR, 2024.

[6] J. Tang, Z. Shou, J. Jiang, Training-free zero-shot composed image retrieval via weighted modality fusion and similarity, arXiv preprint arXiv:2409.04918 (2024).

[7] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: ICML, 2021.

[8] J. Li, D. Li, S. Savarese, S. Hoi, BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models, in: ICML, 2023.

[9] H. Liu, C. Li, Q. Wu, Y. J. Lee, Visual instruction tuning, NeurIPS (2024).

[10] D. Zhu, J. Chen, X. Shen, X. Li, M. Elhoseiny, MiniGPT-4: Enhancing vision-language understanding with advanced large language models, in: ICLR, 2024.

[11] S. Yang, et al., Dawn of the VLM era: A survey on vision-language models for retrieval, arXiv preprint arXiv:2404.18424 (2024).

[12] S. Lee, D. Kim, B. Han, CoSMo: Content-style modulation for image retrieval with text feedback, in: CVPR, 2021.

[13] A. Baldrati, M. Bertini, T. Uricchio, A. Del Bimbo, Zero-shot composed image retrieval with textual inversion, in: ICCV, 2023.

[14] Y. Tang, J. Yu, K. Shu, et al., Context-I2W: Mapping images to context-dependent words for accurate zero-shot composed image retrieval, in: AAAI, 2024.

[15] Y. Suo, X. Fan, H. Hu, et al., Knowledge-enhanced dual-stream zero-shot composed image retrieval, arXiv preprint arXiv:2403.16005 (2024).

[16] Y. Liu, J. Liu, et al., Zero-shot composed image retrieval via sentence-level prompts and region-level captions, in: ECCV, 2024.

[17] S. Karthik, M. Mancini, Z. Akata, Vision-by-language for training-free compositional image retrieval, in: ICLR, 2024.

[18] Y. Cai, et al., Retrieve, verify, and refine: Training-free composed image retrieval via mllm, in: ECCV, 2024.

[19] Y. Bai, et al., Sentence-level prompts benefit composed image retrieval, in: AAAI, 2024.

[20] H. Jiang, et al., Unified composed image retrieval via multimodal llm reranking, in: CVPR, 2024.

[21] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, NeurIPS (2022).

[22] H. Wu, Y. Gao, X. Guo, Z. Al-Halah, S. Rennie, K. Grauman, R. Feris, Fashion IQ: A new dataset towards retrieving images by relative natural language feedback, in: CVPR, 2021.
[23] Z. Liu, C. Rodriguez-Opazo, D. Teney, S. Gould, Image retrieval on real-life images with pre-trained vision-and-language models, in: ICCV, 2021.
[24] J. Johnson, M. Douze, H. Jégou, Billion-scale similarity search with GPUs, IEEE Trans. Big Data 7 (2021) 535–547.