

# PDV: Prompt Directional Vectors for Zero-shot Composed Image Retrieval

Osman Tursun<sup>1</sup>, Sinan Kalkan<sup>2</sup>, Simon Denman<sup>1</sup> and Clinton Fookes<sup>1</sup>

<sup>1</sup>Queensland University of Technology

<sup>2</sup>Middle East Technical University

{osman.tursun, s.denman, c.fookes}@qut.edu.au, skalkan@metu.edu.tr

## Abstract

*Zero-shot Composed Image Retrieval (ZS-CIR) enables image search using a reference image and a text prompt without requiring specialized text-image composition networks trained on large-scale paired data. However, current ZS-CIR approaches suffer from three critical limitations in their reliance on composed text embeddings: static query embedding representations, insufficient utilization of image embeddings, and suboptimal performance when fusing text and image embeddings. To address these challenges, we introduce the **Prompt Directional Vector (PDV)**, a simple yet effective training-free enhancement that captures semantic modifications induced by user prompts. PDV enables three key improvements: (1) Dynamic composed text embeddings where prompt adjustments are controllable via a scaling factor, (2) composed image embeddings through semantic transfer from text prompts to image features, and (3) weighted fusion of composed text and image embeddings that enhances retrieval by balancing visual and semantic similarity. Our approach serves as a plug-and-play enhancement for existing ZS-CIR methods with minimal computational overhead. Extensive experiments across multiple benchmarks demonstrate that PDV consistently improves retrieval performance when integrated with state-of-the-art ZS-CIR approaches, particularly for methods that generate accurate compositional embeddings. The code will be released upon publication.*

## 1. Introduction

Composed Image Retrieval (CIR) involves searching for images using a combination of a reference image and a prompt that describes how the target image should differ from the reference [3, 6, 22, 26]. Compared to traditional content-based image retrieval (CBIR) systems, CIR offers increased flexibility and precision by allowing users to articulate complex, multi-modal queries that combine visual and semantic information [8, 14, 22].

The core challenge in CIR lies in effectively integrat-

ing information from two distinct modalities: image and text. With the rapid progress in vision and language models (VLMs), CIR has attracted significant attention in the computer vision community [3, 6, 14, 18, 22]. Early approaches to CIR were primarily supervised in nature [1, 7, 15, 16, 26, 28]. However, as highlighted by Saito et al. [22], the labeling cost for supervised datasets in this domain is prohibitively high, prompting researchers to explore more efficient alternatives, namely zero-shot composed image retrieval (ZS-CIR). In this work, we provide a simple and training-free approach to improve the controllability and accuracy of existing ZS-CIR approaches.

ZS-CIR leverages VLMs, denoted by  $\Psi$ , which operate through a dual-pathway architecture. The first pathway consists of a vision branch,  $\Psi_I$ , that extracts feature representations from target images,  $I_{target}$ . The second pathway employs a language branch,  $\Psi_T$ , that processes a textual composition of reference images,  $I_{ref}$ , and user-provided text prompts,  $P$ . This composition, represented by  $\mathcal{F}(I_{ref}, P)$ , can be achieved through two primary methods: (1) *Caption Generation*, where a caption is generated for the reference image using a VLM, and this caption is merged with the text-prompt using Large Language Models (LLMs), as demonstrated in CIREVL [14]; or (2) *Pseudo Tokenization*, which uses CLIP’s [20] visual branch to process  $I_{ref}$  and a mapping network (consisting of a lightweight multi-layer perceptron) to tokenise the image, as demonstrated in Pic2Word [22]. The resulting  $\mathcal{F}(I_{ref}, P)$  is a textual query representation that encompasses both the provided visual and text information, and facilitates zero-shot retrieval. The aforementioned pipeline is illustrated in Figure 1a.

We identify three major gaps in the literature, despite the promising results [3, 11, 14, 22]:

**Gap 1: Inefficient Iterative Search Due to Static Composed Embeddings** Existing CIR studies primarily emphasize initial retrieval success but neglect users’ need to iteratively refine and steer results when initial searches fail. This limitation stems mainly from the static nature of composed text embeddings. Existing ZS-CIR approaches are unable to directly tune the composed text embedding,

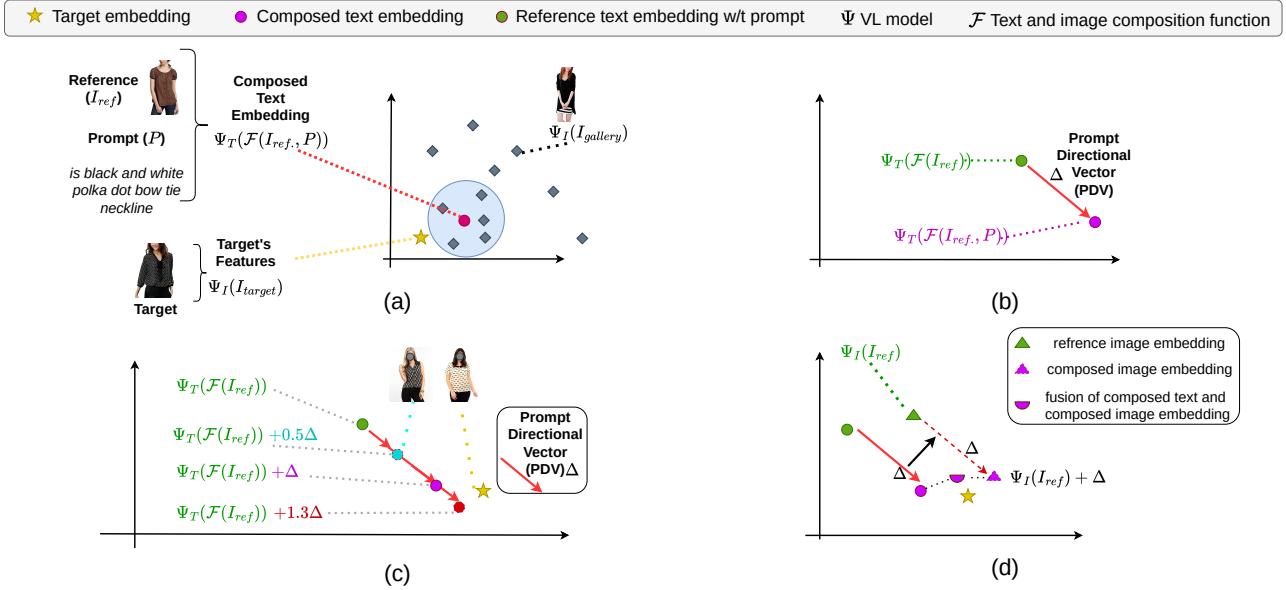


Figure 1. Overview of Prompt Directional Vector (PDV) for Zero-Shot Composed Image Retrieval (ZS-CIR). (a) Standard ZS-CIR pipeline. (b) PDV calculation process. (c) Dynamic text embedding composition using PDV. (d) Fusion of composed text and image embeddings: PDV-modified composed image embedding combined with composed text embedding.

$\Psi_T(\mathcal{F}(I_{ref}, P))$ , to bring it closer to the target image embedding,  $\Psi_T(\mathcal{F}(I_{ref}, P))$ . With existing ZS-CIR approaches users need to refine their prompts to iteratively improve retrieval results. However, this approach incurs additional manual effort for prompt construction and computational overhead from expensive feature extraction.

**Gap 2: Underutilisation of Reference Image Embedding.** Current methods generally do not utilize the embedding  $\Psi_I(I_{ref})$  of the reference image directly for retrieval; instead,  $\Psi_I(I_{ref})$  is used solely for composition. This omission stems from consistently poor retrieval performance (refer to Image-only results in Table S8) when incorporating these embeddings, as documented in multiple studies [3, 14, 22].

**Gap 3: Suboptimal Performance of Image-Text Embedding Fusion.** While the fusion of image and text embeddings outperforms single-modality approaches (image-only or text-only) [3, 14, 22], it still underperforms compared to composed text embeddings (refer to “Image+Text” results in Tables S5, S6 in the supplementary.).

**Promp Directional Vector (PDV): a Plug-and-Play Solution.** We propose the *Promp Directional Vector* (PDV) as a straightforward, training-free approach to address the aforementioned gaps. Denoted by  $\Delta_{PDV}$ , the PDV represents the residual vector between two text embeddings: the composed text embedding  $\Psi_T(\mathcal{F}(I_{ref}, P))$  and the reference image text embedding  $\Psi_T(\mathcal{F}(I_{ref}))$ . The latter is equivalent to  $\Psi_T(\mathcal{F}(I_{ref}, P_{Empty}))$ , where  $P_{Empty}$  represents an empty input string, corresponding to the unprompted base-

line. As illustrated in Figure 1b and shown via a red arrow, this PDV captures the semantic modification induced by the prompt. In the following, we summarize how the PDV effectively addresses the three aforementioned challenges.

**PDV: Addressing Gap 1.** To change the static nature of the composed text embedding and increase flexibility and utility for users, we generalise the synthesis of the composed text embeddings  $\Psi_T(\mathcal{F}(I_{ref}, P))$ . We interpret  $\Psi_T(\mathcal{F}(I_{ref}, P))$  as a shift from the reference image text embedding without the prompt,  $\Psi_T(\mathcal{F}(I))$ , by a vector  $\Delta_{PDV}$ . Under this formulation, the baseline ZS-CIR approach can be viewed as a special case where  $\Psi_T(\mathcal{F}(I_{ref}, P)) = \Psi_T(\mathcal{F}(I_{ref}, P)) + \alpha\Delta_{PDV}$  with  $\alpha = 1$ . We hypothesize that when  $\Delta_{PDV}$  captures the desired modifications, but not their precise magnitude (particularly with less descriptive prompts), adjusting  $\alpha$  can enhance retrieval performance and controllability. As demonstrated in Figure 1c, increasing  $\alpha$  to 1.3 produces results more closely aligned with the target compared to the default  $\alpha = 1$ .

**PDV: Addressing Gap 2.** Although image embeddings  $\Psi_I(I_{ref})$  contain valuable visual content regarding the reference image, they lack prompt-specific semantic information, leading to poor performance when used in ZS-CIR. By leveraging the shared semantic space learned by Vision-Language models, we can transfer prompt semantics to the image embedding by adding the Prompt Vector  $\Delta$ , obtaining  $\Psi_I(I_{ref}) + \alpha\Delta_{PDV}$ , as illustrated in Figure 1d. We denote this augmented representation as the *composed image embedding*. Similar to the dynamic composed text embed-

ding, this representation can be adjusted through a scaling factor,  $\alpha$  to offer controllability to enhance retrieval.

**PDV: Addressing Gap 3.** Lastly, several studies demonstrate that the direct fusion of image and text embeddings outperforms using either input feature (image or text-prompt) alone [3, 14, 22]. However, this fusion approach still underperforms compared to using the composed text embeddings. This performance gap exists because prompt embeddings are significantly changed by incorporating context from the reference image. Specifically,  $\Delta_{PDV}$  is not equivalent to  $\Psi_T(P)$ . To address this, we propose fusing the composed text and composed image embeddings, as illustrated in Figure 1d. Through varying the fusion weight factor  $\beta$ , we can dynamically control the balance between visual similarity to the reference image and semantic alignment with the prompt without needing to craft new prompts, or modify reference images. Lower  $\beta$  values prioritize visual fidelity, while higher values emphasize semantic modifications specified in the prompt. The previous work, CompoDiff [12], also demonstrates similar controllability through adjusting visual and semantic weights. However, their method cannot be applied to other approaches as ours can, as our method is an add-on approach while theirs is specifically tailored to their own framework.

PDV serves as a plug-and-play enhancement for most ZS-CIR approaches, offering a simple and training-free solution. The computational overhead is minimal, requiring only the calculation of text and image embeddings from the reference image. We evaluate PDV by integrating it with four distinct ZS-CIR methods across various CIR benchmarks. Our experimental results demonstrate that all three use cases of PDV consistently improve upon baseline approaches, particularly when the baseline method already generates accurate compositional embeddings.

**Contributions.** Our main contributions are as follows: (i) We introduce the Prompt Directional Vector (PDV), a simple and training-free enhancement that overcomes limitations of current Zero-Shot CIR methods. (ii) We propose three novel applications of PDV: (1) dynamic composed text embedding synthesis through PDV scaling, which offers enhanced control over retrieval results without tedious prompt modification; (2) composed image embedding synthesis via semantic transfer of prompts to visual features through PDV addition, which prioritizes visual similarity; and (3) effective fusion of composed text and image embeddings, which improves overall performance and enables controllable balancing of visual and semantic similarity. (iii) Through extensive experiments on multiple benchmarks with four ZS-CIR methods, we demonstrate that PDV consistently improves retrieval performance with minimal computational overhead.

## 2. Related Work

**Vision-Language (VL) models** have revolutionized computer vision by effectively bridging visual and textual modalities. The emergence of powerful models such as CLIP [20], ALIGN [13], and Florence [32] has enabled remarkable advances in multi-modal understanding. Trained on large-scale image-text pairs through contrastive learning, these models learn rich visual-semantic representations that generalize across domains and tasks. Building upon these advances, Composed Image Retrieval (CIR) has shown significant progress [6, 14, 22]. Early approaches leveraged VL models and either trained a combiner network to compose text and image features [6], or fine-tuned a text encoder [4] to extract task-specific text features. However, these methods required expensive domain-specific triplets (reference image, modified image, and text description) that must be manually verified. Recent work has explored alternative approaches to reduce the data collection burden, such as using synthetic triplets [12] or mining triplets from large-scale image-text datasets [17]. However, these methods still incur significant computational costs during training.

**Zero-shot CIR with Text Inversion** Recent research has focused on zero-shot approaches to address these challenges. Many methods adopt *text inversion*, a technique initially proposed for personalized image generation [10, 21], which maps images to pseudo-tokens or words. Pic2Word [22] introduced a self-supervised text inversion network trained with cyclic contrastive loss, though it requires a large-scale image dataset. SEARLE [3] reduces the cost of training Pic2Word and improves the efficiency of the text inversion network. KEDs [23] implicitly models the attributes of the reference images by incorporating a database; thus, tokens obtained through inversion include attributes such as color, object number and layout. To further improve scalability, LinCIR [11] proposed a language-only approach that reduces training costs and increases scalability. Most recently, CIReVL [14] introduced a more direct approach that leverages image captioning models to generate natural language descriptions of reference images, which are then combined with text that specifies desired modifications to form queries. Subsequently proposed methods, such as LDRE [30] and SEIZE [29], leverage multiple captions over a single caption to increase the diversity and also take the semantic increment during the composition into consideration.

**Composition with a Residual** In contrast to ZS-CIR, early supervised CIR approaches learned prompt-induced modifications by training on labelled triplet data (reference image, prompt, and target image). Vo et al. [26] pioneered this approach by introducing a residual learning module based on an LSTM network. Subsequently, several methods [7, 28, 31] adopted similar residual learning strategies for text-image composition. Baldrati et al. [4] further ad-

vanced this approach by fine-tuning CLIP’s text encoder to learn residual embeddings. While these prior works explored residual-based approaches, they all relied on supervised training. In contrast, our proposed PDV achieves similar capabilities by directly leveraging pre-trained VL models, eliminating the need for task-specific training.

### 3. Methodology

#### 3.1. Baseline ZS-CIR Framework

Composed Image Retrieval (CIR) enables users to search for target images  $I_{target}$  by providing a reference image,  $I_{ref}$ , and a text prompt,  $P$ , describing desired modifications. Zero-shot composed image retrieval (ZS-CIR) leverages Vision-Language (VL) models,  $\Psi$ , such as CLIP [20], whose vision branch,  $\Psi_I$ , and text branch,  $\Psi_T$ , are trained to learn a shared embedding space where semantically similar image and text pairs are mapped close to each other. In this framework, as shown in Figure 1a, target images are encoded using the vision branch,  $\Phi_I$ , while the query is composed by processing both  $I_{ref}$  and  $P$  through the text branch  $\Psi_T$ , as composition operations are more naturally handled in the text modality.

Recent ZS-CIR approaches generate the composed text embedding from  $I_{ref}$  and  $P$  using one of two methods: direct image captioning (CIReVL, LDRE and SEIZE) or pseudo tokenization (Pic2Word, LinCIR, SEARLE and KEDs). We denote this composition process as  $\mathcal{F}$ , resulting in a composed text embedding  $\Psi_T(\mathcal{F}(I_{ref}, P))$ .

In an ideal ZS-CIR scenario, the target image  $I_{target}$  should appear within the top-k results retrieved from the gallery  $\mathcal{D}$ . This retrieval is formalized as:

$$\mathbb{I}_{top-k} = \arg \max_{I \in \mathcal{D}} \frac{\Psi_T(\mathcal{F}(I_{ref}, P))^T \cdot \Psi_I(I)}{\|\Psi_T(\mathcal{F}(I_{ref}, P))\| \cdot \|\Psi_I(I)\|}. \quad (1)$$

If  $I_{target} \notin \mathbb{I}_{top-k}$ , the user must reformulate the prompt and repeat the feature extraction process to obtain alternative retrieval results, incurring time and computational resource costs. Notably, as shown in Eq. 1, only the composed feature embedding  $\Psi_T(\mathcal{F}(I_{ref}, P))$  directly influences the computation of  $\mathbb{I}_{top-k}$ . Although the gallery images are represented by their image embeddings, the image embedding of the reference image  $\Psi_I(I_{ref})$  do not contribute to the retrieval process.

#### 3.2. Our Approach: Prompt Directional Vector

Rather than simply employing the composed embedding alone,  $\Psi_T(\mathcal{F}(I_{ref}, P))$ , as depicted in Figure 1b, we propose a generalized formulation of composed text embeddings by considering the embedding modification direction,  $\delta_{PDV}$ , which is derived from the difference between the provided prompt,  $P$ , and the reference image,  $I_{ref}$ . For-

mally, we define  $\delta_{PDV}$  as,

$$\Delta_{PDV} = \Psi_T(\mathcal{F}(I_{ref}, P)) - \Psi_T(\mathcal{F}(I_{ref})). \quad (2)$$

We then form the composed text embedding as follows,

$$\Psi_T(\mathcal{F}(I_{ref}, P)) = \Psi_T(\mathcal{F}(I_{ref})) + \alpha_T \Delta_{PDV}, \quad (3)$$

where  $\alpha$  controls the movement along the prompt vector  $\Delta_{PDV}$  and  $\Psi_T(\mathcal{F}(I_{ref}))$  is the original text embedding.

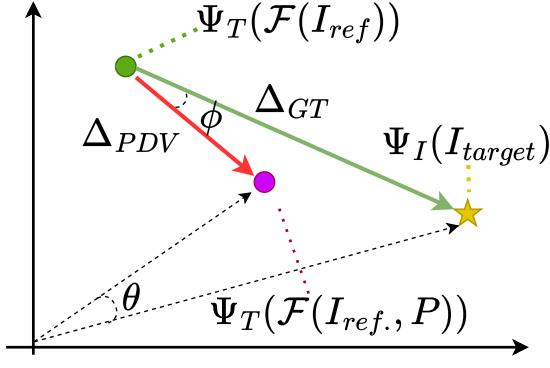
#### 3.3. Strategies for Using PDV

We explore three strategies for using  $\Delta_{PDV}$ :

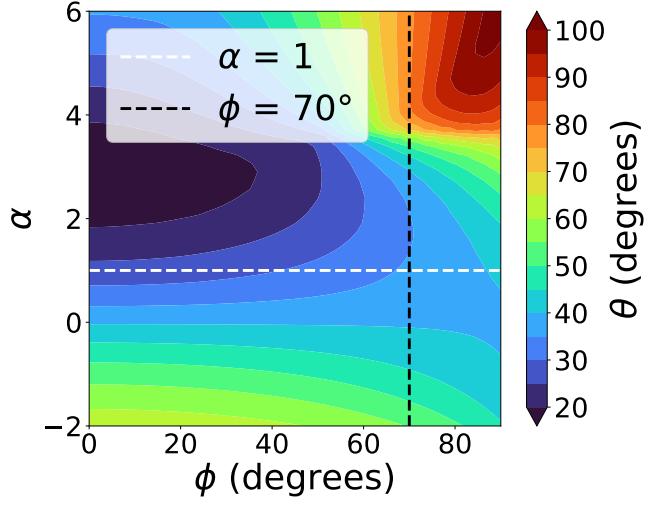
**(1) Prompt Directional Vector for Text (PDV-T),** which enhances controllability in ZS-CIR. While baseline ZS-CIR approaches represent a special case where  $\alpha = 1$ , varying  $\alpha$  provides users with additional control over the retrieval process (refer to Figure 1c). Setting  $\alpha > 1$  amplifies the modification specified by the prompt, while  $\alpha < 1$  reduces its effect. This approach offers a more efficient alternative to modifying the prompt directly, as it requires neither new feature extraction nor prompt reformulation. Note that we use the notation  $\Phi_{PDV-T}$  to represent the composed text embedding.

**When is PDV-T effective?** In CIR, better retrieval performance is directly correlated with a smaller angle,  $\theta$ , between the target embedding vector  $\Psi_I(I_{target})$  and the composed embedding vector  $\Psi_T(\mathcal{F}(I_{ref}, P))$  (refer to Figure 2a). When  $\phi$ , the angle between the calculated prompt directional vector  $\Delta_{PDV}$  and the ground truth prompt directional vector  $\Delta_{GT}$ , is small, adjusting the parameter  $\alpha$  can effectively reduce  $\theta$ . To demonstrate this relationship, we conducted a simulation based on a common real-world scenario where the magnitude of  $\Delta_{PDV}$  is smaller than  $\Delta_{GT}$  due to users writing less descriptive prompts (see Figure 2a). Based on simulation results (Figure 2b), when  $\phi$  is less than 70 degrees, increasing  $\alpha$  up to 3 reduces  $\theta$ . However, when  $\phi$  is between 70 and 90 degrees, decreasing  $\alpha$  is more effective in reducing  $\theta$ . These findings indicate that PDV’s performance is highly dependent on the accuracy of the baseline CIR method. With a strong baseline system, adjusting  $\alpha$  becomes a straightforward and effective way to tune CIR results.

**(2) Prompt Directional Vector for Image (PDV-I),** which extends the modification principle to visual embeddings. While previous approaches primarily relied on composed text embeddings, experimental results show that direct fusion of image and text features yields inferior performance compared to composed features. We hypothesize that this performance gap arises because the direct text embedding,  $\Phi_T(P)$ , differs significantly from the prompt vector  $\Delta_{PDV}$ , as illustrated in Figure 3. This difference occurs because the semantic meaning of natural language is context-sensitive, where in our case the context is provided



(a) Visualisation of  $\theta$  and  $\phi$ .



(b)  $\theta$  vs.  $\alpha$  and  $\phi$

Figure 2. A visualisation of how scaling  $\alpha$  affects the  $\theta$  angle between the composed embedding and the target embedding.

by the reference image embedding  $\Psi_T(\mathcal{F}(I_{ref}))$ . To address this limitation, we propose combining  $\Delta_{PDV}$  with visual embeddings. Specifically, we compute the composed visual embedding  $\Phi_{PDV-I}$  as  $\Psi_I(I_{ref}) + \alpha_I \Delta_{PDV}$ , where  $\Psi_I(I_{ref})$  represents the original visual embedding obtained from the reference image, and the same prompt vector obtained via Eq. 2 is used to modify this visual representation.

**(3) Prompt Directional Vector Fusion (PDV-F)**, which calculates the final similarity score between a query and target image which combines both composed embeddings. This fusion embedding,  $\Phi_{PDV-F}$ , can be defined as,

$$\Phi_{PDV-F} = (1 - \beta)\Phi_{PDV-I} + \beta\Phi_{PDV-T}, \quad (4)$$

where  $\beta$  is a weighting parameter balancing the contribution of the composed visual and textual embeddings.

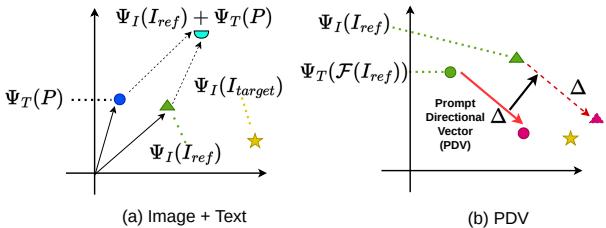


Figure 3. Comparison of Image + Text (a) vs PDV-I (b).

### 3.4. Parameter Tuning and Efficient Searching

PDV introduces three hyperparameters:  $\alpha_T$ ,  $\alpha_I$ , and  $\beta$ . Among these,  $\alpha_T$  is most critical, and adjusting it alone typically suffices. Fine-tuning  $\alpha_I$  and  $\beta$  can further enhance

retrieval performance and controllability. Detailed tuning guidelines are in Section S1.1<sup>1</sup>, and our automatic tuning method for  $\alpha_I$  is described in Section S1.1.1.

PDV mainly reduces feature extraction costs in iterative search. We also find that filtering gallery images by prior ranking results lowers ranking costs, another key factor in CIR retrieval. Section S1.2 provides discussion and supporting experiments.

## 4. Experiments

**Implementation Details.** We utilize the official implementations of four ZS-CIR baseline methods: CIReVL<sup>2</sup> and LDRE<sup>3</sup> as representative caption-based feature extraction approaches and Pic2Word<sup>4</sup> and SEARL<sup>5</sup> as representative pseudo tokenization-based methods. All feature extraction processes follow the original implementations provided by baseline methods. However, to calculate  $\Delta_{PDV}$ , we need text embeddings without prompts, which are not provided in the original implementations. For CIReVL and LDRE, we obtain these embeddings by passing the generated image captions directly to CLIP. For Pic2Word and SEARL, we construct the base text embedding by passing the phrase “a photo of ⟨token⟩” to CLIP, where ⟨token⟩ represents the extracted image token obtained via text inversion.

**Datasets and Base Vision-Language Models.** Following previous work, we evaluated our method on a suite of datasets including Fashion-IQ [28], CIRR [18] and CIRCO

<sup>1</sup>S represent supplementary material

<sup>2</sup>[https://github.com/ExplainableML/Vision\\_by\\_Language](https://github.com/ExplainableML/Vision_by_Language)

<sup>3</sup><https://github.com/yzy-bupt/LDRE>

<sup>4</sup>[https://github.com/google-research/composed\\_image\\_retrieval](https://github.com/google-research/composed_image_retrieval)

<sup>5</sup><https://github.com/miccunifn/SEARL>

[3]. Our proposed method is a plug-and-play approach requiring no additional training, leveraging only pre-trained models. For feature extraction, we use three CLIP variants: ViT-B/32, ViT-L/14, and ViT-G/14, and used the same pre-trained weights as used by the baseline methods. For image tokenization, we employ the pre-trained Pic2Word model.

#### 4.1. Effect of Using the PDV

We now explore the impact of the three proposed uses of the PDV: Using the PDV to augment text queries (PDV-T, see Sec. 4.1), using the PDV to augment image queries (PDV-I, see Sec. 4.1), and using the PDV in queries that fuse image and text data (PDV-F, see Sec. 4.1).

**Analysing the PDV for Text (PDV-T)** To investigate how scaling the prompt vector,  $\Delta_{PDV}$ , affects retrieval performance with composed text embeddings, we conducted experiments using two zero-shot approaches (CIRVL and Pic2Word) with different backbone networks across three datasets. We evaluated the performance by varying the scaling parameter,  $\alpha$  (Eq. 3), from -0.5 to 3 by an interval of 0.1.

The results are presented in Figure 4a. To account for scale variations across different experiments, we report relative recall values, where a baseline of zero is established at  $\alpha = 1$ . As shown in Figure 4a, varying  $\alpha$  leads to significant changes in relative recall performance<sup>6</sup>. Our analysis reveals method-specific patterns across datasets. With CIRVL, increasing  $\alpha$  improves relative recall on both FashionIQ and CIRCO datasets. In contrast, Pic2Word shows no significant improvement on FashionIQ and CIRR when varying  $\alpha$ , while CIRCO’s performance improves when  $\alpha$  is reduced to 0.8-1.0. This divergent behavior is fundamentally linked to each method’s ability to generate an accurate  $\Delta_{PDV}$ . As demonstrated in Tables 1 and 2, CIRVL consistently outperforms Pic2Word across various benchmarks, indicating its superior ability to generate a more accurate composed query, and thus a more accurate  $\Delta_{PDV}$ . Consequently, increasing  $\alpha$  yields greater benefits for CIRVL compared to Pic2Word.

We visualize the top-5 retrieval results using CIRVL with a ViT-B-32 backbone across three datasets (one reference image from each) under varying  $\alpha$  values, as shown in Figure 5a. As  $\alpha$  increases, the retrieved results show stronger alignment with the prompt. Conversely, when  $\alpha$  exceeds 1, the results include semantically related but unseen variations, while  $\alpha$  values below 0.5 yields results opposite to the prompt’s intent. For instance, “brighter blue and sleeveless” retrieves “dark blue with sleeves,” “plain background” yields “natural/dark background,” and “young boy” returns “adult” images.

**Analysing the PDV for Image (PDV-I)** To evaluate whether  $\Delta_{PDV}$  enhances the retrieval performance of image embeddings, we conducted experiments following the

protocol described in Section 4.1. We modified image embeddings by adding  $\Delta_{PDV}$  scaled with  $\alpha$  values ranging from -0.5 to 2.0, where  $\alpha = 0$  represents the original image-only embeddings. As shown in Figure 4b, Recall@K exhibits a positive correlation with  $\alpha$  for values below 1. This upward trend continues until  $\alpha = 2.0$  for CIRVL, while Pic2Word’s performance peaks when  $\alpha$  reaches 1.4. The performance of PDV-I was evaluated on the FashionIQ, CIRR and CIRCO datasets by comparing it with other visual embedding-based methods, as detailed in Tables S5, S6 in the supplementary material. The results reveal that PDV-I achieved significant improvements over existing approaches.

Following the methodology in Section 4.1, we conduct similar visualizations, with results shown in Figure 5b. As with PDV-T, increasing  $\alpha$  leads to stronger alignment between retrieved results and the prompt. When  $\alpha$  exceeds 0.5, the results exhibit semantic relationships to the query, while  $\alpha$  values below 0.5 yield results opposing the prompt’s intent. Notably, PDV-I’s top retrievals demonstrate higher visual similarity to reference images compared to PDV-F, as evidenced by the preserved design elements in the clothing item (left) and laptop (middle). This characteristic is particularly valuable for applications include fashion search [28] and logo retrieval [25], where visual similarity plays a crucial role.

**Analysing PDV Fusion (PDV-F)** Finally, we evaluate the effectiveness of fusing image and text-composed embeddings by varying the fusion parameter,  $\beta$ , from 0 to 1 while maintaining  $\alpha = 1$  for both PDV-I and PDV-F. At  $\beta = 0$ , the model relies solely on composed image embeddings, while at  $\beta = 1$ , it uses only composed text embeddings. As shown in Figure 4c, the fusion of both embeddings consistently outperforms using either embedding type alone. Optimal retrieval performance is typically achieved when  $\beta$  is between 0.4 and 0.8.

We similarly visualize the top-5 retrieved results across different  $\beta$  values. As shown in Figure 5c, when  $\beta$  is small, the retrieved results maintain high visual similarity to the reference image. Conversely, as  $\beta$  exceeds 0.5, the results demonstrate stronger semantic alignment with the prompt.

#### 4.2. ZS-CIR Benchmark Comparison

We evaluated PDV-F alongside four baseline approaches (CIRVL, LDRE, Pic2Word, and SEARLE) across three benchmarks. Notably, CIRVL was tested with three different backbones on three datasets, as its models and intermediate results are publicly available. However, for the remaining methods, we conducted evaluations using the models which are publicly available.

Numerical results are presented in Tables 1 and 2. The hyperparameters shown in these tables were selected based on the valid ranges identified in Figure 4. Since these pa-

<sup>6</sup>See supplementary material for Recall@10 and Recall@50 figures

Fashion-IQ				Shirt		Dress		Top tee		Average		
Backbone	Method	$\beta$	$\alpha_I$	$\alpha_T$	R@10	R@50	R@10	R@50	R@10	R@50	R@10	R@50
ViT-B/32	SEARLE	-	-	-	24.14	41.81	18.39	38.08	25.91	47.02	22.81	42.30
	SEARLE + <b>PDV-F</b>	0.9	1.1	0.9	<b>24.83</b>	41.71	<b>20.13</b>	<b>41.40</b>	<b>25.96</b>	<b>47.17</b>	<b>23.64</b>	<b>43.43</b>
	CIReVL †	-	-	-	28.36	47.84	25.29	46.36	31.21	53.85	28.29	49.35
	CIReVL + <b>PDV-F</b>	0.75	1.4	1.4	<b>32.88</b>	<b>52.80</b>	<b>32.67</b>	<b>54.49</b>	<b>38.91</b>	<b>61.81</b>	<b>34.82</b>	<b>56.37</b>
	LDRE †	-	-	-	27.38	46.27	19.97	41.84	27.07	48.78	24.81	45.63
	SEIZE †	-	-	-	<u>29.38</u>	<u>47.97</u>	<u>25.37</u>	<u>46.84</u>	<u>32.07</u>	<u>54.78</u>	<u>28.94</u>	<u>49.86</u>
ViT-L/14	Pic2Word	-	-	-	25.96	43.52	19.63	40.90	27.28	47.83	24.29	44.08
	Pic2Word + <b>PV-F</b>	0.8	1.0	1.0	<b>28.21</b>	<b>44.55</b>	<b>20.92</b>	<b>42.24</b>	<b>29.02</b>	<b>48.90</b>	<b>26.05</b>	<b>45.23</b>
	SEARLE	-	-	-	26.84	45.19	20.08	42.19	28.40	49.62	25.11	45.67
	SEARLE + <b>PDV-F</b>	0.8	1.2	1.0	<b>28.66</b>	<b>46.76</b>	<b>23.60</b>	<b>46.41</b>	<b>31.00</b>	<b>52.32</b>	<b>27.75</b>	<b>48.50</b>
	CIReVL †	-	-	-	29.49	47.40	24.79	44.76	31.36	53.65	28.55	48.57
	CIReVL + <b>PDV-F</b>	0.55	1	1.3	<b>37.78</b>	<b>54.22</b>	<b>33.61</b>	<b>56.07</b>	<b>41.61</b>	<b>62.16</b>	<b>37.67</b>	<b>57.48</b>
ViT-G/14	LinCIR	-	-	-	29.10	46.81	20.92	42.44	28.81	50.18	26.82	46.49
	SEIZE	-	-	-	<u>33.04</u>	<u>53.22</u>	<u>30.93</u>	<u>50.76</u>	<u>35.57</u>	<u>58.64</u>	<u>33.18</u>	<u>54.21</u>
	Pic2Word	-	-	-	33.17	50.39	25.43	47.65	35.24	57.62	31.28	51.89
	SEARLE	-	-	-	36.46	55.35	28.16	50.32	39.83	61.45	34.81	55.71
	CIReVL †	-	-	-	33.71	51.42	27.07	49.53	35.80	56.14	32.19	52.36
	CIReVL + <b>PV-F</b>	0.6	1.4	1.4	<b>41.90</b>	<b>58.19</b>	<b>40.70</b>	<b>62.82</b>	<b>48.09</b>	<b>67.77</b>	<b>43.56</b>	<b>62.93</b>

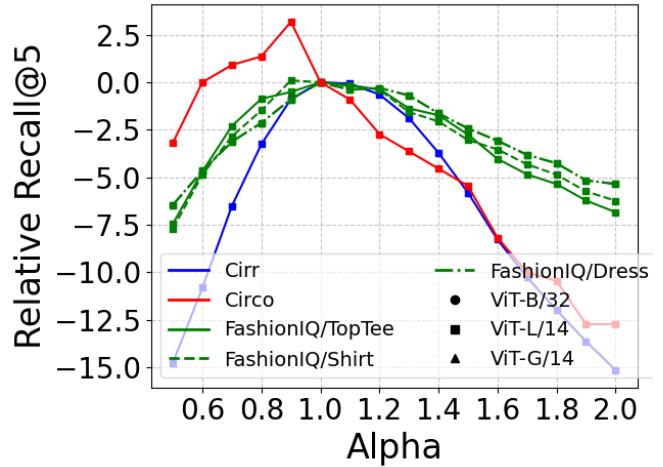
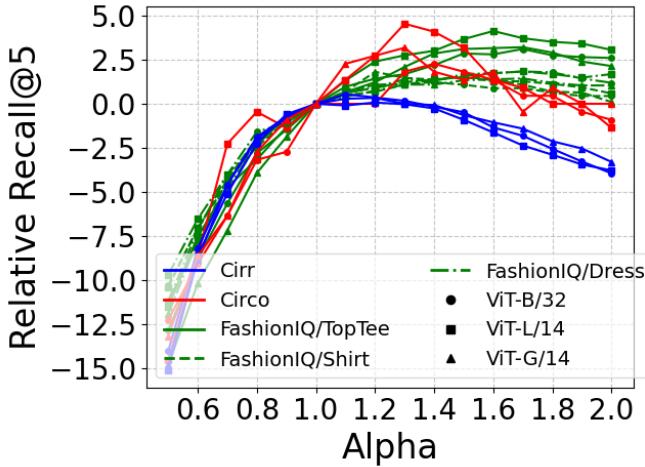
Table 1. Average recall for different methods on Fashion-IQ validation dataset. Yellow highlighting shows PDV improvements over baseline. Orange = PDV improvements, **bold** = best, underline = second. †Numbers from original paper.

Dataset			CIRCO				CIRR								
Metric			mAP@k				Recall@k				$R_s @ k$				
Arch	Method	$\beta$	$\alpha_I$	$\alpha_T$	k=5	k=10	k=25	k=50	k=1	k=5	k=10	k=50	k=1	k=2	k=3
ViT-B/32	PALAVRA [8] †	-	-	-	4.61	5.32	6.33	6.80	16.62	43.49	58.51	83.95	41.61	65.30	80.94
	SEARLE †	-	-	-	9.35	9.94	11.13	11.84	24.00	53.42	66.82	89.78	54.89	76.60	88.19
	SEARLE + <b>PDV-F</b>	0.9	1.4	1.2	<b>9.99</b>	<b>10.50</b>	<b>11.70</b>	<b>12.40</b>	<b>24.53</b>	<b>53.71</b>	<b>67.33</b>	<b>89.81</b>	<b>56.94</b>	<b>78.05</b>	<b>88.99</b>
	CIReVL †	-	-	-	14.94	15.42	17.00	17.82	23.94	52.51	66.00	86.95	60.17	80.05	90.19
	CIReVL + <b>PDV-F</b>	0.75	1.4	1.2	<b>19.90</b>	<b>20.61</b>	<b>22.64</b>	<b>23.52</b>	<b>33.25</b>	<b>64.15</b>	<b>75.23</b>	<b>92.43</b>	<b>65.81</b>	<b>83.76</b>	<b>92.10</b>
	LDRE	-	-	-	17.81	18.04	19.73	20.67	25.69	55.52	68.77	89.86	60.10	80.58	91.04
ViT-L/14	LDRE + <b>PDV-F</b>	0.75	1.4	1.4	17.80	<b>18.78</b>	20.61	21.56	<b>29.30</b>	<b>60.39</b>	<b>72.51</b>	<b>91.42</b>	<b>63.06</b>	<b>82.36</b>	<b>91.54</b>
	SEIZE	-	-	-	19.04	19.64	21.55	22.49	27.47	57.42	70.17	-	<b>65.59</b>	<b>84.48</b>	<b>92.77</b>
	Pic2Word	-	-	-	6.81	7.49	8.51	9.07	23.69	51.32	63.66	86.21	53.61	74.34	87.28
	Pic2Word + <b>PDV-F</b>	0.85	1.2	1.0	<b>7.74</b>	<b>8.67</b>	<b>9.77</b>	<b>10.37</b>	<b>23.90</b>	<b>51.95</b>	<b>64.63</b>	<b>87.04</b>	53.16	74.07	87.08
	SEARLE †	-	-	-	11.68	12.73	14.33	15.12	24.24	52.48	66.29	88.84	53.76	75.01	88.19
	SEARLE + <b>PDV-F</b>	0.85	1.4	1.2	<b>12.58</b>	<b>13.57</b>	<b>15.30</b>	<b>16.07</b>	25.64	53.61	66.58	88.55	<b>55.83</b>	<b>76.48</b>	<b>88.53</b>
ViT-G/14	CIReVL †	-	-	-	18.57	19.01	20.89	21.80	24.55	52.31	64.92	86.34	59.54	79.88	89.69
	CIReVL + <b>PDV-F</b>	0.75	1.4	1.2	<b>25.67</b>	<b>26.61</b>	<b>28.81</b>	<b>29.95</b>	<b>36.24</b>	<b>66.17</b>	<b>76.96</b>	<b>92.29</b>	<b>68.07</b>	<b>85.35</b>	<b>93.47</b>
	LDRE	-	-	-	22.32	23.75	25.97	27.03	26.68	55.45	67.49	88.65	60.39	80.53	90.15
	LDRE + <b>PDV-F</b>	0.75	1.4	1.4	<b>25.23</b>	<b>26.52</b>	<b>28.94</b>	<b>29.95</b>	<b>30.16</b>	<b>59.98</b>	<b>71.90</b>	<b>90.87</b>	<b>63.66</b>	<b>82.87</b>	<b>91.57</b>
	LinCIR	-	-	-	12.59	13.58	15.00	15.85	25.04	53.25	66.68	-	57.11	77.37	88.89
	SEIZE	-	-	-	24.98	25.82	28.24	29.35	28.65	57.16	69.23	-	<b>66.22</b>	<b>84.05</b>	<b>92.34</b>
ViT-G/14	CIReVL †	-	-	-	26.77	27.59	29.96	31.03	34.65	64.29	75.06	91.66	67.95	84.87	93.21
	CIReVL + <b>PDV-F</b>	0.75	1.4	1.2	<b>30.02</b>	<b>31.46</b>	<b>34.01</b>	<b>35.08</b>	<b>38.15</b>	<b>67.93</b>	<b>77.90</b>	<b>92.77</b>	<b>69.37</b>	<b>85.37</b>	<b>93.45</b>
	LDRE	-	-	-	<u>33.30</u>	34.32	37.17	38.27	37.40	66.96	78.17	93.66	68.84	85.64	93.90
	LDRE + <b>PDV-F</b>	0.75	1.4	1.4	<b>34.88</b>	<b>36.41</b>	<b>39.12</b>	<b>40.23</b>	<b>42.51</b>	<b>72.22</b>	<b>81.71</b>	<b>94.94</b>	<b>72.39</b>	<b>88.34</b>	<b>94.80</b>
	SEARLE	-	-	-	13.20	13.85	15.32	16.04	34.80	64.07	75.11	-	68.72	84.70	93.23
	LinCIR	-	-	-	19.71	21.01	23.13	24.18	35.25	64.72	76.05	-	63.35	82.22	91.98
	SEIZE	-	-	-	32.46	33.77	36.46	37.55	38.87	69.42	79.42	-	<b>74.15</b>	<b>89.23</b>	<b>95.71</b>

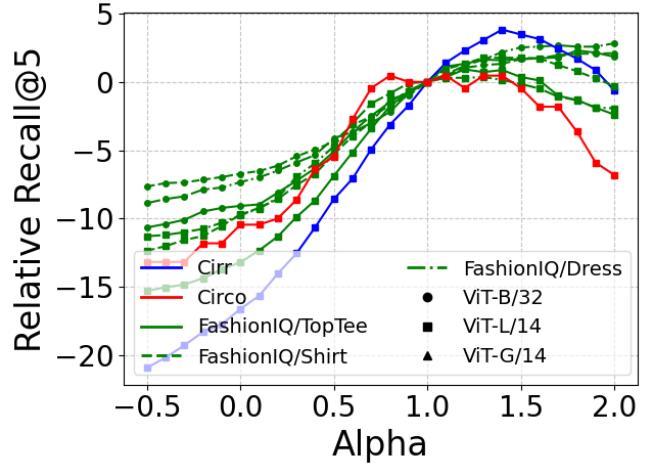
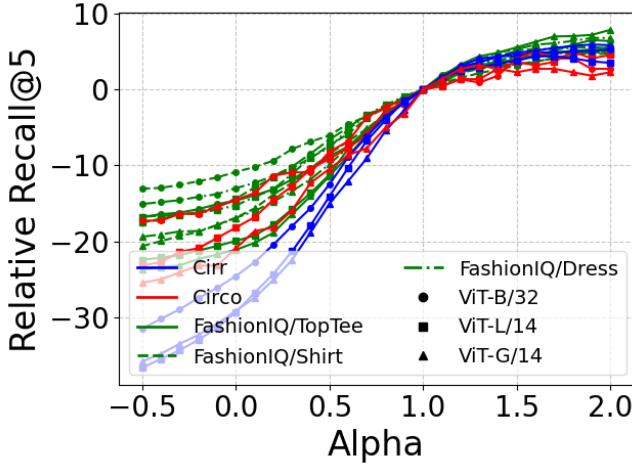
Table 2. Performance comparison on CIRCO and CIRR test datasets. As in previous works, for CIRCO, mAP@k is reported, while for CIRR both Recall@k and  $R_s @ k$  metrics are used. Orange = PDV improvements, **bold** = best, underline = second. †Numbers from original paper.

rameters are intended for user adjustment and users are expected to tune them in response to initial retrieval results, we present the optimal hyperparameters from the three trials. On the FashionIQ benchmark, PDV-F yields substantial

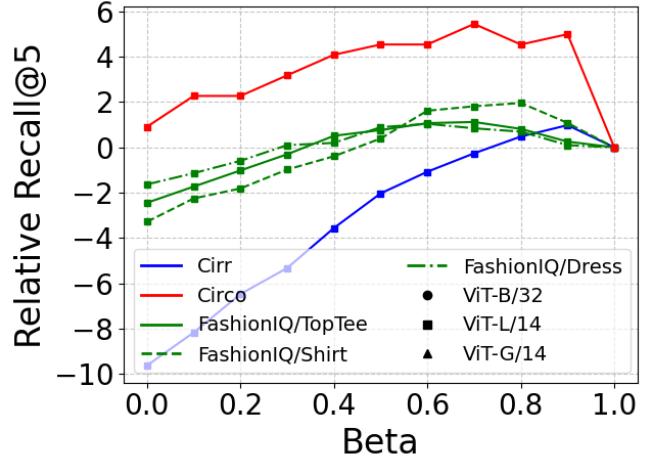
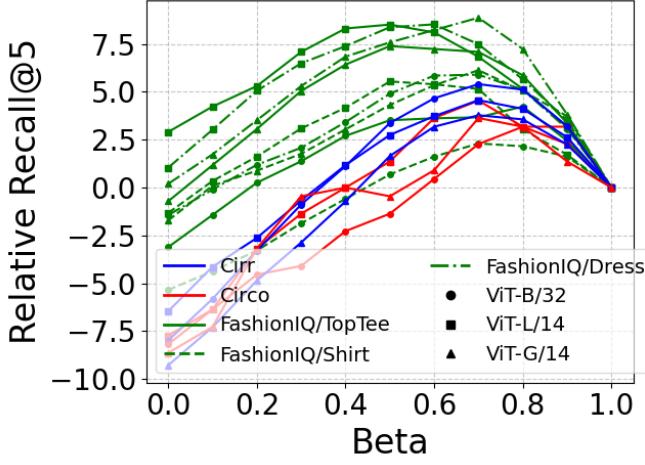
improvements for all baseline approaches, with CIReVL showing particularly strong gains that scale with backbone size. Similarly, all methods demonstrate significant performance improvements on CIRCO and CIRR datasets. No-



(a) PDV-T: Impact of  $\alpha$  scaling on composed text embeddings



(b) PDV-I: Impact of  $\alpha$  scaling on composed image embeddings



(c) PDV-F: Impact of varying  $\beta$  with on composed fused embeddings

Figure 4. Impact of changing  $\alpha/\beta$  on Recall@5 performance across different PDV applications. For each row, results are shown for the CIReVL (left) and Pic2Word (right) baseline methods.

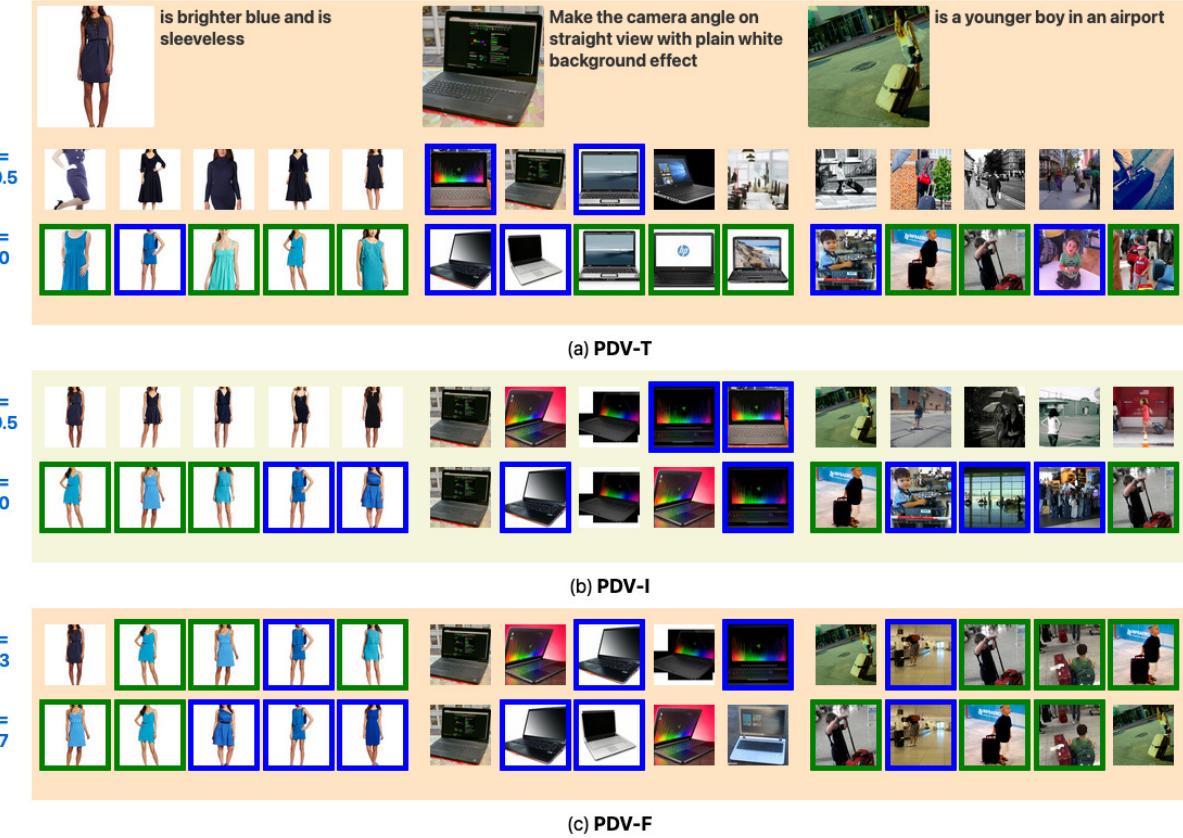


Figure 5. Visualisation of the impact of  $\alpha/\beta$  scaling on top-5 retrieval results. CIReVL with ViT-B-32 Clip model is the baseline method used. Representative examples with prompts from three datasets: FashionIQ (left), CIRR (middle), and CIRCO (right) are shown at the top. **Green** and **blue** bounding boxes indicate true positives and near-true positives, respectively.

table, CIReVL achieves larger improvements compared to other methods, with the most substantial gains observed when using small and medium backbone architectures. Our PDV-F implementation within the CIReVL framework consistently outperformed other state-of-the-art methods, including LinCIR and SEIZE, across most evaluation metrics. Similar to SEIZE, PDV-F offers the advantage of being entirely training-free; however, unlike SEIZE, it does not significantly increase feature extraction computational costs. While LinCIR demonstrates exceptional inference speed, it lacks the training-free nature of our approach, requiring dedicated model training before deployment.

## 5. Conclusion & Future Applications

We introduce the Prompt Directional Vector (PDV), a simple yet effective approach for enhancing Zero-Shot Composed Image Retrieval. PDV captures semantic modifications induced by user prompts without requiring additional training or expensive data collection. Through extensive experiments across multiple benchmarks, we demonstrated three successful applications of PDV: dynamic text

embedding synthesis, composed image embedding through semantic transfer, and effective multi-modal fusion.

Our approach not only improves retrieval performance consistently, but also provides enhanced controllability through the use of scaling factors. PDV serves as a plug-and-play enhancement that can be readily integrated with existing ZS-CIR methods while incurring minimal computational overhead.

We note that PDV’s effectiveness correlates strongly with the underlying method’s ability to generate accurate compositional embeddings. This insight suggests promising future research directions, including developing more robust compositional embedding techniques and exploring adaptive scaling strategies for PDV. The simplicity and effectiveness of PDV also open possibilities for its application in multi-prompt composed image retrieval (*i.e.* dialogue-based search) and other multi-modal tasks where semantic modifications play a crucial role.

## Supplement A. Additional Experiments and Results

### A.1. Regarding Hyper-parameter Tuning

In this work, we introduce three parameters:  $\alpha_I$ ,  $\alpha_T$ , and  $\beta$ . Once their semantic roles are understood, manually adjusting them becomes intuitive and straightforward:

- $\alpha_T \rightarrow$  Influence of the text prompt on the semantic content of the reference image
- $\alpha_I \rightarrow$  Influence of the text prompt on the visual content of the reference image
- $\beta \rightarrow$  Controls trade-off between visual and semantic content

We also examined the potential for automatic tuning. However, due to several uncertain factors—such as prompt quality, baseline performance on the target dataset, and user expectations—it remains highly challenging to optimise all three parameters automatically. Nevertheless, we find that  $\alpha_I$  can be tuned automatically to reduce the gap between PDV-I and PDV-T. In the following subsections, we discuss the manual adjustment of each parameter and the automatic tuning of  $\alpha_I$ .

#### A.1.1 Tuning $\alpha_T$ : Influence of Text Prompt

Among these parameters, the most important is  $\alpha_T$ , which primarily controls the influence of the text prompt. If a user observes that the semantic changes in the top retrieved results are insufficient,  $\alpha_T$  should be increased; conversely, if the changes are too strong, it should be decreased. For example, in the top case of Figure S6, when  $\alpha_T = 1$ , the skirt does not yet display clear white stripes. Increasing  $\alpha_T$  produces results with more distinct white stripes. In contrast, in the middle and bottom examples of Figure S6, the retrieved results with  $\alpha_T = 1$  are already valid. Further increasing  $\alpha_T$  in these cases makes the semantic changes too strong, leading the method to return invalid results.

#### A.1.2 Manual Tuning $\alpha_I$ : Influence of Text Prompt

The role of  $\alpha_I$  is similar to that of  $\alpha_T$ . It also controls the strength of the prompt, but in this case, the composition is with the original visual embedding  $\Psi_I(I_{ref})$ . When  $\alpha_I = 0$ , PDV-I reduces to content-based image retrieval. From the ablation results shown in Figure S13, we observe that setting  $\alpha_I = 1$  is generally safe, as most methods achieve consistent improvements when  $\alpha_I$  is increased from  $-0.5$ . Nevertheless, further increases in  $\alpha_I$  can also be beneficial. Users should continue increasing  $\alpha_I$  when the top retrieved results are overly similar to the reference image and fail to incorporate the semantic concepts specified in the prompt. For instance, in the top and bottom examples of Figure S7,

when  $\alpha_I > 1$ , the top-1 retrieval results successfully present the semantic elements described in the user prompt.

#### A.1.3 Tuning $\beta$ : Fusion Factor

The parameter  $\beta$  is used in PDV-F, which fuses PDV-I and PDV-T. Its value ranges from 0 to 1. When  $\beta = 1$ , PDV-F is equivalent to PDV-T, and when  $\beta = 0$ , it reduces to PDV-I. From the ablation results shown in Figure S14, we observe that most methods achieve improved performance when  $\beta$  lies between 0.6 and 0.9. Beyond performance optimization,  $\beta$  also plays a crucial role in balancing retrieval characteristics. As illustrated in Figure S8, lower  $\beta$  values ( $\beta < 0.5$ ) emphasize visual similarity, producing top results that closely resemble the reference image  $I_{ref}$  in terms of appearance. In contrast, higher  $\beta$  values ( $\beta > 0.5$ ) prioritize semantic alignment, incorporating conceptual elements described in the text prompt. This provides fine-grained control over whether the retrieval system favors visual fidelity or semantic relevance.

#### A.1.4 Automatically Tuning $\alpha_I$

Based on the experimental results, we observe that PDV-T consistently outperforms PDV-I. If the features of PDV-I, denoted as  $\Phi_{PDV-I}$ , are more closely aligned with those of PDV-T,  $\Phi_{PDV-T}$ , the performance of PDV-I can approach that of PDV-T. Motivated by this observation, we tune the parameter  $\alpha_I$  (while keeping  $\Phi_{PDV-T}$  fixed) to minimize the  $\ell_2$  distance between  $\Phi_{PDV-I}$  and  $\Phi_{PDV-T}$ , as expressed in Equation S5. To determine the optimal value of  $\alpha_I$ , we employ the Nelder–Mead optimization method [19], which is a derivative-free and straightforward approach, making it particularly convenient to implement.

$$\alpha_I = \arg \min_{\alpha} \mathcal{L}(\Phi_{PDV-T}, \Phi_{PDV-I}(\alpha)), \quad (S5)$$

To evaluate the effectiveness of the proposed method, we fix  $\alpha_T = 1$ , making PDV-T equivalent to the baseline, and compare the performance of tuned  $\alpha_I$  against the fixed setting  $\alpha_I = 1$ . We conduct experiments on the FashionIQ dataset using three different methods with multiple backbone architectures. As shown in Table S3, our approach successfully determines customized  $\alpha_I$  values for each setting. In all cases, R@50 shows consistent improvements over the baseline, with the largest gain of 23% achieved by CIReVL with the ViT-B/32 backbone on the Toptee subset. R@10 also improves steadily in most scenarios, particularly with the CIReVL method. However, for Pic2Word, R@10 decreases by 3.42% on the Dress subset.

## A.2. Efficient Retrieval with PDV

PDV is designed to enhance the retrieval performance of baseline methods in a subsequent search, triggered when an

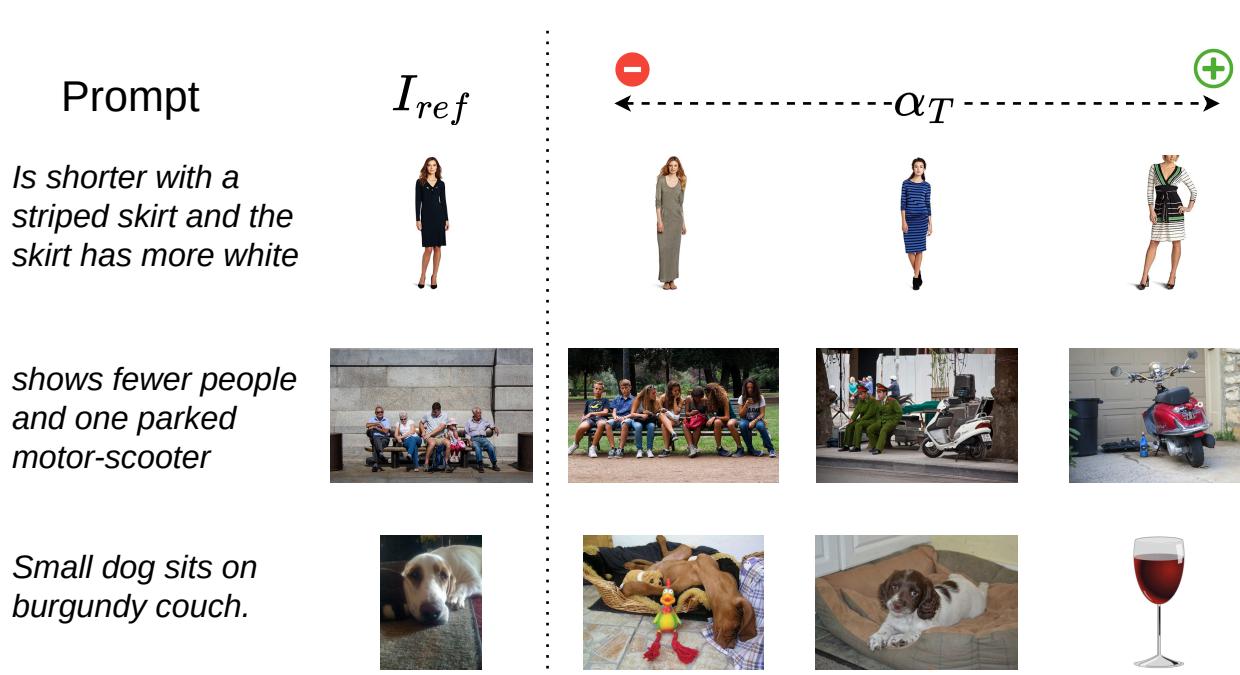


Figure S6. Qualitative results of PDV-T showing the effect of different  $\alpha_T$  values. For each query, we display the top-1 retrieval result for three different  $\alpha_T$  settings. The middle result uses  $\alpha_T = 1$  (baseline), the left result uses a smaller  $\alpha_T$  value, and the right result uses a larger  $\alpha_T$  value. All  $\alpha_T$  values are within the range  $[-0.5, 2]$ .

Backbone	Method	$\alpha_I$	Shirt		Dress		Toptee	
			R@10	R@50	R@10	R@50	R@10	R@50
ViT-B/32	SEARLE	1.57/1.65/1.58	15.68%	15.62%	20.04%	20.51%	14.52%	11.18%
	CIReVL	1.92/2.24/2.02	25.65%	18.82%	24.95%	16.86%	27.81%	23.11%
ViT-L/14	CIReVL	1.90/2.16/1.95	12.96%	9.58%	15.12%	10.50%	17.70%	12.80%
	Pic2Word	1.47/1.46/1.48	0.00%	3.09%	-3.42%	2.07%	0.00%	0.74%
	SEARLE	1.67/1.82/1.73	5.84%	10.62%	16.57%	9.41%	6.15%	9.86%
ViT-G/14	CIReVL	1.50/1.63/1.53	10.43%	6.61%	19.15%	14.30%	17.51%	12.14%

Table S3. Performance differences with automatic  $\alpha_I$  tuning compared to the fixed setting  $\alpha_I = 1$  on the FashionIQ datasets. The  $\alpha_I$  column reports the tuned values for the Shirt, Dress, and Toptee subsets.

initial query fails, without incurring the high computational cost typically associated with iterative search processes.

The computational bottleneck in Zero-Shot Composed Image Retrieval (ZS-CIR) systems stems from two primary operations: feature extraction and similarity ranking.

Regarding feature extraction, the cost is dictated by the model employed. Recent ZS-CIR approaches rely on large vision-language models, whose feature extraction overhead is significant, as detailed in Table S4. In contrast, PDV generates new features for subsequent trials by building upon the embeddings from the initial retrieval. This process involves only efficient scalar multiplications and matrix additions, making its per-trial feature extraction cost nearly negligible. The only substantial computational overhead is

the one-time initial calculation of the reference image embedding,  $\Psi_I(I_{ref})$ .

The cost of similarity ranking, on the other hand, is primarily a function of the feature dimension and the gallery size. While the feature dimension is fixed by the base model, the gallery size can be reduced for subsequent searches. To improve efficiency, we integrate a simple filtering strategy with PDV: items whose distance from the query exceeds a predefined threshold are removed from the gallery for subsequent ranking. While not unique to PDV, we believe this is the first discussion of such an optimization in a ZS-CIR context.

We evaluated this approach on the FashionIQ dataset using two baseline methods, CIReVL and Pic2Word. As

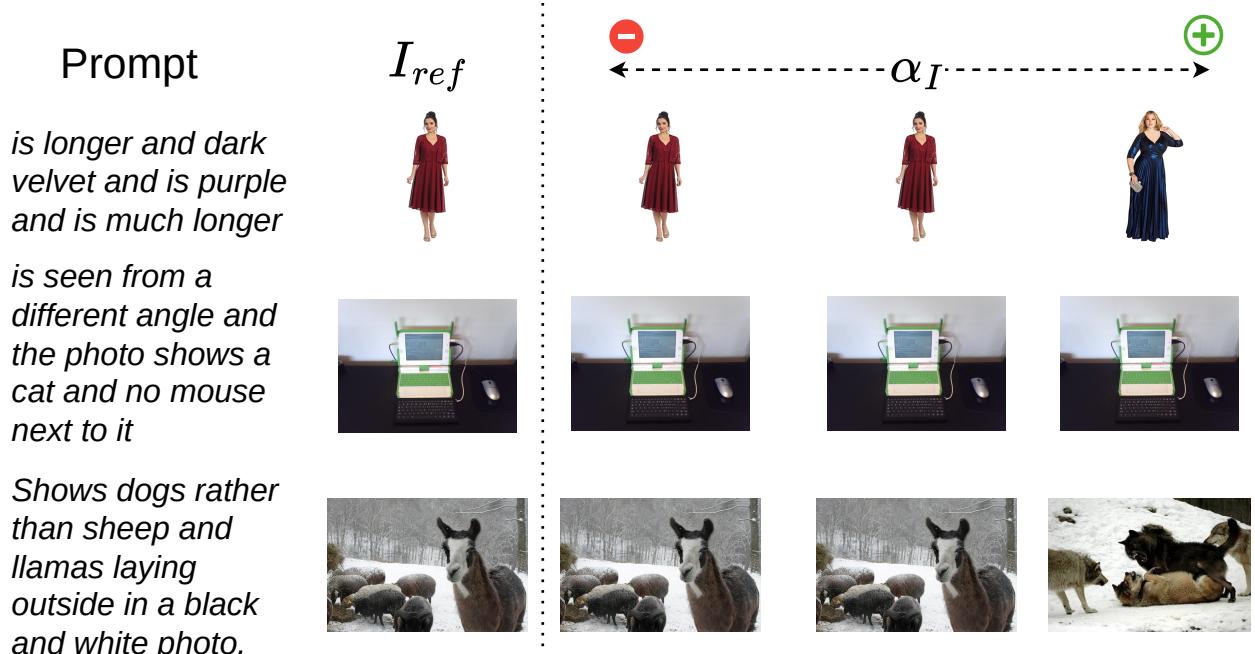


Figure S7. Qualitative results of PDV-I showing the effect of different  $\alpha_I$  values. For each query, we display the top-1 retrieval result for three different  $\alpha_I$  settings. The middle result uses  $\alpha_I = 1$  (baseline), the left result uses a smaller  $\alpha_I$  value, and the right result uses a larger  $\alpha_I$  value. All  $\alpha_I$  values are within the range  $[-0.5, 2]$ .

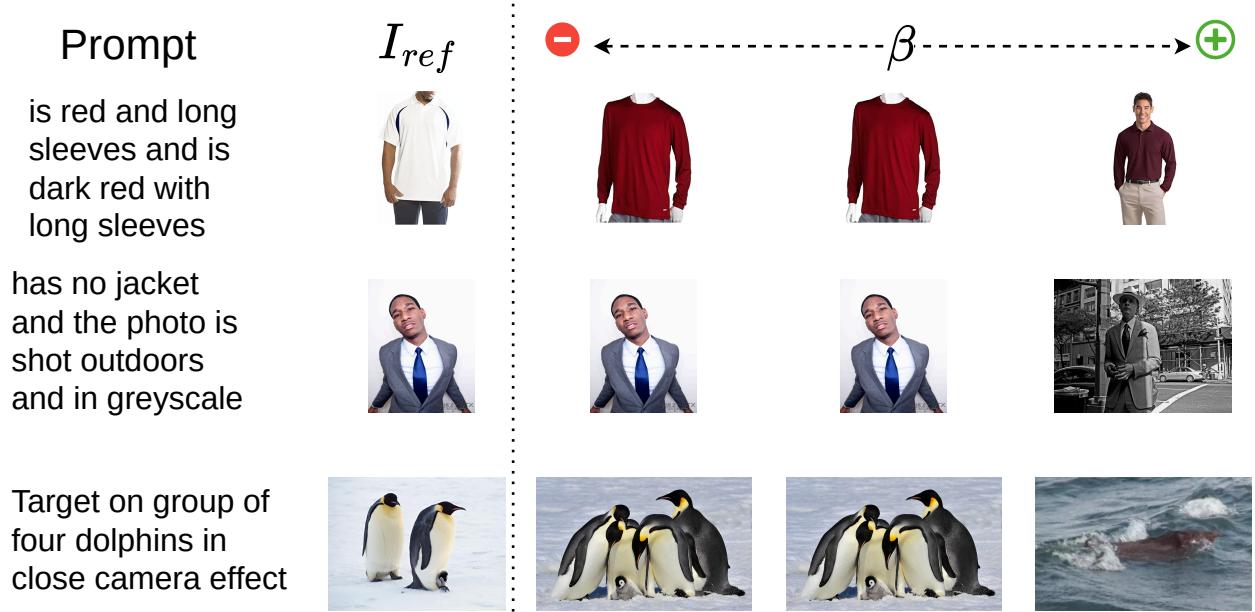


Figure S8. Qualitative results of PDV-F illustrating the effect of different  $\beta$  values. For each query, we show the top-1 retrieval result under three settings:  $\beta = 0$  (left),  $\beta = 0.5$  (middle), and  $\beta = 1$  (right).

Method	Feature Extraction Time (Sec.)	
	Initial	Retrial
Pic2Word	0.02	0.02
+ PDV	0.03	0.00
LinCIR	0.02	0.02
+ PDV	0.03	0.00
KEDs	0.03	0.04
CIReVL (2 Captions)	1.23	1.23
+ PDV	1.24	0.00
LDRE (20 Captions)	17.30	17.30
+ PDV	17.31	0.00

Table S4. Comparison of computation efficiency of the baseline ZS-CIR approaches on NVIDIA A100 GPU.

shown in Table S5, for CIReVL, a threshold of 0.8 filters out over 80% of the gallery items while degrading the R@50 metric by at most 1.31%. For Pic2Word, a threshold of 0.75 filters out over 68% of the gallery with a maximum R@50 decrease of 3.73%. These results demonstrate that PDV, combined with gallery filtering, offers a highly effective trade-off, significantly accelerating retrieval speed while maintaining competitive accuracy.

### A.3. $\phi$ Angles of the Baselines

In Section 3.3 of the main paper, we discussed that PDV’s performance is highly correlated with baseline performance and provided theoretical justification through simulation results as shown in Figure 2. We have introduced a new parameter  $\phi$ , which is the angle between the calculated prompt directional vector  $\Delta_{\text{PDV}}$  and the ground truth prompt directional vector  $\Delta_{\text{GT}}$ . When  $\phi$  is small, adjusting the parameter  $\alpha$  can effectively reduce  $\theta$  which is the angle between the target embedding vector  $\Psi_I(I_{\text{target}})$  and the composed embedding vector  $\Psi_T(\mathcal{F}(I_{\text{ref}}, P))$ .

Here, we present the actual  $\phi$  values for three baseline methods—CIReVL, Pic2Word, and SEARLE—using different backbones across three subdatasets of the FashionIQ dataset. Figure S9 presents the  $\phi$  values for three baseline methods—CIReVL, Pic2Word, and SEARLE—across the FashionIQ subdatasets. While we observed the expected trend of stronger models exhibiting smaller  $\phi$  values, we were surprised to find that the state-of-the-art CIReVL baseline maintains a large  $\phi$  angle of approximately 65°. This finding suggests that PDV has not yet been evaluated with optimal baseline models, despite already yielding considerable gains with CIReVL. We therefore anticipate that PDV will be even more effective when paired with future baselines that achieve a smaller  $\phi$  (ideally  $< 60^\circ$ ), the regime where, according to Figure 2b, our method is most impactful.

### A.4. Additional Quantitative Results

In this supplementary section, we present additional quantitative results that were omitted from the main paper

due to space constraints.

#### A.4.1 Ablation Analysis

While Figure 3 in the main paper illustrates the effects of scaling factor  $\alpha$  and fusion factor  $\beta$  on Recall@5 performance across various PDV applications, Figures S12, S13, and S14 present complementary results for Recall@10 and Recall@50 metrics.

The Recall@10 and Recall@50 results demonstrate consistent trends with the Recall@5 findings presented in the main paper, thus validating our conclusions across multiple evaluation metrics.

#### A.4.2 PDV-I Results

We also provide additional PDV-I results achieved on the validation set of the FashionIQ dataset, as shown in Tables S7 and S8. PDV-I also achieved significant improvements over existing approaches that directly leverage image embeddings for retrieval.

Lastly, we provide a detailed visualization of the impact of  $\alpha/\beta$  scaling on top-5 retrieval results. Figure S11 illustrates the performance of CIReVL with the ViT-B-32 CLIP model across three different datasets.

#### A.4.3 Zero-shot methods with PDV Vs. Supervised Methods

We compared state-of-the-art zero-shot composed image retrieval (ZS-CIR) methods enhanced with PDV against supervised methods on the FashionIQ and CIRR datasets. Our evaluation included early supervised methods such as TIRG [26] and ARTEMIS citedelmas2022artemis, as well as recent state-of-the-art approaches like CCIN [24] and SPRC [2]. The comparative results presented in Table S6 demonstrate that PDV achieves remarkably competitive performance against supervised methods. The PDV-based approaches significantly outperform early supervised baselines, with substantial improvements over TIRG (41.90% vs 14.13% R@10 on FashionIQ Dress) and ARTEMIS [9] (41.90% vs 25.68%). While PDV methods do not quite match the performance of recent state-of-the-art approaches like CCIN and SPRC, the performance gap remains relatively modest—typically within 7-9 percentage points on FashionIQ and approximately 8-9 points on CIRR’s mean score (72.85% vs 81.66% for CCIN). This narrow performance gap is particularly noteworthy given that ZS-CIR methods with PDV operate without human-annotated training data. These results suggest that unsupervised approaches are reaching a level of effectiveness that positions them as viable alternatives to supervised methods.

Backbone	Method	Dataset	Threshold	Filtered Ratio	Change in R@10	Change in R@50
ViT-B/32	CIReVL + PDV-F	Toptee	0.8	90.85%	-2.88%	-1.31%
		Dress	0.8	82.31%	0.00%	-0.73%
		Shirt	0.8	87.96%	-1.22%	-1.31%
ViT-L/14	Pic2Word + PDV-F	Toptee	0.75	85.62%	-1.24%	-0.94%
		Dress	0.75	68.79%	0.00%	0.12%
		Shirt	0.75	88.36%	-1.91%	-3.73%

Table S5. Performance changes of PDV-F after applying filtering to the initial retrieval results on the FashionIQ dataset.

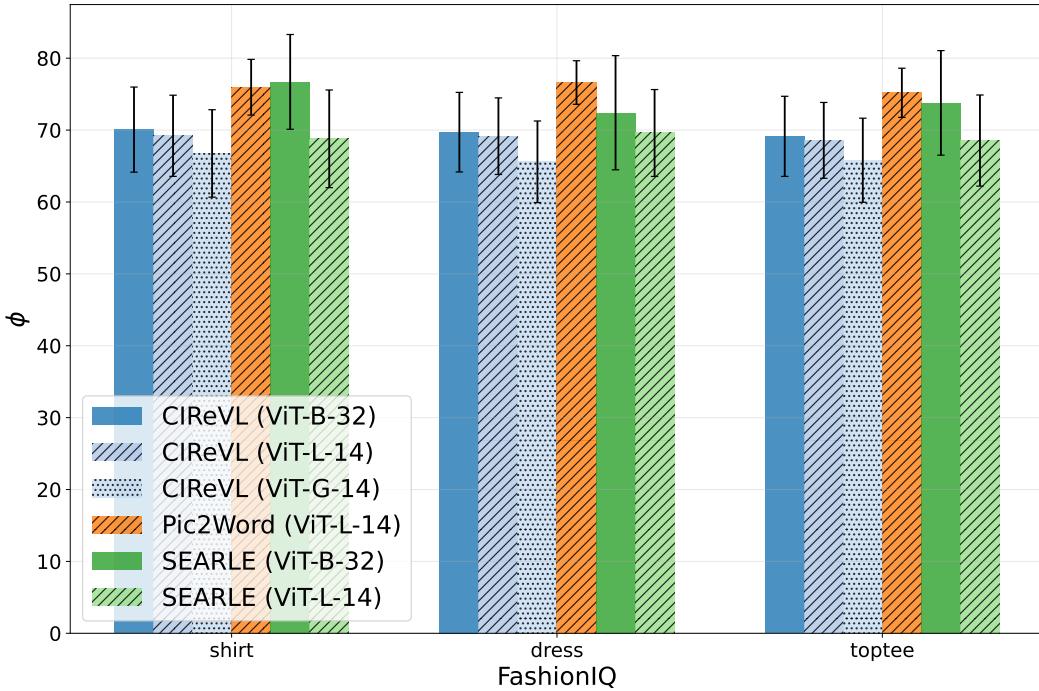


Figure S9. Performance Comparison:  $\phi$  Angles by Method and Model Across FashionIQ Datasets

Method	FashionIQ							CIRR						
	Dress		Shirt		Toptee		Mean	R@1		R@5		R@10	R@50	Mean
	R@10	R@50	R@10	R@50	R@10	R@50		R@1	R@5	R@10	R@50			
TIRG [26]	14.13	34.61	13.10	30.91	14.79	34.37	23.66	14.61	48.37	64.08	90.03	54.27		
ARTEMIS [9]	25.68	51.05	21.57	44.13	25.89	55.06	37.68	16.96	46.10	61.31	87.73	53.03		
CLIP4CIR [5]	33.81	59.40	39.99	60.45	41.41	65.37	50.03	38.53	69.98	81.86	95.93	71.58		
CompoDiff [12]	40.65	57.14	36.87	57.39	43.93	61.17	49.53	22.35	54.36	73.41	91.77	60.47		
TG-CIR [27]	45.22	69.66	52.60	72.52	56.14	77.10	58.05	45.25	78.29	87.16	97.30	77.00		
SPRC [2]	48.83	72.09	53.83	74.14	<b>58.13</b>	<b>78.58</b>	64.27	51.96	82.12	89.74	97.69	80.37		
CCIN [24]	<b>49.38</b>	<b>72.58</b>	<b>55.93</b>	<b>74.14</b>	57.93	77.56	<b>64.59</b>	<b>53.41</b>	<b>84.05</b>	<b>91.17</b>	<b>98.00</b>	<b>81.66</b>		
CIReVL + PDV	41.90	58.19	40.70	62.82	48.09	67.77	53.25	38.15	67.93	77.90	92.77	69.19		
LDRE + PDV	-	-	-	-	-	-	-	42.51	72.22	81.71	94.94	72.85		

Table S6. Comparison of PDV with supervised methods on FashionIQ and CIRR datasets.

## Supplement B. PDV Algorithm and Code

The PDV algorithm is given in Algorithm 1, and the code is shown in Figure S10. The implementation of PDV is very intuitive, and it could be easily integrated with any ZS-CIR approaches.

---

### Algorithm 1 Calculate PDV Features

---

```
1: function CALCULATEPDVFEATURES( $\mathbf{f}_{\text{text}}$ ,  $\mathbf{f}_{\text{text\_composed}}$ ,  $\mathbf{f}_{\text{image}}$ ,  $\alpha_i$ ,  
     $\alpha_t$ ,  $\beta$ )  
2:    $\mathbf{f}_{\text{text}} \leftarrow \text{normalize}(\mathbf{f}_{\text{text}})$   
3:    $\mathbf{f}_{\text{text\_composed}} \leftarrow \text{normalize}(\mathbf{f}_{\text{text\_composed}})$   
4:    $\mathbf{f}_{\text{image}} \leftarrow \text{normalize}(\mathbf{f}_{\text{image}})$   
5:    $\mathbf{pdv} \leftarrow \mathbf{f}_{\text{text\_composed}} - \mathbf{f}_{\text{text}}$   
6:    $\mathbf{f}_{\text{PDVI}} \leftarrow \mathbf{f}_{\text{image}} + \alpha_i \cdot \mathbf{pdv}$   
7:    $\mathbf{f}_{\text{PDVT}} \leftarrow \mathbf{f}_{\text{text}} + \alpha_t \cdot \mathbf{pdv}$   
8:    $\mathbf{f}_{\text{PDVF}} \leftarrow (1 - \beta) \cdot \mathbf{f}_{\text{PDVI}} + \beta \cdot \mathbf{f}_{\text{PDVT}}$   
9:   return  $\text{normalize}(\mathbf{f}_{\text{PDVF}})$   
10: end function
```

---

Fashion-IQ				Shirt		Dress		Toptee		Average	
Backbone	Method	$\alpha_I$	R@10	R@50	R@10	R@50	R@10	R@50	R@10	R@50	
ViT-B/32	Image-only †	-	6.92	14.23	4.46	12.19	6.32	13.77	5.90	13.37	
	Text-only †	-	19.87	34.99	15.42	35.05	20.81	40.49	18.70	36.84	
	Image + Text †	-	13.44	26.25	13.83	30.88	17.08	31.67	14.78	29.60	
	SEARLE + <b>PDV-I</b>	2	18.25	31.84	18.49	39.17	21.32	37.74	19.35	36.25	
	CIReVL + <b>PDV-I</b>	2	28.95	45.88	29.00	49.13	34.22	56.09	30.72	50.37	

Table S7. PDV-I performance on FashionIQ val datasets. † denotes that numbers are taken from the original paper.

Dataset		Metric	CIRCO				CIRR				$R_s @ k$		
			mAP@k				Recall@k						
Arch	Method	$\alpha_I$	k=5	k=10	k=25	k=50	k=1	k=5	k=10	k=50	k=1	k=2	k=3
ViT-B/32	Image-only †	-	1.34	1.60	2.12	2.41	6.89	22.99	33.68	59.23	21.04	41.04	60.31
	Text-only †	-	2.56	2.67	2.98	3.18	21.81	45.22	57.42	81.01	62.24	81.13	90.70
	Image + Text †	-	2.65	3.25	4.14	4.54	11.71	35.06	48.94	77.49	32.77	56.89	74.96
	SEARLE + <b>PDV-I</b>	1.5	4.77	5.23	6.31	6.82	16.65	42.53	55.16	81.42	44.68	67.78	82.94
	CIReVL + <b>PDV-I</b>	2.0	<b>10.29</b>	<b>10.80</b>	<b>12.23</b>	<b>12.93</b>	<b>27.18</b>	<b>56.53</b>	<b>67.76</b>	<b>87.64</b>	<b>59.81</b>	<b>79.59</b>	<b>90.15</b>
	LDRE + <b>PDV-I</b>	2.0	<u>8.00</u>	<u>8.88</u>	<u>10.06</u>	<u>10.72</u>	<u>23.37</u>	<u>51.21</u>	<u>63.69</u>	<u>85.57</u>	<u>55.57</u>	<u>76.63</u>	<u>88.15</u>

Table S8. PDV-I performance on CIRCO and CIRR test datasets. Note that the image-only approach utilizes the visual embedding of the reference image, whereas the text-only approach employs the text embedding of the prompt. **Bold** = best results, underline = second best. †Numbers from original paper.

```

def calculate_pdv_features(feature_text, feature_text_composed, feature_image,
                           alpha_i=1, alpha_t=1, beta=1):
    """
    Calculates enhanced multimodal features using Prompt Difference Vector (PDV) approach.

    Parameters:
    - feature_text: Features extracted from the text branch of the VLM,
                    representing the text-only encoding (e.g., from text inversion or captioning)
    - feature_text_composed: Features of text with compositional prompt, representing
                            text encoding with additional prompt information
    - feature_image: Features extracted from the visual branch of the VLM,
                     representing the visual-only encoding
    - alpha_i: Scaling factor for applying PDV to image features (default=1)
    - alpha_t: Scaling factor for applying PDV to text features (default=1)
    - beta: Weighting factor for combining PDV-enhanced features (default=1)

    Returns:
    - Normalized combined feature vector enhanced with PDV
    """
    # Normalize all input features to unit length
    feature_text = normalize(feature_text, dim=-1)
    feature_text_composed = normalize(feature_text_composed, dim=-1)
    feature_image = normalize(feature_image, dim=-1)

    # Calculate the Prompt Difference Vector (PDV)
    # This captures the semantic difference added by the compositional prompt
    pdv = feature_text_composed - feature_text

    # Apply PDV to image features with scaling factor alpha_i
    # This enhances the image representation with prompt-related information
    feature_PDVI = feature_image + alpha_i * pdv

    # Apply PDV to text features with scaling factor alpha_t
    # This enhances the text representation with additional prompt influence
    feature_PDVT = feature_text + alpha_t * pdv

    # Combine the PDV-enhanced features with weighting factor beta
    # Higher beta values emphasize text features, lower values emphasize image features
    feature_PDVF = (1 - beta) * feature_PDVI + beta * feature_PDVT

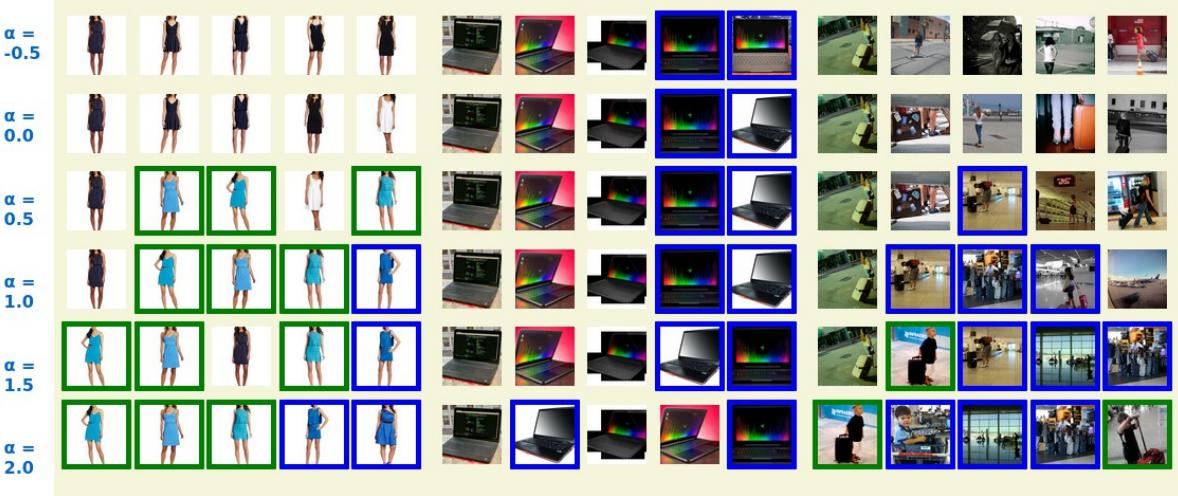
    # Normalize and return the final feature vector
    return normalize(feature_PDVF, dim=-1)

```

Figure S10. Python function for calculating PDV features.



(a) PDV-T



(b) PDV-I



(c) PDV-F

Figure S11. Visualisation of the impact of  $\alpha/\beta$  scaling on top-5 retrieval results. CIReVL with ViT-B-32 Clip model is the baseline method used. Representative examples with prompts from three datasets: FashionIQ (left), CIRR (middle), and CIRCO (right) are shown at the top. **Green** and **blue** bounding boxes indicate true positives and near-true positives, respectively.

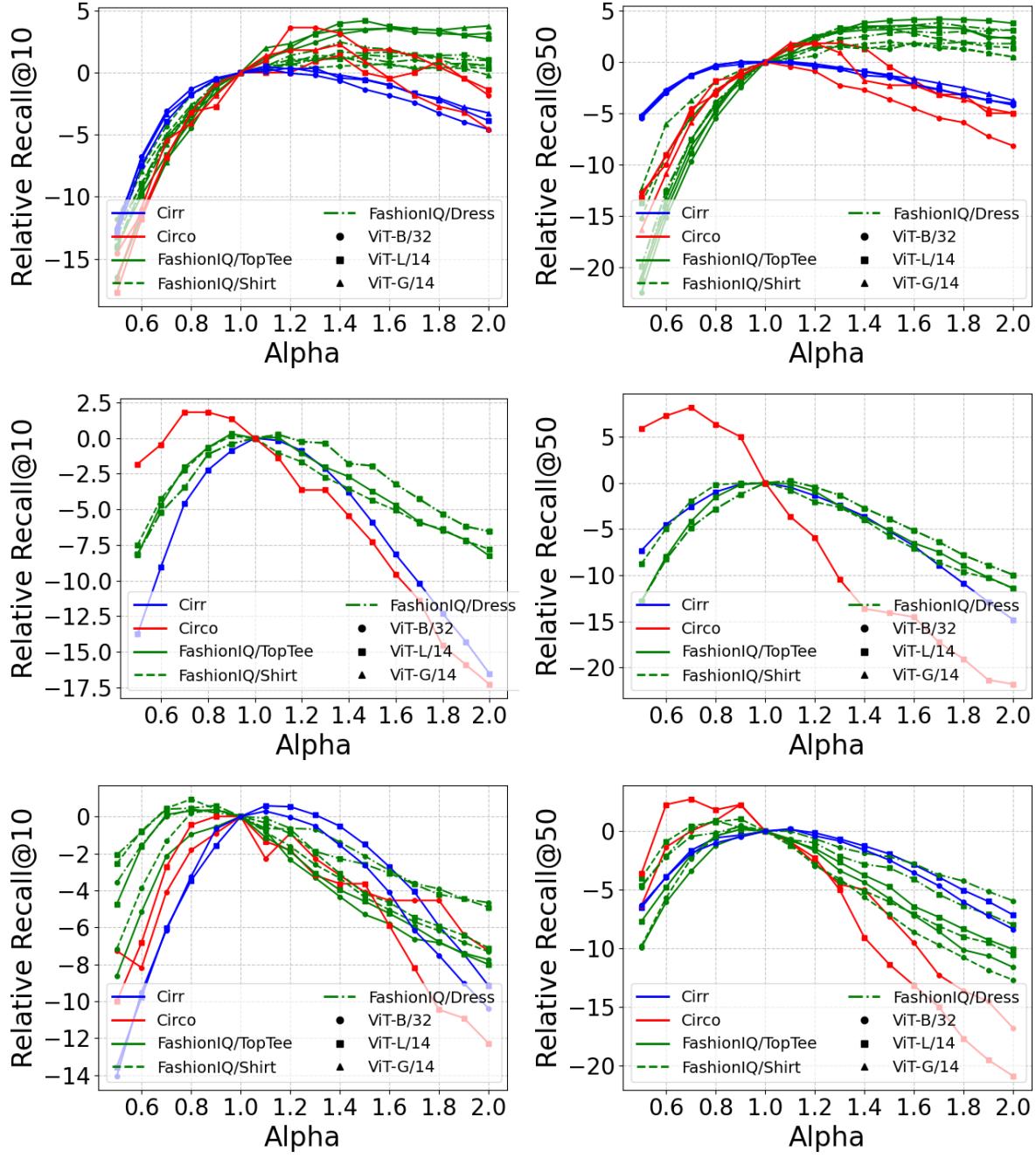


Figure S12. PDV-T: Impact of  $\alpha$  scaling on Recall@10 (left) and Recall@50 (right) performance. Results shown for three baseline methods: CIRr (top), Pic2Word (middle) and SEARLE (bottom).

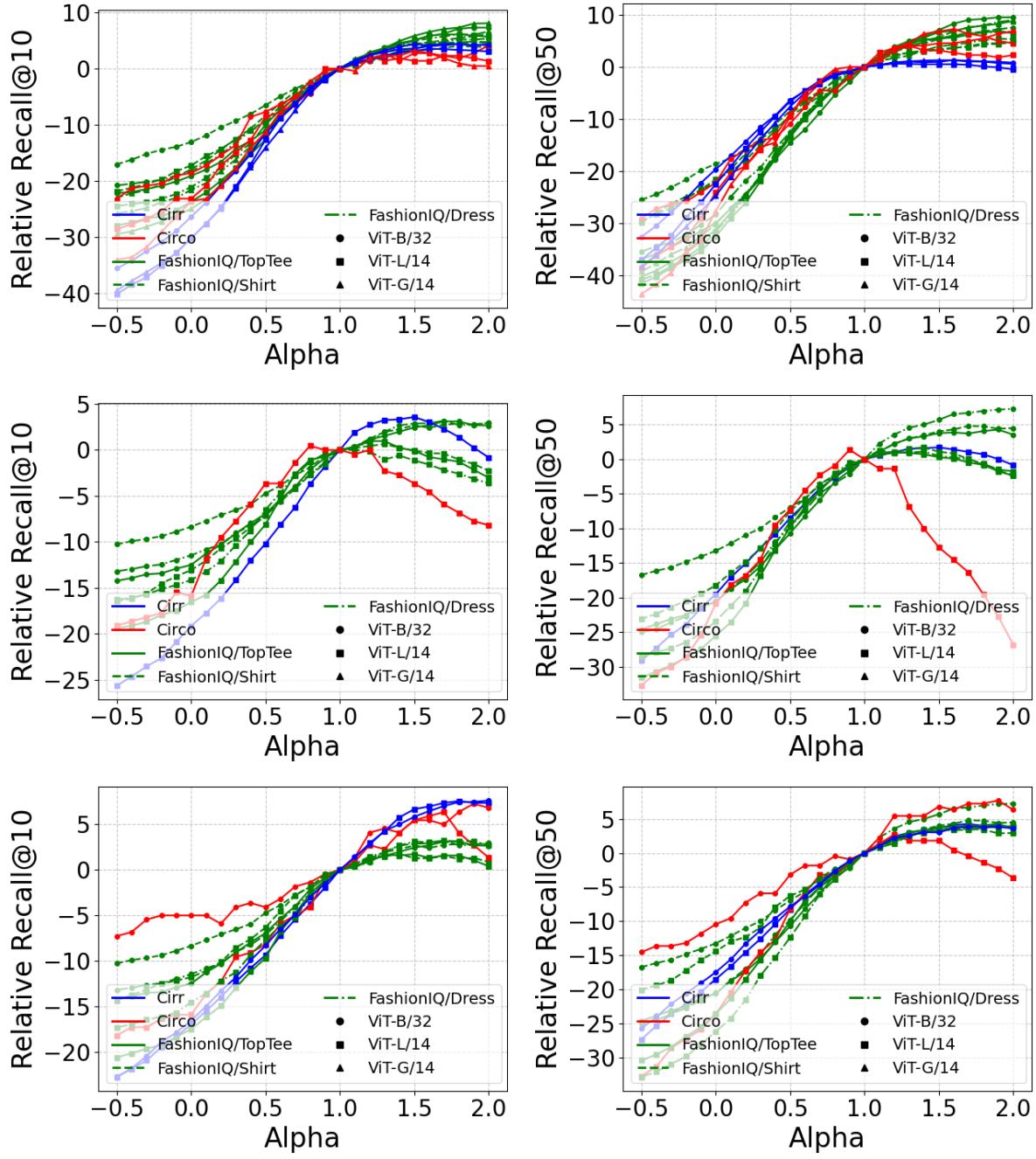


Figure S13. PDV-I: Impact of  $\alpha$  scaling on Recall@10 (left) and Recall@50 (right) performance. Results shown for three baseline methods: CIRrVL (top), Pic2Word (middle) and SEARLE (bottom).

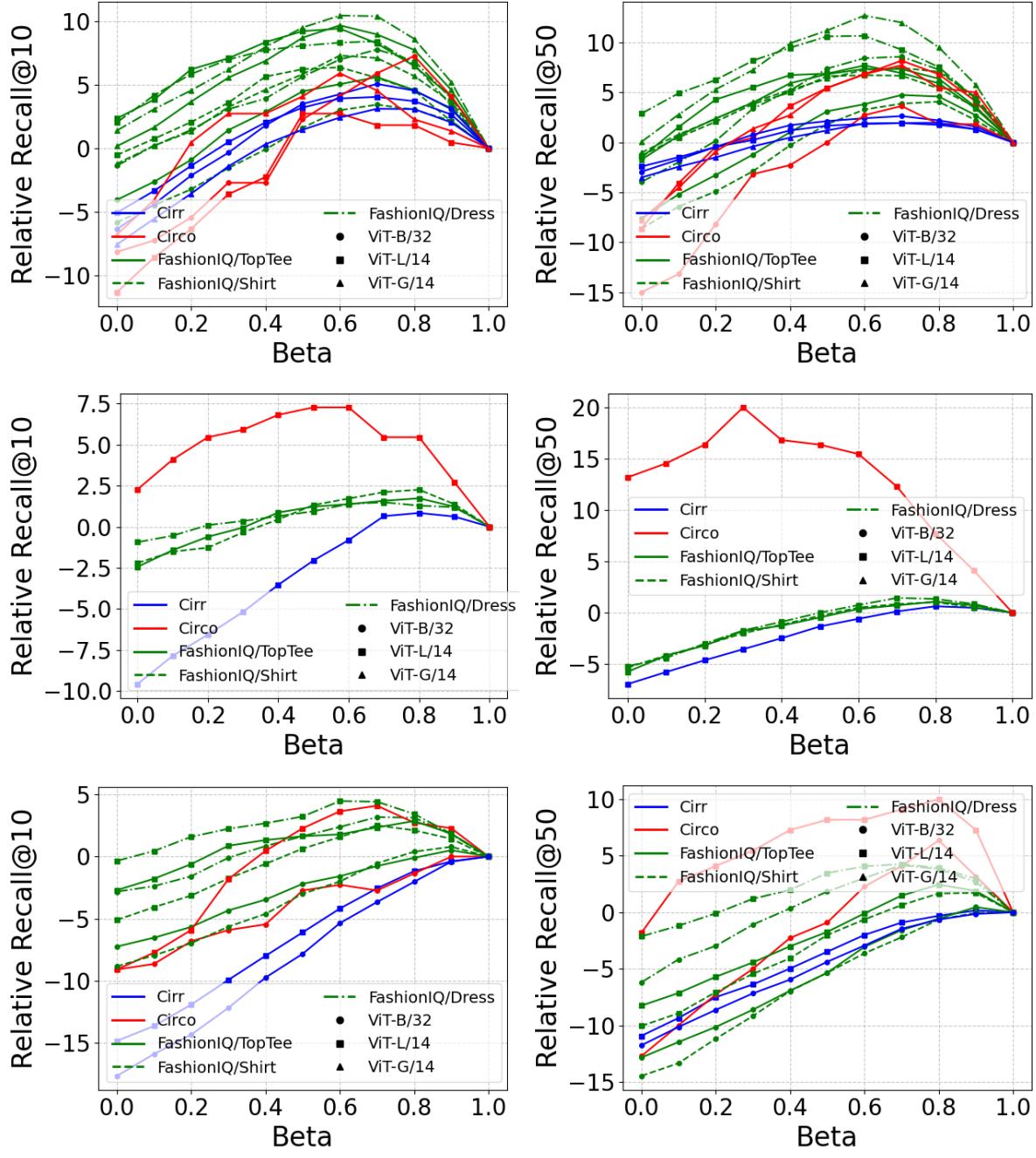


Figure S14. PDV-F: Impact of  $\beta$  scaling on Recall@10 (left) and Recall@50 (right) performance. Results shown for three baseline methods: CIRr (top), Pic2Word (middle) and SEARLE (bottom).

## References

- [1] Muhammad Umer Anwaar, Egor Labintcev, and Martin Kleinstuber. Compositional learning of image-text query for image retrieval. In *Proceedings of the IEEE/CVF Winter conference on Applications of Computer Vision*, pages 1140–1149, 2021. 1
- [2] Yang Bai, Xinxing Xu, Yong Liu, Salman Khan, Fahad Khan, Wangmeng Zuo, Rick Siew Mong Goh, and Chun-Mei Feng. Sentence-level prompts benefit composed image retrieval. *arXiv preprint arXiv:2310.05473*, 2023. 13, 14
- [3] Alberto Baldrati, Lorenzo Agnolucci, Marco Bertini, and Alberto Del Bimbo. Zero-shot composed image retrieval with textual inversion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15338–15347, 2023. 1, 2, 3, 6
- [4] Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. Conditioned and composed image retrieval combining and partially fine-tuning clip-based features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4959–4968, 2022. 3
- [5] Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. Conditioned and composed image retrieval combining and partially fine-tuning clip-based features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 4959–4968, June 2022. 14
- [6] Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. Effective conditioned and composed image retrieval combining clip-based features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21466–21474, 2022. 1, 3
- [7] Yanbei Chen and Loris Bazzani. Learning joint visual semantic matching embeddings for language-guided retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 136–152. Springer, 2020. 1, 3
- [8] Niv Cohen, Rinon Gal, Eli A Meirom, Gal Chechik, and Yuval Atzmon. “this is my unicorn, fluffy”: Personalizing frozen vision-language representations. In *European conference on computer vision*, pages 558–577. Springer, 2022. 1, 7
- [9] Ginger Delmas, Rafael Sampaio de Rezende, Gabriela Csurka, and Diane Larlus. Artemis: Attention-based retrieval with text-explicit matching and implicit similarity. *arXiv preprint arXiv:2203.08101*, 2022. 13, 14
- [10] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022. 3
- [11] Geonmo Gu, Sanghyuk Chun, Wonjae Kim, , Yoohoon Kang, and Sangdoo Yun. Language-only training of zero-shot composed image retrieval. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 3
- [12] Geonmo Gu, Sanghyuk Chun, Wonjae Kim, HeeJae Jun, Yoohoon Kang, and Sangdoo Yun. Compodiff: Versatile composed image retrieval with latent diffusion. *arXiv preprint arXiv:2303.11916*, 2023. 3, 14
- [13] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 3
- [14] Shyamgopal Karthik, Karsten Roth, Massimiliano Mancini, and Zeynep Akata. Vision-by-language for training-free compositional image retrieval. *International Conference on Learning Representations (ICLR)*, 2024. 1, 2, 3
- [15] Jongseok Kim, Youngjae Yu, Hoeseong Kim, and Gunhee Kim. Dual compositional learning in interactive image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021. 1
- [16] Seungmin Lee, Dongwan Kim, and Bohyung Han. Cosmo: Content-style modulation for image retrieval with text feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 802–812, 2021. 1
- [17] Yikun Liu, Jiangchao Yao, Ya Zhang, Yanfeng Wang, and Weidi Xie. Zero-shot composed text-image retrieval. *arXiv preprint arXiv:2306.07272*, 2023. 3
- [18] Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. Image retrieval on real-life images with pre-trained vision-and-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2125–2134, 2021. 1, 5
- [19] John A Nelder and Roger Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965. 10
- [20] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 3, 4
- [21] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 3
- [22] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19305–19314, 2023. 1, 2, 3
- [23] Yucheng Suo, Fan Ma, Linchao Zhu, and Yi Yang. Knowledge-enhanced dual-stream zero-shot composed image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26951–26962, 2024. 3
- [24] Likai Tian, Jian Zhao, Zechao Hu, Zhengwei Yang, Hao Li, Lei Jin, Zheng Wang, and Xuelong Li. Ccin: Compositional conflict identification and neutralization for composed image retrieval. In *Proceedings of the IEEE/CVF Conference*

- on Computer Vision and Pattern Recognition (CVPR)*, pages 3974–3983, June 2025. 13, 14
- [25] Osman Tursun, Simon Denman, Sabesan Sivapalan, Sridha Sridharan, Clinton Fookes, and Sandra Mau. Component-based attention for large-scale trademark retrieval. *IEEE Transactions on Information Forensics and Security*, 17:2350–2363, 2019. 6
- [26] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval—an empirical odyssey. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6439–6448, 2019. 1, 3, 13, 14
- [27] Haokun Wen, Xian Zhang, Xuemeng Song, Yinwei Wei, and Liqiang Nie. Target-guided composed image retrieval. In *Proceedings of the 31st ACM international conference on multimedia*, pages 915–923, 2023. 14
- [28] 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 *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11307–11317, 2021. 1, 3, 5, 6
- [29] 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 *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 1245–1254, 2024. 3
- [30] 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 *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 80–90, 2024. 3
- [31] Youngjae Yu, Seunghwan Lee, Yuncheol Choi, and Gunhee Kim. Curlingnet: Compositional learning between images and text for fashion iq data. *arXiv preprint arXiv:2003.12299*, 2020. 3
- [32] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 3