CompoDiff: Versatile Composed Image Retrieval With Latent Diffusion

Geonmo Gu*,1, Sanghyuk Chun*,2, Wonjae Kim2, HeeJae Jun1, Yoohoon Kang1, Sangdoo Yun2

1NAVER Vision 2NAVER AI Lab  * Equal contribution

Abstract

We propose a novel diffusion-based model, CompoDiff, for solving zero-shot Composed Image Retrieval (ZS-CIR) with latent diffusion. This paper also introduces a new synthetic dataset, named SynthTriplets18M, with 18.8 million reference images, conditions, and corresponding target image triplets. CompoDiff and SynthTriplets18M tackle the short-ages of the previous CIR approaches, such as poor generalizability due to the small dataset scale and the limited types of conditions. CompoDiff not only achieves a new state-of-the-art on four ZS-CIR benchmarks, including FashionIQ, CIRR, CIRCO, and GeneCIS, but also enables a more versatile and controllable CIR by accepting various conditions, such as negative text, and image mask conditions. Code and dataset are available at https://github.com/navervision/CompoDiff

1. Introduction

Imagine a customer seeking a captivating cloth serendipitously found on social media but not the most appealing materials and colors. In this scenario, the customer needs a search engine that can process composed queries, e.g., the reference garment image along with text specifying the preferred material and color. This task has been recently formulated as Composed Image Retrieval (CIR). CIR systems offer the benefits of searching for visually similar items while providing a high degree of freedom to depict text queries as text-to-image retrieval. CIR can also improve the search quality by iteratively taking user feedback.

The existing CIR methods address the problem by combining image and text features using additional fusion models, e.g., \( z_i = \text{fusion}(z_{IR}, z_c) \) where \( z_i, z_{IR}, z_c \) are the target image, conditioning text, and reference image features, respectively. Although the fusion methods have shown great success, they have fundamental limitations. First, the fusion module is not flexible; it cannot handle versatile conditions beyond a limited textual one. For instance, a user might want to include a negative text that is not desired for the search \( (x_{cT}) \) (e.g., an image + “with cherry blossom” – “France”, as in Fig. 1 (b)), indicate where \( (x_{cT}) \) the condition is applied (e.g., an image + “balloon” + indicator, as in Fig. 1 (c)), or construct a complex condition with a mixture of them. Furthermore, once the fusion model is trained, it will always produce the same \( z_i \) for the given \( z_{IR} \) and \( z_c \) to users. However, a practical retrieval system needs to control the strength of conditions by its applications or control the level of serendipity. Second, they need a pre-collected human-verified dataset of triplets \( (x_{IR}, x_c, x_t) \) consisting of a reference image \( (x_{IR}) \), a text condition \( (x_c) \), and the corresponding target image \( (x_t) \). However, obtaining such triplets is costly and sometimes impossible; therefore, the existing CIR datasets are small-scale (e.g., 30K [30] or 36K [16] triplets), resulting in a lack of generalizability to other datasets.

We aim to achieve a generalizable CIR model with diverse and versatile conditions by using latent diffusion. We treat the CIR task as a conditional image editing task on the latent space, i.e., \( z_i = \text{Edit}(z_{IR}|z_c, \ldots). \) Our diffusion-based CIR model, named CompoDiff, can easily deal with versatile and complex conditions, benefiting from the flexibility of the latent diffusion model [23] and the classifier-free guidance [10]. We train a latent diffusion model that translates the embedding of the reference image \( (z_{IR}) \) into the embedding of the target image \( (z_t) \) guided by the em-
CompoDiff uses a two-stage training strategy (Fig. 2). In stage 1, we train a text-to-image latent diffusion model on LAION-2B. In stage 2, we fine-tune the model on our synthetic triplet dataset, SynthTriplets18M, and LAION-2B. Below, we describe the details of each stage.

In **stage 1**, we train a transformer decoder to convert CLIP textual embeddings into CLIP visual embeddings. This stage is similar to training the Dalle-2 prior, but our model takes only two tokens; a noised CLIP image embedding and a diffusion timestep embedding. The Dalle-2 prior model is computationally inefficient because it also takes 77 encoded CLIP text embeddings as an input. However, CompoDiff uses the encoded text embeddings as conditions through cross-attention mechanisms, which speeds up the process by a factor of three while maintaining similar performance (See Sec. 4.4). Instead of using the noise prediction of Ho et al. [11], we train the transformer decoder to predict the denoised $z_i$ directly due to the stability.

Now, we introduce the objective of the first stage with CLIP image embeddings of an input image $z_i$, encoded CLIP text embeddings for text condition $z_c$, and the de-
Figure 3. Inference overview. Using the denoising transformer $\theta_d$, we perform composed image retrieval (CIR). We use the classifier-free guidance to transform the input reference image to the target image feature, and perform image-to-image retrieval on the retrieval DB.

Inference

As shown in Fig. 3, given a reference image feature $z_{iR}$, a text condition feature $z_{CT}$, and a mask embedding $z_{CM}$, we apply a denoising diffusion process as follows:

$$
\mathbb{E}_{t \sim [1, T]} \| z_i - \theta_d(z_i^{(t)}, t \mid z_{CT}, z_{iR}, z_{CM}) \|^2
$$

During training, we randomly drop the text condition by replacing $z_{CT}$ with a null text embedding $\varnothing_{CT}$ in order to induce CFG. We use the empty text CLIP embedding ("""") for the null embedding.

In stage 2, we incorporate condition embeddings, injected by cross-attention, into CLIP text embeddings, along with CLIP reference image visual embeddings and mask embeddings (See Fig. 2). We fine-tune the model with three different tasks: a conversion task that converts textual embeddings into visual embeddings, a mask-based conversion task, and the triplet-based CIR task. The first two tasks are trained on LAION-2B, and the last on SynthTriplets18M.

The mask-based conversion task learns a diffusion process that recovers the full image embedding from a masked image embedding. As we do not have mask annotations, we extract masks using a zero-shot text-conditioned segmentation model, CLIPSeg [18]. We use the nouns of the given caption for the CLIPSeg conditions. Then, we add a Gaussian random noise to the mask region of the image and extract $z_{i,masked}$. We also introduce mask embedding $z_{CM}$ by projecting a $64 \times 64$ resized mask to the CLIP embedding dimension using an MLP, where $z_{CM}$ is used for CFG. Now, the mask-based conversion task is defined as follows:

$$
\mathbb{E}_{t \sim [1, T]} \| z_i - \theta_d(z_i^{(t)}, t \mid z_{CT}, z_{iR}, z_{CM}) \|^2
$$

Finally, we introduce the triplet-based training objective to solve CIR tasks on SynthTriplets18M as follows:

$$
\mathbb{E}_{t \sim [1, T]} \| z_{i}_{trp} - \theta_d(z_{i}^{(t)}, t \mid z_{CT}, z_{iR}, z_{CM}) \|^2
$$

where $z_{i}_{trp}$ is a reference image feature and $z_{i}_{trp}$ is a modified target image feature.

We update the model by randomly using one of the conversion task, the mask-based conversion task, or the triplet-based CIR task with the proportions 30%, 30%, 40%. As stage 1, the stage 2 conditions are randomly dropped except for the mask conditions. We use an all-zero mask condition for the tasks that do not use a mask condition.

2.2. Inference

As shown in Fig. 3, given a reference image feature $z_{iR}$, a text condition feature $z_{CT}$, and a mask embedding $z_{CM}$, we apply a denoising diffusion process as follows:

$$
\mathbb{E}_{t \sim [1, T]} \| z_i - \theta_d(z_i^{(t)}, t \mid z_{CT}, z_{iR}, z_{CM}) \|^2
$$

where $\varnothing$ denotes null embeddings, i.e., the empty text ("""") CLIP textual embedding for the text null embedding and an all-zero vector for the image null embedding. One of the advantages of Eq. (4) is the ability to handle various conditions at the same time. When using negative text, we simply replace $\varnothing_{i,masked}$ with the CLIP text embeddings $c_{T}$ for the negative text.

Another advantage of CFG is the controllability of the queries without training, e.g., it allows to control the degree of focus on image features to preserve the visual similarity with the reference by simply adjusting the weights $w_t$ or $w_{TR}$. In practice, we use $(w_t, w_{TR}) = (1.5, 7.5)$.

As CompoDiff is based on a diffusion process, we can easily control the balance between the inference time and the retrieval quality of the modified feature by varying step size. In practice, we set the step size to 5 or 10.


CIR requires a dataset of triplets $(x_{iR}, x_c, x_t)$ of a reference image $(x_{iR})$, a condition $(x_c)$, and the corresponding target image $(x_t)$. Instead of collecting a dataset by humans, we propose to automatically generate massive triplets by using generative models. We follow the main idea of InstructPix2Pix (IP2P) [4]. First, we generate $(x_{tR}, x_c, x_t)$, where $x_{tR}$ is a reference caption, $x_c$ is a modification instruction text, and $x_t$ is the caption modified by $x_c$. We use two strategies to generate $(x_{tR}, x_c, x_t)$: (1) We collect massive captions from the existing caption datasets and generate the modified captions by replacing the keywords in
Table 1. **Dataset statistics.** \( \langle x_{IR}, x_c, x_t \rangle \) denotes the triplet of captions, i.e., \{original caption, modification instruction, and modified caption\}, and \( \langle x_{IR}, x_c, x_t \rangle \) denotes the CIR triplet of \{original image, modification instruction, and modified image\}.

<table>
<thead>
<tr>
<th>IP2P</th>
<th>SynthTriplets18M</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \langle x_{IR}, x_c, x_t \rangle ) (before filtering)</td>
<td>452k</td>
</tr>
<tr>
<td>( \langle x_{IR}, x_c, x_t \rangle ) (after filtering)</td>
<td>313k</td>
</tr>
<tr>
<td>Unique object terms</td>
<td>47,345</td>
</tr>
<tr>
<td>( \langle x_{IR}, x_c, x_t \rangle ) (Keyword-based)</td>
<td>-</td>
</tr>
<tr>
<td>( \langle x_{IR}, x_c, x_t \rangle ) (LLM-based)</td>
<td>1M</td>
</tr>
<tr>
<td>( \langle x_{IR}, x_c, x_t \rangle ) (Total)</td>
<td>1M</td>
</tr>
</tbody>
</table>

Compared to the synthetic dataset of IP2P, our generation process is more scalable due to the keyword-based diverse caption generation process: Our caption triplets are synthesized based on keywords, SynthTriplets18M covers more diverse keywords than IP2P (47k vs. 586k as shown in Tab. 1). As a result, SynthTriplets18M contains more massive triplets (1M vs. 18M), and CIR models trained on our dataset achieve better scores even in the same scale (1M).

### 3.1. Keyword-based diverse caption generation

As the first approach to generating caption triplets, we collect captions from the existing caption datasets and modify the captions by replacing the object terms in the captions, e.g., \{"a strawberry tart is ...", "covert strawberry to pak choi", "a pak choi tart is ..."\} in Fig. 4. For the caption dataset, We use the captions from COYO 700M [5], StableDiffusion Prompts (user-generated prompts that make the quality of StableDiffusion better), LAION-2B-en-aesthetic (a subset of LAION-5B [25]) and LAION-COCO datasets [26] (synthetic captions for LAION-5B subsets with COCO style captions [6]). LAION-COCO less uses proper nouns than the real web texts).

We extract the object terms from the captions using the part-of-speech (POS) tagger provided by Spacy. After frequency filtering, we have 586k unique object terms (Tab. 1).
generates the samples only using SD 1.5, our generation process uses multiple DMs, for more diverse images not biased towards a specific model.

### 3.4. CLIP-based filtering

Our generation process can include low-quality triplets, e.g., broken images or non-related image-text pairs. To prevent the issue, we apply a filtering process following Brooks et al. [4] to remove the low-quality \( \langle x_{tR}, x_c, x_t \rangle \). First, we filter the generated images for an image-to-image CLIP threshold of 0.70 (between \( x_{tR} \) and \( x_t \)) to ensure that the images are not too different, an image-caption CLIP threshold of 0.2 to ensure that the images correspond to their captions (i.e., between \( x_{tR} \) and \( x_{iR} \), and between \( x_t \) and \( x_i \)), and a directional CLIP similarity [8] of 0.2 (\( L_{\text{direction}} := 1 - \sim(x_{iR}, x_t) \cdot \sim(x_{tR}, x_i) \)), where \( \sim(\cdot) \) is the CLIP similarity) to ensure that the change in before/after images correspond with the change in before/after images. For keyword-based data generation, we filter out for a keyword-image CLIP threshold of 0.20 to ensure that images contain the keyword (e.g., image-text CLIP similarity between the strawberry tart image and the keyword “strawberry” in Fig. 4). For instruction-based data generation, we filter out for an instruction-modified image CLIP threshold of 0.20 to ensure consistency with the given instructions.

After the filtering, we have 11.4M \( \langle x_{tR}, x_c, x_t \rangle \) from the keyword-based generated captions and 7.4M \( \langle x_{iR}, x_c, x_i \rangle \) from the LLM-based generated captions. It implies that the fidelity of our keyword-based method is higher than OPT fine-tuning in terms of T2I generation. As a result, SynthTriplets18M contains 18.8M synthetic \( \langle x_{tR}, x_c, x_t \rangle \). Examples of our dataset are shown in Fig. 7.

### 3.5. Dataset Statistics

We show the statistics of our generated caption dataset (i.e., before T2I generation, \( x_{tR} \) and \( x_t \)). We use the CLIP tokenizer to measure the statistics of the captions. Fig. 5 shows the cumulative ratio of captions with tokens less than X. About half of the captions have less than 13 tokens, and 90% of the captions have less than 20 tokens. Only 0.8% of the captions have more than 40 tokens.

We also compare SynthTriplets18M, FashionIQ, and CIRR in the instruction tokens (i.e., \( x_i \)). Fig. 6 shows that the instruction statistics vary across different datasets. We presume that this is why the zero-shot CIR is still difficult to outperform the task-specific supervised CIR methods.

### 4. Experiments

#### 4.1. Implementation details

**Encoders.** We use three different CLIP models for image encoder (Fig. 3 “CLIP Img Enc”), the official CLIP ResNet-50 and ViT-L/14 [20], and CLIP ViT-G/14 by OpenCLIP

---

**Table 2. The full 48 keyword converting templates.**

**Figure 5. Statistics of SynthTriplets18M instructions.**

**Figure 6. Statistics of instructions of the CIR datasets.**
Compared to the previous ZS-CIR methods (Pic2Word and SEARLE), we evaluate the zero-shot (ZS) capability of CompoDiff on four CIR benchmarks, including FashionIQ [30], CIRR [16], CIRCO [3] and GeneCIS [29]. We compare CompoDiff to the recent ZS CIR methods, including Pic2Word [24] and SEARLE [3]. We also reproduce the fusion-based methods, such as ARTEMIS [7] and Combiner [2], on SynthTriplets18M and report their ZS performances. Note that the current CIR benchmarks are somewhat insufficient to evaluate the effectiveness of CompoDiff, particularly considering real-world CIR queries. Our work is the first study that shows the impact of the dataset scale and the zero-shot CIR performances with various methods, such as our method, ARTEMIS and Combiner.

4.3. Qualitative comparisons on four Zero-shot CIR (ZS-CIR) benchmarks

Tab. 3 shows the overview of ZS-CIR comparison results. CLIP + IP2P denotes the naive editing-based approach by editing the reference image with the text condition using IP2P and performing image-to-image retrieval using CLIP ViT-L. In the table, CompoDiff outperforms all the existing methods with significant gaps. The table shows the effectiveness both of our diffusion-based CIR approach and our massive synthetic dataset. In the SynthTriplets18M-trained group, CompoDiff outperforms previous SOTA fusion-based CIR methods with a large gap, especially on CIRR and CIRCO, which focus on real-life images and complex descriptions. Our improvement is not main due to the architecture, as CompoDiff already outperforms the fusion methods in RN50. We also can observe that the SynthTriplets18M-trained group also enables the fusion-based methods to have the ZS capability competitive to the SOTA ZS-CIR methods, Pic2Word and SEARLE.
and SEARLE). CompoDiff achieves remarkable improvements on the same architecture scale (i.e., ViT-L), except on CIRR. We argue that it is due to the noisiness of the CIRR dataset. Instead, CompoDiff outperforms the other methods on FashionIQ, CIRCO and GeneCIS with a significant gap. We believe that it is because CompoDiff explicitly utilizes the diverse and massive synthetic triplets, while Pic2Word and SEARLE only employ images and the “a photo of” caption during training, resulting in a lack of diversity and generalizability.

### 4.4. Impact of dataset scale

Tab. 4 shows the impact of the dataset scale by SynthTriplets18M and LAION-2B (ARTEMIS and Combiner are trained solely on SynthTriplets18M, while CompoDiff is trained on both).

<table>
<thead>
<tr>
<th>IP2P(1M)</th>
<th>1M</th>
<th>5M</th>
<th>10M</th>
<th>18.8M</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FashionIQ Avg(R@10, R@50)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARTEMIS</td>
<td>26.03</td>
<td>27.44</td>
<td>36.17</td>
<td>41.35</td>
</tr>
<tr>
<td>Combiner</td>
<td>29.83</td>
<td>29.64</td>
<td>35.23</td>
<td>41.81</td>
</tr>
<tr>
<td>CompoDiff</td>
<td>27.24</td>
<td>31.91</td>
<td>38.11</td>
<td>42.41</td>
</tr>
<tr>
<td><strong>CIRR Avg(R@1, R_s@1)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ARTEMIS</td>
<td>14.91</td>
<td>15.12</td>
<td>15.84</td>
<td>17.56</td>
</tr>
<tr>
<td>Combiner</td>
<td>16.50</td>
<td>16.88</td>
<td>17.21</td>
<td>18.77</td>
</tr>
<tr>
<td>CompoDiff</td>
<td>27.42</td>
<td>28.32</td>
<td>31.50</td>
<td>37.25</td>
</tr>
</tbody>
</table>

Table 4. Impact of dataset scale. IP2P denotes the public 1M synthetic dataset by [4].

![Table 4. Impact of dataset scale. IP2P denotes the public 1M synthetic dataset by [4].](image)

CompoDiff shows consistent performance improvements from 1M to 18.8M, where manually collecting triplets in this scale is infeasible and nontrivial. Thanks to our diversification strategy, particularly keyword-based generation, we can scale up the triplet to 18.8M without manual human labor.

Tab. 4 shows that the massive data points are not necessary for training CompoDiff, but all methods are consistently improved by scaling up the data points. Also, although the FashionIQ and CIRR scores look somewhat saturated after 10M, these scores cannot represent authentic CIR performances due to the limitations of the datasets. As far as we know, this is the first study that shows the impact of the dataset scale on the ZS-CIR performances.

### 4.5. Qualitative examples

We qualitatively show the versatility of CompoDiff for handling various conditions. For example, CompoDiff not only can handle a text condition, but it can also handle a negative text condition (e.g., removing specific objects or patterns in the retrieval results), masked text condition (e.g., specifying the area for applying the text condition). CompoDiff even can handle all conditions simultaneously. To show the quality of the retrieval results, we conduct a zero-shot CIR on the entire LAION-2B [25] using FAISS [14].

Fig. 8 shows qualitative comparisons of zero-shot CIR results by Pic2Word and CompoDiff. CompoDiff results in semantically high-quality retrieval results (e.g., understanding the “crowdedness” of the query image and the meaning of the query text at the same time). However, Pic2Word shows poor understanding of the given queries, resulting in unfortunate retrieval results (e.g., ignoring “grown up” of text query, or the “crowdedness” of the query image).

Finally, it is worth noting that CompoDiff generates a feature belonging to the CLIP visual latent space. It means...
Figure 8. **Qualitative comparison of zero-shot CIR for Pic2Word and CompoDiff.** We conduct CIR on LAION. As Pic2Word cannot take a simple instruction, we made a simple modification for the given instruction.

<table>
<thead>
<tr>
<th>Reference Image</th>
<th>Conditions</th>
<th>unCLIP Generated</th>
<th>LAION Top-1</th>
<th>Reference Image</th>
<th>Conditions</th>
<th>unCLIP Generated</th>
<th>LAION Top-1</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="As 4k image" /></td>
<td>“As 4k image”</td>
<td><img src="image2" alt="unCLIP Generated" /></td>
<td><img src="image3" alt="LAION Top-1" /></td>
<td><img src="image4" alt="Reference Image" /></td>
<td>“meteor”</td>
<td><img src="image5" alt="unCLIP Generated" /></td>
<td><img src="image6" alt="LAION Top-1" /></td>
</tr>
<tr>
<td><img src="image7" alt="As 4k image - pink rabbit" /></td>
<td>“As 4k image - pink rabbit”</td>
<td><img src="image8" alt="unCLIP Generated" /></td>
<td><img src="image9" alt="LAION Top-1" /></td>
<td><img src="image10" alt="Reference Image" /></td>
<td>“meteor”</td>
<td><img src="image11" alt="unCLIP Generated" /></td>
<td><img src="image12" alt="LAION Top-1" /></td>
</tr>
<tr>
<td><img src="image13" alt="make the dog a cat" /></td>
<td>“make the dog a cat”</td>
<td><img src="image14" alt="unCLIP Generated" /></td>
<td><img src="image15" alt="LAION Top-1" /></td>
<td><img src="image16" alt="Reference Image" /></td>
<td>“Make it long sleeve”</td>
<td><img src="image17" alt="unCLIP Generated" /></td>
<td><img src="image18" alt="LAION Top-1" /></td>
</tr>
</tbody>
</table>

Figure 9. **Generated and retrieved images by CompoDiff.** Images are generated by unCLIP decoder and retrieved from LAION using transformed features by CompoDiff.

unCLIP [22], which decodes a CLIP image feature to an image, can be applied to our composed features. We compare the top-1 retrieval results from LAION and the generated images in Fig. 9. We use the community version ViT-L unCLIP decoder [15], by replacing the original Prior module to CompoDiff. As shown in the figures, CompoDiff can manipulate the given input reflecting the given conditions.

5. Conclusion

We have introduced CompoDiff, a novel diffusion-based method for solving complex CIR tasks. We have created a large and diverse dataset named SynthTriplets18M, consisting of 18.8M triplets of images, modification texts, and modified images. CompoDiff has demonstrated impressive ZS-CIR capabilities, as well as remarkable versatility in handling diverse conditions, such as negative text or image masks, and the controllability to enhance user experience, such as adjusting image text query weights. Furthermore, by training the existing CIR methods on SynthTriplets18M, the models became comparable ZS predictors to the ZS-CIR methods. We strongly encourage future researchers to leverage our dataset to advance the field of CIR.
References


