CompoDiff: Versatile Composed Image Retrieval With Latent Diffusion

Geonmo Gu*,[1], Sanghyuk Chun*,[2], Wonjae Kim[2], HeeJae Jun[1], Yoohoon Kang[1], Sangdoo Yun[2]

[1]NAVER Vision [2]NAVER 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 shortages 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.*, $z_i = \texttt{fusion}(z_{i_R}, z_c)$ where z_i, z_c, z_{i_R} 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 ($x_{c_{T^-}}$) (*e.g.*, an image + "with cherry blossom" − "France", as in Fig. 1 (b)), indicate where (x_{c_M}) 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_{i_R} 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 $\langle x_{i_R}, x_c, x_i \rangle$ consisting of a reference image (x_{i_R}), a text condition (x_c), and the corresponding target image (x_i). 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.

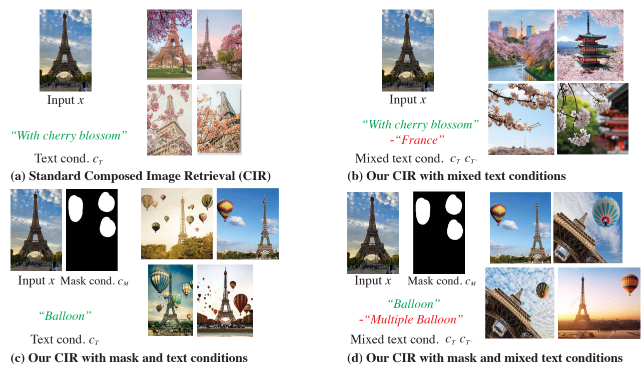

Figure 1. **CIR scenarios.** (a) A standard CIR scenario. (b-d) Our versatile CIR scenarios with mixed conditions (*e.g.*, negative text and mask). Results by CompoDiff on LAION-2B.

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.*, $z_i = \texttt{Edit}(z_{i_R} | z_c, \ldots)$. 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_{i_R}) into the embedding of the target image (z_i) guided by the em-

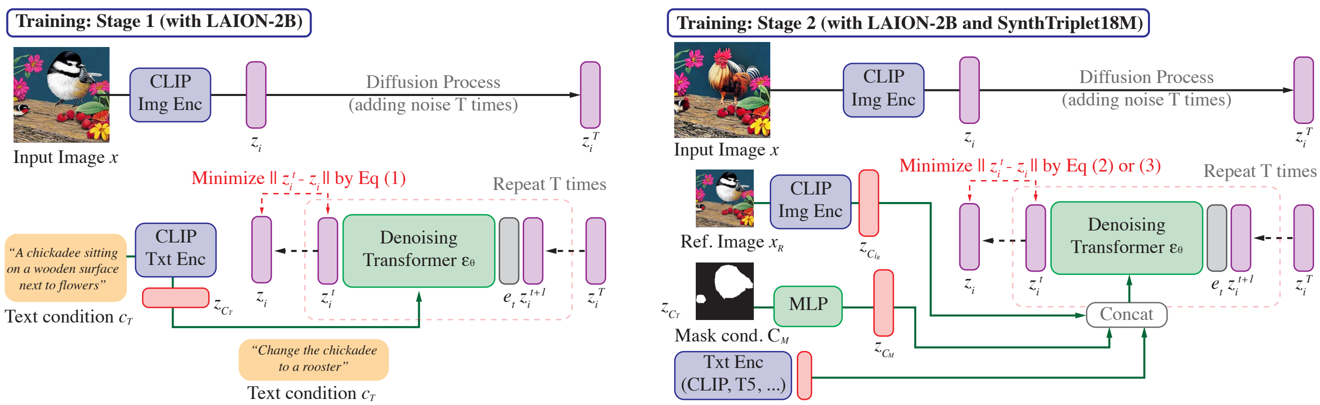

Figure 2. **Training overview.** Stage 1 is trained on LAION-2B with text-to-image generation. For stage 2, we alternatively update Denoising Transformer ϵ_θ on LAION-2B with text-to-image generation and SynthTriplets18M.

bedding of the given text condition (z_c). As shown in Fig. 1, CompoDiff can handle various conditions, which is not possible with the standard CIR scenario with the limited text condition x_{c_T}. Although our method has an advantage over the existing fusion-based CIR methods in terms of versatility, CompoDiff also needs to be trained with triplet datasets.

We address the dataset scale issue by synthesizing a vast set of high-quality **18.8M** triplets of $\langle x_{i_R}, x_c, x_i \rangle$. Our approach is fully automated without human verification; hence, it is scalable even to 18.8M. We follow InstructPix2Pix (IP2P) [4] for synthesizing triplets, while our dataset contains $\times 40$ more triplets and $\times 12.5$ more keywords (*e.g.*, objects, background details, or textures) than IP2P. Our **SynthTriplets18M** dataset is over 500 times larger than existing CIR datasets and covers a diverse and extensive range of conditioning cases, resulting in a notable performance improvement for any CIR model. For example, ARTEMIS [7] trained exclusively with SynthTriplets18M shows outperforming zero-shot performance even than its FashionIQ-trained counterpart (40.6 vs. 38.2).

To show the generalizability of the models, we evaluate the models on the "zero-shot" (ZS) CIR scenario using four CIR benchmarks: FashionIQ [30], CIRR [16], CIRCO [3], and GeneCIS [29]; *i.e.*, we report the retrieval results by the models trained on our SynthTriplets18M and a large-scale image-text paired dataset *without access to the target triplet datasets*. In all experiments, CompoDiff achieves the best zero-shot performances with significant gaps (See Tab. 3). Moreover, we observe that the fusion-based approaches solely trained on SynthTriplets18M (*e.g.*, Combiner [2]) show comparable or outperforming zero-shot CIR performances compared to the previous SOTA ZS-CIR methods [3, 24]. Furthermore, we qualitatively observe that the retrieval results of CompoDiff are semantically better than previous zero-shot CIR methods, such as Pic2Word, on a large-scale image database, *e.g.*, LAION-2B.

Another notable advantage of CompoDiff is the controllability of various conditions during inference, which is inherited from the nature of diffusion models. Users can adjust the weight of conditions to make the model focus on their preference. Users can also manipulate randomness to vary the degree of serendipity. In addition, CompoDiff can control the speed of inference with minimal sacrifice in retrieval performance, accomplished by adjusting the number of sampling steps in the diffusion model. As a result, CompoDiff can be deployed in various scenarios with different computational budgets. All of these controllability features are achievable by controlling the inference parameters of classifier-free guidance without any model training.

2. CompoDiff: CIR with Latent Diffusion

2.1. Training

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_T}, and the de-

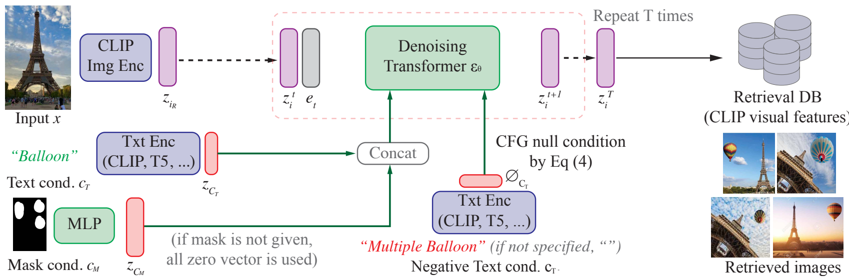

Figure 3. **Inference overview.** Using the denoising transformer ε_θ, we perform composed image retrieval (CIR). We use the classifier-free guidance to transform the input reference image to the target image feature, and perform image-to-image retrieval on the retrieval DB.

noising Transformer ϵ_θ:

$$\mathbb{E}_{t\sim[1,T]}\|z_i - \epsilon_\theta(z_i^{(t)}, t|z_{c_T})\|^2 \tag{1}$$

During training, we randomly drop the text condition by replacing z_{c_T} with a null text embedding $\varnothing_{c_T}$ 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 $z_{i,\text{masked}}$. We also introduce mask embedding z_{c_M} by projecting a 64×64 resized mask to the CLIP embedding dimension using an MLP, where z_{c_M} is used for CFG. Now, the mask-based conversion task is defined as follows:

$$\mathbb{E}_{t\sim[1,T]}\|z_i - \epsilon_\theta(z_{i,\text{masked}}^{(t)}, t|z_{c_T}, z_{i,\text{masked}}, z_{c_M})\|^2, \tag{2}$$

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

$$\mathbb{E}_{t\sim[1,T]}\|z_{i_T} - \epsilon_\theta(z_{i_T}^{(t)}, t|z_{c_T}, z_{i_R}, z_{c_M})\|^2, \tag{3}$$

where z_{i_R} is a reference image feature and z_{i_T} 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_{i_R}, a text condition feature z_{c_T}, and a mask embedding z_{c_M}, we apply a denoising diffusion process as follows:

$$\begin{aligned}\tilde{\epsilon}_\theta(z_i^{(t)}, t|z_{c_T}, z_{i_R}, z_{c_M}) &= \epsilon_\theta(z_i^{(t)}, t|\varnothing_{c_T}, \varnothing_{i_R}, z_{c_M})\\ +w_I(\epsilon_\theta(z_i^{(t)}, t|\varnothing_{c_T}, z_{i_R}, z_{c_M}) &- \epsilon_\theta(z_i^{(t)}, t|\varnothing_{c_T}, \varnothing_{i_R}, z_{c_M}))\\ +w_T(\epsilon_\theta(z_i^{(t)}, t|z_{c_T}, z_{i_R}, z_{c_M}) &- \epsilon_\theta(z_i^{(t)}, t|\varnothing_{c_T}, z_{i_R}, z_{c_M}))\end{aligned} \tag{4}$$

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 $\varnothing_{i_T}$ 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_I or w_T. In practice, we use (w_I, w_T) = (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.

3. SynthTriplets18M: Massive High-Quality Synthesized Dataset

CIR requires a dataset of triplets $\langle x_{i_R}, x_c, x_i\rangle$ of a reference image (x_{i_R}), a condition (x_c), and the corresponding target image (x_i). Instead of collecting a dataset by humans, we propose to automatically generate massive triplets by using generative models. We follow the main idea of Instuct Pix2Pix (IP2P) [4]. First, we generate $\langle x_{t_R}, x_c, x_t\rangle$ where x_{t_R} 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 $\langle x_{t_R}, x_c, x_t\rangle$: (1) We collect massive captions from the existing caption datasets and generate the modified captions by replacing the keywords in

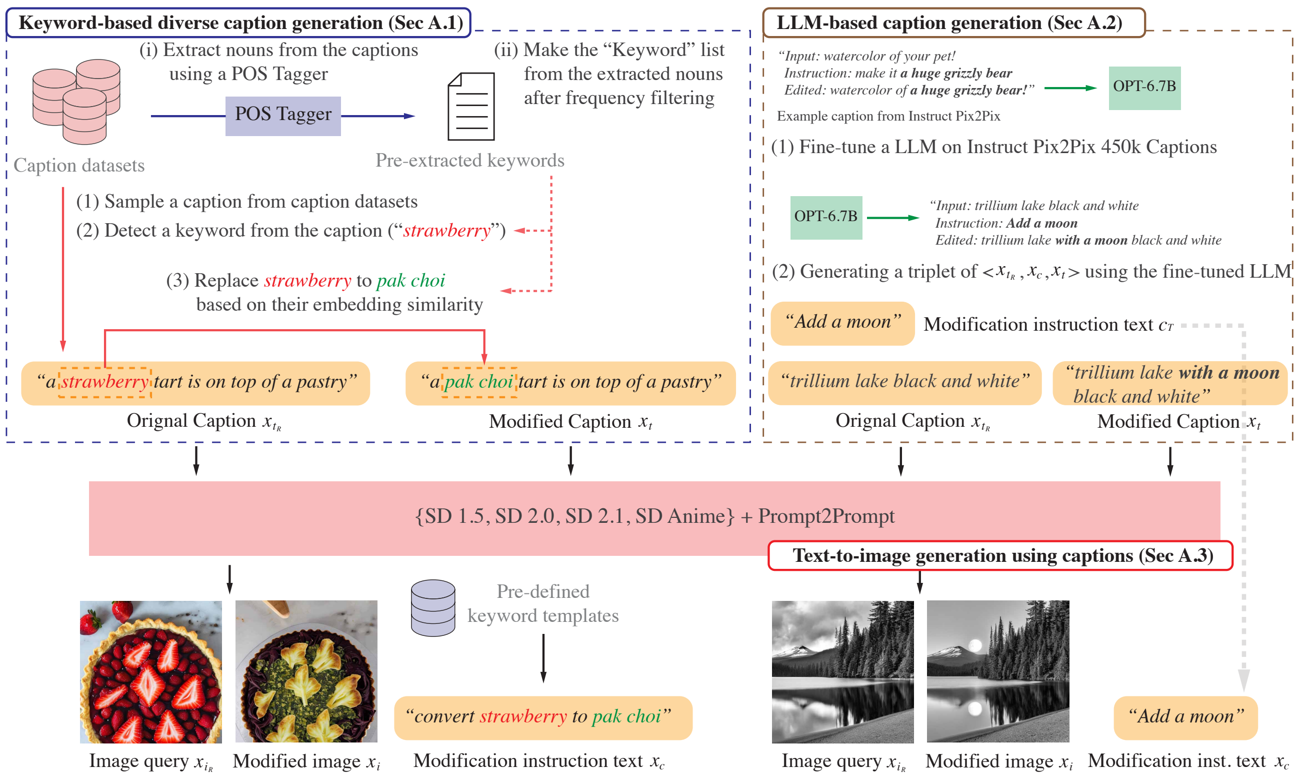

Figure 4. **Overview of the generation process for SynthTriplets18M.** $\langle x_{i_R}, x_c, x_i\rangle$ from $\langle x_{t_R}, x_c, x_t\rangle$.

	IP2P	SynthTriplets18M
$\langle x_{t_R}, x_c, x_t\rangle$(before filtering)	452k	500M
$\langle x_{t_R}, x_c, x_t\rangle$(after filtering)	313k	60M
Unique object terms	47,345	**586,369**
$\langle x_{i_R}, x_c, x_i\rangle$ (Keyword-based)	-	11.4M
$\langle x_{i_R}, x_c, x_i\rangle$ (LLM-based)	1M	7.4M
$\langle x_{i_R}, x_c, x_i\rangle$ (Total)	1M	**18.8M**

Table 1. **Dataset statistics.** $\langle x_{t_R}, x_c, x_t\rangle$ denotes the triplet of *captions*, *i.e.*, {original caption, modification instruction, and modified caption}, and $\langle x_{i_R}, x_c, x_i\rangle$ denotes the *CIR triplet* of {original image, modification instruction, and modified image}.

the reference caption (Sec. 3.1). (2) We fine-tune a large language model, OPT-6.7B [31], on the generated caption triplets from Brooks et al. [4] (Sec. 3.2). After generating massive triplets of $\langle x_{t_R}, x_c, x_t\rangle$, we generate images from the caption triplets using StableDiffusion (SD) and Prompt-to-Prompt Hertz et al. [9] following IP2P (Sec. 3.3). We employ CLIP-based filtering to ensure high-quality triplets (Sec. 3.4). The entire generation process is shown in Fig. 4.

Compared to manual dataset collections [16, 30], our approach can easily generate more diverse triplets even if a triplet rarely occurs in reality (See the examples in Fig. 4).

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).

Templates to change ${source} to ${target}		
"replace ${source} with ${target}"	*"substitute ${target} for ${source}"*	*"change ${source} to ${target}"*
"${target}"	*"${source} is removed and ${target} takes its place"*	*"alter ${source} to ${target}"*
"apply ${target}"	*"modify ${source} to become ${target}"*	*"swap ${source} for ${target}"*
"convert ${source} to ${target}"	*"customize ${source} to become ${target}"*	*"redesign ${source} as ${target}"*
"replace ${source} with ${target}"	*"change ${source} to match ${target}"*	*"turn ${source} into ${target}"*
"update ${source} to ${target}"	*"${target} is introduced after ${source} is removed"*	*"adapt ${source} to fit ${target}"*
"substitute ${target} for ${source}"	*"${target} is added in place of ${source}"*	*"choose ${target} instead"*
"alter ${source} to match ${target}"	*"${target} is introduced as the new option after*	*"${target} is the new choice"*
"upgrade ${source} to ${target}"	*"${source} is removed and ${target} is added"*	*"${target} is the new selection"*
"amend ${source} to fit ${target}"	*"${source} is removed and ${target} is introduced"*	*"${target} is the new option"*
"opt for ${target}"	*"${target} is added as a replacement for ${source}"*	*"use ${target} from now on"*
${source} is removed"	*"${target} is the new option available"*	*"remodel ${source} into ${target}"*
"add ${target}"	*"${target} is added after ${source} is removed"*	*"revamp ${source} into ${target}"*
"if it is ${target}"	*"${target} is introduced after ${source} is retired"*	*"exchange ${source} with ${target}"*
"${target} is the updated option"	*"tweak ${source} to become ${target}"*	*"transform ${source} into ${target}"*
"${target} is the updated choice"	*"${source} is replaced with ${target}"*	*"${target} is the updated version"*

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

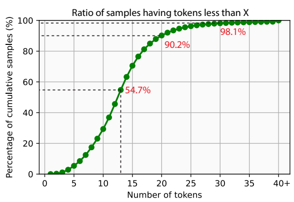

Figure 5. **Statistics of SynthTriplets18M instructions.**

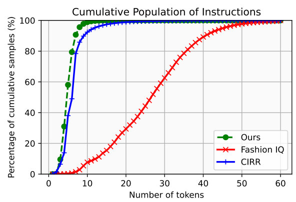

Figure 6. **Statistics of instructions of the CIR datasets.**

To make the caption triplet $\langle x_{t_R}, x_c, x_t \rangle$, we replace the object term of each caption with other similar keywords by using the CLIP similarity score. More specifically, we extract the textual feature of keywords using the CLIP ViT-L/14 text encoder [20], and we choose an alternative keyword from keywords with a CLIP similarity between 0.5 and 0.7. By converting the original object to a similar object, we have caption pairs of $\langle x_{t_R}, x_t \rangle$.

Using the caption pair $\langle x_{t_R}, x_t \rangle$, we generate the modification instruction text x_{c_T} based on a randomly chosen template from 48 pre-defined templates shown in Tab. 2. After this process, we have the triplet of $\langle x_{t_R}, x_c, x_t \rangle$. We generate ≈30M caption triplets by the keyword-based method.

3.2. Amplifying IP2P triplets by LLM

We also re-use the generated $\langle x_{t_R}, x_c, x_t \rangle$ by IP2P. We amplify the number of IP2P triplets by applying the efficient LoRA fine-tuning [12] to OPT-6.7B [31] on the generated 452k caption triplets provided by Brooks et al. [4]. Using the fine-tuned OPT, we generate ≈30M caption triplets.

3.3. Triplet generation from caption triplets

We generate 60M caption triplets $\langle x_{t_R}, x_c, x_t \rangle$ by the keyword-based generation process (Sec. 3.1) and the LLM-based generation process (Sec. 3.2). We generate images for x_{t_R} (original caption) and x_t (modified caption) using state-of-the-art text-to-image generation models, such as StableDiffusion (SD) 1.5, 2.0, 2.1, and SD Anime. Following Brooks et al. [4], we apply Prompt-to-Prompt [9], which aims to generate similar images while keeping the identity of the original image (*e.g.*, the examples in Fig. 4). As a result, we generate 60M $\langle x_{i_R}, x_c, x_i \rangle$ (z_{c_T} is given; x_{i_R} and x_i are generated by x_{t_R} and x_t, respectively). While IP2P 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 $\langle x_{i_R}, x_c, x_i \rangle$. First, we filter the generated images for an image-to-image CLIP threshold of 0.70 (between x_{i_R} and x_i) 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_{t_R} and x_{i_R}, and between x_t and x_i), and a directional CLIP similarity [8] of 0.2 ($L_{\text{direction}} := 1 - \text{sim}(x_{i_R}, x_i) \cdot \text{sim}(x_{t_R}, x_t)$, where $\text{sim}(\cdot)$ is the CLIP similarity) to ensure that the change in before/after captions 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 $\langle x_{i_R}, x_c, x_i \rangle$ from the keyword-based generated captions and 7.4M $\langle x_{i_R}, x_c, x_i \rangle$ 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 $\langle x_{i_R}, x_c, x_i \rangle$. 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_{t_R} 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_c). 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

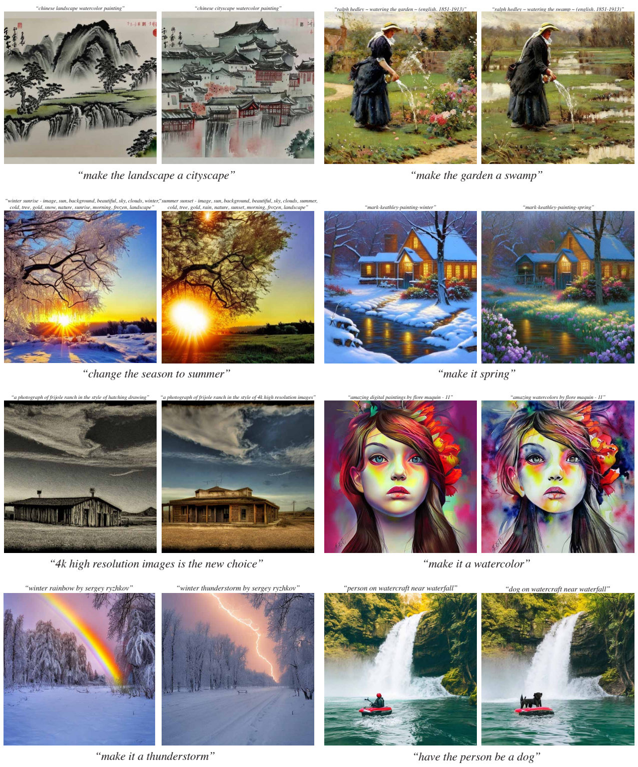

Figure 7. **Examples of SynthTriplets18M.** We show examples of $\langle x_{i_R}, x_c, x_i \rangle$, *i.e.*, {original image, modification instruction, and modified image}, as well as the generation prompt for x_{i_R} and x_i.

[13], whose feature dimensions are 768, 1024, and 1280, respectively. Beyond the backbone size, we observe the choice of the text condition encoder is also important (Fig. 3 "Txt Enc"). As shown Balaji et al. [1], using a text-oriented model such as T5 [21] in addition to the CLIP textual encoder results in improved performance of text-to-image generation models. Motivated by this observation, we also use both the CLIP textual encoder and the language-oriented encoder for small image encoder (*i.e.*, CLIP ViT-L/14). We also observed the positive effect of the text-oriented model and experiment results showed that T5-XL, which has 3B parameters, could improve the performance by a large margin in the overall evaluation metrics.

Denoiser. We use a simple Transformer architecture for the denoising procedure, instead of the denoising U-Net [23]. We empirically observe that our transformer architecture performs slightly better than the U-Net architecture, but is much simpler. We use the multi-head self-attention blocks as the original Transformer [28]. We set the depth, the number of heads, and the dimensionality of each head to 12, 16, and 64, respectively. The hidden dimension of the Transformer is set to 768 and 1280 for ViT-L and ViT-G, respectively. The denoising Transformer takes two inputs: a noisy visual embedding and a time-step embedding. The conditions (*e.g.*, text, mask and image conditions) are applied only to the cross-attention layer; thereby it is computationally efficient even using many conditions. CompoDiff is similar to the "DiT with cross-attention" by Peebles and Xie [19], but handles more various conditions.

Training details. For the efficient training, all visual features are pre-extracted and frozen. All training text embeddings are extracted at every iteration. To improve computational efficiency, we reduced the number of input tokens of the T5 models to 77, as in CLIP. A single-layer perceptron was employed to align the dimension of text embeddings extracted from T5-XL with that of CLIP ViT-L/14.

4.2. Experiment settings

All models were trained using AdamW [17]. We used DDIM [27] for the sampling variance method. We did not apply any image augmentation but used pre-extracted CLIP image features for computational efficiency; text features were extracted on the fly as text conditions can vary.

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.

Compared to the previous ZS-CIR methods (Pic2Word

Method	Arch	Fashion IQ (Avg) R@10	R@50	CIRR R@1	R_s@1	CIRCO mAP@5	mAP@10	mAP@25	GeneCIS R@1
CLIP + IP2P†	ViT-L	7.01	12.33	4.07	6.11	1.83	2.10	2.37	2.44
Previous zero-shot methods (without SynthTriplets18M)									
Pic2Word†	ViT-L	24.70	43.70	23.90	53.76	8.72	9.51	10.65	11.16
SEARLE-OTI†	ViT-L	27.51	47.90	<u>24.87</u>	53.80	10.18	11.03	12.72	-
SEARLE†	ViT-L	25.56	46.23	24.24	53.76	11.68	12.73	14.33	12.31
Zero-shot results with the models trained with SynthTriplets18M									
ARTEMIS	RN50	33.24	47.99	12.75	21.95	9.35	11.41	13.01	13.52
Combiner	RN50	34.30	49.38	12.82	24.12	9.77	12.08	13.58	14.93
CompoDiff	RN50	35.62	48.45	18.02	57.16	12.01	13.28	15.41	14.65
CompoDiff	ViT-L	36.02	48.64	18.24	57.42	12.55	13.36	<u>15.83</u>	14.88
CompoDiff	ViT-L & T5-XL	<u>37.36</u>	<u>50.85</u>	19.37	<u>59.13</u>	<u>12.31</u>	<u>13.51</u>	15.67	<u>15.11</u>
CompoDiff	ViT-G	**39.02**	**51.71**	**26.71**	**64.54**	**15.33**	**17.71**	**19.45**	**15.48**

Table 3. **Zero-shot CIR comparisons.** † denotes the results by the official model weight, otherwise, models are trained on SynthTriplets18M and LAION-2B (ARTEMIS and Combiner are trained solely on SynthTriplets18M, while CompoDiff is trained on both).

	IP2P(1M)	1M	5M	10M	18.8M
FashionIQ Avg(R@10, R@50)					
ARTEMIS	26.03	27.44	36.17	41.35	40.62
Combiner	29.83	29.64	35.23	41.81	41.84
CompoDiff	27.24	31.91	38.11	42.41	42.33
CIRR Avg(R@1, R_s@1)					
ARTEMIS	14.91	15.12	15.84	17.56	17.35
Combiner	16.50	16.88	17.21	18.77	18.47
CompoDiff	27.42	28.32	31.50	37.25	37.83

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

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 noiseness 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 on ARTEMIS, Combiner and CompoDiff. First, at a scale of 1M, models trained on our 1M subset significantly outperformed the IP2P triplets. This result indicates that our dataset has a more diverse representation capability. As the size of our dataset increases, the performance gradually improves. Notably, SynthTriplets18M 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 comparsions 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.

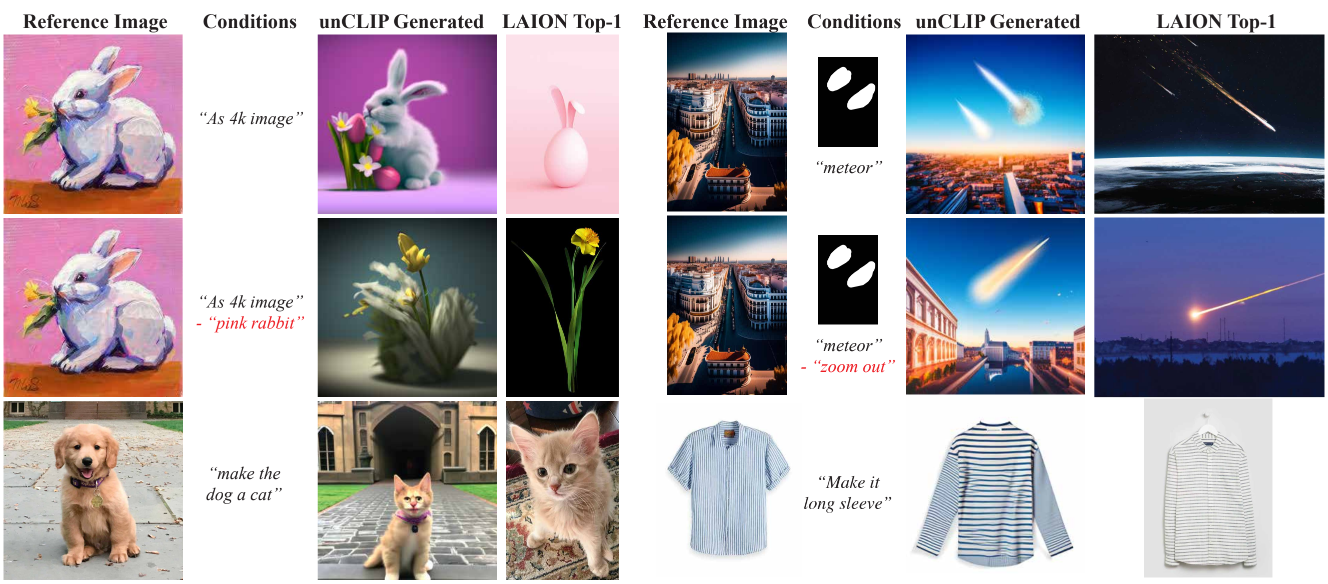

Figure 9. **Generated and retrieved images by CompoDiff.** Images are generated by unCLIP decoder and retrieved from LAION using transformed features by CompoDiff.

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

[1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 6

[2] 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. 2, 6

[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*, 2023. 2, 6

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 2, 3, 4, 5, 7

[5] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 4

[6] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 4

[7] Ginger Delmas, Rafael S Rezende, Gabriela Csurka, and Diane Larlus. Artemis: Attention-based retrieval with text-explicit matching and implicit similarity. In *International Conference on Learning Representations*, 2022. 2, 6

[8] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 5

[9] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 4, 5

[10] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 1

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[12] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 5

[13] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. If you use this software, please cite it as below. 6

[14] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547, 2019. 7

[15] Donghoon Lee, Jiseob Kim, Jisu Choi, Jongmin Kim, Minwoo Byeon, Woonhyuk Baek, and Saehoon Kim. Karlo-v1.0.alpha on coyo-100m and cc15m. https://github.com/kakaobrain/karlo, 2022. 8

[16] 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, 2, 4, 6

[17] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[18] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7086–7096, 2022. 3

[19] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022. 6

[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. 5

[21] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020. 6

[22] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 8

[23] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1, 6

[24] 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. *arXiv preprint arXiv:2302.03084*, 2023. 2, 6

[25] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022. 4, 7

[26] Christoph Schuhmann, Andreas Köpf, Richard Vencu, Theo Coombes, and Romain Beaumont. Laion coco: 600m synthetic captions from laion2b-en. https://huggingface.co/datasets/laion/laion-coco, 2022. 4

[27] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 6

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia

Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 6

[29] Sagar Vaze, Nicolas Carion, and Ishan Misra. Genecis: A benchmark for general conditional image similarity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 2, 6

[30] 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, 2, 4, 6

[31] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 4, 5