DiffusionCLIP: Text-Guided Diffusion Models for Robust Image Manipulation

Gwanghyun Kim[1] Taesung Kwon[1] Jong Chul Ye[2,1]
Dept. of Bio and Brain Engineering[1], Kim Jaechul Graduate School of AI[2]
Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea
{gwang.kim, star.kwon, jong.ye}@kaist.ac.kr

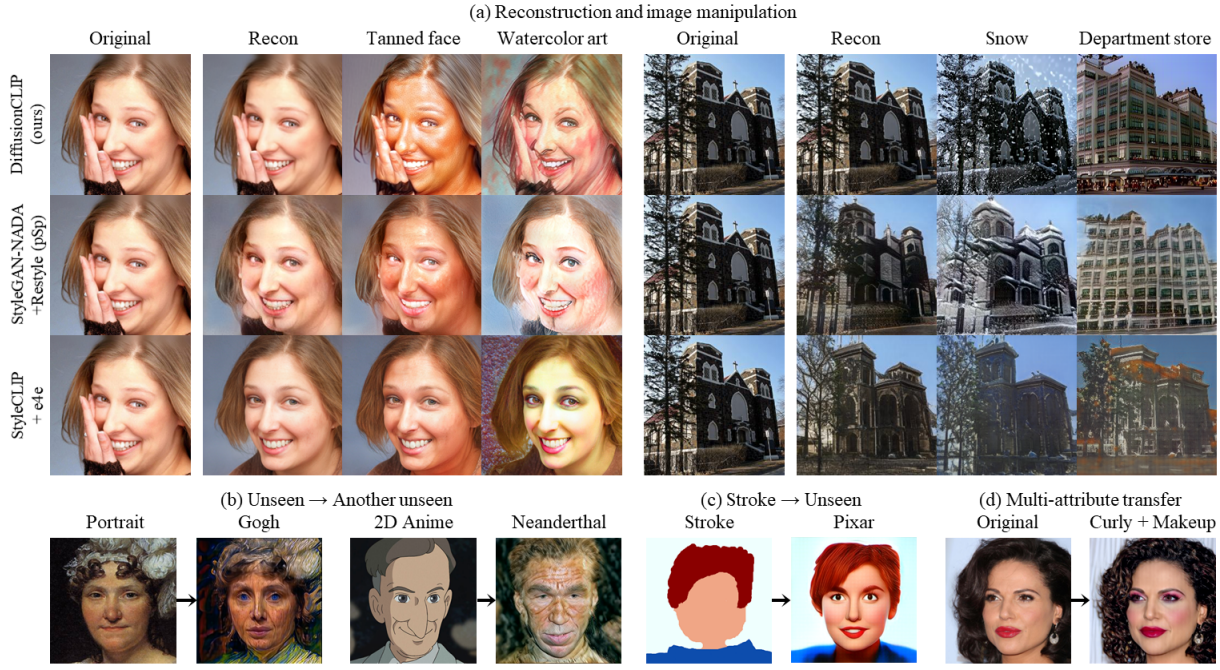

Figure 1. DiffusionCLIP enables faithful text-driven manipulation of real images by (a) preserving important details when the state-of-the-art GAN inversion-based methods fail. Other novel applications include (b) image translation between two unseen domains, (c) stroke-conditioned image synthesis to an unseen domain, and (d) multi-attribute transfer.

Abstract

Recently, GAN inversion methods combined with Contrastive Language-Image Pretraining (CLIP) enables zero-shot image manipulation guided by text prompts. However, their applications to diverse real images are still difficult due to the limited GAN inversion capability. Specifically, these approaches often have difficulties in reconstructing images with novel poses, views, and highly variable contents compared to the training data, altering object identity, or producing unwanted image artifacts. To mitigate these problems and enable faithful manipulation of real images, we propose a novel method, dubbed DiffusionCLIP, that performs text-driven image manipulation using diffusion models. Based on full inversion capability and high-quality image generation power of recent diffusion models, our method performs zero-shot image manipulation successfully even between unseen domains and takes another step towards general application by manipulating images from a widely varying ImageNet dataset. Furthermore, we propose a novel noise combination method that allows straightforward multi-attribute manipulation. Extensive experiments and human evaluation confirmed robust and superior manipulation performance of our methods compared to the existing baselines. Code is available at https://github.com/gwang-kim/DiffusionCLIP.git

This research was supported by Field-oriented Technology Development Project for Customs Administration through the National Research Foundation of Korea(NRF) funded by the Ministry of Science & ICT and Korea Customs Service (NRF-2021M3I1A1097938), and supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00075, Artificial Intelligence Graduate School Program (KAIST)).

1. Introduction

Recently, GAN inversion methods [1–4, 7, 32, 40] combined with Contrastive Language-Image Pretraining (CLIP)

2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 978-1-6654-6946-3/22/$31.00 ©2022 IEEE | DOI: 10.1109/CVPR52688.2022.00246

[29] has become popular thanks to their ability for zero-shot image manipulation guided by text prompts [16, 28]. Nevertheless, its real-world application on diverse types of images is still tricky due to the limited GAN inversion performance.

Specifically, successful manipulation of images should convert the image attribute to that of the target without unintended changes of the input content. Unfortunately, the current state-of-the-art (SOTA) encoder-based GAN inversion approaches [3, 32, 40] often fail to reconstruct images with novel poses, views, and details. For example, in the left panel of Fig. 1(a), e4e [40] and ReStyle [3] with pSp encoder [32] fail to reconstruct unexpected hand on the cheek, inducing the unintended change. This is because they have rarely seen such faces with hands during the training phase. This issue becomes even worse in the case of images from a dataset with high variance such as church images in LSUN-Church [46] and ImageNet [35] dataset. As shown in the right panel of Fig. 1(a) for the conversion to a department store, existing GAN inversion methods produce artificial architectures that can be perceived as different buildings.

Recently, diffusion models such as denoising diffusion probabilistic models (DDPM) [18,36] and score-based generative models [38, 39] have achieved great successes in image generation tasks [18, 19, 37, 39]. The latest works [14, 39] have demonstrated even higher quality of image synthesis performance compared to variational autoencoders (VAEs) [24,27,30], flows [15,23,31], auto-regressive models [26,41], and generative adversarial networks (GANs) [6, 17, 21, 22]. Furthermore, a recent denoising diffusion implicit models (DDIM) [37] further accelerates sampling procedure and enables nearly perfect inversion [14].

Inspired by this, here we propose a novel DiffusionCLIP - a CLIP-guided robust image manipulation method by diffusion models. Here, an input image is first converted to the latent noises through a forward diffusion. In the case of DDIM, the latent noises can be then inverted nearly perfectly to the original image using a reverse diffusion if the score function for the reverse diffusion is retained the same as that of the forward diffusion. Therefore, the key idea of DiffusionCLIP is to fine-tune the score function in the reverse diffusion process using a CLIP loss that controls the attributes of the generated image based on the text prompts.

Accordingly, DiffusionCLIP can successfully perform image manipulation both in the trained and unseen domain (Fig. 1(a)). We can even translate the image from an unseen domain into another unseen domain (Fig. 1(b)), or generate images in an unseen domain from the strokes (Fig. 1(c)). Moreover, by simply combining the noise predicted from several fine-tuned models, multiple attributes can be changed simultaneously through only one sampling process (Fig. 1(d)). Furthermore, DiffsuionCLIP takes another step towards general application by manipulating images from a widely varying ImageNet [35] dataset (Fig. 6), which has been rarely explored with GAN-inversion due to its inferior reconstruction. [5, 12]

Additionally, we propose a systematic approach to find the optimal sampling conditions that lead to high quality and speedy image manipulation. Qualitative comparison and human evaluation results demonstrate that our method can provide robust and accurate image manipulation, outperforming SOTA baselines.

2. Related Works

2.1. Diffusion Models

Diffusion probabilistic models [18,36] are a type of latent variable models that consist of a forward diffusion process and a reverse diffusion process. The forward process is a Markov chain where noise is gradually added to the data when sequentially sampling the latent variables $\boldsymbol{x}_t$ for $t = 1, \cdots, T$. Each step in the forward process is a Gaussian transition $q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) := \mathcal{N}(\sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \mathbf{I})$, where $\{\beta_t\}_{t=0}^T$ are fixed or learned variance schedule. The resulting latent variable $\boldsymbol{x}_t$ can be expressed as:

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_0 + (1-\alpha_t)\boldsymbol{w}, \quad \boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

where $\alpha_t := \prod_{s=1}^{t}(1-\beta_s)$. The reverse process $q(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t)$ is parametrized by another Gaussian transition $p_\theta(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t) := \mathcal{N}(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t),\ \sigma_\theta(\boldsymbol{x}_t, t)\mathbf{I})$. $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t)$ can be decomposed into the linear combination of $\boldsymbol{x}_t$ and a noise approximation model $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$, which can be learned by solving the optimization problem as follows:

$$\min_\theta \mathbb{E}_{\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0), \boldsymbol{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t} ||\boldsymbol{w} - \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)||_2^2. \tag{2}$$

After training $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}, t)$, the data is sampled using following reverse diffusion process:

$$\boldsymbol{x}_t = \frac{1}{\sqrt{1-\beta_t}}\left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right) + \sigma_t \boldsymbol{z}, \tag{3}$$

where $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. It was found that the sampling process of DDPM corresponds to that of the score-based generative models [38, 39] with the following relationship:

$$\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) = -\sqrt{1-\alpha_t}\nabla_{\boldsymbol{x}_t} \log p_\theta(\boldsymbol{x}_t). \tag{4}$$

Meanwhile, [37] proposed an alternative non-Markovian noising process that has the same forward marginals as DDPM but has a distinct sampling process as follows:

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}}\boldsymbol{f}_\theta(\boldsymbol{x}_t, t) + \sqrt{1-\alpha_{t-1}-\sigma_t^2}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) + \sigma_t^2 \boldsymbol{z}, \tag{5}$$

where, $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\boldsymbol{f}_\theta(\boldsymbol{x}_t, t)$ is a the prediction of $\boldsymbol{x}_0$ at t given $\boldsymbol{x}_t$ and $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$:

$$\boldsymbol{f}_\theta(\boldsymbol{x}_t, t) := \frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)}{\sqrt{\alpha_t}}. \tag{6}$$

This sampling allows using different samplers by changing the variance of the noise σ_t. Especially, by setting this noise to 0, which is a DDIM sampling process [37], the sampling process becomes deterministic, enabling full inversion of the latent variables into the original images with significantly fewer steps [14, 37]. In fact, DDIM can be considered as an Euler method to solve an ordinary differential equation (ODE) by rewriting Eq. 5 as follows:

$$\sqrt{\frac{1}{\alpha_{t-1}}}\boldsymbol{x}_{t-1} - \sqrt{\frac{1}{\alpha_t}}\boldsymbol{x}_t = \left(\sqrt{\frac{1}{\alpha_{t-1}} - 1} - \sqrt{\frac{1}{\alpha_t} - 1}\right)\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) \tag{7}$$

For mathematical details, see Supplementary Section A.

2.2. CLIP Guidance for Image Manipulation

CLIP [29] was proposed to efficiently learn visual concepts with natural language supervision. In CLIP, a text encoder and an image encoder are pretrained to identify which texts are matched with which images in the dataset. Accordingly, we use a pretrained CLIP model for our text-driven image manipulation.

To effectively extract knowledge from CLIP, two different losses have been proposed: a global target loss [28], and local directional loss [16]. The global CLIP loss tries to minimize the cosine distance in the CLIP space between the generated image and a given target text as follows:

$$\mathcal{L}_{\text{global}}(\boldsymbol{x}_{\text{gen}}, y_{\text{tar}}) = D_{\text{CLIP}}(\boldsymbol{x}_{\text{gen}}, y_{\text{tar}}), \tag{8}$$

where y_{tar} is a text description of a target, $\boldsymbol{x}_{\text{gen}}$ denotes the generated image, and D_{CLIP} returns a cosine distance in the CLIP space between their encoded vectors. On the other hand, the local directional loss [16] is designed to alleviate the issues of global CLIP loss such as low diversity and susceptibility to adversarial attacks. The local directional CLIP loss induces the direction between the embeddings of the reference and generated images to be aligned with the direction between the embeddings of a pair of reference and target texts in the CLIP space as follows:

$$\mathcal{L}_{\text{direction}}\left(\boldsymbol{x}_{\text{gen}}, y_{\text{tar}}; \boldsymbol{x}_{\text{ref}}, y_{\text{ref}}\right) := 1 - \frac{\langle \Delta I, \Delta T \rangle}{\|\Delta I\| \|\Delta T\|}, \tag{9}$$

where

$$\Delta T = E_T(y_{\text{tar}}) - E_T(y_{\text{ref}}),\ \Delta I = E_I(\boldsymbol{x}_{\text{gen}}) - E_I(\boldsymbol{x}_{\text{ref}}).$$

Here, E_I and E_T are CLIP's image and text encoders, respectively, and $y_{\text{ref}}, \boldsymbol{x}_{\text{ref}}$ are the source domain text and image, respectively. The manipulated images guided by the directional CLIP loss are known robust to mode-collapse issues because by aligning the direction between the image representations with the direction between the reference text and the target text, distinct images should be generated. Also, it is more robust to adversarial attacks because the perturbation will be different depending on images [29]. More related works are illustrated in Supplementary Section A.

3. DiffusionCLIP

The overall flow of the proposed DiffusionCLIP for image manipulation is shown in Fig. 2. Here, the input image $\boldsymbol{x}_0$ is first converted to the latent $\boldsymbol{x}_{t_0}(\theta)$ using a pretrained diffusion model $\boldsymbol{\epsilon}_\theta$. Then, guided by the CLIP loss, the diffusion model at the reverse path is fine-tuned to generate samples driven by the target text y_{tar}. The deterministic forward-reverse processes are based on DDIM [37]. For translation between unseen domains, the latent generation is also done by forward DDPM [18] process as will be explained later.

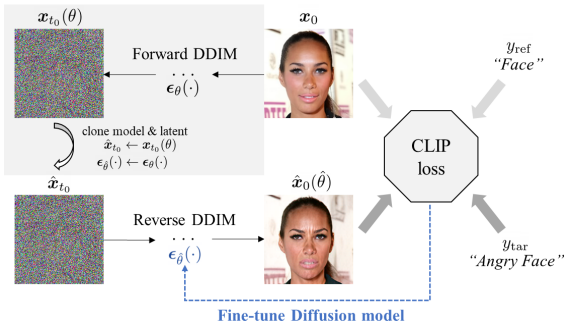

Figure 2. Overview of DiffusionCLIP. The input image is first converted to the latent via diffusion models. Then, guided by directional CLIP loss, the diffusion model is fine-tuned, and the updated sample is generated during reverse diffusion.

3.1. DiffusionCLIP Fine-tuning

In terms of fine-tuning, one could modify the latent or the diffusion model itself. We found that direct model fine-tuning is more effective, as analyzed in Supplementary Section D. Specifically, to fine-tune the reverse diffusion model $\boldsymbol{\epsilon}_\theta$, we use the following objective composed of the directional CLIP loss $\mathcal{L}_{\text{direction}}$ and the identity loss $\mathcal{L}_{\text{ID}}$:

$$\mathcal{L}_{\text{direction}}\left(\hat{\boldsymbol{x}}_0(\hat{\theta}), y_{\text{tar}}; \boldsymbol{x}_0, y_{\text{ref}}\right) + \mathcal{L}_{\text{id}}(\hat{\boldsymbol{x}}_0(\hat{\theta}), \boldsymbol{x}_0), \tag{10}$$

where $\boldsymbol{x}_0$ is the original image, $\hat{\boldsymbol{x}}_0(\hat{\theta})$ is the generated image from the latent $\boldsymbol{x}_{t_0}$ with the optimized parameter $\hat{\theta}$, y_{ref} is the reference text, y_{tar} is the target text given for image manipulation.

Here, the CLIP loss is the key component to supervise the optimization. Of two types of CLIP losses as discussed above, we employ directional CLIP loss as a guidance thanks to the appealing properties as mentioned in Section 2.2. For the text prompt, directional CLIP loss requires a reference text y_{ref} and a target text y_{tar} while training. For example, in the case of changing the expression of a given face image into an angry expression, we can use 'face' as a reference text and 'angry face' as a target text. In this paper, we often use concise words to refer to each text prompt (e.g. 'tanned face' to 'tanned').

The identity loss $\mathcal{L}_{\text{id}}$ is employed to prevent the unwanted changes and preserve the identity of the object. We generally use ℓ_1 loss as identity loss, and in case of human face image manipulation, face identity loss in [13] is added:

$$\mathcal{L}_{\text{id}}(\hat{\boldsymbol{x}}_0(\hat{\theta}), \boldsymbol{x}_0) = \lambda_{\text{L1}} \|\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0(\hat{\theta})\| + \lambda_{\text{face}} \mathcal{L}_{\text{face}}(\hat{\boldsymbol{x}}_0(\hat{\theta}), \boldsymbol{x}_0), \tag{11}$$

where $\mathcal{L}_{\text{face}}$ is the face identity loss [13], and $\lambda_{\text{L1}} \geq 0$ and $\lambda_{\text{face}} \geq 0$ are weight parameters for each loss. The necessity of identity losses depends on the types of the control. For some controls, the preservation of pixel similarity and the human identity are significant (e.g. expression, hair color) while others prefer the severe shape and color changes (e.g. artworks, change of species).

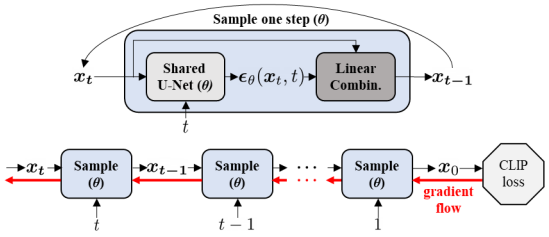

Figure 3. Gradient flows during fine-tuning the diffusion model with the shared architecture across t.

Existing diffusion models [14, 18, 37] adopt the shared U-Net [33] architecture for all t, by inserting the information of t using sinusoidal position embedding as used in the Transformer [42]. With this architecture, the gradient flow during DiffusionCLIP fine-tuning can be represented as Fig. 3, which is a similar process of training recursive neural network [34].

Once the diffusion model is fine-tuned, any image from the pretrained domain can be manipulated into the image corresponding to the target text y_{tar} as illustrated in Fig. 4(a). For details of the fine-tuning procedure and the model architecture, see Supplementary Section B and C.

3.2. Forward Diffusion and Generative Process

As the DDPM sampling process in Eq. 3 is stochastic, the samples generated from the same latent will be different every time. Even if the sampling process is deterministic, the forward process of DDPM, where the random Gaussian noise is added as in Eq. 1, is also stochastic, hence the reconstruction of the original image is not guaranteed. To fully leverage the image synthesis performance of diffusion models with the purpose of image manipulation, we require the deterministic process both in the forward and reverse direction with pretrained diffusion models for successful image manipulation. On the other hand, for the image translation between unseen domains, stochastic sampling by DDPM is often helpful, which will be discussed in more detail later.

For the full inversion, we adopt deterministic reverse DDIM process [14, 37] as generative process and ODE approximation of its reversal as a forward diffusion process. Specifically, the deterministic forward DDIM process to obtain latent is represented as:

$$\boldsymbol{x}_{t+1} = \sqrt{\alpha_{t+1}} \boldsymbol{f}_\theta(\boldsymbol{x}_t, t) + \sqrt{1 - \alpha_{t+1}} \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) \tag{12}$$

and the deterministic reverse DDIM process to generate sample from the obtained latent becomes:

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \boldsymbol{f}_\theta(\boldsymbol{x}_t, t) + \sqrt{1 - \alpha_{t-1}} \boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) \tag{13}$$

where $\boldsymbol{f}_\theta$ is defined in Eq. 6. For the derivations of ODE approximation, see Supplementary Sec A.

Another important contribution of DiffusionCLIP is a fast sampling strategy. Specifically, instead of performing forward diffusion until the last time step T, we found that we can accelerate the forward diffusion by performing up to $t_0 < T$, which we call 'return step'. We can further accelerate training by using fewer discretization steps between $[1, t_0]$, denoted as S_{for} and S_{gen} for forward diffusion and generative process, respectively [37]. Through qualitative and quantitative analyses, we found the optimal groups of hyperparameters for t_0, S_{for} and S_{gen}. For example, when T is set to 1000 as a common choice [14, 18, 37], the choices of $t_0 \in [300, 600]$ and $(S_{\text{for}}, S_{\text{gen}}) = (40, 6)$ satisfy our goal. Although $S_{\text{gen}} = 6$ may give imperfect reconstruction, we found that the identity of the object that is required for training is preserved sufficiently. We will show the results of quantitative and qualitative analyses on $S_{\text{for}}, S_{\text{gen}}$ and t_0 later through experiments and Supplementary Section F.

Lastly, if several latents have been precomputed (grey square region in Fig. 2), we can further reduce the time for fine-tuning by recycling the latent to synthesize other attributes. With these settings, the fine-tuning is finished in 1∼7 minutes on NVIDIA Quardro RTX 6000.

3.3. Image Translation between Unseen Domains

The fine-tuned models through DiffusionCLIP can be leveraged to perform the additional novel image manipulation tasks as shown in Fig. 4.

First, we can perform image translation from an unseen domain to another unseen domain, and stroke-conditioned image synthesis in an unseen domain as described in Fig. 4(b) and (c), respectively. A key idea to address this difficult problem is to bridge between two domains by inserting the diffusion models trained on the dataset that is relatively easy to collect. Specifically, in [8, 25], it was found that with pretrained diffusion models, images trained from the unseen domain can be translated into the images in the trained domain. By combining this method with DiffsuionCLIP, we can now translate the images in zero-shot settings for both source and target domains. Specifically, the images in the source unseen domain $\boldsymbol{x}_0$ are first perturbed through the forward DDPM process in Eq. 1 until enough time step t_0 when

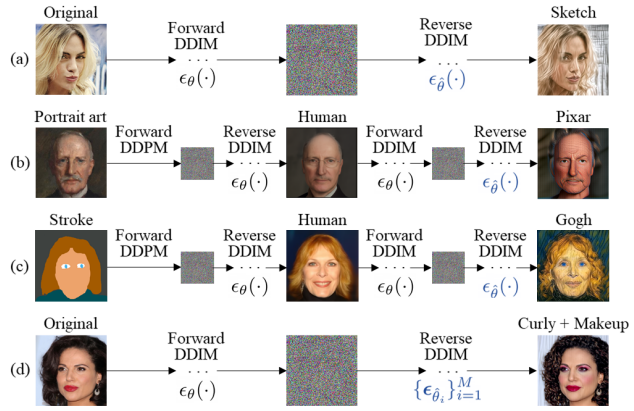

Figure 4. Novel applications of DiffusionCLIP. (a) Manipulation of images in pretrained domain to CLIP-guided domain. (b) Image translation between unseen domains. (c) Stroke-conditioned image generation in an unseen domain. (d) Multi-attribute transfer. ϵ_θ and $\epsilon_{\hat{\theta}}$ indicate the original pretrained and fine-tuned diffusion models, respectively.

the domain-related component are blurred but the identity or semantics of object is preserved. This is usually set to 500. Next, the images in the pretrained domain x'_0 are sampled with the original pretrained model ϵ_θ using reverse DDIM process in Eq. 13. Then, x'_0 is manipulated into the image $\hat{x_0}$ in the CLIP-guided unseen domain as we do in Fig. 4(a) with the fine-tuned model $\epsilon_{\hat{\theta}}$.

3.4. Noise Combination

Multi-attribute transfer. We discover that when the noises predicted from multiple fine-tuned models $\{\epsilon_{\hat{\theta}_i}\}_{i=1}^M$ are combined during the sampling, multiple attributes can be changed through only one sampling process as described in Fig. 4(d). Therefore, we can flexibly mix several single attribute fine-tuned models with different combinations without having to fine-tune new models with target texts that define multiple attributes. In detail, we first invert the image with the original pretrained diffusion model and use the multiple diffusion models by the following sampling rule:

$$\begin{aligned} x_{t-1} =& \sqrt{\alpha_{t-1}} \textstyle\sum_{i=1}^M \gamma_i(t) f_{\hat{\theta}_i}(x_t, t) \\ &+ \sqrt{1-\alpha_{t-1}} \textstyle\sum_{i=1}^M \gamma_i(t) \epsilon_{\hat{\theta}_i}(x_t, t), \end{aligned} \tag{14}$$

where $\{\gamma_i(t)\}_{t=1}^T$ is the sequence of weights of each fine-tuned model $\epsilon_{\hat{\theta}_i}$ satisfying $\sum_{i=1}^M \gamma_i(t) = 1$, which can be used for controlling the degree of each attribute. From Eq. 4, we can interpret this sampling process as increasing the joint probability of conditional distributions as following:

$$\textstyle\sum_{i=1}^M \gamma_i(t) \epsilon_{\hat{\theta}_i}(x_t, t) \propto -\nabla_{x_t} \log \prod_{i=1}^M p_{\hat{\theta}_i}(x_t | y_{\text{tar},i})^{\gamma_i(t)}, \tag{15}$$

where $y_{\text{tar, i}}$ is the target text for each fine-tuned model $\epsilon_{\hat{\theta}_i}$.

In the existing works [9, 10], users require the combination of tricky task-specific loss designs or dataset preparation with large manual effort for the task, while ours enable the task in a natural way without such effort.

Continuous transition. We can also apply the above noise combination method for controlling the degree of change during single attribute manipulation. By mixing the noise from the original pretrained model ϵ_θ and the fine-tuned model $\epsilon_{\hat{\theta}}$ with respect to a degree of change $\gamma \in [0, 1]$, we can perform interpolation between the original image and the manipulated image smoothly.

For more details and pseudo-codes of the aforementioned applications, see Supplementary Section B.

4. Experiments

For all manipulation results by DiffusionCLIP, we use 256^2 size of images. We used the models pretrained on CelebA-HQ [20], AFHQ-Dog [11], LSUN-Bedroom and LSUN-Church [46] datasets for manipulating images of human faces, dogs, bedrooms, and churches, respectively. We use images from the testset of these datasets for the test. To fine-tune diffusion models, we use Adam optmizer with an initial learning rate of 4e-6 which is increased linearly by 1.2 per 50 iterations. We set λ_{L1} and λ_{ID} to 0.3 and 0.3 if used. As mentioned in Section 3.2, we set t_0 in [300, 600] when the total timestep T is 1000. We set $(S_{\text{for}}, S_{\text{gen}}) = (40, 6)$ for training; and to $(200, 40)$ for the test time. Also, we precomputed the latents of 50 real images of size 256^2 in each training set of pretrained dataset. For more detailed hyperparameter settings, see Supplementary Section F.

Table 1. Quantitative comparison for face image reconstruction.

Method	MAE ↓	LPIPS ↓	SSIM ↑
Optimization	0.061	0.126	0.875
pSp	0.079	0.169	0.793
e4e	0.092	0.221	0.742
ReStyle w pSp	0.073	0.145	0.823
ReStyle w e4e	0.089	0.202	0.758
HFGI w e4e	0.062	0.127	0.877
Diffusion ($t_0 = 300$)	**0.020**	**0.073**	**0.914**
Diffusion ($t_0 = 400$)	0.021	0.076	0.910
Diffusion ($t_0 = 500$)	0.022	0.082	0.901
Diffusion ($t_0 = 600$)	0.024	0.087	0.893

Table 2. Human evaluation results of real image manipulation on CelebA-HQ [20]. The reported values mean the preference rate of results from DiffusionCLIP against each method.

vs		StyleGAN-NADA (+ Restyle w pSp)	StyleCLIP (+ e4e)
Hard cases	In-domain	69.85%	69.65%
	Out-of-domain	79.60%	94.60%
	All domains	73.10%	77.97%
General cases	In-domain	58.05%	50.10%
	Out-of-domain	71.03%	88.90%
	All domains	62.47%	63.03%

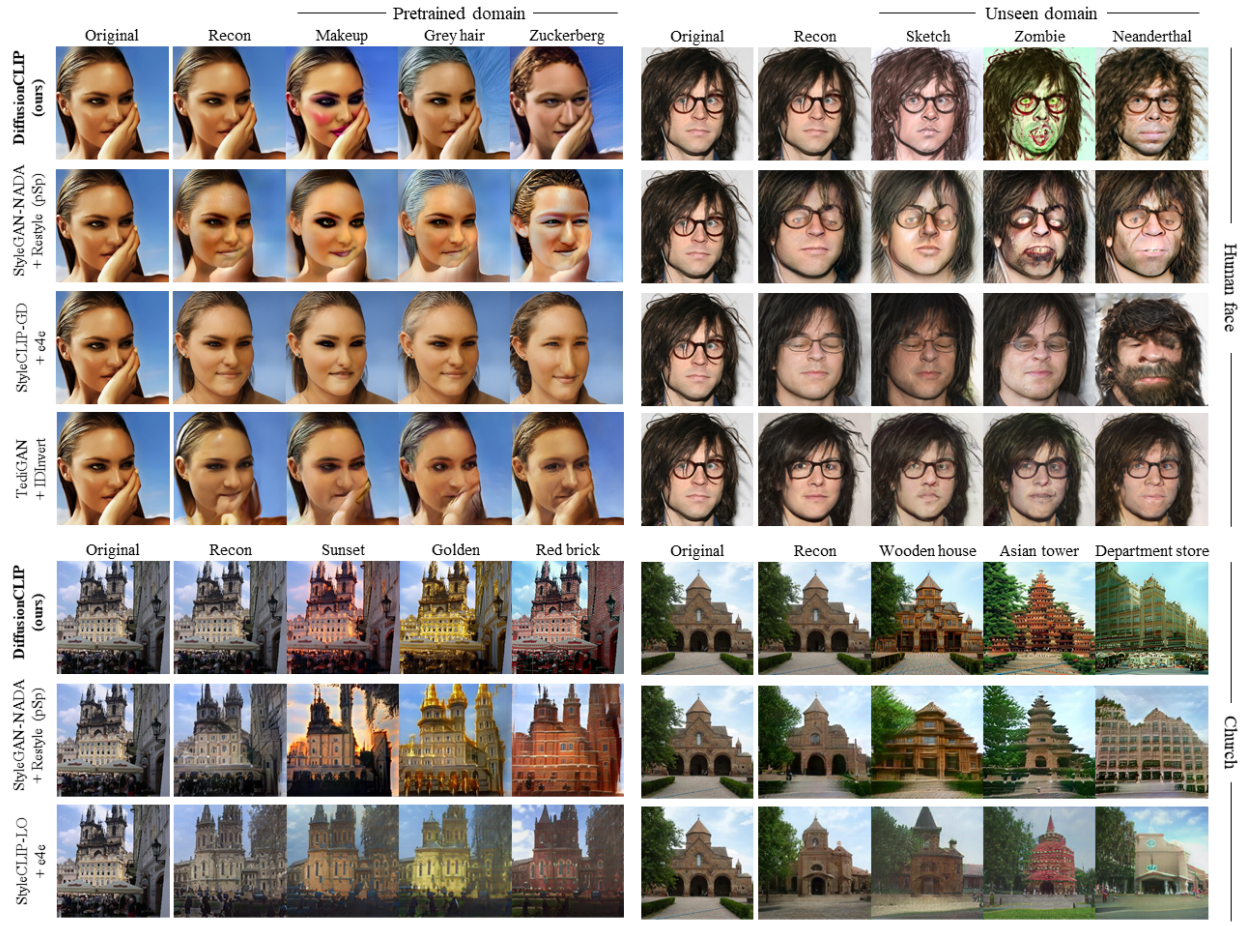

Figure 5. Comparison with the state-of-the-art text-driven manipulation methods: TediGAN [44], StyleCLIP [28] and StyleGAN-NADA [16]. StyleCLIP-LO and StyleCLIP-GD refer to the latent optimization (LO) and global direction (GD) methods of StyleCLIP.

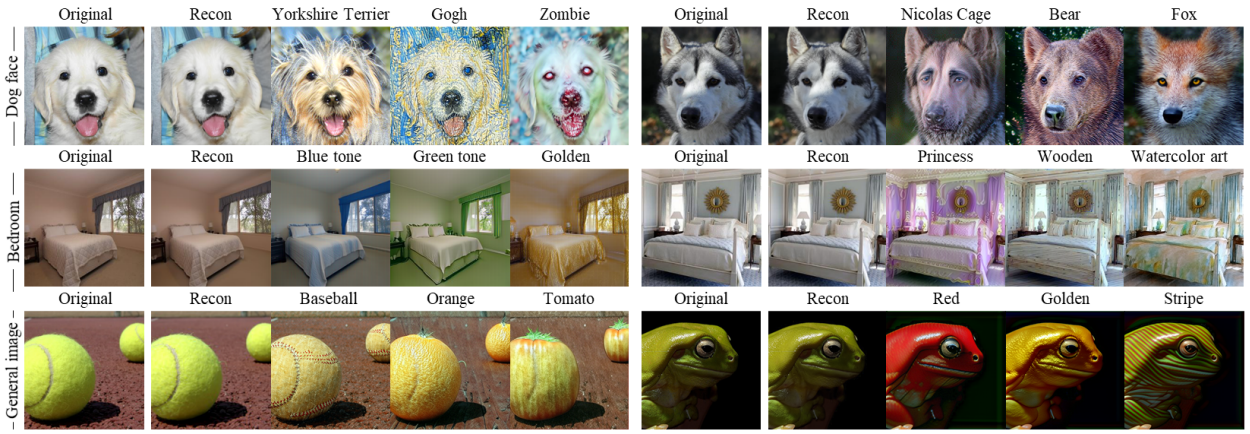

Figure 6. Manipulation results of real dog face, bedroom and general images using DiffusionCLIP.

4.1. Comparison and Evaluation

Reconstruction. To demonstrate the nearly perfect reconstruction performance of our method, we perform the quantitative comparison with SOTA GAN inversion methods, pSp [32], e4e [40], ReStyle [3] and HFGI [43]. As in Tab. 1, our method shows higher reconstruction quality than all baselines in terms of all metrics: MAE, SSIM and LPIPS [47].

Qualitative comparison. For the qualitative comparison of manipulation performance with other methods, we use the state-of-the-art text manipulation methods, TediGAN [44], StyleCLIP [28] and StyleGAN-NADA [16] where images

Table 3. Quantitative evaluation results. Our goal is to achieve the better score in terms of Directional CLIP similarity ($\mathcal{S}_{\text{dir}}$), segmentation-consistency (SC), and face identity similarity (ID).

	CelebA-HQ			LSUN-Church	
	$\mathcal{S}_{\text{dir}}\uparrow$	SC$\uparrow$	ID$\uparrow$	$\mathcal{S}_{\text{dir}}\uparrow$	SC$\uparrow$
StyleCLIP	0.13	86.8%	0.35	0.13	67.9%
StyleGAN-NADA	0.16	89.4%	0.42	0.15	73.2%
DiffusionCLIP (Ours)	**0.17**	**93.7%**	**0.70**	**0.20**	**78.1%**

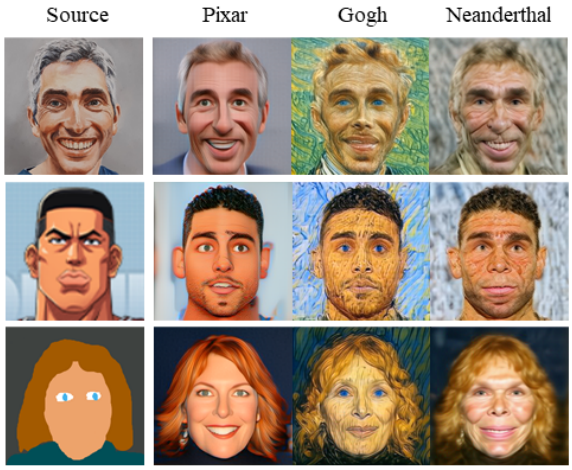

Figure 7. Results of image translation between unseen domains.

Figure 8. Results of multi-attribute transfer.

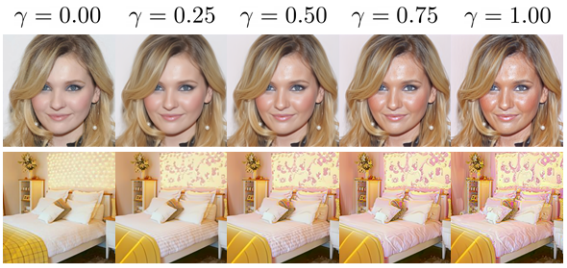

Figure 9. Results of continuous transition.

for the target control is not required similar to our method. StyleGAN2 [22] pretrained on FFHQ-1024 [21] and LSUN-Church-256 [46] is used for StyleCLIP and StyleGAN-NADA. StyleGAN [21] pretrained on FFHQ-256 [21] is used for TediGAN. For GAN inversion, e4e encoder [40] is used for StyleCLIP latent optimization (LO) and global

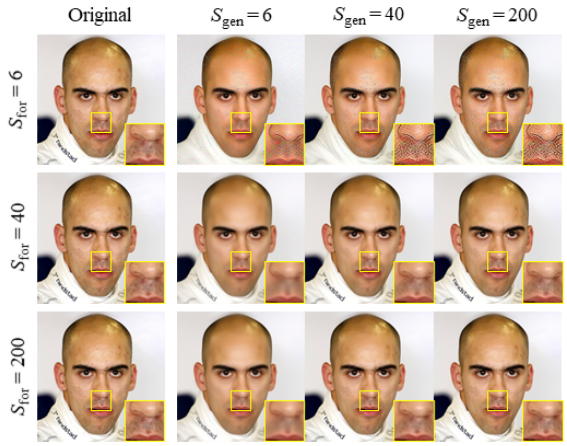

Figure 10. Reconstruction results varying the number of forward diffusion steps S_{for} and generative steps S_{gen}.

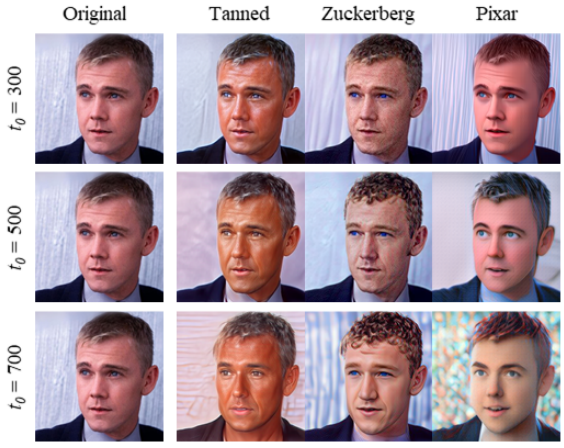

Figure 11. Manipulation results depending on t_0 values.

direction (GD), Restyle encoder [3] with pSp [32] is used for StyleGAN-NADA, and IDInvert [50] is used for TediGAN, as in their original papers. Face alignment algorithm is used for StyleCLIP and StyleGAN-NADA as their official implementations. Our method uses DDPM pretrained on CelebA-HQ-256 [20] and LSUN-Church-256 [46].

As shown in Fig. 5, SOTA GAN inversion methods fail to manipulate face images with novel poses and details producing distorted results. Furthermore, in the case of church images, the manipulation results can be recognized as the results from different buildings. These results imply significant practical limitations. On the contrary, our reconstruction results are almost perfect even with fine details and background, which enables faithful manipulation. In addition to the manipulation in the pretrained domain, DiffusonCLIP can perform the manipulation into the unseen domain successfully, while StyleCLIP and TediGAN fail.

User study. We conduct user study to evaluate real face image manipulation performance on CelebA-HQ [20] with our method, StyleCLIP-GD [28] and StyleGAN-NADA [16].

We get 6000 votes from 50 people using a survey platform. We use the first 20 images in CelebA-HQ testset as general cases and use another 20 images with novel views, hand pose, and fine details as hard cases. For a fair comparison, we use 4 in-domain attributes (angry, makeup, beard, tanned) and 2 out-of-domain attributes (zombie, sketch), which are used in the studies of baselines. Here, we use official pretrained checkpoints and implementation for each approach. As shown in Tab. 2, for both general cases and hard cases, all of the results from DiffusionCLIP are preferred compared to baselines ($> 50\%$). Of note, in hard cases, the preference rates for ours were all increased, demonstrating robust manipulation performance. It is remarkable that the high preference rates ($\approx 90\%$) against StyleCLIP in out-of-domain manipulation results suggest that our method significantly outperforms StyleCLIP in out-of-domain manipulation.

Quantitative evaluation. We also compare the manipulation performance using the following quality metrics: Directional CLIP similarity ($\mathcal{S}_{\text{dir}}$), segmentation-consistency (SC), and face identity similarity (ID). To compute each metric, we use a pretrained CLIP [29], segmentation [45, 48, 49] and face recognition models [13], respectively. Then, during the translation between three attributes in CelebA-HQ (makeup, tanned, gray hair) [20] and LSUN-Church (golden, red brick, sunset) [46], our goal is to achieve the better score in terms of $\mathcal{S}_{\text{dir}}$, SC, and ID. As shown in Tab. 3, our method outperforms baselines in all metrics, demonstrating high attribute-correspondence ($\mathcal{S}_{\text{dir}}$) as well as well-preservation of identities without unintended changes (SC, ID).

For more experimental details and results of the comparison, see Supplementary Section D and E.

4.2. More Manipulation Results on Other Datasets

Fig. 6 presents more examples of image manipulations on dog face, bedroom and general images using the diffusion models pretrained on AFHQ-Dog-256 [11], LSUN-Bedroom-256 [46] and ImageNet-512 [35] datasets, respectively. The results demonstrate that the reconstruction is nearly flawless and high-resolution images can be flexibly manipulated beyond the boundary of the trained domains. Especially, due to the diversity of the images in ImageNet, GAN-based inversion and its manipulation in the latent space of ImageNet show limited performance [5, 12]. DiffusionCLIP enables the zero-shot text-driven manipulation of general images, moving a step forward to the general text-driven manipulation. For more results, see Supplementary Section E.

4.3. Image Translation between Unseen Domains

With the fine-tuned diffusion models using DiffusionCLIP, we can even translate the images in one unseen domain to another unseen domain. Here, we are not required to collect the images in the source and target domains or introduce external models. In Fig. 7, we perform the image translation results from the portrait artworks and animation images to other unseen domains, Pixar, paintings by Gogh and Neanderthal men. We also show the successful image generation in the unseen domains from the stroke which is the rough image painting with several color blocks. These applications will be useful when enough images for both source and target domains are difficult to collect.

4.4. Noise Combination

As shown in Fig. 8 we can change multiple attributes in one sampling. As discussed before, to perform the multi-attribute transfer, complex loss designs, as well as specific data collection with large manual efforts, aren't required. Finally, Fig. 9 shows that we can control the degree of change of single target attributes according to γ by mixing noises from the original model and the fine-tuned model.

4.5. Dependency on Hyperparameters

In Fig. 10, we show the results of the reconstruction performance depending on S_{for}, S_{gen} when $t_0 = 500$. Even with $S_{\text{for}} = 6$, we can see that the reconstruction preserves the identity well. When $S_{\text{for}} = 40$, the result of $S_{\text{gen}} = 6$ lose some high frequency details, but it's not the degree of ruining the training. When $S_{\text{for}} = 200$ and $S_{\text{gen}} = 40$, the reconstruction results are so excellent that we cannot differentiate the reconstruction with the result when the original images. Therefore, we just use $(S_{\text{for}}, S_{\text{gen}}) = (40, 6)$ for the training and $(S_{\text{for}}, S_{\text{gen}}) = (200, 40)$ for the inference.

We also show the results of manipulation by changing t_0 while fixing other parameters in Fig. 11. In case of skin color changes, 300 is enough. However, in case of the changes with severe shape changes such as the Pixar requires stepping back more as $t_0 = 500$ or $t_0 = 700$. Accordingly, we set different t_0 depending on the attributes. The additional analyses on hyperparameters and ablation studies are provided in Supplementary Section F.

5. Discussion and Conclusion

In this paper, we proposed DiffusionCLIP, a method of text-guided image manipulation method using the pretrained diffusion models and CLIP loss. Thanks to the near-perfect inversion property, DiffusionCLIP has shown excellent performance for both in-domain and out-of-domain manipulation by fine-tuning diffusion models. We also presented several novel applications of using fine-tuned models by combining various sampling strategies.

There are limitations and societal risks on DiffusionCLIP. Therefore, we advise users to make use of our method carefully for proper purposes. Further details on limitations and negative social impacts are given in Supplementary Section G and H.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 1

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8296–8305, 2020. 1

[3] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6711–6720, 2021. 1, 2, 6, 7

[4] David Bau, Hendrik Strobelt, William Peebles, Jonas Wulff, Bolei Zhou, Jun-Yan Zhu, and Antonio Torralba. Semantic photo manipulation with a generative image prior. *arXiv preprint arXiv:2005.07727*, 2020. 1

[5] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pages 537–546. PMLR, 2017. 2, 8

[6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 2

[7] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016. 1

[8] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. *arXiv preprint arXiv:2108.02938*, 2021. 4

[9] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018. 5

[10] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020. 5

[11] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 5, 8

[12] Giannis Daras, Augustus Odena, Han Zhang, and Alexandros G Dimakis. Your local gan: Designing two dimensional local attention mechanisms for generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14531–14539, 2020. 2, 8

[13] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 4, 8

[14] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *arXiv preprint arXiv:2105.05233*, 2021. 2, 3, 4

[15] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016. 2

[16] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *arXiv preprint arXiv:2108.00946*, 2021. 2, 3, 6, 7

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 2

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020. 2, 3, 4

[19] Alexia Jolicoeur-Martineau, Rémi Piché-Taillefer, Rémi Tachet des Combes, and Ioannis Mitliagkas. Adversarial score matching and improved sampling for image generation. *arXiv preprint arXiv:2009.05475*, 2020. 2

[20] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 5, 7, 8

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 2, 7

[22] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 2, 7

[23] Diederik P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *arXiv preprint arXiv:1807.03039*, 2018. 2

[24] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[25] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 4

[26] Jacob Menick and Nal Kalchbrenner. Generating high fidelity images with subscale pixel networks and multidimensional upscaling. *arXiv preprint arXiv:1812.01608*, 2018. 2

[27] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *arXiv preprint arXiv:1711.00937*, 2017. 2

[28] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. *arXiv preprint arXiv:2103.17249*, 2021. 2, 3, 6, 7

[29] 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. *arXiv preprint arXiv:2103.00020*, 2021. 2, 3, 8

[30] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *Advances in neural information processing systems*, pages 14866–14876, 2019. 2

[31] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015. 2

[32] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2287–2296, 2021. 1, 2, 6, 7

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 4

[34] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science, 1985. 4

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 2, 8

[36] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 2

[37] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3, 4

[38] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*, 2019. 2

[39] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[40] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 1, 2, 6, 7

[41] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International Conference on Machine Learning*, pages 1747–1756. PMLR, 2016. 2

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 4

[43] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity gan inversion for image attribute editing. *arXiv preprint arXiv:2109.06590*, 2021. 6

[44] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2256–2265, 2021. 6

[45] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018. 8

[46] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 2, 5, 7, 8

[47] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6

[48] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 8

[49] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal on Computer Vision*, 2018. 8

[50] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain gan inversion for real image editing. In *European conference on computer vision*, pages 592–608. Springer, 2020. 7