DIFFUSIONCLIP: TEXT-GUIDED IMAGE MANIPULATION USING DIFFUSION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models are recent generative models that have shown great success in image generation with the state-of-the-art performance. However, only a few researches have been conducted for image manipulation with diffusion models. Here, we present a novel DiffusionCLIP which performs text-driven image manipulation with diffusion models using Contrastive Language–Image Pre-training (CLIP) loss. Our method has a performance comparable to that of the modern GAN-based image processing methods for in and out-of-domain image processing tasks, with the advantage of almost perfect inversion even without additional encoders or optimization. Furthermore, our method can be easily used for various novel applications, enabling image translation from an unseen domain to another unseen domain or stroke-conditioned image generation in an unseen domain, etc. Finally, we present a novel multiple attribute control with DiffusionCLIP by combining multiple fine-tuned diffusion models.

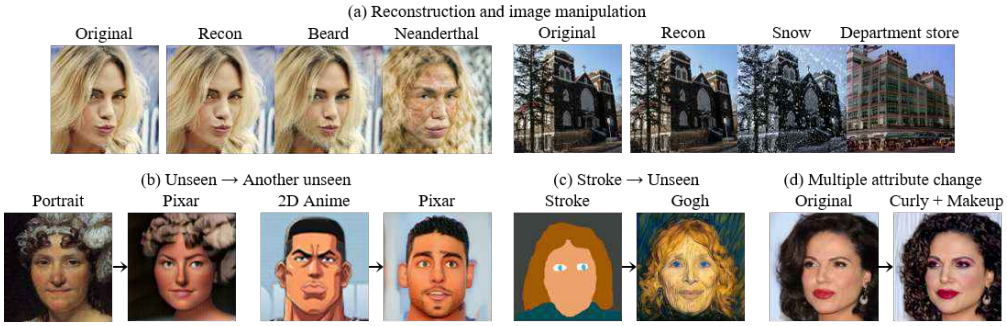


Figure 1: Examples of text-guided image manipulation using DiffusionCLIP.

1 INTRODUCTION

Diffusion models such as denoising diffusion probabilistic models (DDPM) (Ho et al., 2020; Sohl-Dickstein et al., 2015) and score-based generative models (Song & Ermon, 2019; Song et al., 2020b) have shown great success in image generation (Song & Ermon, 2019; Ho et al., 2020; Song et al., 2020a;b; Jolicoeur-Martineau et al., 2020; Dhariwal & Nichol, 2021). The latest works (Song et al., 2020b; Dhariwal & Nichol, 2021) have demonstrated comparable or even superior image sampling performance compared with the current state-of-the-art generative adversarial networks (GANs) (Goodfellow et al., 2014), with additional advantages of great mode coverage and stable training.

Recently, a few studies (Choi et al., 2021; Meng et al., 2021) have been carried out for image manipulation with diffusion models, such as local editing (Meng et al., 2021) and the image translation from unseen domain to the trained domain (Choi et al., 2021). However, it is not clear how these methods can be extended for more general image manipulation applications, such as attribute manipulation (e.g. facial expression change), the image translation from trained domain to multiple unseen domains (e.g. human photo to sketch/Pixar character), etc.

Image manipulation methods for such general tasks have been mainly investigated using GAN (Zhu et al., 2017; Isola et al., 2017; Chen et al., 2018; Goetschalckx et al., 2019; Shen et al., 2020; Bau et al., 2020; Patashnik et al., 2021; Gal et al., 2021). Moreover, the combination of GAN inversion (Zhu et al., 2016; Brock et al., 2016) with contrastive language-image pre-training (CLIP) models (Radford et al., 2021) has recently achieved in- and out-of-domain manipulation of images given simple text prompts without extra manual labors (Patashnik et al., 2021; Gal et al., 2021). Unfortunately, due to difficulty of perfect GAN inversion rooted in limited model capacity and optimization difficulty, the use of the GAN inversion for latent extaction and further image manipulation is still limited.

To address this, here we propose a novel DiffusionCLIP - a CLIP-based text-guide image manipulation method for diffusion models. DiffuionCLIP leverages the sampling and inversion processes based on denoising diffusion implicit models (DDIM) sampling (Song et al., 2020a) and its reversal, which not only accelerate the manipulation but also enable nearly perfect inversion. Accordingly, we can reduce the gap of the scope of applications between diffusion models and GANs. For example, DiffusonCLIP can performs manipulation of images both in trained domain and to unseen domain successfully as Fig. 1(a). We can even translate the image from unseen domain into another unseen domain as illustrated in Fig. 1(b), or generates images in an unseen domain from the strokes as shown Fig. 1(c). Furthermore, by simply combining the noise predicted from several fine-tuned models, multiple attributes can be changed simultaneously through only one sampling process as shown in Fig. 1(d).

Through extensive experiments, we demonstrate that our models show comparable manipulation performance to GAN-based manipulation, having comparative advantage of nearly perfect inversion without additional encoder or optimization.

2 Related Works

2.1 Diffusion models

Diffusion probabilistic models are latent variable models that are composed of two processes: the forward process and the reverse 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 $q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) := \mathcal{N}(\boldsymbol{x}_t; \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \mathbf{I})$ where $\beta_0, \ldots, \beta_T$ is a fixed or learned variance schedule. Then the 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 $p_\theta(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t) := \mathcal{N}(\boldsymbol{x}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t),\ \sigma_\theta(\boldsymbol{x}_t, t)\mathbf{I})$, where $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t)$ can be decomposed into the linear combination of $\boldsymbol{x}_t$ and a noise approximation model $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$. Under these parametrization, the reverse process can be trained by solving the following optimization problem:

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

After training $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}, t)$, the data is sampled using following sampling rule.:

$$\boldsymbol{x}_t = \frac{1}{\sqrt{1-\beta_t}}\left(\boldsymbol{x}_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right) + \sigma_t \boldsymbol{z}, \ \text{ where } \boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{3}$$

It was found that the sampling process of DDPM corresponds to that of the score-based generative models (Song & Ermon, 2019; Song et al., 2020b) if the score function can be represented as a rescaled version of $\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)$ as follows:

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

Specifically, Song et al. (Song & Ermon, 2019; Song et al., 2020b) proposed an alternative non-Markovian noising process that has the same forward marginals as DDPM and corresponding 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 $\boldsymbol{f}_\theta(\boldsymbol{x}_t, t)$ 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 to use different reverse samplers by changing the variance of the reverse noise σ_t. Especially, by setting this noise to 0, which is a DDIM sampling process (Song et al., 2020a), the sampling process become deterministic, enabling to convert latent variables into the data consistently and to sample with fewer steps. In fact, DDIM can be considered as a Euler method to solve ODE by rewriting (5) as follows:

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

2.2 CLIP GUIDANCE FOR TEXT-DRIVEN IMAGE MANIPULATION

CLIP (Radford et al., 2021) was proposed to efficiently learn visual concepts from natural language supervision. Specifically, CLIP pre-trains an image encoder and a text encoder to predict which images were paired with which texts in the dataset. Accordingly, it can be applied to any visual classification benchmark by simply providing the names of the visual categories to be recognized.

As the source of guidance for our target manipulation, we use a pre-trained CLIP model. In order to effectively extract knowledge from CLIP, two different losses are proposed in (Patashnik et al., 2021; Gal et al., 2021): a global target loss, and a local directional loss. The global CLIP loss (Patashnik et al., 2021) tries to to minimize the cosine distance in CLIP space between the generated images and a given target text as follows:

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

where t_{tar} is the description in text of some target control, $\boldsymbol{x}_{\text{gen}}$ denotes the generated image, and D_{CLIP} returns the cosine distance in CLIP-space between them. On the other hand, the local directional loss (Gal et al., 2021) is designed in order to alleviate the issues of global CLIP loss such as low diversity and susceptibility to adversarial solutions. The local directional CLIP loss induces the direction between the encoded vectors of the reference and generated images to be aligned with the direction between the encoded vectors of a pair of reference and target texts in CLIP space as follows:

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

where

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

where E_I and E_T are CLIP's image and text encoders, respectively, and $t_{\text{ref}}, \boldsymbol{x}_{\text{ref}}$ are the source domain text and image, respectively. The manipulated images guided by the directional CLIP loss is 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 (Radford et al., 2021).

3 DIFFUSIONCLIP

The overall flow of DiffusionCLIP for image manipulation is shown in Fig. 2. Here, the input image $\boldsymbol{x}_0$ is first converted to the latent $\boldsymbol{x}_l$ via forward diffusion. Then, guided by the CLIP loss, the diffusion model is fine-tuned, and the updated sample is generated from the fine-tune diffusion model. In terms of diffusion model fine-tuning, one could modify the latent or the diffusion model. In this work, we found that direct model fine-tuning is more effective, as will be shown later in experiments.

To fine-tune the diffusion model $\boldsymbol{\epsilon}_\theta$, we use the following objective composed of CLIP loss and the identity loss:

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

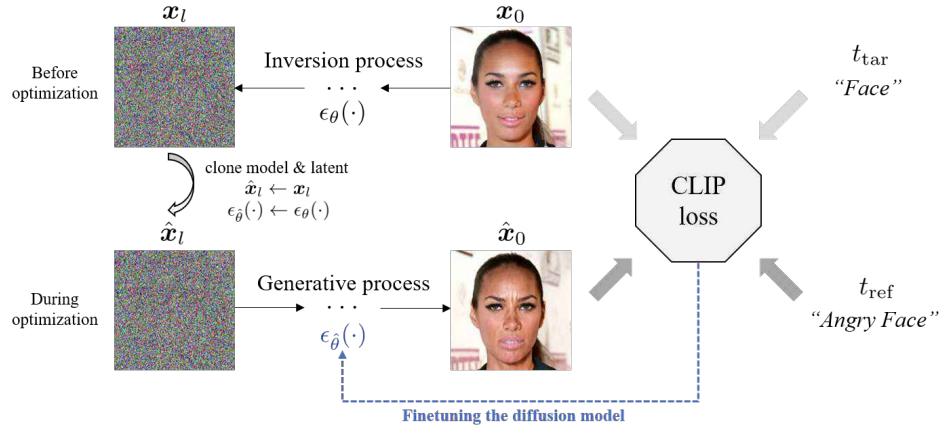


Figure 2: Overview of DiffusionCLIP. The input image is first converted to the latent via diffusion. Then, guided by directional CLIP loss, the diffusion model is fine-tuned, and the updated sample is generated from the fine-tuned diffusion model.

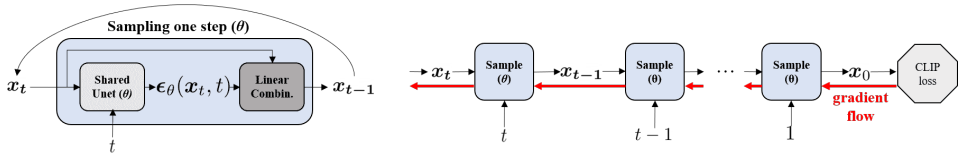


Figure 3: Fine-tuning the general diffusion models with the shared architecture.

where $\boldsymbol{x}_0$ is the original image, $\hat{\boldsymbol{x}}_0(\theta)$ is the manipulated image with the optimized parameter θ, t_{ref} is the reference text, t_{tar} is the target text to manipulate.

Here, CLIP loss is the key component to supervise the optimization for the target control. Of two types of two types of CLIP losses as discussed above, we employ directional CLIP loss as a guidance owing to the appealing properties as mentioned in Section 2.2. For the text prompt, directional CLIP loss requires a reference text t_{ref} and a target text t_{tar} while training. For example, in case of changing the expression of given face image into angry expression, we can use 'face' as a reference text and 'angry face' as a target text. In this paper, we often use the concise word to refer each text prompt (e.g. 'tanned face' to 'tanned'). The identity loss $\mathcal{L}_{\text{id}}$ is employed to prevent from the unwanted changes and preserve the identity of the object. We generally use ℓ_1 loss as identity loss, and in case of the human face image manipulation, face identity loss proposed in (Deng et al., 2019) is added:

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

where $\mathcal{L}_{\text{face}}$ is the face identity loss (Deng et al., 2019). and $\lambda_{\text{L1}} > 0$ and $\lambda_{\text{face}} > 0$ 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, the species of Animal).

Existing diffusion models (Ho et al., 2020; Song et al., 2020a; Dhariwal & Nichol, 2021) adopt the shared Unet architecture for all t, by inserting the information of t using sinusoidal position embedding as used in the Transformer (Vaswani et al., 2017). With this shared architecture, the gradient flow during DiffusionCLIP training can be represented as Fig. 3, which is a similar process of training recursive neural network (Rumelhart et al., 1985).

3.1 Inversion and Generative Process

As DDPM sampling process is stochastic, the samples generated from the same latent will be different every time. This will hinder optimization by confusing the changes from the stochastic of the process or our optimization. Even if the sampling process is the deterministic, the forward process of DDPM,

where the image is noised randomly with different noise scale, is also stochastic, so 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 there require the deterministic process both in forward and reverse direction with pretrained diffusion models for the successful image manipulation.

Accordingly, we adopt DDIM deterministic generative process of (Song et al., 2020a) and ODE approximation of its reverse process as inversion process. The authors in (Dhariwal & Nichol, 2021) founded that through these inversion and generative process, nearly perfect reconstruction can be achieved with enough number of iterations. Specifically, the inversion process 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 generative process 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 (6).

To accelerate training, we found that the image is not required to be inverted until the last steps T, so we invert the image into $t_0 \in [0, T]$, which we call 'return step'. We can further accelerate training by using smaller steps of the inversion and generative process, denoted as S_{inv} and S_{gen}, respectively, both in $[0, t_0]$. We tried to find the optimal t_0, S_{inv} and S_{gen} satisfying both of good quality of image manipulation and the fast training. We discovered that when T is set to 1000 as common choice (Ho et al., 2020; Song et al., 2020a; Dhariwal & Nichol, 2021), we can set t_0 in [300, 600], and choose S_{inv} and S_{gen} to 40, 6, respectively. Although this may give imperfect reconstruction, the identity of the object that is required for training is preserved sufficiently. We show the results of hyperparameter studies for S_{inv}, S_{gen} and t_0 in Section 4.3. With these settings, the fine-tuning is finished in $1 \sim 8$ minuites on NVIDIA Quardro RTX 6000.

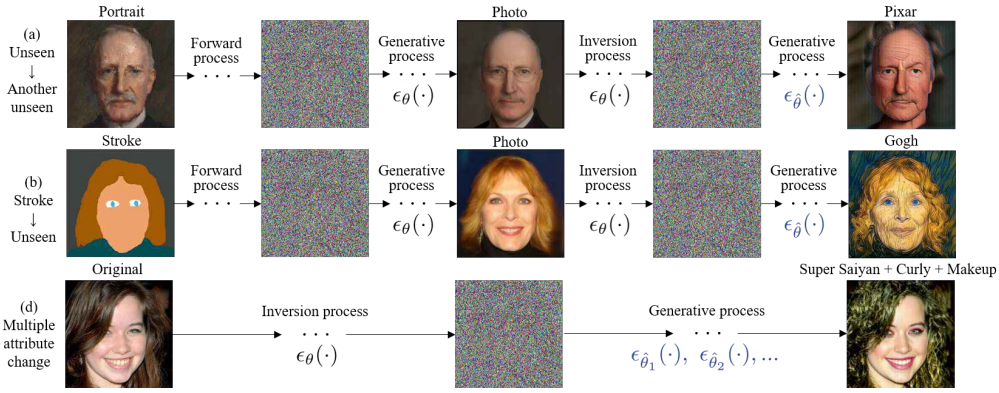


Figure 4: Novel applications of DiffusionCLIP.

3.2 Novel Manipulation Applications

The fine-tuned models through DiffusionCLIP can be leveraged to carry out the several novel applications. With existing works, users often need the combination of multiple models, tricky task-specific loss designs or dataset preparation with large manual effort. On the other hand, our method is free of such effort and enables applications in a natural way with the original pretrained and fine-tuned models by DiffusionCLIP. The specific applications of our DiffusionCLIP are as follows.

Image translation into unseen domains First, we can translate the images from one domain into another domain without seeing any images in both domains during the training of the models. This application may be useful if it is hard to collect enough images to train with in both domains. Our insight to solve this tough problem is to bridge between two other domains by inserting the diffusion models trained on the dataset that is relatively easy to be collected. The researches in (Choi et al., 2021; Meng et al., 2021) found that with pretrained diffusion models, images trained from unseen

domain can be translated into the images in the trained domain. By combining this method with DiffsuionCLIP where the images in trained domain can be translated into the unseen domain, we can now translate the images from unseen domain to another unseen domain. Two representative image translation, storke-based image generation in the unseen domain and portrait to Pixar characters are described in Fig. 4(a)(b). In detail, the images in the source unseen domain (e.g. a stroke or portrait) are first perturbed through the stochastic forward process of DDPM until enough time steps when the domain-related component are blurred but the identity or semantics of object is preserved. Next, the images in the trained domain (e.g. a photo of human face) is sampled with the original pretrained model $\boldsymbol{\epsilon}_\theta$. Then, the images are converted to the latent which can be reconstructed through the ODE inversion process. Finally, the images in the target unseen domain (e.g. Gogh painting, Pixar character) are generated through the generative process with the fine-tuned models $\boldsymbol{\epsilon}_{\hat{\theta}}$.

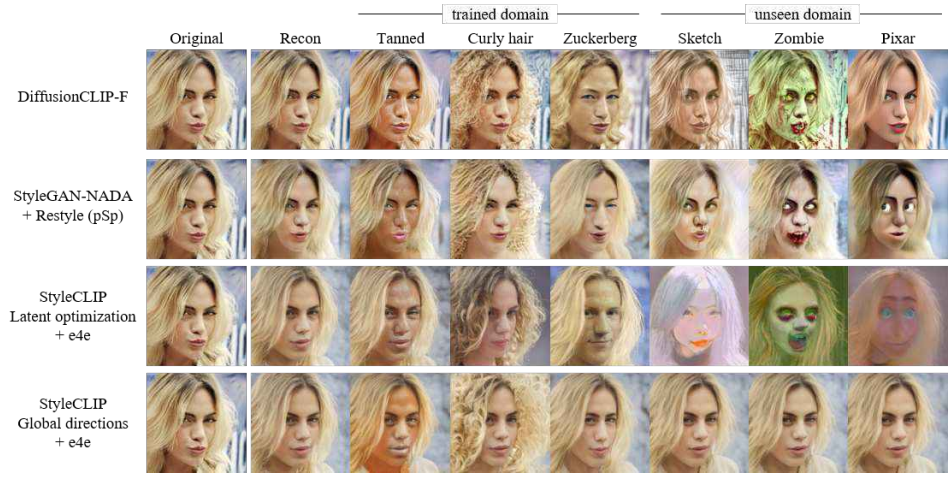


Figure 5: Comparison of the manipulation methods for a variety of text prompt. We use DDPM models pretrained on CelebA-HQ 256×256 for DiffusonCLIP. We follow the official codes of StyleCLIP and StyleGAN-NADA that use StyleGAN-ADA pretrained on FFHQ 1024×1024.

Multiple attribute change by noise combination. We discover that by combining the noise predicted from multiple (say, M) fine-tuned models during the sampling, multiple attributes can be changed through only one sampling process without new training. As described in Fig. 4(c), we first invert the image using normal image manipulation with the original pretrained diffusion model and use the multiple diffusion models fine-tuned for the different controls. In specific, we can use the following sampling rule.

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \sum_{i=1}^{M} \gamma_i \boldsymbol{f}_{\theta_i}(\boldsymbol{x}_t, t) + \sqrt{1-\alpha_{t-1}} \sum_{i=1}^{M} \gamma_i \boldsymbol{\epsilon}_{\theta_i}(\boldsymbol{x}_t, t) \tag{14}$$

where γ_i is the time dependent weights of the model satisfying $\sum_{i=1}^{M} \gamma_i = 1$, which can be used for controlling the degree of each change. We found that several main changes take place at different steps depending on the types of controls. Therefore, by applying different weights of the models at different step, we can change multiple attributes more successfully.

4 Experimental Results

For all manipulation results by DiffusionCLIP, we use 256×256 size of images. We used the models pretrained on CelebA-HQ (Karras et al., 2017), AFHQ-Dog (Choi et al., 2020), LSUN-Bedroom, LSUN-Church (Yu et al., 2015) datasets for manipulating images of human faces, dogs, churches and bedrooms, respectively. For Celeba-HQ, LSUN-Church and LSUN-bedroom models, we used the pretrained models in (Meng et al., 2021) and for AFHQ-Dog, we used models in (Choi et al., 2021).

For all experiments of DiffusionCLIP, we use Adam optmizer with an initial learning rate of 1e-6 which is increased linearly by 1.2 per 50 iterations normally. We set the weights of ℓ_1 loss and

the face identity loss to 0.3 and 0.1 if used. We precomputed the latents of 50 real images of size 256×256 through the inversion process and use them for fine-tuning the models.

As mentioned in Section 3.1, we set t_0 in [300, 600] when total timestep T is 1000 and we set S_{inv} and S_{gen} to 40 and 6, respectively, for training; and $S_{\text{gen}} = 40, S_{\text{inv}} = 40$ for the test time.

4.1 Comparison with other CLIP-guided manipulation methods

For the comparison of DiffusionCLIP with other methods, we use the state-of-the-art CLIP guided text manipulation methods, StyleCLIP (Patashnik et al., 2021) and StyleGAN-NADA (Gal et al., 2021) where the manual labors for the target control is not required as ours. In StyleCLIP, the latent of the pretrained StyleGAN is manipulated using the CLIP models with 3 approaches: latent optimization, latent mapper, and global directions. Of those models, we use latent optimization and global directions for the comparison. In StyleGAN-NADA, the pretrained StyleGAN is fine-tuned guided by CLIP loss. We use the official pretrained models trained on 1024×1024 images in FFHQ (Karras et al., 2019) for both models. For GAN inversion of two methods, e4e (Tov et al., 2021),

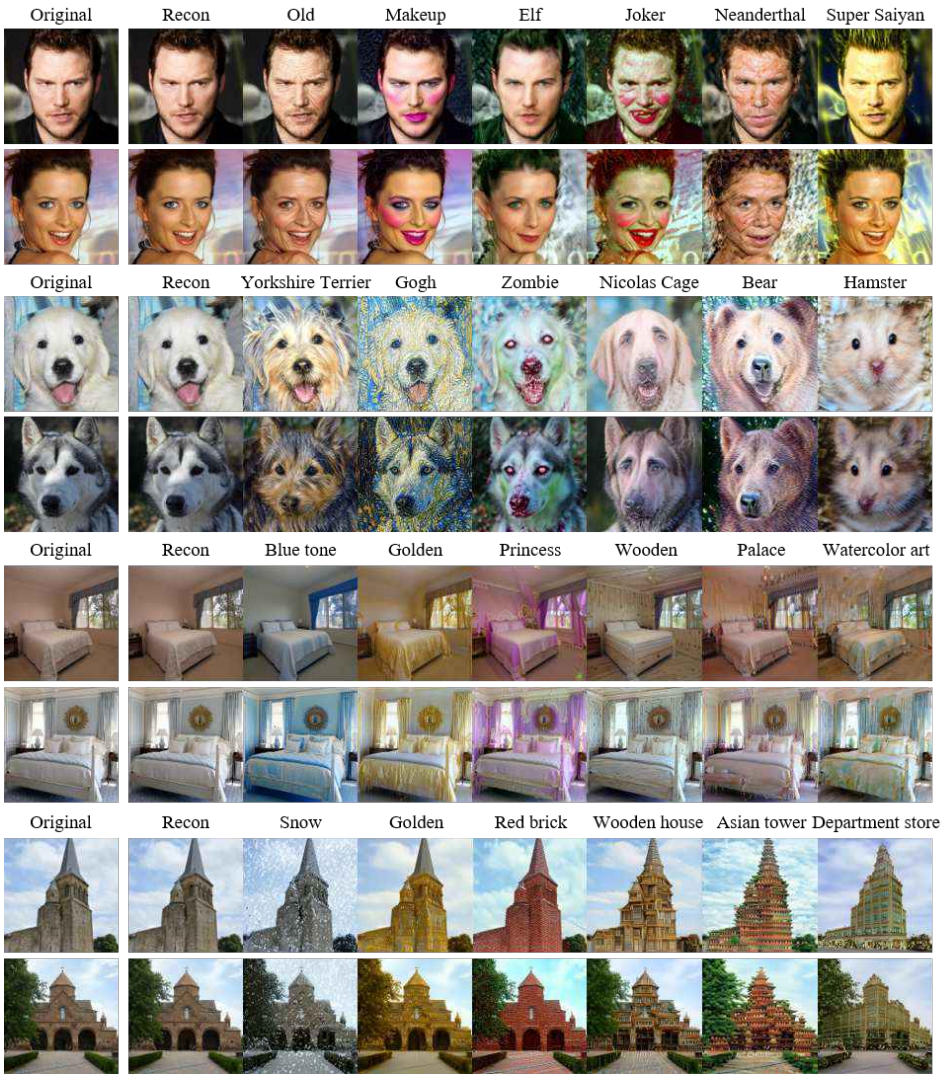


Figure 6: More results of manipulation using DiffusionCLIP where the original pretrained models are trained on CelebA-HQ, AFHQ-Dog, LSUN-Bedroom and LSUN-Church, respectively. The results demonstrate various text-guided manipulation over the trained domains.

and Restyle (Alaluf et al., 2021) with pSp (Richardson et al., 2021) inversion methods are used for StyleCLIP and StyleGAN-NADA, respectively, as in the original papers.

In the second column of Fig. 5, we can see that the inversion method used in StyleCLIP and StyleGAN-NADA fails to reconstruct the original images from the inverted latent, implying the significant practical limitations. On the contrary, our method inverts the image almost perfectly into the latent without any encoder or additional optimization. From the inversion, we can see that DiffusonCLIP is comparable to the existing methods for in-domain manipulation, and to StyleGAN-NADA for out-of-domain manipulation.

4.2 More results of DiffusionCLIP

In-domain and out-of-domain mage manipulations on various datasets. We show more examples of image manipulations by DiffusionCLIP using the diffusion models trained on three more dataset: CelebA-HQ, AFHQ-Dog, LSUN-Bedroom, LSUN-Church. The results in Fig. 6 demonstrate that the reconstruction is nearly flawless and images can be flexibly manipulated beyond the boundary of the trained domains.

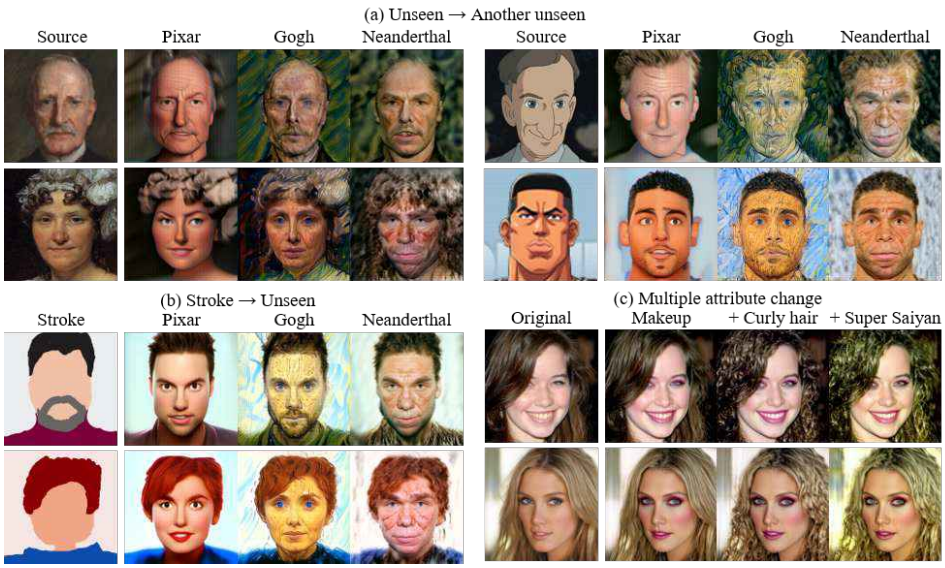


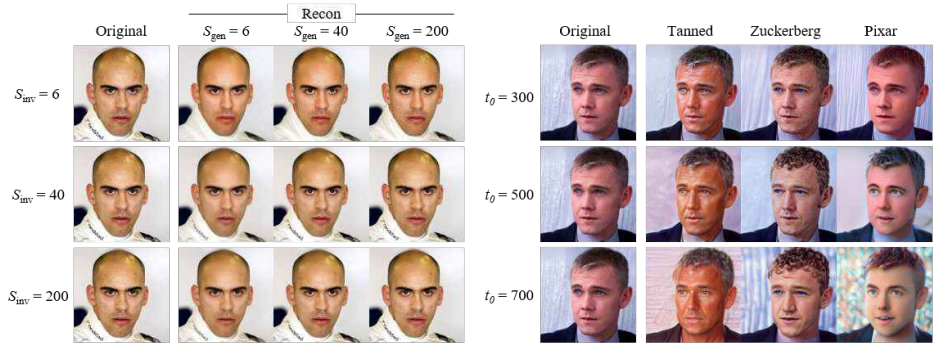
Figure 7: Examples of the novel applications of fine-tuned models using DiffusionCLIP.

Image translation into 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 Figs. 7(a)(b), we show the image translation results from the portrait artworks and animation images to other unseen domains, Pixar characters, paintings by Gogh and Neanderthal men. Next, we 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 the enough images for both source and target domains are hard to be collected.

Multiple attribute change by noise combination. Finally, as shown in Fig. 7(c), we can change multiple attributes in one sampling. We show the multiple manipulations of hair style, makeup and Super Saiyan are conducted successfully.

4.3 Ablation Study

Dependency on hyperparameters. We show the results of the reconstruction performance depending on S_{inv}, S_{gen} in Fig. 8(a) when $t_0 = 600$. Even with $S_{\text{inv}} = 6$, we can see that the reconstuction seems to preserve identity well and it can be used for training. However, there are minute artifacts.



(a) varying the number of inversion steps S_{inv} and sampling steps S_{gen}

(b) varying the return point t_0

Figure 8: Comparison with respect to various hyperparameters.

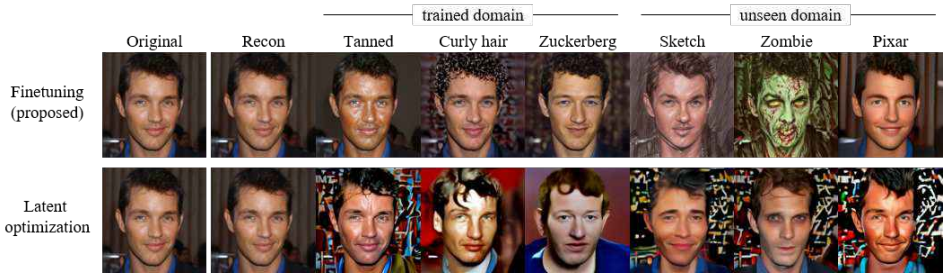


Figure 9: Comparison between fine-tuning and latent optimization.

When $S_{\text{inv}} = 40$, the result of $S_{\text{gen}} = 6$ looks excellent but lose some high frequency details, but it's not the degree of ruining the training. When $S_{\text{gen}} = 40$, the reconstruction seems excellent as we cannot differentiate the reconstruction with the result when $S_{\text{gen}} = 200$ and the original images. Therefore, we just use ($S_{\text{inv}} = 40$, $S_{\text{gen}} = 6$) for the training and ($S_{\text{inv}} = 40$,$S_{\text{gen}} = 40$) in the inference time. We also show two results of manipulation by changing t_0 while fixing other parameters. 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 to change. Accordingly, we set different t_0 depending on the attributes.

Comparison between Latent Optimization and Model Fine-tuning Instead of fine-tuning the diffusion model, one may consider latent manipulation. In Fig. 9, we display a series of the real image manipulation given the text prompt by our fine-tuning method and latent optimization. We can see that the manipulation results via latent optimization method failed to manipulate the images to the unseen domain. This may be because the manipulation using latent optimization is restricted by the learned distribution of the pretrained model. On the other hand, the proposed model fine-tuning method shows superior manipulation performance.

5 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 the 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 diffusion models.

BIBLIOGRAPHY

Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. *arXiv preprint arXiv:2104.02699*, 2021.

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.

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

Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3291–3300, 2018.

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.

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.

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*, pp. 4690–4699, 2019.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *arXiv preprint arXiv:2105.05233*, 2021.

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.

Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola. Ganalyze: Toward visual definitions of cognitive image properties. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5744–5753, 2019.

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.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

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.

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.

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*, pp. 4401–4410, 2019.

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.

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.

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.

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*, pp. 2287–2296, 2021.

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.

Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9243–9252, 2020.

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

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

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

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*, 2020b.

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.

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*, pp. 5998–6008, 2017.

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.

Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *European conference on computer vision*, pp. 597–613. Springer, 2016.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.