

# CtlGAN: Few-shot Artistic Portraits Generation with Contrastive Transfer Learning

Yue Wang<sup>1</sup>, Ran Yi<sup>1</sup>, Ying Tai<sup>2</sup>, Chengjie Wang<sup>2</sup>, and Lizhuang Ma<sup>1</sup>

<sup>1</sup> Shanghai Jiao Tong University {imwangyue,ranyi,ma-lz}@sjtu.edu.cn

<sup>2</sup> Tencent YouTu Lab {yingtai,jasoncjwang}@tencent.com

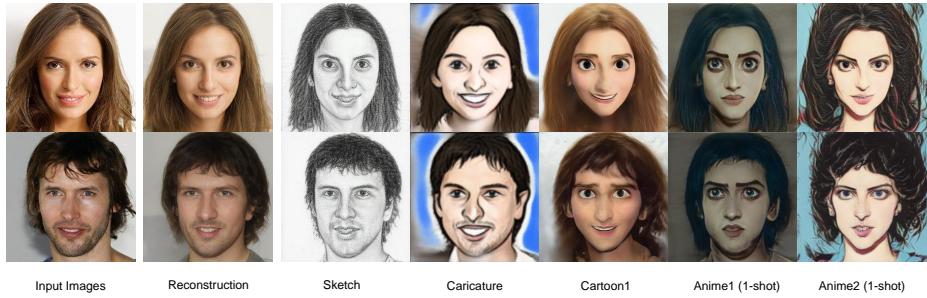


Fig. 1: Our few-shot artistic portraits generation results on different artistic styles (*10-shot or 1-shot*). We eliminate overfitting using a novel contrastive transfer learning strategy. With our style encoder, real face photos are embedded into the latent space shared by our decoders on different artistic domains.

**Abstract.** Generating artistic portraits is a challenging problem in computer vision. Existing portrait stylization models that generate good quality results are based on Image-to-Image Translation and require abundant data from both source and target domains. However, without enough data, these methods would result in overfitting. In this work, we propose CtlGAN, a new few-shot artistic portraits generation model with a novel contrastive transfer learning strategy. We adapt a pretrained StyleGAN in the source domain to a target artistic domain with no more than 10 artistic faces. To reduce overfitting to the few training examples, we introduce a novel Cross-Domain Triplet loss which explicitly encourages the target instances generated from different latent codes to be distinguishable. We propose a new encoder which embeds real faces into  $\mathcal{Z}+$  space and proposes a dual-path training strategy to better cope with the adapted decoder and eliminate the artifacts. Extensive qualitative, quantitative comparisons and a user study show our method significantly outperforms state-of-the-arts under 10-shot and 1-shot settings and generates high quality artistic portraits. The code will be made publicly available.

**Keywords:** Artistic portraits generation, Few-shot domain adaptation, Cross-domain triplet, StyleGAN, StyleGAN inversion

## 1 Introduction

Portrait art is a longstanding art form that captures human facial features in expressive art styles, such as painting, cartoon, sketch, and caricature. However, even for professional artists, it takes hours to paint a good artistic portrait. Developing computer programs to automatically generate artistic portraits can free artists from time-consuming and repeated works, and has the advantage of automatic portraits production with efficiency streamline.

With the development of machine learning, neural style transfer algorithms [11,19,15] are developed to transfer the style of a style exemplar to a content image. However, these methods are unable to stylize portraits well since they tend to deform facial structures. As Generative Adversarial Networks (GANs) gain success in various vision tasks, Image-to-Image Translation (IIT) methods leverage GANs to translate images from a source domain to a target domain by learning from paired [17,44] or unpaired data [54,51]. Based on this, some works [40,18,35,50,55,7] formulate the artistic portraits generation problem as the translation from real faces domain to artistic faces domain, and develop IIT algorithms to learn from a group of artistic faces.

However, these artistic portraits generation algorithms need abundant data, which is often difficult to acquire in real application scenarios. For example, the Artistic-Faces Dataset [49] collects 160 artistic portraits of 16 different artists, only 10 for each artist, while existing methods often need at least 100 training images. Although there are some research on few-shot Image-to-Image Translation [29], they mainly deal with translation between different object classes and few-shot generation for unseen classes, which is different from our problem.

We aim at learning a photo to artistic portrait translation by learning from a few artistic faces (e.g., no more than 10). We observe that humans can learn artistic portraits of a certain style after seeing a small number of artistic samples, since they gain knowledge about faces in daily life, and apply it to portraits painting. Similar to this, with transfer learning, machine applies knowledge gained in one problem to another related problem. Although transfer learning has been explored in image generation with limited data [47,31,46,26], most methods still cannot generate good results when training examples are very few [33]. Recent research [33] studies the image generation given only 10 training examples, by adapting a pretrained GAN to a target image domain via cross-domain correspondence. However, it didn't explicitly enforce the generations of different latent codes to be different, which leads to a certain degree of overfitting (Fig. 2(a) middle). Recently, some one-shot or text-guided methods [10,55] were proposed leveraging the semantic power of CLIP [36], but these methods are worse in identity preservation.

In this paper, we propose CtlGAN, a novel few-shot artistic portraits generation model with a *contrastive transfer learning strategy*. We adapt a pretrained StyleGAN2 [24] on real faces to a target artistic domain with no more than 10 artistic faces. To prevent overfitting to the few training examples, we explicitly enforce the generations of different latent codes to be distinguishable

with a new Cross-Domain Triplet loss. To translate real faces into artistic portraits, we propose a new encoder to *invert* real faces into the StyleGAN2 latent space, which uses  $\mathcal{Z}+$  latent space instead of  $\mathcal{W}+$  and proposes a dual-path training strategy to cope with our decoder. Our CtlGAN automatically generates high quality artistic portraits from real face photos under 10-shot or 1-shot settings (Figs. 1, 7-9, 15-20).

In summary, our main contributions are three-fold:

- We propose CtlGAN, a new model for artistic portraits generation from real face photos under few-shot setting. With no more than 10 training examples, our model generates high-quality artistic portraits for various artistic domains.
- We present a novel contrastive transfer learning strategy that adapts a pretrained StyleGAN2 to a target artistic domain with Cross-Domain Triplet loss, and avoids overfitting to the few training samples.
- We propose a novel style encoder which embeds real photos into  $\mathcal{Z}+$  latent space and proposes a new dual-path training strategy to better cope with the adapted decoder and generate high-quality artistic portraits.

## 2 Related Works

**Generative Adversarial Networks.** GANs [13] achieve great success and are widely used in synthesizing images. Conditional GANs [17] control the network outputs by conditional setting or inputs. Recently, Karras et al. proposed StyleGAN series [23,24,21,22] to improve the image synthesis quality and constructed a high quality face dataset named FFHQ. Due to their high generation quality, StyleGAN series have achieved great success in many face generation tasks [42,3,48]. We utilize StyleGAN2 [24] as the decoder and transfer a pretrained model on FFHQ to a target artistic portraits domain using no more than 10 examples with a novel contrastive transfer strategy.

**GAN Inversion.** GAN inversion is the process of embedding real images into the latent space of GANs. In this paper, we focus on StyleGAN, where a  $\mathcal{Z}$  space latent code is first translated into an intermediate  $\mathcal{W}$  space by a mapping network, and then used to control the generator via AdaIN blocks [15] and output an image. There are two main ways to realize GAN inversion: optimization based methods and learning based methods. Optimization based methods try to optimize the latent code with some specialized loss functions. Some works [24,1,2] find extending  $\mathcal{W}$  space to  $\mathcal{W}+$  space and directly optimizing  $\mathcal{W}+$  space latent code can help improve the reconstruction quality. Optimization based methods have higher quality in reconstructing images, while learning based methods process faster and use fewer computation resources. Richardson et al. proposed pSp [37], an encoder based on a Feature Pyramid Network (FPN) [27] to embed images to  $\mathcal{W}+$  space. Based on pSp architecture, e4e [43] proposed a new encoder for better StyleGAN-based image manipulation, and restyle [4] proposed iterative refinement for higher quality of reconstructed images. Recently, Song

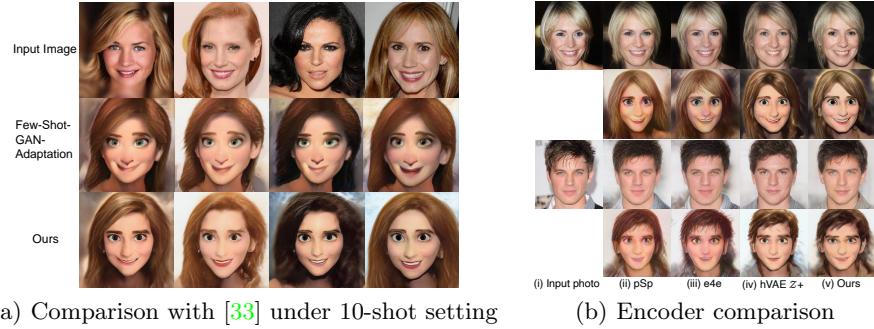


Fig. 2: (a) Few-Shot-GAN-Adaptation [33] results show a certain degree of overfitting (similar faces in the middle row), while ours generates diverse results and well preserves the identity. (b) Results of combining different encoders [37,43,40] with our StyleGAN-based artistic decoder. For each input face, the top row shows reconstruction results, and the second row shows cartoonization results.

et al. proposed a hierarchical Variational Autoencode (hVAE) in AgileGAN [40] to embed face photos into latent codes that follow Gaussian distribution, and extend  $\mathcal{Z}$  space to  $\mathcal{Z}+$ . We adopt a similar  $\mathcal{Z}+$  space setting, but instead of using a VAE, we propose a new encoder with dual-path training strategy, which eases the training and better copes with our adapted decoder.

**Transfer Learning for GANs.** Transfer learning utilizes knowledge gained in solving one problem to solve related problems, and helps the network training with limited data. To help training GANs with limited data, some methods have been proposed to transfer GANs. Transferring GANs (TGANs) [47] adapts a pretrained GAN model to a target domain by fine-tuning the original objective function. BSA [32] only updates the scale and shift parameters in the generator during transfer. FreezeD [31] freezes the lower layers of discriminator during adaptation. MineGAN [46] proposes a miner network to find the knowledge that is most beneficial to a target domain from pretrained GANs. EWC [26] regularizes the weights changes during the adaptation, to best preserve the source “information”. However, these methods fail to generate good results when the training examples are very few [33]. Few-Shot-GAN-Adaptation[33] transfers a pretrained GAN to a target domain with very few training samples, by preserving pairwise similarity before and after adaptation. However, it is prone to overfitting (Fig. 2(a) second row). StyleGAN-NADA [10] uses text to guide the domain adaptation by leveraging the semantic knowledge in the pretrained Contrastive-Language-Image-Pretraining (CLIP) [36] model. Mind-the-gap [55] also leverages the CLIP model, but uses a reference image (instead of text) to guide the domain adaptation. It builds upon StyleGAN-NADA and considers the semantic difference within domain. These two methods utilize external knowledge from CLIP and achieve good adaptation results, but they are weaker in identity preservation. JoJoGAN [7] proposes a style mixing strategy

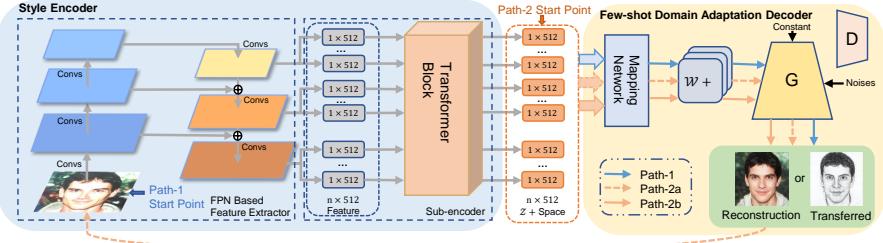


Fig. 3: Our CtlGAN contains two parts, including a Style Encoder, and a Domain Adaptation Decoder which is based on StyleGAN2 [24]. We design a dual path training for our encoder with path-1 shown in blue and path-2 (cycle path) shown in orange.

to generate a large training set of paired face images from a single reference, and then finetunes StyleGAN using pixel loss, but sometimes suffers from artifacts. In contrast, our CtlGAN transfers a pretrained GAN via contrastive transfer learning to address the overfitting and well preserves the identity.

**Image-to-Image Translation.** Image-to-Image Translation aims at translating images from a source domain to a target domain. Early research works [17,44] rely on paired data in source and target domain. When paired data is not available, unpaired Image-to-Image Translation methods [54,51,28,16,6,29] are proposed to utilize cycle consistency loss to learn from unpaired data. Recently, Park et al. proposed a patch-based contrastive loss to constrain the patches of the translated images to match that of the source images at the sampled image locations. Our work also utilizes the contrastive learning, but with a different format (adapt a source generator to a target generator) and a different purpose (prevent overfitting to the few training images).

**StyleGAN-based Artistic Portrait Generation.** Recently, some methods leverage StyleGAN for artistic portraits generation. StyleCariGAN[18] and Toonify[35] first train two StyleGAN networks on real face photos domain and artistic portraits domain, and swap some layers of the two StyleGANs to generate caricatures and cartoonized portraits. AgileGAN[40] transfers a pretrained StyleGAN on FFHQ to a stylistic domain and proposes a hierarchical VAE to embed real face photos to an extended  $\mathcal{Z}$ , i.e.,  $\mathcal{Z} + \text{Space}$ , to translate real face photos into artistic styles. However, the above methods need abundant data to train StyleGAN on artistic portraits domain or transfer a pretrained StyleGAN<sup>3</sup>. In comparison, our CtlGAN generates high quality results by learning from no more than 10 artistic examples.

### 3 Method

Given a few (e.g., no more than 10) artistic examples, our task is to learn a model which generates artistic portraits from real face photos. To solve this task, we

<sup>3</sup> StyleCariGAN uses 6,000+ caricatures, Toonify uses about 300 cartoon faces, and AgileGAN uses about 100 artistic faces.

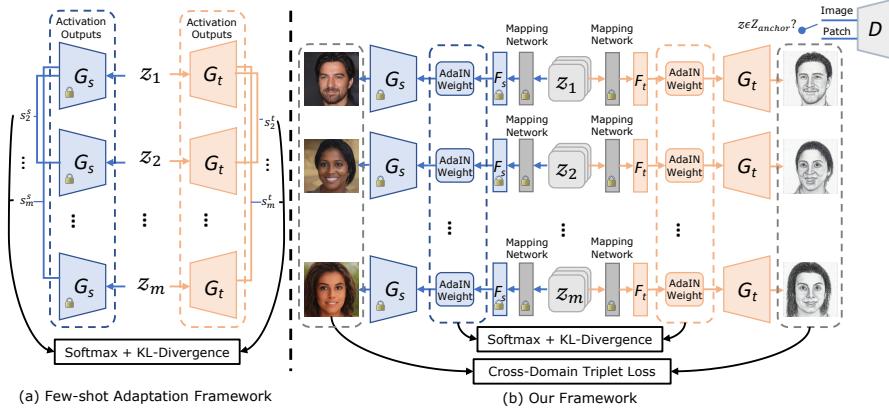


Fig. 4: (a) Few-Shot-GAN-Adaptation [33] framework and (b) Our contrastive transfer learning framework. Given a pretrained source generator and 10 target domain examples, Few-Shot-GAN-Adaptation adapts the model to the target domain by constraining the pairwise similarity before and after adaptation. While we explicitly enforce the target instances generated from two different latent codes to be different to prevent overfitting.

design a novel CtlGAN with a contrastive transfer learning strategy and a style encoder. Given a real face photo as input, our pipeline first encodes the face into a latent code, and then the decoder utilizes the latent code to generate an artistic portrait.

We design a novel contrastive transfer learning strategy to train our decoder for artistic portraits generation. Noticed that StyleGAN series [23,24] have achieved great success in high quality face generation, we leverage the facial knowledge in a pretrained StyleGAN2 [24] in real faces domain, and adapt the model to a target artistic domain. To prevent overfitting to the few training samples, we propose a novel Cross-Domain Triplet loss, which explicitly enforces the target instances generated from different latent codes to be distinguishable.

In order to translate a real face photo into an artistic portrait while keeping the original identity, a decent encoder is needed to map the face photo into the latent space of StyleGAN. Although there are some encoders [37,43] to embed an image into the  $\mathcal{W}$ + latent space (the extended  $\mathcal{W}$  space) of StyleGAN, they can't well cope with our decoder and the results show obvious artifacts (Fig. 2(b)(ii-iii)). Recent research [40] embeds real faces into a  $\mathcal{Z}$ + space (extended  $\mathcal{Z}$ ) to resolve the artifacts, but this does not generate an accurate reconstruction and causes identity loss (Fig. 2(b)(iv)). We follow the  $\mathcal{Z}$ + space setting and propose a novel style encoder to better preserve the identity information, which consists of a feature extractor and a sub-encoder (Fig. 2(b)(v)). We also design a dual path training to constrain the encoder output close to Gaussian distribution.

As described above, our CtlGAN consists of two components: 1) Few-shot Domain Adaptation Decoder (Sec. 3.1), which is transferred from a pretrained

StyleGAN to a target domain, and 2) Style Encoder (Sec. 3.2), which embeds real faces to  $\mathcal{Z}+$  space latent codes [40]. The pipeline is shown in Fig. 3.

### 3.1 Few-shot Domain Adaptation Decoder

Given a pretrained StyleGAN2 model  $\mathcal{G}_s$  trained on FFHQ dataset, which maps a  $\mathcal{Z}$  space latent code to a realistic face image, we aim to transfer  $\mathcal{G}_s$  to a target domain generator  $\mathcal{G}_t$  for an artistic style, *using only a few examples* (e.g., 10). Since adaption using very few examples easily leads to overfitting, to solve this problem, [33] proposes to preserve the relative distance before and after adaptation (Fig. 4(a)). However, its results still show similar facial features (Fig. 2(a)), which lowers the identity similarity.

To prevent the generations from overfitting to the few training examples, we want target generations from two different latent codes  $\mathcal{G}_t(z_i), \mathcal{G}_t(z_j)$  (pair1) to be different. To better preserve identity, we want source and target generations from the same latent code  $\mathcal{G}_s(z_i), \mathcal{G}_t(z_i)$  (pair2) to be similar in content (the same person with different styles). We propose a contrastive learning strategy (Fig. 4(b)) to achieve the above properties as follows:

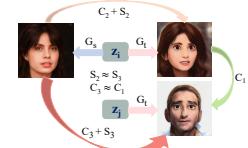
**Cross-Domain Triplet loss.** We propose to use triplet loss[38] to enforce the desired distance between pair1 and pair2. The triplet is an image  $x^a$  (*anchor*) with its *positive* example  $x^p$  (same class) and its *negative* example  $x^n$  (different class), and the triplet loss is calculated as:

$$\mathcal{L}_{triplet} = \max(d(x^a, x^p) - d(x^a, x^n) + \alpha, 0), \quad (1)$$

where  $d$  measures the distance between two images, and  $\alpha$  is a margin enforced between positive and negative pairs.

We regard the distance  $d$  between two images to be the addition of content distance  $C$  and style distance  $S$ , i.e.,  $d = C + S$ , and assume the style distance between the same domain images is 0. As shown in the right figure, we want  $C_1$  to be large,  $C_2$  to be small, and minimize the objective function  $C_2 - C_1$ . However, directly computing content distance  $C_2$  is hard, so we compute  $C_2 + S_2$  instead. Since the style distance between two domains is similar, i.e.,  $S_2 \approx S_3$ , and content differences between two people should be much larger than that of the same person in different forms, i.e.,  $C_3 \approx C_1$ , the objective function  $C_2 - C_1$  becomes  $(C_2 + S_2) - (C_1 + S_2) \approx (C_2 + S_2) - (C_3 + S_3)$ . In our problem, the *anchor* is  $\mathcal{G}_s(z_i)$ , the *positive* example is  $\mathcal{G}_t(z_i)$ , and the *negative* example is  $\mathcal{G}_t(z_j)$ . Then, our Cross-Domain triplet loss  $\mathcal{L}_{cdt}$  is formulated as:

$$\mathcal{L}_{cdt} = \mathbb{E}_{\{z_i \sim p_z(z)\}} \max(d^+(z_i) - d^-(z_i) + \alpha, 0) \quad (2)$$



$$d^+(z_i) = \mathcal{L}_d(\mathcal{G}_s(z_i), \mathcal{G}_t(z_i)) \quad (3)$$

$$d^-(z_i) = \frac{1}{m-1} \sum_{j,j \neq i}^m \mathcal{L}_d(\mathcal{G}_s(z_i), \mathcal{G}_t(z_j)), \quad (4)$$

where  $i, j$  are the index of latent codes;  $\mathcal{G}_s(z_i)$  is the source model output;  $\mathcal{G}_t(z_i)$ ,  $\mathcal{G}_t(z_j)$  are the target model outputs;  $m$  is the total number of sampled latent codes;  $\alpha$  is the margin; and  $\mathcal{L}_d$  is a modified LPIPS [53] which omits the 4-th layer output of VGG16 [39] in LPIPS. To validate the design of Cross-Domain triplet loss, we conduct ablation experiments in Sec. 4.5.

**KL-divergence for Adaptive Instance Normalization Inputs.** The Adaptive Instance Normalization (AdaIN) [15] blocks are important modules in StyleGAN, whose inputs control the “style” of the face output. We thus want the AdaIN’s inputs of source generator and target generator to share similar distribution. We propose a KL loss to preserve the relative distance between generations of two latent codes before and after adaptation (similar to [33]), but compute distance on the inputs to AdaIN modules. The KL-AdaIN Loss is formulated as follows:

$$\mathcal{L}_{kl-adain} = \mathbb{E}_{\{z_i \sim p_z(z)\}} \sum_{l,i} D_{KL}(y_i^{s,l} \| y_i^{t,l}) \quad (5)$$

$$y_i^{s,l} = \text{Softmax}(\{\text{sim}(F_s^l(z_i), F_s^l(z_j))\}_{\forall i \neq j}) \quad (6)$$

$$y_i^{t,l} = \text{Softmax}(\{\text{sim}(F_t^l(z_i), F_t^l(z_j))\}_{\forall i \neq j}), \quad (7)$$

where  $F_s^l$ ,  $F_t^l$  are the  $l$ -th AdaIN blocks’ inputs;  $i, j$  are index of latent codes; sim is the cosine similar function; and  $D_{KL}$  indicates KL-divergence.

**Total loss.** Total loss of our domain adaptation decoder consists of the adversarial loss  $\mathcal{L}_{adv}$ , Cross-Domain Triplet loss  $\mathcal{L}_{cdt}$  and KL-AdaIN loss  $\mathcal{L}_{kl-adain}$ :

$$\mathcal{L}_{decoder} = \lambda_{adv} \mathcal{L}_{adv} + \lambda_{cdt} \mathcal{L}_{cdt} + \lambda_{kl-adain} \mathcal{L}_{kl-adain}, \quad (8)$$

where  $\lambda_*$  are hyper-parameters,  $\mathcal{L}_{adv}$  is calculated using an image discriminator, a patch-level discriminator and an “anchor region” of latent space to decide which discriminator to use, as introduced in [33].

### 3.2 Style Encoder

To translate a real face photo into an artistic domain, a decent GAN inversion to embed the real face into the latent space is needed. We aim at *learning an encoder that embeds images into the latent space of decoders on different artistic domains*, i.e., the encoder is shared among decoders of different domains.

**$\mathcal{Z}+$  instead of  $\mathcal{W}+$ .** Traditional GAN inversion methods mostly embed images into the  $\mathcal{W}+$  space of StyleGAN [1,37,43]. The  $\mathcal{W}+$  space is designed for better reconstruction. After domain adaptation, the encoder’s goal is to find

latent codes best suitable for stylization. The difference between these two tasks (stylization vs reconstruction) leads to the difference of the most suitable  $\mathcal{W}+$ : as shown in Fig. 2(b)(ii-iii), combining reconstruction-based  $\mathcal{W}+$  encoders with our transferred decoder causes some artifacts. In our decoder adaptation, the input to the source and target decoders is the same  $z$  latent code, so  $\mathcal{Z}$  space remains the same after adaptation and is suitable for inversion. But directly embedding image into  $\mathcal{Z}$  space is difficult and sometimes can't maintain some key information, we follow [40] to extend  $\mathcal{Z}$  space to  $\mathcal{Z}+$ , by stacking  $n$  different  $z$  vectors, one for each layer of StyleGAN2 ( $n = 18$  for  $1024^2$ ,  $14$  for  $256^2$ ).

**Architecture.** The encoder is divided into two parts as in Fig. 3: a feature extractor and a sub-encoder. The feature extractor extracts multi-level features, and the sub-encoder maps multi-level features into  $z+$  latent code. 1) Feature Extractor: Since FPN has excellent performance in extracting multi-level features[27], and we adopt the FPN as our feature extractor. 2) Sub-encoder: To map the features into latent space, pSp [37] uses simple linear layers, we argue that the linear layers have weak performance when there is a big gap between the latent space and features. Recently, some image segmentation works [5,41] demonstrate that transformer block has excellent ability to translate features (extracted by CNN) to semantic information, we also found that adopting a transformer block instead of linear layers in the sub-decoder helps improve our style encoder's performance (the experiment is in appendix).

**Dual Path Training.** We utilize a pretrained StyleGAN2 on FFHQ as the decoder and fix its weights during training our encoder. We observe that during StyleGAN training and domain adaptation, the  $z$  latent code is sampled from Gaussian distribution, so the ideal  $\mathcal{Z}+$  space latent codes should also follow Gaussian distribution. The hVAE in AgileGAN[40] leverages a variational loss to ensure the output latent codes to follow the Gaussian distribution. However, we found it inferior in reconstruction task (Fig. 2(b)(iv)). We constrain the encoder output to follow Gaussian distribution by dual path training (Fig. 3). In path-1, a real face photo is fed into our encoder and then the decoder to reconstruct the input face, and we constrain the reconstructed face to be similar to the input face. In path-2, a  $z$  code is sampled, then extended to a  $z+$  code by simply repeating  $n$  times, and fed into the decoder to output a synthesized face (the process is denoted as path-2a); and the synthesized face then goes through path-1 (denoted as path-2b). We constrain both the reconstructed  $z+$  code and reconstructed image. Since path-2 contains a cycle, we name it *cycle path*.

For path-1, we use  $\mathcal{L}_2$  loss, LPIPS [53] loss  $\mathcal{L}_{lpips}$ , identity loss  $\mathcal{L}_{iden}$  based on ArcFace [8], and regularization loss  $\mathcal{L}_{reg}$ , with the same setting as [37]. The loss function is formulated as:

$$\mathcal{L}_{path1} = \lambda_{L_2} \mathcal{L}_{L_2} + \lambda_{lpips} \mathcal{L}_{lpips} + \lambda_{reg} \mathcal{L}_{reg} + \lambda_{iden} \mathcal{L}_{iden}. \quad (9)$$

For path-2, we use the smooth  $L_1$  loss [12] to measure the difference between original  $\mathcal{Z}+$  latent codes and embedded ones, which can be described as:

$$\mathcal{L}_{z\_predict} = \mathcal{L}_{smooth-L_1}(z_o, z_e), \quad (10)$$

where  $z_o$  is the latent code sampled from Gaussian distribution, and  $z_e$  is the output of our encoder. We use the same reconstruction loss as path-1 to supervise the image reconstruction in path-2. The overall loss for training path-2 is:

$$\mathcal{L}_{path2} = \lambda_{path1} \mathcal{L}_{path1} + \lambda_{z\_predict} \mathcal{L}_{z\_predict}, \quad (11)$$

where the  $\lambda_*$  are hyper-parameters. Dual path is a regularization term, and we conduct a t-SNE [30] experiment in appendix to show its effects.

## 4 Experiment

We implement the proposed method in PyTorch. All experiments are run on a PC with an NVIDIA RTX 3090 GPU. We use the StyleGAN model pretrained on faces images with  $256 \times 256$  resolution. We conduct extensive qualitative, quantitative comparison and a perceptual study to demonstrate that the proposed method outperforms state-of-the-arts in artistic portrait generation on various styles under 10-shot and 1-shot settings. Then, we conduct ablation studies to show the power of the three most important components and analysis of cross-domain triplet loss. More results and network details are presented in the appendix.

### 4.1 Training Details, Datasets and Metrics

**Training details and Datasets.** The decoder and encoder are trained separately. The encoder is trained only once, and shared among multiple adapted decoders, while one decoder is adapted for each artistic domain.

**Stage-1: encoder training.** To train our style encoder, we use a fixed StyleGAN2 trained on FFHQ dataset [23] as the decoder, and train the encoder on FFHQ by optimizing loss function  $\mathcal{L}_{path1}$  and  $\mathcal{L}_{path2}$ . We then use the CelebA-HQ dataset [20] as test data.

**Stage-2: decoder adapting.** For the few-shot domain adaptation, we use the pretrained StyleGAN2 as the base model, and adapt the base decoder from source domain to a target artistic domain by optimizing  $\mathcal{L}_{decoder}$  in Eq. 8. We adapt one decoder for each of the following target artistic domains, *all using no more than 10 artistic portraits images as training data* (Fig. 5): 1) Sketches from CUHK face sketch dataset[45]; 2) Caricature from web; 3) Cartoon from Toonify cartoon dataset[35]; 4) Raphael from Artistic-Faces[49]; 5) Roy Lichtenstein from Artistic-Faces. We further extend to 6) Sunglasses from FFHQ dataset.

**Metrics.** To quantitatively evaluate our method, we randomly sample 5000 images from CelebA-HQ dataset [20] as test data and evaluate the generated images using the following three metrics: **1) Fréchet Inception Distance (FID)**[14] is a widely used metric to evaluate the similarity between the distribution of generated images and the distribution of real data. Lower FID indicates higher similarity and better generation. We use FID to measure the distribution similarity between the generated artistic portrait images and the real

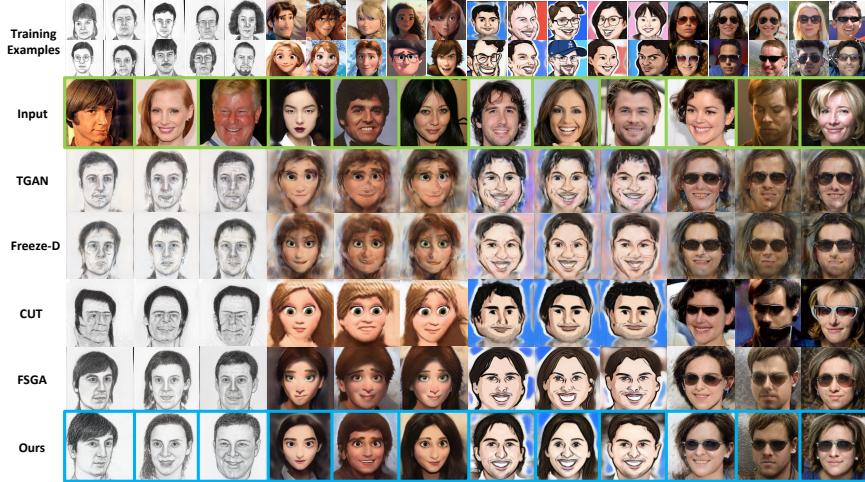


Fig. 5: Comparison with FSGA[33], TGAN[47], Freeze-D[31], CUT[34] under 10 training images setting.

Table 1: Quantitative comparison with different domain adaptation and unpaired Image-to-Image Translation methods on multiple domains. All methods are trained with 10 images. The metrics are computed from 5,000 generated images.

Models	Sketches			Cartoon			Sunglasses		
	FID↓	Ld↓	Lc↑	FID↓	Ld↓	Lc↑	FID↓	Ld↓	Lc↑
TGAN [47]	134.21	0.71	0.285/0.01	199.52	0.61	0.431/0.01	88.98	0.60	0.443/0.03
Freeze-D [31]	132.92	0.72	0.294/0.01	176.06	0.62	0.437/0.02	87.94	0.61	0.409/0.02
CUT [34]	82.21	0.66	0.405/0.09	176.96	0.56	0.431/0.04	85.65	<b>0.45</b>	<b>0.587/0.02</b>
FSGA [33]	52.35	0.69	0.322/0.02	104.07	0.58	0.460/0.03	61.40	0.51	0.475/0.02
Ours	<b>49.85</b>	<b>0.58</b>	<b>0.424/0.03</b>	<b>84.93</b>	<b>0.51</b>	<b>0.515/0.03</b>	<b>48.29</b>	0.50	0.482/0.01

artistic data. Real data source: for sketch, we use 295 face sketches from CUHK face sketch dataset; for cartoon, we use 252 cartoons from Toonify dataset and web; for sunglasses, we use 2,683 sunglasses images from FFHQ. **2) LPIPS Distance** [53] is a widely used metric to evaluate the perceptual similarity between two images. We calculate the LPIPS score between the input face photo and the generated artistic image. We use this metric to evaluate the identity preservation of generated results. **3) LPIPS Cluster:** Deep models could easily overfit under few-shot setting. Following [33], we measure the overfitting extent of the generation model using the Intra-cluster pairwise LPIPS distance, which we abbreviate as LPIPS cluster. The metric assigns the generated images to the nearest training image (by LPIPS distance) and obtains 10 clusters, and calculates the average intra-cluster pairwise distance. The higher the average distance, the lower the overfitting degree.

Table 2: Quantitative comparison with different encoders.

Encoders	Sketches	Cartoon	Sunglasses	FID ↓	Ld↓	FID ↓	Ld↓	FID ↓	Ld↓
pSp [37]	90.96	0.597	92.22	0.565	66.08	0.531			
e4e [43]	101.19	0.596	93.37	0.542	62.03	0.528			
hVAE $\mathcal{Z}+$ [40]	51.49	0.589	94.63	0.554	57.39	0.504			
Ours	<b>49.85</b>	<b>0.586</b>	<b>90.80</b>	<b>0.537</b>	<b>48.29</b>	<b>0.504</b>			

Table 3: User study results.

	Ours	FSGA	Freezed	TGAN	CUT
Rank1	<b>59.6%</b>	25.7%	2.9%	3.7%	8.2%
Rank2	26.2%	56.4%	5.3%	5.3%	6.8%
Rank3	7.6%	9.7%	36.6%	26.1%	20.0%
Rank4	3.5%	4.3%	34.2%	45.7%	12.3%
Rank5	3.1%	3.9%	21.0%	19.3%	52.7%

## 4.2 Comparisons with Few-Shot Generation Models

**Comparison Methods.** We compare our few-shot domain adaptation decoder with domain adaptation methods and an unpaired Image-to-Image Translation method *under few-shot setting (10 training images)*: (i) Few-Shot-GAN-Adaptation (FSGA) [33]: adapts a pretrained model in source domain to target domain via cross-domain correspondence. (ii) TGAN [47]: transfers a pretrained source domain model to target domain by finetuning the original loss function. (iii) FreezeD [31]: freezes the first three layers of the discriminator during adaptation. (iv) CUT [34]: an unpaired Image-to-Image Translation method based on patch-wise contrastive learning. We use the author implementations for (i), (iii), (iv) and implement (ii) by ourselves. For (ii)(iii), we use the patch-wise discriminator the same as (i) and Ours to ensure a fair comparison.

For encoder, we compare our encoder with (i) pSp [37] and e4e [43] encoder, which encode real face photos into  $\mathcal{W}+$  space; and (ii) the hVAE  $\mathcal{Z}+$  encoder in AgileGAN [40]. We use author implementations for (i) and since (ii) AgileGAN is not open-sourced, we implement its encoder following the paper description.

**Qualitative Comparison.** Fig. 5 shows qualitative comparisons with different domain adaptation methods and unpaired Image-to-Image Translation methods on multiple target domains, i.e., Sketches, Cartoon, Caricature, and Sunglasses. Results of CUT show clear overfitting, except sunglasses domain; FreezeD and TGAN results contain cluttered lines in all domains; Few-Shot-GAN-Adaptation results preserve the identity but still show overfitting; while our results well preserve the input facial features, show the least overfitting, and significantly outperform the comparison methods on all four domains.

We show qualitative comparisons with different encoders in Fig. 2(b). Results of pSp and e4e show clear artifacts in generation results, and hVAE  $\mathcal{Z}+$  encoder is worse in identity preservation. In contrast, our encoder has good performance in reducing artifacts and outperforms the comparison encoders.

**Quantitative Comparison.** Table 1 shows the FID, LPIPS distance (Ld), and LPIPS cluster (Lc) scores of ours and different domain adaptation methods and unpaired Image-to-Image Translation methods on multiple target domains, i.e., Sketches, Cartoon and Sunglasses. Our few-shot domain adaptation decoder achieves the best FID on all three domains. We also achieve the best LPIPS distance and LPIPS cluster on Sketches and Cartoon domain. For Sunglasses domain, our LPIPS distance and LPIPS cluster are worse than CUT, but

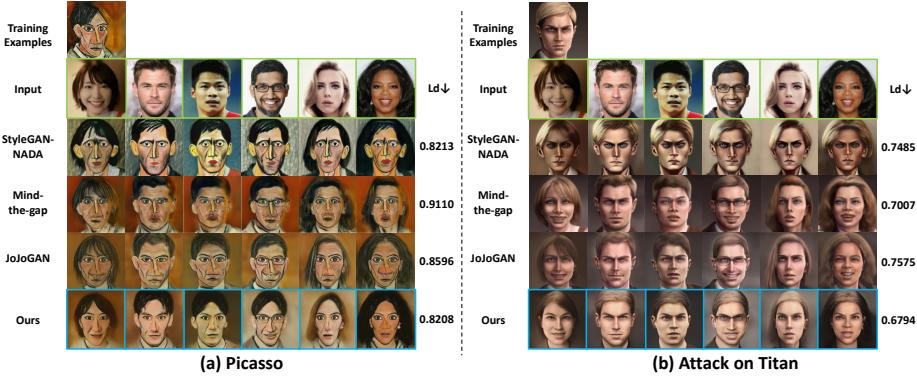


Fig. 6: Comparison with StyleGAN-NADA[3], Mind-the-gap[55], JoJoGAN[7] under one-shot setting. All models utilize a pretrained StyleGAN, and StyleGAN-NADA, Mind-the-gap leverage external knowledge in a pretrained CLIP. Ld denotes LPIPS distance to input photos.

qualitative results (Fig. 5) show CUT simply blackens the eye regions and leads to obvious artifacts.

Table 2 shows the FID, LPIPS distance of ours and different encoders on multiple target domains, i.e., Sketches, Cartoon and Sunglasses. Our encoder is better than all three comparison encoders on all three domains. Results show that our encoder better copes with our artistic domain decoder and generates better results in distribution similarity and identity preservation.

**User study.** We conduct a user study to compare our method with TGAN, Freeze-D, CUT and Few-Shot-GAN-Adaptation in 10-shot setting. We randomly sample 120 images from CelebA-HQ dataset, and generate artistic portraits in 4 domains (Sketches, Cartoon, Caricature, Sunglasses). Participants are required to rank the results of comparison methods and ours considering generation quality, style consistency and identity preservation. 62 participants attended the user study and each of them compared 30 groups of results (randomly sampled from 120). Results of the percentages of each method ranked as 1-5 are summarized in Table 3. Our method ranks the best in 59.6% of votes, which significantly outperforms other methods. The average rankings of different methods are: ours 1.64, Few-Shot-GAN-Adaptation 2.04, FreezeD 3.65, TGAN 3.72 and CUT 3.94.

### 4.3 Comparisons with One-Shot Domain Adaptation Models

**Comparison Methods.** Recently, some notable one-shot domain adaptation methods are developed based on pretrained StyleGAN and CLIP models. 1) StyleGAN-NADA[10]: leverages the semantic power of CLIP[36] model and uses text to guide the domain adaptation, its code also implements one-shot domain adaptation based on a reference image; 2) Mind-the-GAP [55]: a one-shot domain adaptation method that leverages CLIP model and StyleGAN



Fig. 7: Our 1-shot results on multiple artistic domains, each with 1 training image shown in the top row.



Fig. 8: Our 1-shot results on multiple artistic domains, each with 1 training image shown in the top row.

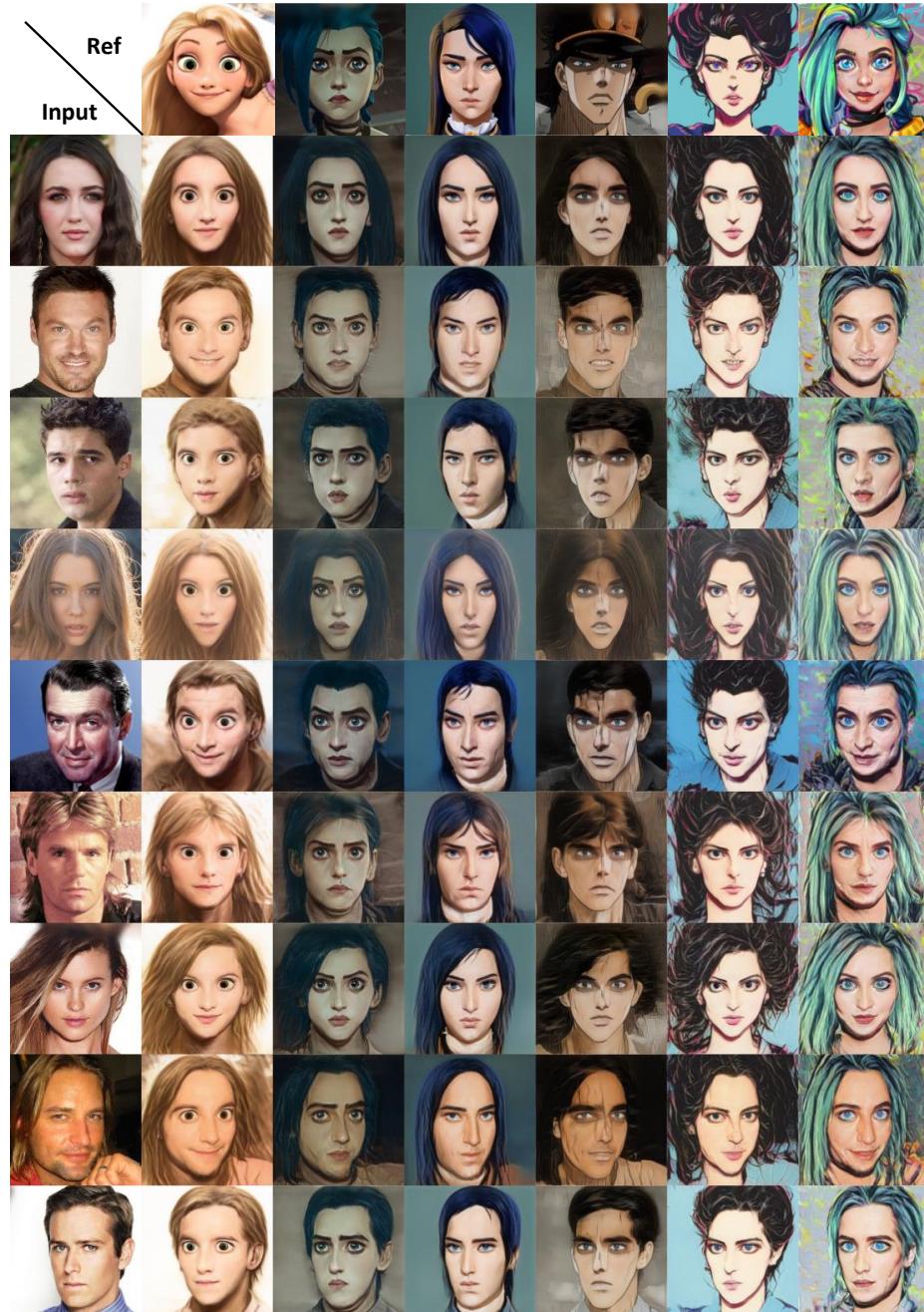


Fig. 9: Our 1-shot results on multiple artistic domains, each with 1 training image shown in the top row.

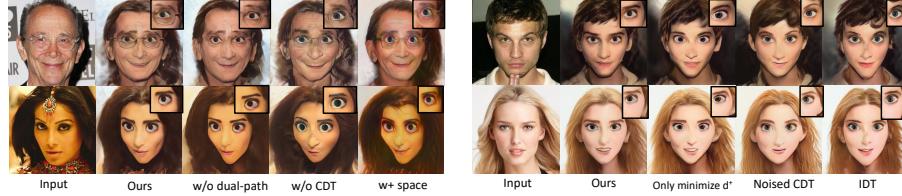


Fig. 10: (a) Ablation study on three key components;(b)Analysis of Cross-Domain Triplet loss.

Table 4: Ablation study quantitative scores.

Models	FID ↓	Ld ↓	Lc ↑
w/o CDT	113.56	0.56	0.498/0.04
w/o dual-path	88.89	0.51	0.498/0.05
$z+ \rightarrow w+$	92.23	0.56	0.474/0.03
<b>Ours</b>	<b>84.93</b>	<b>0.51</b>	<b>0.515/0.03</b>

Table 5: Analysis of Cross-domain Triplet Loss.

Models	FID ↓	Ld ↓	Lc ↑
only minimizing $d^+$	108.07	0.54	0.511/0.02
close $z$ instead of same $z$	99.64	0.53	0.495/0.04
only $C$ instead of $C + S$	106.49	0.59	0.489/0.03
<b>Ours</b>	<b>84.93</b>	<b>0.51</b>	<b>0.515/0.03</b>

inversion and computes the domain gap as a direction in CLIP embedding space; 3) JoJoGAN [7]: a one-shot domain adaptation method that generates a large dataset from a single reference by mixing style latent codes and then finetunes StyleGAN using pixel-level loss.

In Fig. 6, we compare with these methods under one-shot setting on two artistic domains. StyleGAN-NADA results are worse in style similarity; Mind-the-gap and JoJoGAN are unstable for some domain (Fig. 6(a)), because they first invert the reference image of target domain back to FFHQ faces domain, and this is difficult for abstract style like Picasso. CtlGAN achieves good stylization and has the lowest LPIPS distance (Ld) to input photos.

**More 1-shot results are shown in Figs 7, 8, 9, including 27 test photos and six different artistic domains, where the training examples are shown in the top row.**

#### 4.4 Ablation Study on Three Key Components

We conduct ablation studies on three key components of our method: the cross-domain triplet loss (CDT) in our decoder, the  $Z+$  space and the dual training path in our encoder. We train the ablated models by removing each component and evaluate the metrics. As shown in Fig. 10(a) and Table. 4, each component plays an important role in our final results.

#### 4.5 Analysis of Cross-Domain Triplet Loss

To validate the design of cross-domain triplet loss, we conduct comparison with three different designs: (1) only minimizing  $d^+$ , i.e., the distances between images generated by the same latent code for source and target domain; (2) using a

close  $z$  instead of the same  $z$  in positive pair (Noised CDT); (3) only concerning content distance without style distance (In-Domain Triplet Loss, IDT).

As shown in Table. 5, our Cross-Domain Triplet loss has better FID, Ld and Lc score than other settings. As shown in Fig. 10(b), the model trained with our CDT has the best visual quality.

## 5 Conclusion and Discussion

In this paper, we propose CtlGAN, a new framework for few-shot artistic portraits generation (no more than 10 artistic faces). With a new contrastive transfer learning strategy, we effectively avoid overfitting in few-shot generation. And with a new encoder with  $Z+$  latent space setting and dual path training, we generate high-quality artistic portraits while keeping the identity. Extensive qualitative, quantitative comparisons and a user study show our method achieves state-of-the-art performance. Our model mainly targets artistic portraits generation and has some limitations for local editing, as shown in Fig. 5, the FFHQ→Sunglasses model sometimes changes the haircut and skin details. In the future, we wish to develop a model suitable for both global style change and local editing.

## Appendix A Overview

This appendix includes:

- more 1-shot and 10-shot results on multiple artistic domains (Sec. B);
- results on other domains (Sec. C);
- more analysis on each component: 1) t-SNE visualization of the dual-path impact (Sec. D.1); 2) analysis of sub-encoder (Sec. D.2); 3) more ablation studies on decoder (Sec. D.3); 4) detailed analysis on triplet loss (Sec. D.4);
- detailed network architecture, training settings and hyper-parameters (Sec. E);
- comparison with neural style transfer and image-to-image translation methods (Sec. F).

## Appendix B 1-shot and 10-shot results on multiple artistic domains

In this section, we show more results on multiple artistic domains under 1-shot and 10-shot training.

**1-shot results** are shown in Figs. 7, 8, 9, including 27 test photos and six different artistic domains, where the training examples are shown in the top row.

**10-shot results** are shown in Figs. 15, 16, 17, 18, 19, 20, including 54 test photos and five different artistic domains, where the 10 training images of each artistic domain are shown in Fig. 14.

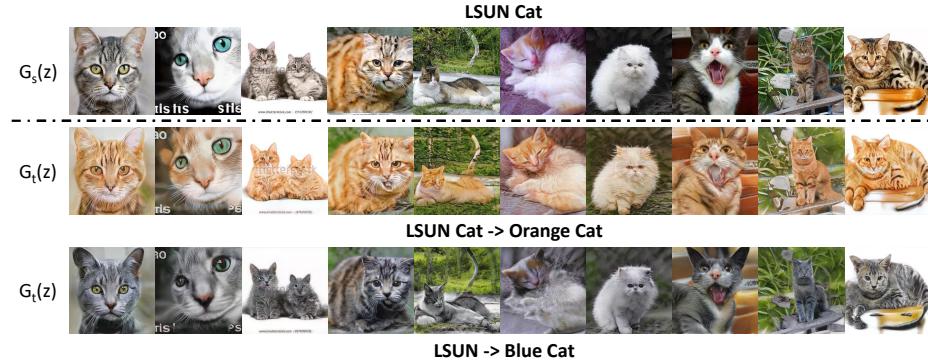


Fig. 11: Results on Cats. The source domain is LSUN[52] Cat, and the target domain is Orange Cat or Blue Cat.



Fig. 12: Results on Churches. The source domain is LSUN[52] Church, and the target domain is Van Gogh House or Haunted House.

## Appendix C Results on Other Domains

We test our model on other domains, e.g., Cats and Churches. The source domain is LSUN [52] Cat or LSUN Church, where we use the StyleGAN2 models pretrained on these datasets. The target domains include Orange Cat, Blue Cat, Van Gogh House, and Haunted House, where the 10 training images are collected from web and shown in Fig. 14. We directly sample  $z+$  latent codes to validate the correctness of our contrastive strategy in these domains. The testing results are shown in Fig 11 and Fig 12, our models generate good stylization results and keep the content well.

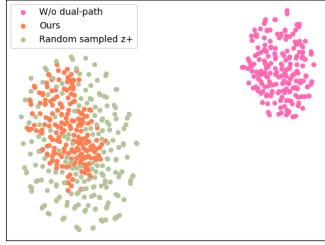


Fig. 13: t-SNE visualization of dual path training.

## Appendix D More Analysis on Each Component

### D.1 t-SNE Visualization of the Dual-path Impact

In this section, we conduct a t-SNE [30] experiment to visualize the influence of dual-path training strategy used in our encoder. As shown in Fig. 13, we compare three groups of  $z+$  latent codes:

1. Latent codes generated by our encoder (orange);
2. Latent codes generated by our encoder without dual-path (pink);
3. Random sampled latent codes under Gaussian distribution (green).

The results show the dual-path training strategy helps constrain the output latent distribution to follow Gaussian distribution (which is the sampling distribution of decoder input), so that it can better cope with our decoder.

### D.2 Analysis of Sub-encoder

We compare the reconstruction quality of using linear layers (used in pSp) or attention module, or transformer block as the sub-encoder in our encoder<sup>4</sup>. We separately study the three kinds of sub-encoder in  $\mathcal{W}+$  and  $\mathcal{Z}+$  space:

1. Linear layers (fully-connected layers): To explore the effect of linear layers, we try linear settings with one single linear layer and 8 linear layers, and set the input dimension as 512, output dimension as 512, a LeakyReLU following each linear layer;
2. Attention module: To explore the performance of attention module (we use the implementation in ViT [9] code), we design the sub-encoder block with 6 attention layers, and set the input dimension as 512, the heads number as 14, the head dimension as 64;
3. Transformer block: We use ViT [9] implementation, and set the input dimension as 512, the heads number as 14, the head dimension as 64, the MLP dimension as 1024, and the number of layers as 6.

From Table 6, we find that transformer block achieves the best reconstruction quality in both kinds of latent spaces, with a closer LPIPS distance than 1 linear

<sup>4</sup> Because the feature extractor contains convolution layers and the sub-encoder is after the feature extractor, we don't try convolution layers here.

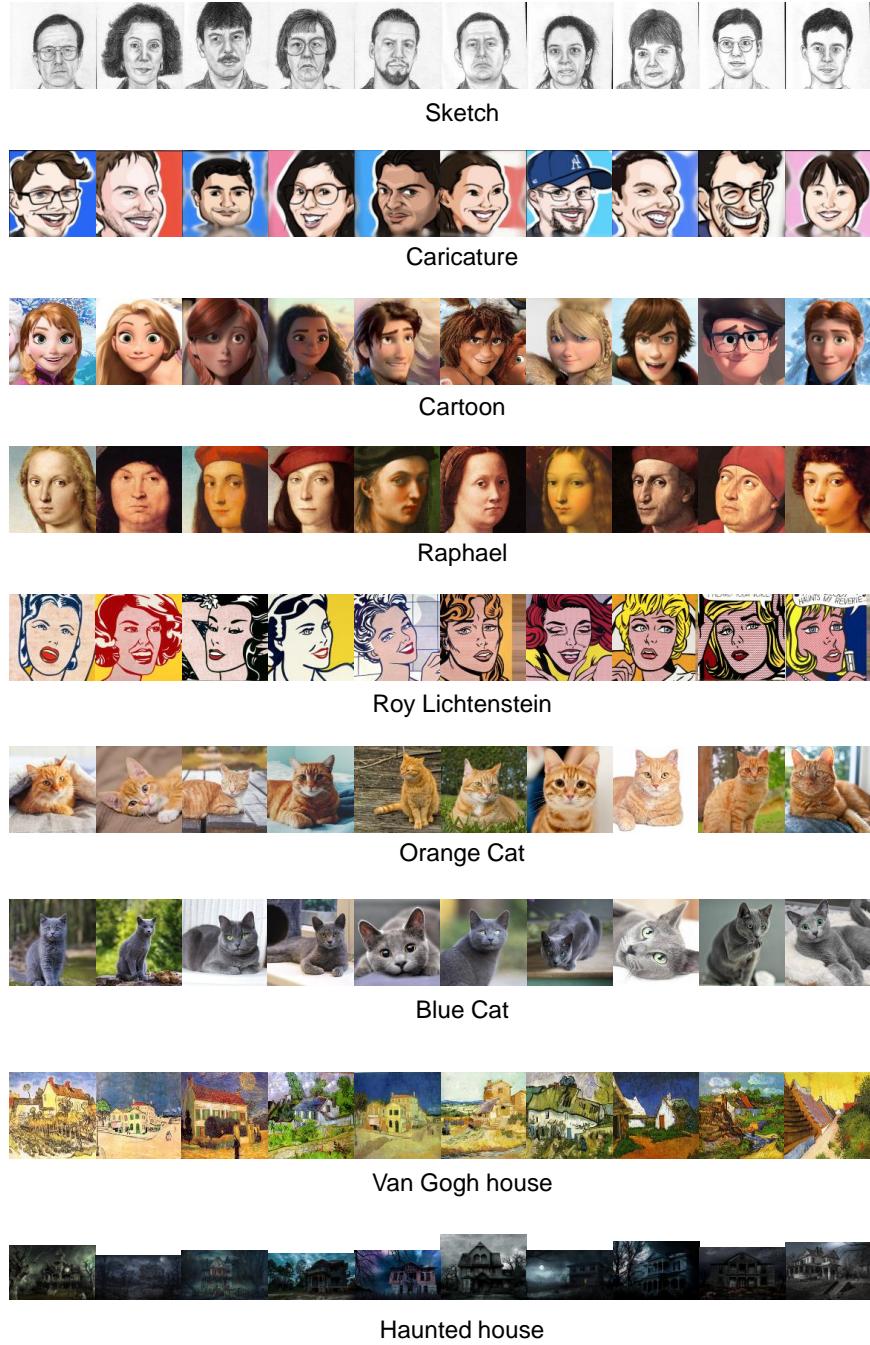


Fig. 14: The 10 training images used for different artistic domains.

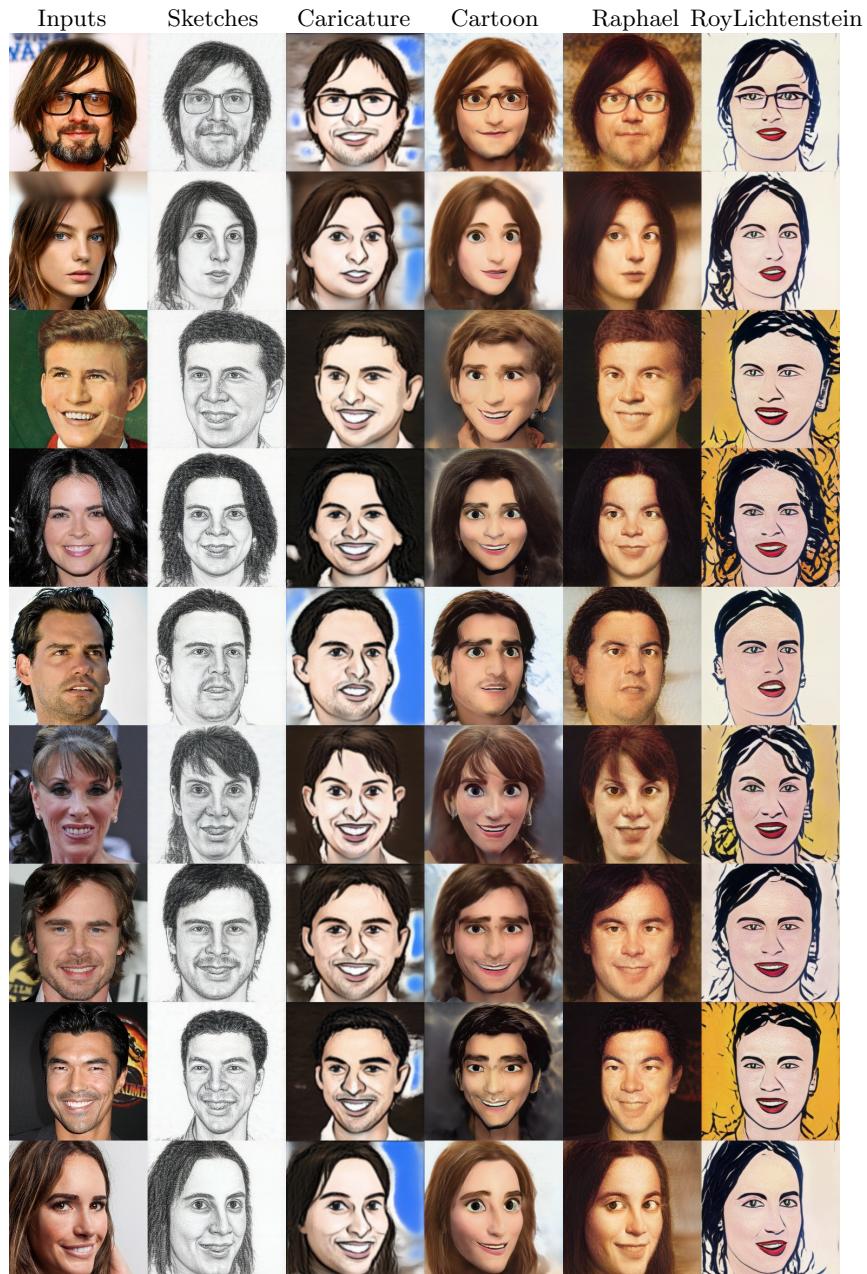


Fig. 15: Our 10-shot results on multiple artistic domains. From left to right: input face photos, Sketches, Caricature, Cartoon, Raphael, and Roy Lichtenstein results.

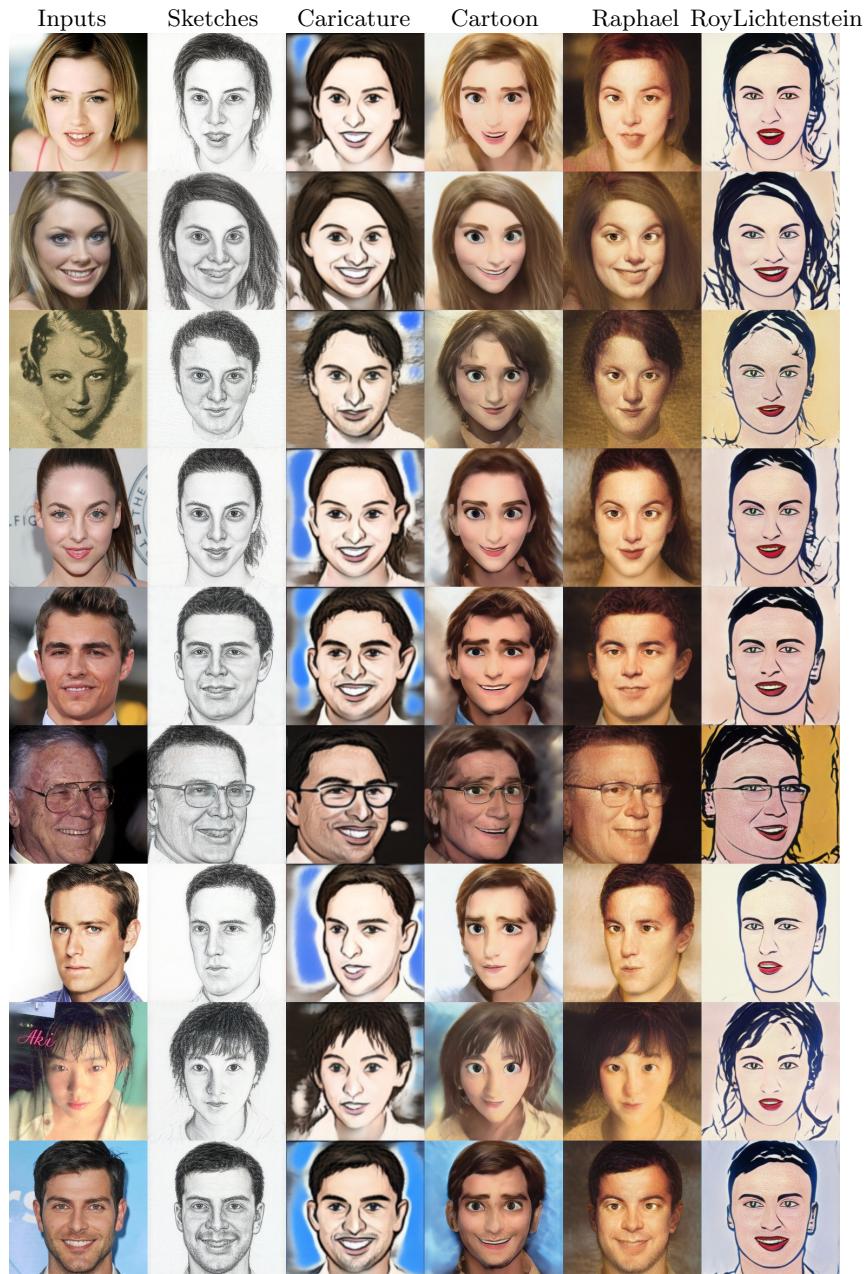


Fig. 16: Our 10-shot results on multiple artistic domains. From left to right: input face photos, Sketches, Caricature, Cartoon, Raphael, and Roy Lichtenstein results.

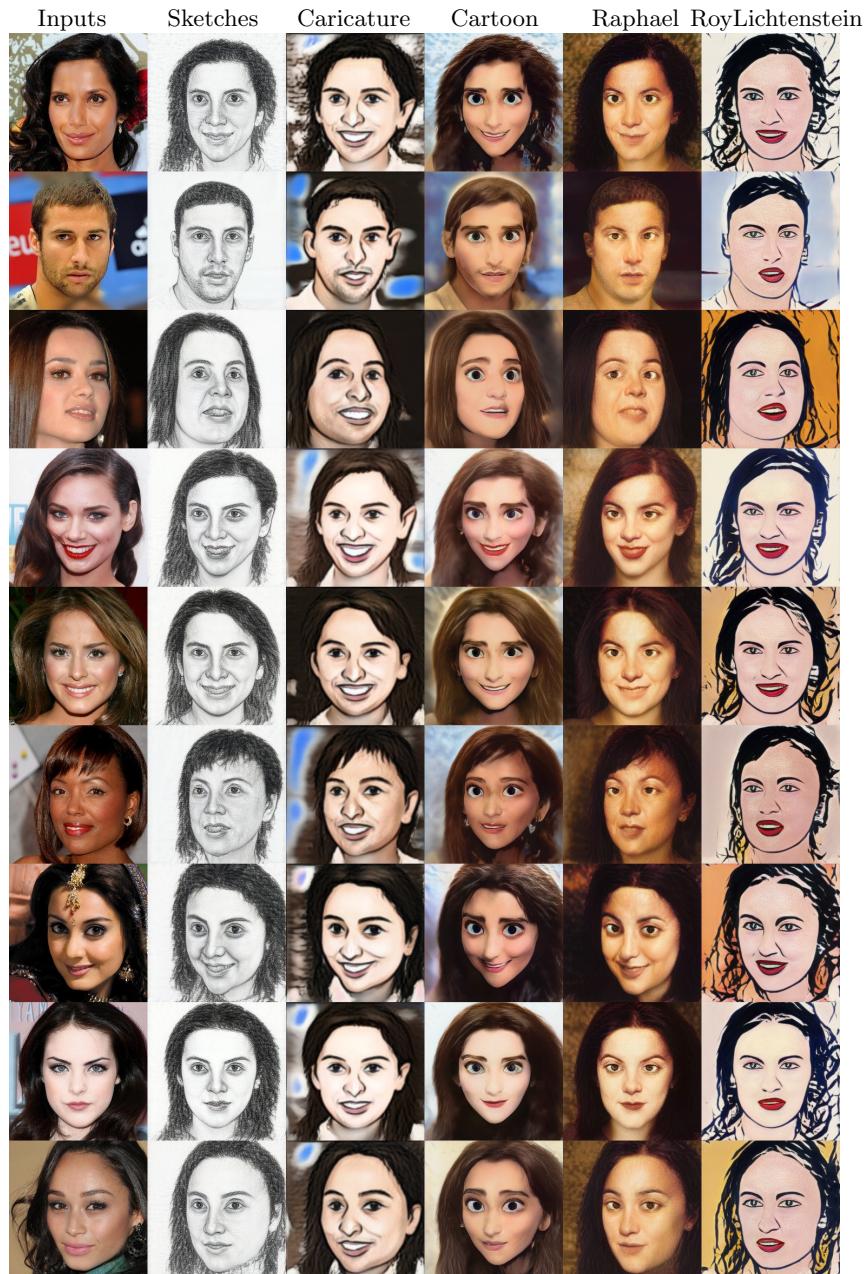


Fig. 17: Our 10-shot results on multiple artistic domains. From left to right: input face photos, Sketches, Caricature, Cartoon, Raphael, and Roy Lichtenstein results.

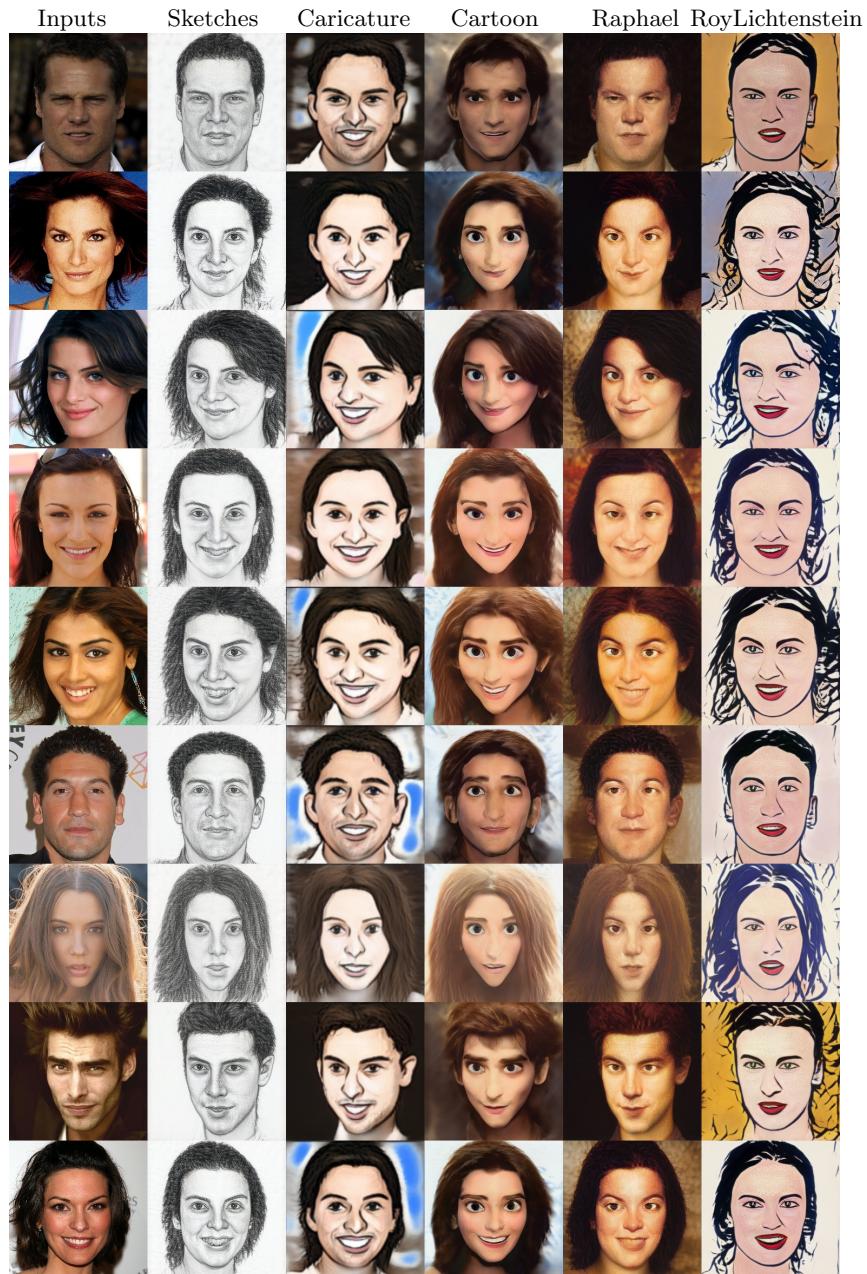


Fig. 18: Our 10-shot results on multiple artistic domains. From left to right: input face photos, Sketches, Caricature, Cartoon, Raphael, and Roy Lichtenstein results.

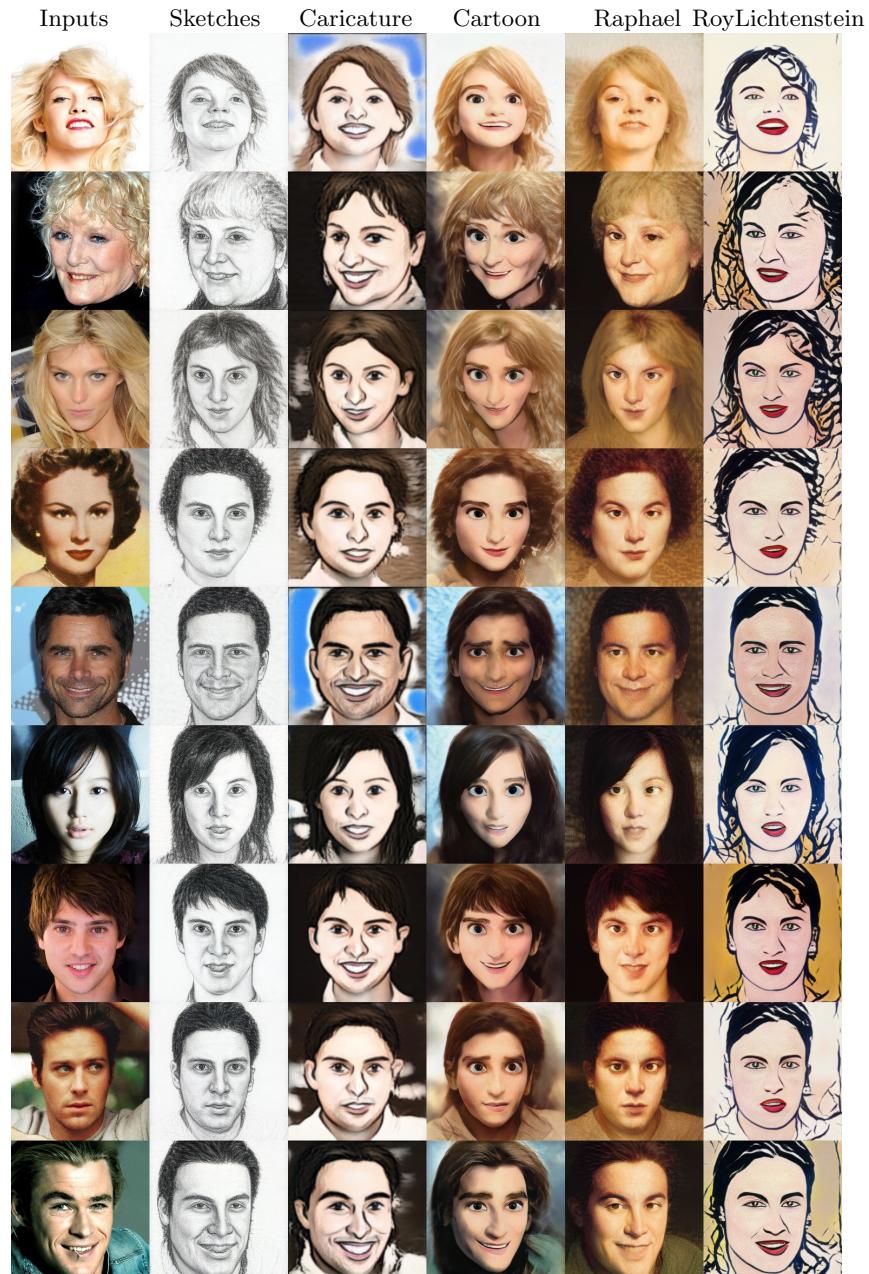


Fig. 19: Our 10-shot results on multiple artistic domains. From left to right: input face photos, Sketches, Caricature, Cartoon, Raphael, and Roy Lichtenstein results.

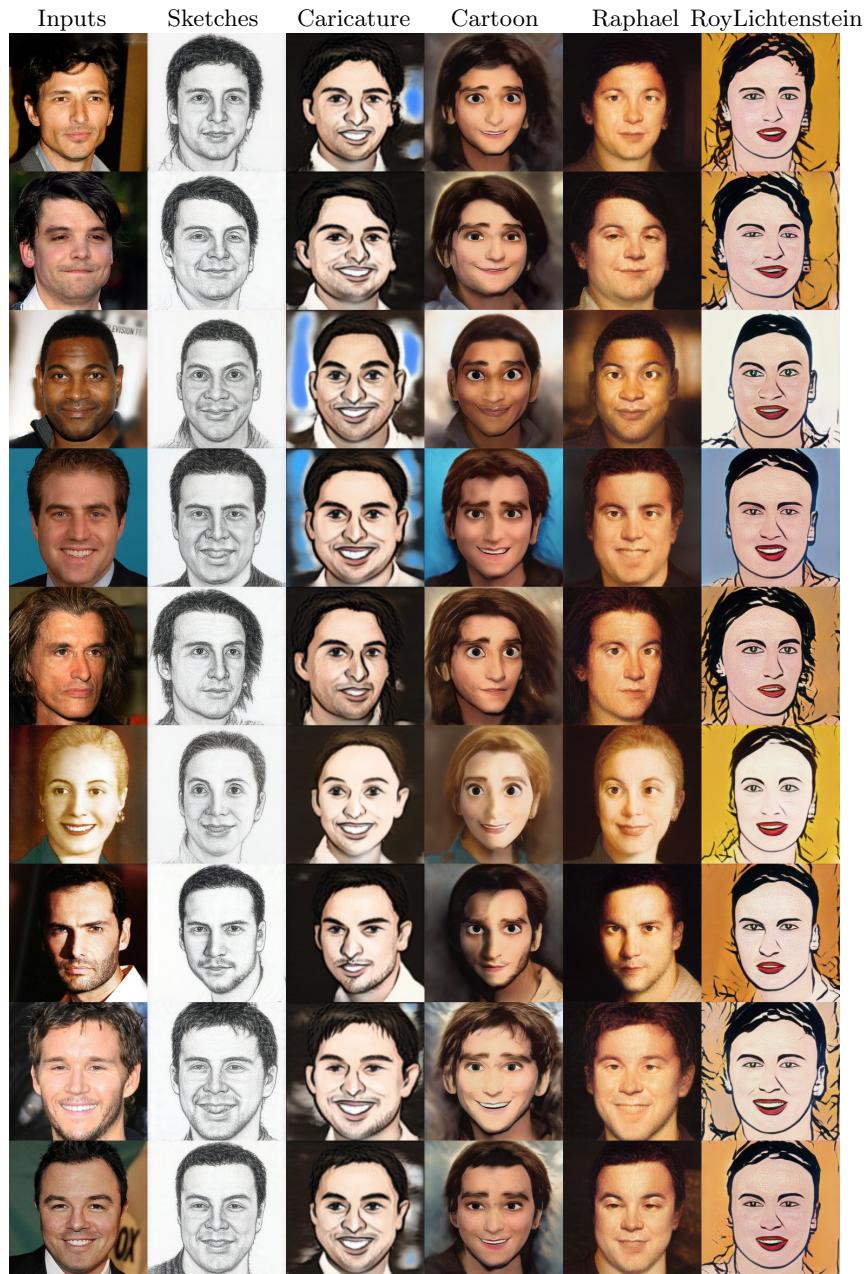


Fig. 20: Our 10-shot results on multiple artistic domains. From left to right: input face photos, Sketches, Caricature, Cartoon, Raphael, and Roy Lichtenstein results.

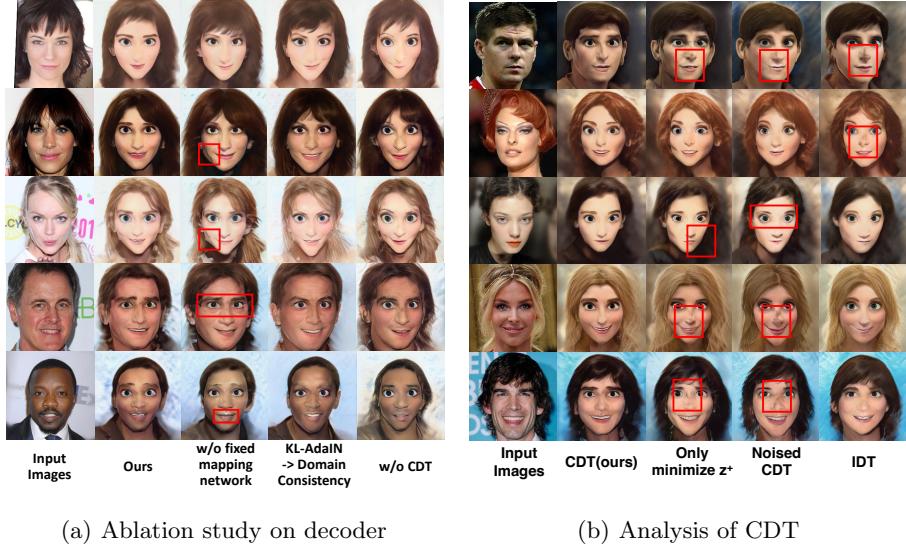


Fig. 21: (a) More ablation study results on decoder. The 10 training images are displayed on the left. (b) Analysis of Cross-Domain Triplet loss.

Table 6: Analysis of sub-encoder: av- Table 7: Ablation study results on average LPIPS distance of reconstruction decoder.  
and cartoon FID score.

Models	recon LPIPS ↓		FID↓	Models	FID ↓ ld ↓	
	$z+$	$w+$			Ours w/o CDT	113.56 0.56
linear layers (1 layer)	0.25	0.21	100.18	Ours w/o fixed mapping network	96.87	0.54
linear layers (8 layers)	0.26	0.28	86.55	Ours w/o KL-AdaIN	101.57	<b>0.49</b>
attention module (6 layers)	0.48	0.32	282.03	Ours	<b>84.93</b>	0.51
transformer block (6 layers)	<b>0.23</b>	<b>0.19</b>	<b>84.93</b>			

layer (-0.02 for  $\mathcal{Z}+$  and -0.02 for  $\mathcal{W}+$ ), 8 linear layers (-0.03 for  $\mathcal{Z}+$  and -0.09 for  $\mathcal{W}+$ ) and attention module (-0.25 for  $\mathcal{Z}+$  and -0.13 for  $\mathcal{W}+$ ). We also test the FID scores of different encoders on cartoon domain, and results show that transformer block achieves the best stylization.

### D.3 More Ablation Studies on Decoder

In the main paper Sec. 4.4, we evaluate the effectiveness of cross-domain triplet loss in our decoder. In this section we further analyze other components in our decoder.

- KL-AdaIN loss: Apart from CDT loss, we introduce KL-AdaIN loss in our decoder. In this ablation, we replace KL-AdaIN with KL loss (i.e., the cross-domain distance consistency) in [33]. As shown in Fig. 21(a)(column4), the ablated version has worse style similarity. As shown in Table. 7, Ours has much better FID and similar ld with the ablated version.

- Fixed mapping network: We found fixing the  $Z$ -to- $W$  mapping network during adaption helps ease the training. In this ablation, we make the mapping network trainable during adaptation. As shown in Fig. 21(a)(column3), the ablated version contains more artifacts than ours. As shown in Table. 7, Ours outperforms the ablated version on both metrics.

#### D.4 Detailed Analysis on Triplet Loss

In the main paper Sec. 3.1, we describe our Cross-Domain Triplet loss (CDT).

$$\mathcal{L}_{cdt} = \mathbb{E}_{\{z_i \sim p_z(z)\}} \max(d^+(z_i) - d^-(z_i) + \alpha, 0) \quad (12)$$

$$d^+(z_i) = \mathcal{L}_d(\mathcal{G}_s(z_i), \mathcal{G}_t(z_i)) \quad (13)$$

$$d^-(z_i) = \frac{1}{m-1} \sum_{j,j \neq i}^m \mathcal{L}_d(\mathcal{G}_s(z_i), \mathcal{G}_t(z_j)), \quad (14)$$

And In the main paper Sec. 4.5 and Table 5, we validate the the design of cross-domain triplet loss with three different designs.

In this section, we describe the three comparison designs in detail, and provide more qualitative comparisons (Fig. 21(b)):

1. **Only minimizing  $d^+$ .** In this ablation, given two sampled latent codes  $z_i$ , we directly minimize the LPIPS distance between the source domain image and target domain image ( $d^+$ ). Fig. 21(b)(column3) shows this hurts the stylization, while our CDT ( $d^-$  counteracts the style difference in  $d^+$ ) achieves better stylization.
2. **In-Domain Triplet loss (IDT).** In this ablation, we only compare generated results within the target domain, instead of between source and target domains. Given two sampled latent codes  $z_i$  and  $z_j$ , we sample a close  $z_i^* = z_i + \Delta z_i$ ,  $\Delta z_i \sim \mathcal{N}(0, 0.1)$ . We set the anchor as  $\mathcal{G}_t(z_i)$ , the positive example as  $\mathcal{G}_t(z_i^*)$ , and the negative example as  $\mathcal{G}_t(z_j)$ , where  $\mathcal{G}_t$  is the target decoder. Fig. 21(b)(column5) shows its results contain artifacts, while our CDT (cross-domain distance) achieves better results.
3. **Noised Cross-Domain Triplet loss (Noised CDT).** In this ablation, we change the positive example from  $\mathcal{G}_t(z_i)$  to  $\mathcal{G}_t(z_i^*)$ , where  $z_i^*$  is a close latent code to  $z_i$ ,  $z_i^* = z_i + \Delta z_i$ ,  $\Delta z_i \sim \mathcal{N}(0, 0.1)$ . The anchor is  $\mathcal{G}_s(z_i)$ , the positive example is  $\mathcal{G}_t(z_i^*)$ , and the negative example is  $\mathcal{G}_t(z_j)$ . Fig. 21(b)(column4) shows that the results are worse in keeping identity, while our CDT (same  $z_i$ ) achieves better results.

## Appendix E Detailed Network Architecture and Hyper-Parameters

### E.1 Few-shot Domain Adaptation Decoder

**Architecture:** We adopt StyleGAN2 architecture [24] for our decoder, to map  $\mathcal{Z}+$  space latent codes into artistic portraits. We use adversarial training

for adapting the decoder, and use two discriminators, an image discriminator, a patch-level discriminator following the implementation from Few-shot-GAN-adaptation [33].

**Training details and hyper-parameters:** We adopt a pretrained StyleGAN2 on FFHQ as the base model and then adapt the base model to our target artistic domain.

For 10-shot training, we set  $\lambda_{adv} = 1.0$ ,  $\lambda_{kladain} = 1000$  in main paper Eq.(8) for all target artistic domains, and set different  $\lambda_{cdt}$  for different artistic domains as follows:

1.  $\lambda_{cdt} = 0.05$  for Sketches and Raphael domains;
2.  $\lambda_{cdt} = 0.02$  for Caricature domain, Cat domain and Church domain tasks;
3.  $\lambda_{cdt} = 0.005$  for Cartoon, Roy Lichtenstein and Sunglasses domains.

We train 5000 iterations for Sketches domain, 3000 iterations for Raphael domain and Caricature domains, 2000 iterations for Sunglasses domain, 1250 iterations for Roy Lichtenstein domain, and 1000 iterations for Cartoon domain. We set learning rate  $lr = 0.002$  for all of the above decoder training.

For 1-shot training, we set  $\lambda_{cdt} = 0.005$  for face domain tasks, and train about 600 iterations for all the target domains.

For Cross-Domain Triplet Loss calculation, we find that adding a weight for  $d^+$  in main paper Eq.(2) helps reduce artifacts, i.e.,

$$\mathcal{L}_{cdt} = \mathbb{E}_{\{z_i \sim p_z(z)\}} \max(w \cdot d^+(z_i) - d^-(z_i) + \alpha, 0) \quad (15)$$

We set  $w = 1.5$  for Sketches domain,  $w = 2.0$  for other domains, and  $\alpha = 2$  for all target artistic domains.

## E.2 Style Encoder

**Architecture:** Our encoder consists of a feature extractor and a sub-encoder. For the feature extractor, we follow pSp [37] and adopt its FPN [27] based feature extractor design. For the sub-encoder, we compared different designs (detailed in Sec. D.2) and found transformer-based architecture achieves the best results.

**Training details:** We first train Style Encoder from scratch on FFHQ [23] dataset for 170,000 iterations in path-1 (mentioned in main paper section 3.2), and use the model as pretrained encoder model. Then, we train Style Encoder in dual path setting (both path-1 and path-2) for 70,000 iterations.

**Training details for encoder ablation studies:** For ablation study on dual-path, we train another 70,000 iterations only in path-1 from the pretrained encoder model to match our full encoder (dual path) setting.

**Hyper-parameters:** In the above training process, we set  $\lambda_{L_2} = 1.0$ ,  $\lambda_{lpips} = 0.8$ ,  $\lambda_{reg} = 0$ ,  $\lambda_{iden} = 0.1$  in main paper Eq.(9). And set  $\lambda_{z-predict} = 0.1$  and  $\lambda_{path1} = 1$  in main paper Eq.(11). We set the learning rate as  $lr = 0.0001$ .

**For more details, we provide the source code for closer inspection.**

## Appendix F Comparison with Neural Style Transfer and Image-to-Image Translation Methods

In this section, we provide more comparisons. For TGAN [47] and FreezeD [31], in the main paper we use an anchor sample space [33] for  $z$  to prevent overfitting. Here, we use the full sample space (i.e. sample  $z$  from Gaussian distribution), to evaluate the two comparison methods in their original settings. As shown in Fig. 22, the results of these two methods are prone to overfitting.

We further compare with 5 neural style transfer, image-to-image translation methods under 10-shot setting: Gatys[11], AdaIN[15], CycleGAN[54], UGATIT [25], and Toonify[35]. Qualitative comparison results are shown in Fig. 23. We find neural style transfer methods (Gatys, AdaIN) sometimes fail to capture the target cartoon style and generate results with artifacts. CycleGAN and UGATIT results are of lower quality under few-shot setting. Toonify results also contain artifacts. In comparison, our method generates high-quality artistic portraits.

## References

- Abdal, R., Qin, Y., Wonka, P.: Image2stylegan: How to embed images into the stylegan latent space? In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4431–4440 (2019) [3](#), [8](#)
- Abdal, R., Qin, Y., Wonka, P.: Image2stylegan++: How to edit the embedded images? In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8293–8302 (2020) [3](#)
- Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. ACM Transactions on Graphics (TOG) **40**(3), 1–21 (2021) [3](#), [13](#)
- Alaluf, Y., Patashnik, O., Cohen-Or, D.: Restyle: A residual-based stylegan encoder via iterative refinement. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021) [3](#)
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020) [9](#)
- Choi, Y., Choi, M., Kim, M., Ha, J., Kim, S., Choo, J.: StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8789–8797 (2018) [5](#)
- Chong, M.J., Forsyth, D.: Jojogan: One shot face stylization. arXiv preprint arXiv:2112.11641 (2021) [2](#), [4](#), [13](#), [17](#)
- Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4690–4699 (2019) [9](#)
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) [20](#)

10. Gal, R., Patashnik, O., Maron, H., Chechik, G., Cohen-Or, D.: Stylegan-nada: Clip-guided domain adaptation of image generators. arXiv preprint arXiv:2108.00946 (2021) [2](#), [4](#), [13](#)
11. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2414–2423 (2016) [2](#), [31](#), [36](#)
12. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015) [9](#)
13. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 2672–2680 (2014) [3](#)
14. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017) [10](#)
15. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1501–1510 (2017) [2](#), [3](#), [8](#), [31](#), [36](#)
16. Huang, X., Liu, M., Belongie, S.J., Kautz, J.: Multimodal unsupervised image-to-image translation. In: 15th European Conference (ECCV). pp. 179–196 (2018) [5](#)
17. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5967–5976 (2017) [2](#), [3](#), [5](#)
18. Jang, W., Ju, G., Jung, Y., Yang, J., Tong, X., Lee, S.: Stylecarigan: caricature generation via stylegan feature map modulation. ACM Transactions on Graphics (TOG) **40**(4), 1–16 (2021) [2](#), [5](#)
19. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: 14th European Conference (ECCV). pp. 694–711 (2016) [2](#)
20. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017) [10](#)
21. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T.: Training generative adversarial networks with limited data. In: Proc. NeurIPS (2020) [3](#)
22. Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. arXiv preprint arXiv:2106.12423 (2021) [3](#)
23. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4396–4405 (2019) [3](#), [6](#), [10](#), [30](#)
24. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of StyleGAN. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020) [2](#), [3](#), [5](#), [6](#), [29](#)
25. Kim, J., Kim, M., Kang, H., Lee, K.H.: U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. In: International Conference on Learning Representations (2020), <https://openreview.net/forum?id=BJlZ5ySKPH> [31](#), [36](#)
26. Li, Y., Zhang, R., Lu, J., Shechtman, E.: Few-shot image generation with elastic weight consolidation. In: Advances in Neural Information Processing Systems (2020) [2](#), [4](#)

27. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 936–944 (2017). <https://doi.org/10.1109/CVPR.2017.106> 3, 9, 30
28. Liu, M., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 700–708 (2017) 5
29. Liu, M.Y., Huang, X., Mallya, A., Karras, T., Aila, T., Lehtinen, J., Kautz, J.: Few-shot unsupervised image-to-image translation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10551–10560 (2019) 2, 5
30. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research 9(11) (2008) 10, 20
31. Mo, S., Cho, M., Shin, J.: Freeze the discriminator: a simple baseline for fine-tuning gans. arXiv preprint arXiv:2002.10964 (2020) 2, 4, 11, 12, 31, 35
32. Noguchi, A., Harada, T.: Image generation from small datasets via batch statistics adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2750–2758 (2019) 4
33. Ojha, U., Li, Y., Lu, C., Efros, A.A., Lee, Y.J., Shechtman, E., Zhang, R.: Few-shot image generation via cross-domain correspondence. In: CVPR (2021) 2, 4, 6, 7, 8, 11, 12, 28, 30, 31
34. Park, T., Efros, A.A., Zhang, R., Zhu, J.Y.: Contrastive learning for unpaired image-to-image translation. In: European Conference on Computer Vision. pp. 319–345 (2020) 11, 12
35. Pinkney, J.N., Adler, D.: Resolution dependent gan interpolation for controllable image synthesis between domains. arXiv preprint arXiv:2010.05334 (2020) 2, 5, 10, 31, 36
36. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763. PMLR (2021) 2, 4, 13
37. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D.: Encoding in style: a stylegan encoder for image-to-image translation. In: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2021) 3, 4, 6, 8, 9, 12, 30
38. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015) 7
39. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) 8
40. Song, G., Luo, L., Liu, J., Ma, W.C., Lai, C., Zheng, C., Cham, T.J.: Agilegan: Stylizing portraits by inversion-consistent transfer learning. ACM Trans. Graph. 40(4) (Jul 2021). <https://doi.org/10.1145/3450626.3459771>, <https://doi.org/10.1145/3450626.3459771> 2, 4, 5, 6, 7, 9, 12
41. Sun, Z., Cao, S., Yang, Y., Kitani, K.M.: Rethinking transformer-based set prediction for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3611–3620 (2021) 9
42. Tewari, A., Elgharib, M., Bharaj, G., Bernard, F., Seidel, H.P., Pérez, P., Zollhofer, M., Theobalt, C.: Stylerig: Rigging stylegan for 3d control over portrait images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6142–6151 (2020) 3

43. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D.: Designing an encoder for stylegan image manipulation. ACM Trans. Graph. **40**(4) (Jul 2021). <https://doi.org/10.1145/3450626.3459838>, <https://doi.org/10.1145/3450626.3459838> 3, 4, 6, 8, 12
44. Wang, T., Liu, M., Zhu, J., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional GANs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8798–8807 (2018) 2, 5
45. Wang, X., Tang, X.: Face photo-sketch synthesis and recognition. IEEE transactions on pattern analysis and machine intelligence **31**(11), 1955–1967 (2008) 10
46. Wang, Y., Gonzalez-Garcia, A., Berga, D., Herranz, L., Khan, F.S., Weijer, J.v.d.: Minegan: effective knowledge transfer from gans to target domains with few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9332–9341 (2020) 2, 4
47. Wang, Y., Wu, C., Herranz, L., van de Weijer, J., Gonzalez-Garcia, A., Raducanu, B.: Transferring gans: generating images from limited data. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 218–234 (2018) 2, 4, 11, 12, 31, 35
48. Yang, T., Ren, P., Xie, X., Zhang, L.: GAN prior embedded network for blind face restoration in the wild. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR. pp. 672–681 (2021) 3
49. Yaniv, J., Newman, Y., Shamir, A.: The face of art: landmark detection and geometric style in portraits. ACM Transactions on graphics (TOG) **38**(4), 1–15 (2019) 2, 10
50. Yi, R., Liu, Y., Lai, Y., Rosin, P.L.: ApdrawingGAN: Generating artistic portrait drawings from face photos with hierarchical GANs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10743–10752 (2019) 2
51. Yi, Z., Zhang, H.R., Tan, P., Gong, M.: Dualgan: Unsupervised dual learning for image-to-image translation. In: IEEE International Conference on Computer Vision (ICCV). pp. 2868–2876 (2017) 2, 5
52. Yu, F., Zhang, Y., Song, S., Seff, A., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015) 19
53. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018) 8, 9, 11
54. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 2242–2251 (2017) 2, 5, 31, 36
55. Zhu, P., Abdal, R., Femiani, J., Wonka, P.: Mind the gap: Domain gap control for single shot domain adaptation for generative adversarial networks. arXiv preprint arXiv:2110.08398 (2021) 2, 4, 13



Fig. 22: More comparisons with TGAN [47] and FreezeD [31]

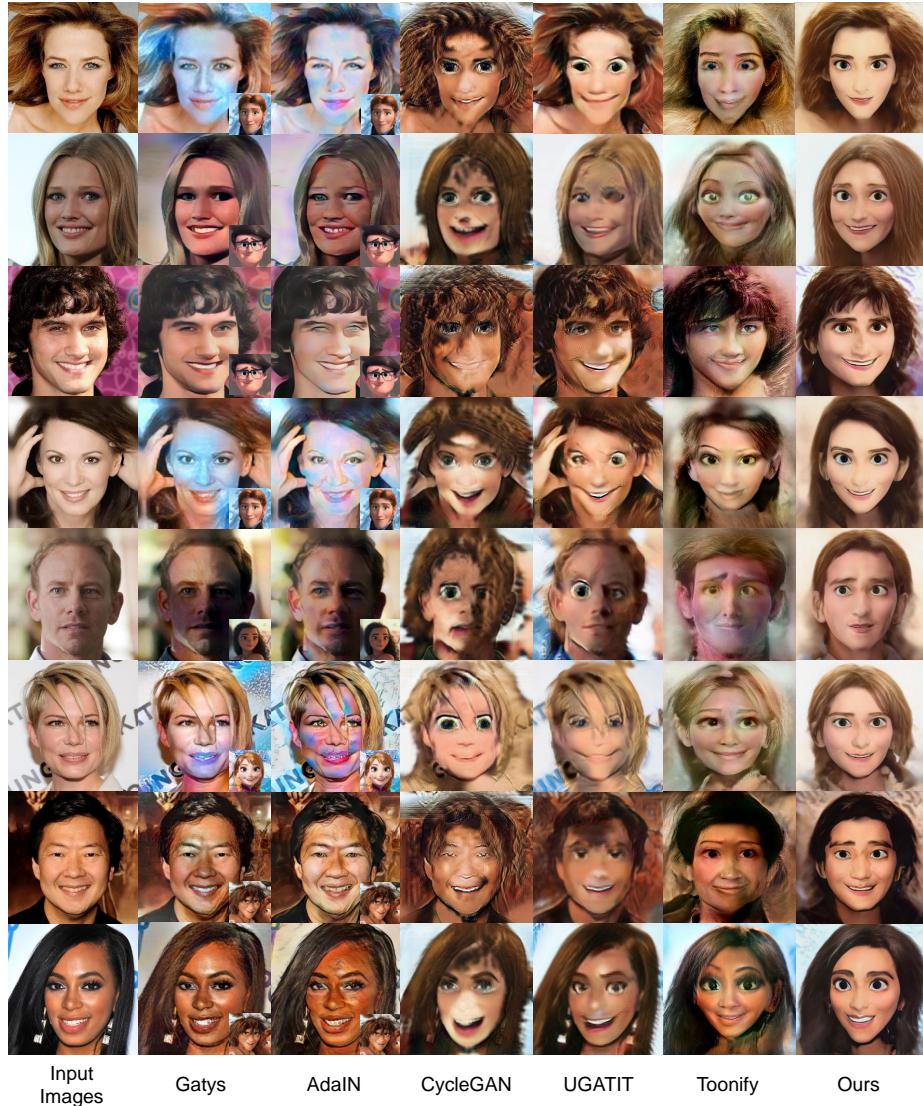


Fig. 23: Comparisons with more neural style transfer and image-to-image translation methods: Gatys [11], AdaIN [15], CycleGAN [54], UGATIT [25], and Toonify [35]