Bring Clipart to Life

Nanxuan Zhao [1], Shengqi Dang [2], Hexun Lin [2], Yang Shi [2], Nan Cao [2]
[1] Adobe Research [2] Tongji University

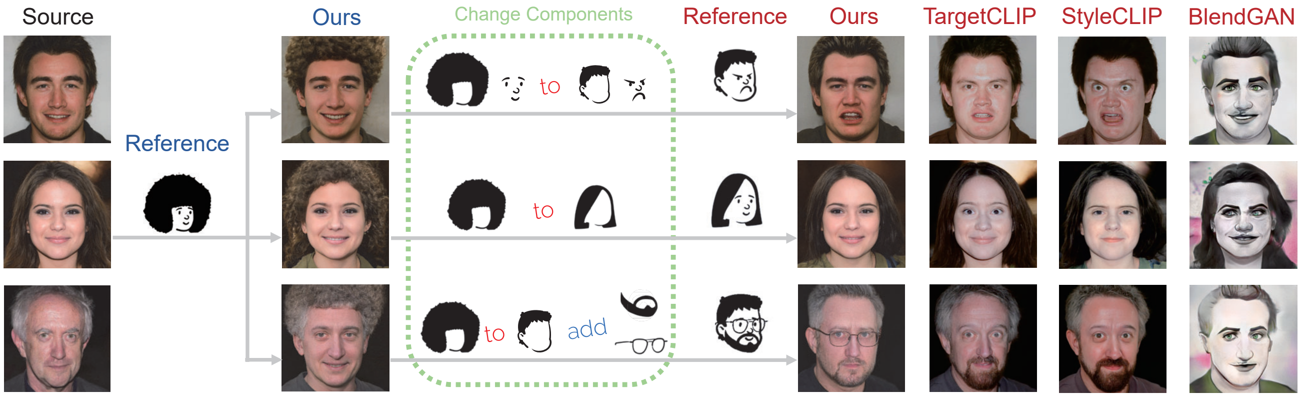

Figure 1: Clipart-driven facial editing. Given a clipart image, our model can transfer its facial attributes (*e.g.*, hair, beard, etc.) to the input source image. It allows the users to efficiently conduct facial editing by simply adding/swapping components in the reference clipart (as shown on the left). Compared with prior works [3, 28], our model can transfer facial attributes while maintaining the identity (as shown on the right). Clipart © from Open Peeps [37].

Abstract

The development of face editing has been boosted since the birth of StyleGAN. While previous works have explored different interactive methods, such as sketching and exemplar photos, they have been limited in terms of expressiveness and generality. In this paper, we propose a new interaction method by guiding the editing with abstract clipart, composed of a set of simple semantic parts, allowing users to control across face photos with simple clicks. However, this is a challenging task given the large domain gap between colorful face photos and abstract clipart with limited data. To solve this problem, we introduce a framework called ClipFaceShop [1] built on top of StyleGAN. The key idea is to take advantage of $\mathcal{W}+$ latent code encoded rich and disentangled visual features, and create a new lightweight selective feature adaptor to predict a modifiable path toward the target output photo. Since no pairwise labeled data exists for training, we design a set of losses to provide supervision signals for learning the modifiable path. Experimental results show that ClipFaceShop generates realistic and faithful face photos, sharing the same facial attributes as the reference clipart. We demonstrate that ClipFaceShop supports clipart in diverse styles, even in form of a free-hand sketch.

1. Introduction

Face editing aims to manipulate the attributes specified by the users while maintaining the non-modifiable attributes unchanged. With the rapid development of Generative Adversarial Network (GAN) [12], this task attracts many recent works [45, 32, 9] for achieving impressive results. In particular, because of the powerful disentangled latent space, StyleGAN [18] has become a de facto building block for generating realistic editing results. To express the user's intention, there are mainly three ways of editing including sketching, text-guiding, and exemplar photo. Unfortunately, these interaction methods may not enable effective and precise editing across different face photos for people without design experience.

[*] Nanxuan Zhao is with Adobe Research. E-mail: nanxuanzhao@gmail.com.

[*] Shengqi Dang, Hexun Lin, Yang Shi and Nan Cao are with Intelligent Big DataVisualization Lab, Tongji University. Nan Cao is the corresponding author. E-mail: dangsq123@tongji.edu.cn, linhexun@pku.edu.cn, {shiyang1230, nan.cao}@gmail.com.

[1] Code: https://github.com/dangsq/ClipFaceShop

More specifically, sketch-based face editing [30, 4, 25] allows users to directly draw strokes on top of the face, such as changing the size of eyes. Although simple, the quality of results often relies on the users' sketching skills and the input sketch cannot directly adapt to different face photos. By taking advantage of well-aligned textual-visual embedding space of CLIP [31], text-based face photo editing [28, 21] has appeared to control the facial attributes with simple textual instructions, such as making a face photo into "Emma Stone" style. Such a condition can flexibly change the facial attributes without finetuning on a specific dataset, but is unable to conduct fine-grained control given the ambiguity and high-level nature of the text description. Another stream of works [24, 13, 44] transfers the facial attributes from a reference photo to the target photo, allowing both fine-grained and cross-photo control. However, finding a perfect examplar photo is not an easy task and such transfer is a one-off operation that is hard to iteratively refine the results.

In this work, we propose a new interaction method for face editing using a mix-and-match clipart. Given a source photo and a reference clipart, we aim to transfer the attributes (*e.g.,* facial expression, beard, hairstyle, etc.) from the clipart to the face photo. Thanks to the growing clipart community, there are many well-curated libraries, such as OpenPeeps [37] and Avataaars [36]. Users can easily create the reference clipart by combining the components through simple clicks, as shown in Fig. 2. However, clipart-driven face editing poses many challenges. First, different from directly editing on the photo, there exists a large domain gap between the clipart and the natural photo to be influenced, not only on the identity. Since clipart is much more abstract than the natural photo and with fewer visual features, how to correctly match the facial attributes across domains and identities is a critical problem to be solved. Second, unlike face photos which have plenty of datasets with annotations (*e.g.,* segmentation), clipart is often limited by the data scale. Collecting pairwise datasets (*i.e.,* reference clipart and source photo) for training is impractical. A model that could learn such attribute transfer only from a single clipart is desired. Furthermore, facial attributes can be diverse, including intrinsic attributes (*e.g.,* beard) and accessories (*e.g.,* glasses). The model needs to transfer these attributes accurately without modifying the face identity.

To this end, we introduce a novel pipeline, namely *ClipFaceShop*, to tackle the above challenges and make the following technical contributions: 1) Rather than finetuning or changing StyleGAN's architecture, we take the strength of its latent code by learning a light-weight selective feature adaptor to conduct feature transformation. This can also effectively avoid overfitting with few training data; 2) To facilitate training without pairwise ground truth data, we design a set of loss terms. Besides the general identity and background preservation constraints, we further design transfer, similarity, and consistency losses by utilizing the semantic info encoded in the CLIP embedding. These terms help our adaptor to find a modifiable path to bridge the domain gap while maintaining the identity of the input source photo; 3) As not all features are equally contributed to different facial attributes, we conduct an experiment to analyze the layer-wise and channel-wise importance among features for selective feature modification to edit the face photo. We also take a two-stage training for inference to allow customization on the final results.

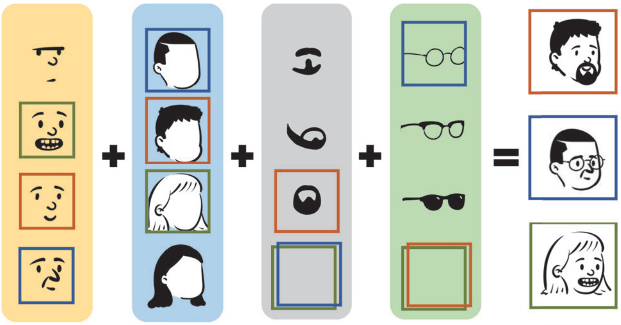

Figure 2: Examples of mix-and-match clipart, ©Open Peeps.

We have conducted extensive experiments and a user study to demonstrate the effectiveness of our model and shown its superiority over state-of-the-art methods of clipart-based face photo editing. Note that our model not only works well on very abstract clipart containing only black and white strokes, but it also works for more cartoon-like clipart with diverse styles. We also show the generality of our model on video in the supplementary material.

2. Related Work

Our paper falls into the domain of face photo editing, which has been actively studied recently. The milestone work StyleGAN [18] can generate high-resolution images and its learned disentangled latent space is friendly for editing tasks.

StyleGAN Inversion. StyleGAN inversion aims to find a latent vector that can reconstruct the input face photo. It can be categorized into optimization-based and encoder-based methods. The optimization-based methods obtain a latent vector of an input image via an online optimization process. For example, Image2StyleGAN [1] optimizes perceptual loss to obtain a latent code. PTI [33] first gets a rough latent vector by optimization, then fine-tunes the generator to incorporate the out-of-domain images. The encoder-based methods design an encoder to learn the latent code mapping. For example, pSp [32] designs an encoder based on a feature pyramid to map the features of input images into the latent space of $W+$. e4e [38] uses a latent discriminator to train an encoder. Besides, some other works like HyperStyle [2] and HyperInverter [7] focus on fine-tuning the weight of the StyleGAN for inversion. Furthermore, InterfaceGAN [35] and StyleFusion [16] study disentangled face representation in order to control properties more precisely. We borrow the idea of StyleGAN inversion and leverage the optimization-based approach to train an adaptor to help select proper features for face photo editing.

Text-driven Editing. Text-driven editing aims to edit a face photo guided by text prompts. TediGAN [40] proposes a multi-modal generation framework and controls images with textual features. By using the CLIP's [31] powerful semantic features, StyleCLIP [28] edits images with shift latent via three ways of text-guided manipulations. StyleMC [21] extends the online optimization process of the style layer in StyleGAN and improves its efficiency. StyleGAN-NADA [11] presents diverse style transfer by shifting the generator to characterized domains through CLIP prompts. Recently, NVIDIA's Textual Inversion [10] and Google's DreamBooth [34] apply prompt inversion to express the information of an image using a pseudoword and improve the generative outputs, increasing the performance on efficiency, image fidelity, and information accuracy. Text-driven algorithms produce reasonable visuals that correspond to text descriptions, showing the surprising power. However, as a high-level interaction format, text prompt has its limitation on expressing local fine-grained changes.

Sketch-driven Editing. Sketch-driven editing aims to edit a face photo referring to a sketch. Pix2Pix [15] and DeepFaceDrawing [4] modify the target image based on sketch editing via a pixel-level generation. FaceShop [30] allows users to directly draw sketches on photos for editing and trains a convolutional neural network with its own-collected dataset to render images with sketches. JoJoGAN [5] and Mind the GAP [47] stylize faces into reference domain (*e.g.,* sketch) by using StyleGAN's latent space. DeepFaceVideoEditing [25] extends sketch-based transfer to videos, and users can edit portrait attributes to each frame with sketches. One downside effect of sketch-based interaction is that users may take effort to draw multiple times before obtaining the satisfied results, and this drawing cannot be adapted to other photos easily.

Exemplar-driven Editing. Exemplar-driven editing aims to transfer the attribute from a reference face to a target face. It usually uses real face photos or facial segmentation as exemplars to guide editing. ELEGANT [41] creates a model that transfers the same sort of properties from one exemplar to another by swapping a portion of their encodings. MulGAN [13] encodes various attributes about specific regions to improve image generation and editing ability. Yin *et al*. [44] apply a geometry-aware flow to implement instance-level attribute transfer. MaskGAN [24] employs semantic masks as intermediate features of references to improve face editing while maintaining image fidelity. Barbershop [46] and Style Your Hair [20] propose models for solving hairstyle transfer with the help of segmentation masks. TargetCLIP [3] makes contributions to manipulating high-level semantic attributes by doing essence transfer with exemplars like real photos or cartoon photos. These works, although effective when using a natural face photo as the reference, will generate serious artifacts when directly referring to a clipart face.

3. Method

Following previous works [28, 1, 3], our *ClipFaceShop* also builds on top of the StyleGAN's $\mathcal{W}+$ space, which has shown promising results on various applications. However, directly finetuning the original StyleGAN model [5, 39] is inadequate for our setting given the limited unpaired training data. To fully utilize the disentangled latent space, we keep the pre-trained StyleGAN fixed and design a lightweight selective feature adaptor and a set of losses for finding the modifiable path along the $\mathcal{W}+$ space, as shown in Fig. 3. Formally, given a source photo I_{face} and the clipart reference I_{art}, we first encode the source photo into the $\mathcal{W}+$ space as w through e4e encoder [38]. Then our adaptor $\mathcal{A}$ transforms w into $w' = \mathcal{A}(w)$ before sending it into the pre-trained StyleGAN generator G for obtaining the transformed face photo.

3.1. Selective Feature Adaptor

The pre-trained StyleGAN has encoded rich facial attributes in the w latent code and has been proven that simple interpolation on w can generate faithful results [19, 1]. Inspired by domain adaption works [42, 47], to preserve this distribution as much as possible, we design our adaptor as a linear transformation of $w \in \mathbb{R}^{18\times512}$ as:

$$\mathcal{A}(w) = a \odot w + b,$$

where $a, b \in \mathbb{R}^{18\times512}$ are learned parameters during training, and $\odot$ denotes the element-wise multiplication. We regard b as the modifiable direction for transferring facial attributes. Note that during the inference time, our model acts differently from the training one, where we only use b for transforming the w for obtaining I_{out}:

$$I_{out} = G(w + b).$$

The insight here is that directly predicting the modifiable path b with w is difficult and parameter a serves as a domain adaption operator, closing the drastic domain gap between natural photo and clipart. In this way, the model can focus on finding the high-level attribute changes instead of low-level texture and style differences.

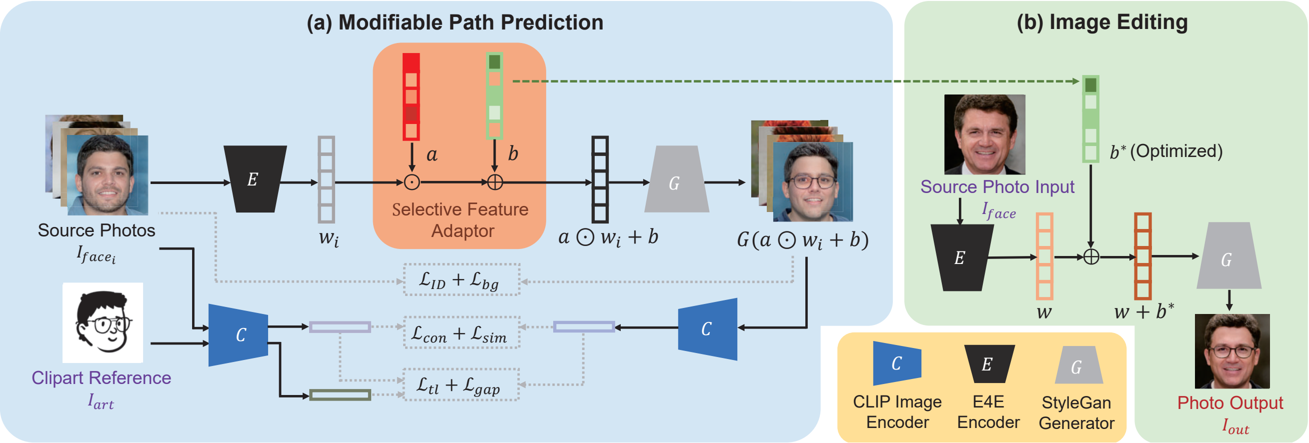

Figure 3: Our pipeline. Given a clipart reference image I_{art}, our model trains to transfer its facial attributes (*i.e.,* expression and red hair) to the source photos $\{I_{face_i}\}$ through a set of specially-designed losses. The lightweight selective feature adaptor aims to learn a modifiable path b in stage (a) for conducting image editing in stage (b) during the inference time.

3.2. Training Objectives

To learn the modifiable path b and domain adaption operator a, we design a set of losses in this subsection. They can be generally categorized into two groups, one for facial attributes transfer, and the other for identity and background preservation.

Transfer Loss. Inspired by TargetCLIP [3], we use the CLIP embedding for capturing semantic info. We also experiment with other backbones [8], and find CLIP performs the best because of its well-learned visual-semantic embedding. After transforming through the adaptor, the image should share high-level semantics with the reference clipart and we use cosine similarity to compute the loss as:

$$\mathcal{L}_{tl} = 1 - \frac{C\left(I_{art}\right) \cdot C(G(\mathcal{A}(w)))}{\|C\left(I_{art}\right)\|_2 \, \|C(G(\mathcal{A}(w)))\|_2}, \tag{1}$$

where C denotes the visual encoder of the CLIP model.

Gap Loss. To further reduce the domain gap, we create a standard clipart reference I_{std} by computing the average face over Open Peeps dataset in the $\mathcal{W}+$ space, and the changing direction in face photo domain should be aligned with the clipart domain and compute the loss as:

$$\Delta \hat{I_{art}} = C(I_{art}) - C(I_{std}), \Delta \hat{I_{face}} = C(G(\mathcal{A}(w))) - C(I_{face}). \tag{2}$$

$$\mathcal{L}_{gap} = 1 - \frac{\Delta \hat{I_{art}} \cdot \Delta \hat{I_{face}}}{\left\|\Delta \hat{I_{art}}\right\|_2 \left\|\Delta \hat{I_{face}}\right\|_2}. \tag{3}$$

Consistency Loss. When transforming facial features of the same clipart reference to different face photos, the shift directions should be correlated, and we measure this shift direction as the semantic difference in CLIP space: $\Delta I_{face} = C(G(\mathcal{A}(w))) - C(G(w))$. We enumerate all pairwise comparisons and define the loss as:

$$\mathcal{L}_{con} = \frac{1}{\binom{N}{2}} \sum_{i,j \in N} 1 - \frac{\Delta I_{face_i} \cdot \Delta I_{face_j}}{\|\Delta I_{face_i}\|_2 \left\|\Delta I_{face_j}\right\|_2}, \tag{4}$$

where N denotes the number of face photos in a training batch.

Identity Loss. To keep the identity of the input photo, we add this term for constraining the identity changes with the help of a pre-trained face recognition model called ArcFace [6] $R(\cdot)$:

$$\mathcal{L}_{ID} = 1 - \frac{R(G(\mathcal{A}(w))) \cdot R(I_{face})}{\|R(G(\mathcal{A}(w)))\|_2 \, \|R(I_{face})\|_2}. \tag{5}$$

Similarity Loss. To strengthen the identity preservation and avoid the influence of domain gap, we add this loss to measure the similarity of the face photo before and after transferring in CLIP visual embedding as:

$$\mathcal{L}_{sim} = 1 - \frac{C(G(\mathcal{A}(w))) \cdot C(I_{face})}{\|C(G(\mathcal{A}(w)))\|_2 \, \|C(I_{face})\|_2}. \tag{6}$$

Background Loss. We find that the background may be altered during the training and add this loss for constraining:

$$\mathcal{L}_{bg} = \|G(\mathcal{A}(w))P(G(\mathcal{A}(w))) - I_{face}P(I_{face})\|_1, \tag{7}$$

where P denotes the pre-trained face parsing model [23], for segmenting the background region out.

In summary, our total loss is:

$$\begin{aligned} L = \; & \lambda_{con}\mathcal{L}_{con} + \lambda_{tl}\mathcal{L}_{tl} + \lambda_{sim}\mathcal{L}_{sim} + \lambda_{ID}\mathcal{L}_{ID} \\ & + \lambda_{gap}\mathcal{L}_{gap} + \lambda_{bg}\mathcal{L}_{bg} + \lambda_{L_2}(\|a\|_2 + \|b\|_2), \end{aligned} \tag{8}$$

where $\{\lambda\}$ are used to balance among different loss terms. Besides, we also add a $L2$ regularization term during the optimization.

3.3. Selective Masking and Two-stage Training

There are many recent works [40, 19] studying the $\mathcal{W}+$ space and finding that different parts of the latent code control different visual features. Some are more suitable to conduct global changes, and some tend to be served for more local modifications. Motivated by StyleCLIP [28], to adopt more fine-grained changes, we compute layer-wise importance over the latent code for facial attributes. Then the importance values can be converted into a binary mask after thresholding (*i.e.,* , the top 5 most activated layers empirically). We impose this binary mask directly on a, b, since they are all in the same dimension with $w \in \mathbb{R}^{18\times 512}$ (*i.e.,* 18 layers with 18×512 channels) and we have fixed w for training.

We regard the clip embedding difference for every attribute i as the target direction Δt_i using a bank of sentence templates. We then perturb each channel c of the latent code with a random noise, the same as StyleCLIP [28] for obtaining the code direction Δw_c. The channel-wise importance for all N attributes can be easily computed by projecting the code direction onto the target direction as $\frac{\sum |\Delta w_c \cdot \Delta t_i|}{N}$. The final importance value of each layer is thus obtained by averaging all the channel importance values in that specific layer.

As shown in Fig. 4, our model can also transfer the color from the reference. To provide such flexibility for user control, we train our model in two stages. The first stage mainly focuses on structural changes of facial attributes by only using black-and-white reference clipart. While in the second stage, we change the standard clipart in L_{gap} to the black-white version of the input reference clipart to learn the color shift. Each stage will learn a separate modifiable path b_i, and the output is calculated as:

$$I_{out} = G(w + \alpha_1 b_1 + \alpha_2 b_2), \tag{9}$$

where users can use α_i to control the extent of each operation.

4. Experiments

We show our results on face photo editing in Fig. 4. As can be seen, our model can work on various face photos and references for changing different facial attributes (*e.g.,* hairstyle, expression, hair color, eyeglasses, etc.) while maintaining the facial identity. It can also generate uncommon faces, such as women with beard. Note that our model is not limited to this particular style of Open Peeps, but a general method for different kinds of reference medias (*e.g.,* cartoon) as shown in Fig. 10 and Fig. 11. In this section, we conduct several experiments and a user study to demonstrate the effectiveness and generality of our model.

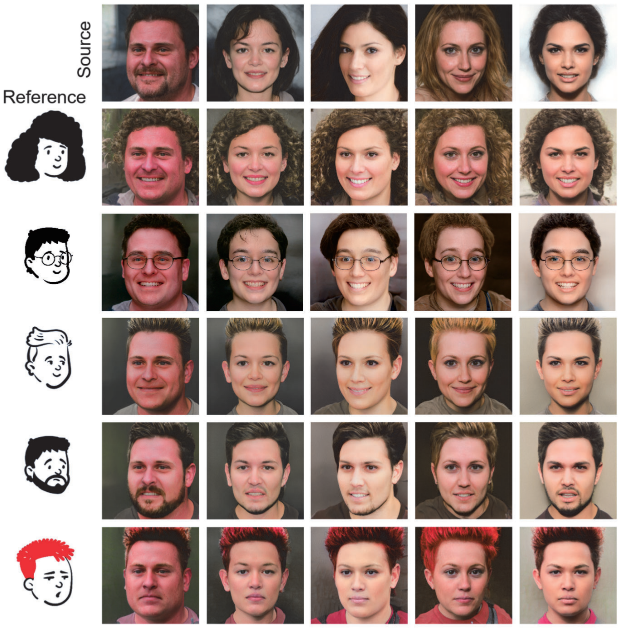

Figure 4: Our results on clipart-driven face photo editing. Our model can edit various facial attributes according to the input reference.

4.1. Implementation Details.

We use the pre-trained StyleGAN2 [19] on the FFHQ dataset [17] as our generator G, and ViT-B/32 as our CLIP model C. To train our model, we randomly sample face photos from FFHQ dataset. We mainly use clipart from Open Peeps [37] for training and testing, but we will show the generality of our model on other diverse data (*e.g.,* digital paintings and cartoons) in the following subsections. The optimizer is set to Adam with a default learning rate of 0.1. And each training takes 6∼8mins with a GPU RTX3090 under 300 epochs. We use different sets of hyperparameters for re-weighting loss in different stages. For the first stage, we set $\lambda_{id} = 1.5, \lambda_{sim} = 0.3, \lambda_{tl} = 1, \lambda_{con} = 0.1, \lambda_{gap} = 1, \lambda_{L_2} = 5e-5, \lambda_{bg} = 1e-7$, and for the second stage, we set $\lambda_{id} = 0.2, \lambda_{sim} = 0.1, \lambda_{tl} = 0.15, \lambda_{con} = 0.05, \lambda_{gap} = 0.5, \lambda_{L_2} = 3e-6, \lambda_{bg} = 1e-7$.

4.2. Comparisons

In this subsection, we compare our pipeline with the state-of-the-art methods both qualitatively and quantitatively.

Baselines. We compare with three methods including StyleCLIP [28], a work that manipulates face attributes based on semantics encoded in CLIP embedding;

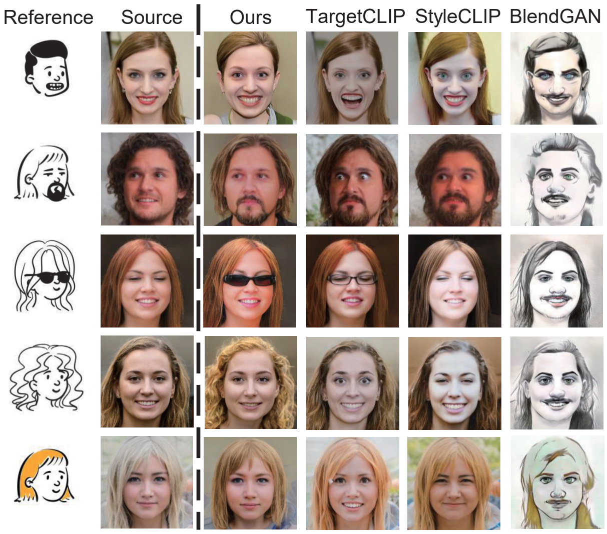

Figure 5: Comparison of different methods for clipart-driven face photo editing. Our method can better transfer the facial attributes from the reference while keeping the identity. The clipart © from Open Peeps [37].

Table 1: Quantitative comparison with state-of-the art methods.

Model	$ID\downarrow$	$FID\downarrow$	$SIM\uparrow$	$Color\uparrow$
BlendGAN	0.8418	105.14	0.66	0.52
StyleCLIP	0.5452	54.54	0.77	0.42
TargetCLIP	0.6642	49.35	0.68	0.60
Ours	0.5750	74.92	0.68	0.65

Table 2: Accuracy on different transferred facial attributes over different compared methods.

Model	$Beard_{Acc}\uparrow$	$Glass_{Acc}\uparrow$	$Hair_{Acc}\uparrow$	$Smile_{Acc}\uparrow$
BlendGAN	0.50	0.50	0.51	0.46
StyleCLIP	0.59	0.52	0.48	0.62
TargetCLIP	**0.78**	0.53	0.46	0.64
Ours	0.64	**0.56**	**0.58**	**0.80**

BlendGAN [27], generating stylized face photo based on digital painting; and TargetCLIP [3], using cartoon photos to conduct essence transfer on human faces. All three works are open-source projects and are easy to adapt to our task. For StyleCLIP, we replace the prompt encoding with the CLIP embedding of input reference clipart tailored to our task.

Testset and Metrics. We evaluate models quantitatively across 2,500 editing cases, covering 25 clipart from Open Peeps and 100 real photos from FFHQ dataset [17] with varied appearances that out of training set. We measure the performance from different perspectives: 1). Identity (**ID**), by implementing with Eq. 5. 2). Quality, by calculating **FID** (Fréchet Inception Distance) [14]. 3). Faithfulness, by computing cosine similarity (**SIM**) of CLIP embeddings between generated face photo and the input source one. 4). **Color**, by evaluating across 500 editing cases, covering 5 clipart (with different hair colors: black, grey, yellow, blue, and red) and 100 real face photo. The results are classified into these five color categories based on CLIP embedding similarity between each output image and color prompt and we use the accuracy to represent the color performance.

Results. We show the qualitative comparison in Fig. 5 over various design cases. As can be seen, our model can generate superior results over compared methods. It can transfer the facial attributes and accessories not only at the semantic level (*i.e.,* whether possess an attribute), but at the appearance/style level. For example, in the second and third row of the first group of results, our model can transfer the beard and eyeglasses (*i.e.,* black lenses) in the exact style of those in the reference clipart. Instead, BlendGAN prefers to generate less realistic results. Though TargetCLIP and StyleCLIP can successfully edit some attributes, they often have artifacts due to the large domain gap, such as glaring eyes and pale skin. Our model can reduce such domain gaps and maintain the identity of the input source photo.

The quantitative comparison is shown in Tab. 1. Since the facial attributes cannot be changed by StyleCLIP very often, it obtains the highest identity score and lower FID. BlendGAN fades the color of photos to mimic the style of input clipart, leading to a lower SIM score but a high FID. Though TargetCLIP gets better results on transferring facial attributes (*i.e.,* high SIM), it sacrifices the identity. Our model balances these terms well and obtains the highest score for the Color metric. As the SIM is influenced by the textural appearance which may not be accurate at the semantic level, we further compute accuracy over different facial attributes including beard (*i.e.,* w/ or w/o), eyeglasses (*i.e.,* w/ or w/o), hair (*i.e.,* long or short), and smile (*i.e.,* w/ or w/o) similar to the Color. The result is shown in Tab. 2. Our model outperforms the others in most cases, validating the effectiveness of our model in transferring facial attributes in such a challenging scenario.

4.3. User Study

We conduct a user study to visually compare results. We invited 30 participants in public from various backgrounds. Each participant was asked to fill up a questionnaire, and each questionnaire contains three sets of questions, covering 20 different editing cases randomly selected from our test set. In each design case, we displayed results of ours and baselines randomly and asked the participants to choose the best one based on 1). **Realistic**; 2). **Faithfulness**, whether it shares the same facial attributes as the reference clipart; and 3). **Identity**, whether it has the same identity

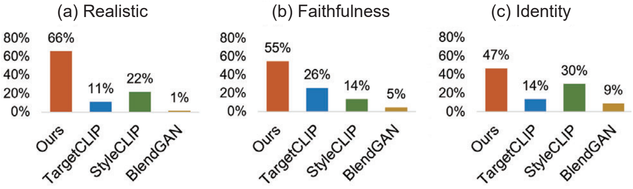

Figure 6: User study result. We show the user preference over different methods from three aspects.

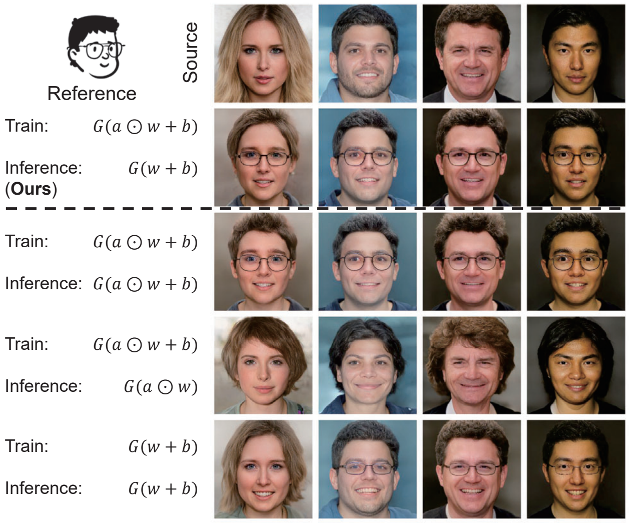
Reference
Source
Train: $G(a \odot w + b)$
Inference: $G(w + b)$
(Ours)
Train: $G(a \odot w + b)$
Inference: $G(a \odot w + b)$
Train: $G(a \odot w + b)$
Inference: $G(a \odot w)$
Train: $G(w + b)$
Inference: $G(w + b)$

Figure 7: The effectiveness of selective feature adaptor. We train and test on different variants of adaptors.

as the input photo. The results are shown in Fig. 6, and our model obtains the best result, outperforming the others by a large margin.

4.4. Ablation Study

The Effectiveness of Selective Feature Adaptor. We test different variants of the selective feature adaptor by training and testing in different ways, as shown in Fig. 7. By visualizing $a \odot w + b$ and $a \odot w$, we can find that a helps the model to obtain a more average face to reduce the domain gap, focusing more on modifiable path b prediction. Without the a by directly training with $w + b$ fails to obtain a consistent modifiable direction and cannot faithfully transfer the facial attributes across all photos.

The Effectiveness of Loss Terms. To validate different loss designs, we start with the transfer loss L_{tl} and gradually add the others one by one. We show the qualitative comparison in Fig. 8 and the quantitative comparison in Tab. 3. By adding L_{gap} and L_{con}, the photo shares more similar attributes as the reference clipart. By adding L_{bg}, L_{sim} and L_{ID}, the identity has been restored properly. Note that the visual quality (Fig. 8) has explicit improvement when adding different loss terms even though the metric increases a little.

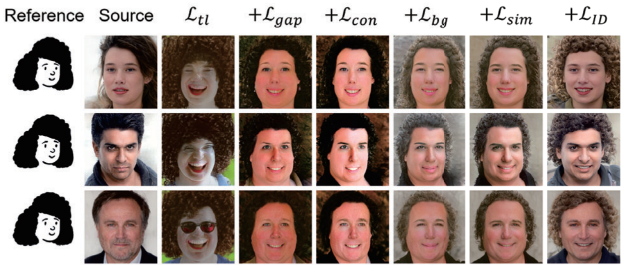

Figure 8: The effectiveness of different loss terms.

Table 3: The effectiveness of masking and different loss terms.

Model	$ID\downarrow$	$FID\downarrow$	$SIM\uparrow$	$Color\uparrow$
w/o mask	0.8865	103.38	0.72	0.59
L_{tl}	0.8858	66.00	**0.74**	0.52
$+L_{con}$	0.8951	66.25	**0.74**	0.52
$+L_{ID}$	0.5941	59.48	0.70	0.62
$+L_{sim}$	**0.5395**	**59.17**	0.70	0.54
$+L_{gap}$	0.6641	72.87	0.69	**0.82**
$+L_{bg}$ (Ours)	0.5750	74.92	0.68	0.65

Table 4: Accuracy on transferred facial attributes over different model variants over masking and loss terms.

Model	$Beard_{Acc}\uparrow$	$Glass_{Acc}\uparrow$	$Hair_{Acc}\uparrow$	$Smile_{Acc}\uparrow$
w/o mask	0.52	0.47	**0.61**	0.33
L_{tl}	**0.75**	0.73	0.49	0.63
$+L_{con}$	**0.75**	0.73	0.55	0.61
$+L_{ID}$	0.58	0.74	0.55	0.60
$+L_{sim}$	0.58	0.71	0.55	0.53
$+L_{gap}$	0.59	**0.75**	0.50	0.79
$+L_{bg}$ (Ours)	0.64	0.56	0.58	**0.80**

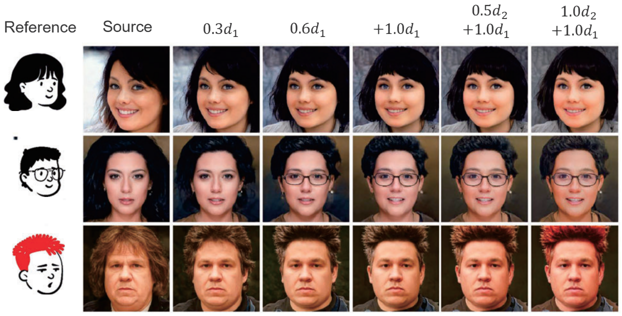

Figure 9: The effectiveness of two-stage training. Users are allowed to control the extent of transferred attributes by adjusting the parameters of each modifiable path.

The Effectiveness of Masking and Two-stage Train-

ing. as shown in Tab. 3, by removing the masks during training, the performance drops a lot, indicating the effectiveness of the masking operation. As mentioned in Sec. 3.3, we use two-stage training to provide users more controllability over results. We alter the weights of learned modifiable paths and show the result in Fig. 9. As the weight increases, the transferred facial attributes become stronger. Users can manipulate a slider bar to flexibly control the extent of modification during editing.

4.5. Generality to Diverse Reference Styles

To examine the generality of our model, we also test on reference images with different styles, such as rough sketches and cartoons. We show the results compared with previous works in Fig. 10 and more results in Fig. 11. As can be seen, rather than being as a customized method for a unique style, our model is a general method that can work well on various types of reference styles. For example, it can transfer hairstyle from a hand-drawn coarse sketch (Fig. 10, fourth row) and hair color with facial expression from colorful cartoon images (Fig. 10, fifth row).

4.6. Discussions

Though our model can achieve promising results, it still has some limitations given such a challenging task. First, since our model is based on a pre-trained StyleGAN, it is hard to generate out-of-distribution photos, such as a person wearing a hat. We believe that using an advanced generator trained on a more diverse dataset will mitigate this problem. Second, some attributes are entangled, such as smile and expression. For example, as shown in the last row of the photos in Fig. 10, the eyes are closed given a sad face, and in Fig. 7, the teeth go with a smiling face. One possible solution is to find latent hyperplanes and map latent to a more disentangled subspace, as discussed in InterfaceGAN [35]. Third, as shown in the supplementary video, when applying our model directly on a talking face, it keeps the mouth close all the time because of the reference clipart. Since our model is designed for static images, adding guidance of optical flow may help in such a dynamic case. Last, to avoid potential ethical issues, we advocate users for choosing adequate clipart for editing.

5. Conclusion

In this work, we introduce *ClipFaceShop*, a novel pipeline for clipart-driven face photo editing. Our model can transfer the facial attributes from abstract clipart to the face photo while preserving the identity. This allows users to edit a photo easily by adding/removing components in clipart through clicks. To achieve this and resolve the large domain gap, we propose a selective feature adaptor with masking and a set of losses. We have demonstrated the effectiveness of our model through extensive experiments.

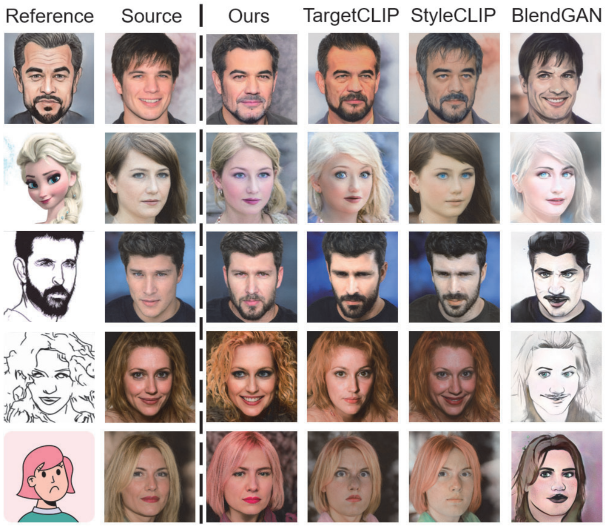

Figure 10: Generality to different reference styles. We show results under diverse source and reference image pairs, and our model can transfer the facial features faithfully. The clipart © from Vue Color Avatar [22], cartoon from Disney Animation, and sketches from Toonify [29], PSP [32] and ArtLine [26].

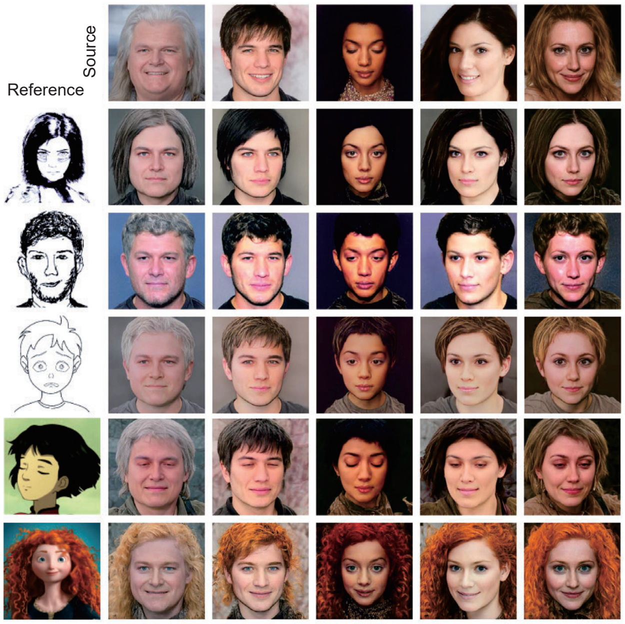

Figure 11: More results on diverse reference images. The cartoon © from Disney Animation, B&T Animation, and sketches from ArtLine [26], PSP [32], EmoG [43].

We believe editing with such abstract art has big potential in practical applications, and we are the very initial step in this direction. We will release our code and hope our model can inspire more future work.

6. Acknowledgments

Nan Cao is the corresponding author. This work was supported by NSFC 62072338 and 62061136003.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4432–4441, 2019. 2, 3

[2] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF conference on computer Vision and pattern recognition (CVPR)*, pages 18511–18521, 2022. 3

[3] Hila Chefer, Sagie Benaim, Roni Paiss, and Lior Wolf. Image-based clip-guided essence transfer. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*, pages 695–711, 2022. 1, 3, 4, 6

[4] Shu-Yu Chen, Wanchao Su, Lin Gao, Shihong Xia, and Hongbo Fu. Deepfacedrawing: Deep generation of face images from sketches. In *ACM Transactions on Graphics (TOG)*, volume 39, pages 72–1. ACM New York, NY, USA, 2020. 2, 3

[5] Min Jin Chong and David Forsyth. Jojogan: One shot face stylization. In *European Conference on Computer Vision (ECCV)*, pages 128–152, 2022. 3

[6] 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 (CVPR)*, pages 4690–4699, 2019. 4

[7] Tan M Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son Hua. Hyperinverter: Improving stylegan inversion via hypernetwork. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11389–11398, 2022. 3

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 4

[9] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 12873–12883, 2021. 1

[10] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2111.09876*, 2022. 3

[11] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022. 3

[12] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *In Advances in neural information processing systems, pages 2672–2680*, 2014. 1

[13] Jingtao Guo, Zhenzhen Qian, Zuowei Zhou, and Yi Liu. Mulgan: Facial attribute editing by exemplar. *arXiv preprint arXiv:1912.12396*, 2019. 2, 3

[14] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, 30, 2017. 6

[15] 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 (CVPR)*, pages 1125–1134, 2017. 3

[16] Omer Kafri, Or Patashnik, Yuval Alaluf, and Daniel Cohen-Or. Stylefusion: Disentangling spatial segments in stylegan-generated images. *ACM Transactions on Graphics (TOG)*, 41(5):15, oct 2022. 3

[17] Tero Karras, Samuli Laine, and Timo Aila. Ffhq dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition(CVPR)*, pages 4401–4410, 2019. 5, 6

[18] 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 (CVPR)*, pages 4401–4410, 2019. 1, 2

[19] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 8110–8119, 2020. 3, 5

[20] Taewoo Kim, Chaeyeon Chung, Yoonseo Kim, Sunghyun Park, Kangyeol Kim, and Jaegul Choo. Style your hair: Latent optimization for pose-invariant hairstyle transfer via local-style-aware hair alignment. In *European Conference on Computer Vision (ECCV)*, pages 188–203, 2022. 3

[21] Umut Kocasari, Alara Dirik, Mert Tiftikci, and Pinar Yanardag. Stylemc: multi-channel based fast text-guided image generation and manipulation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 895–904, 2022. 2, 3

[22] Leo Ku. Vue color avatar. *https://github.com/Codennnn/vue-color-avatar*, 2021. 8

[23] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Celeba dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5549–5558, 2020. 4

[24] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5549–5558, 2020. 2, 3

[25] Feng-Lin Liu, Shu-Yu Chen, Yukun Lai, Chunpeng Li, Yue-Ren Jiang, Hongbo Fu, and Lin Gao. Deepfacevideoediting: Sketch-based deep editing of face videos. In *ACM Transactions on Graphics (TOG)*, volume 41, page 167. Association for Computing Machinery, 2022. 2, 3

[26] Vijish Madhavan. Artline. *https://github.com/vijishmadhavan/ArtLine*, 2020. 8

[27] Liu Mingcong, Li Qiang, Qin Zekui, Zhang Guoxin, Wan Pengfei, and Zheng Wen. Blendgan: implicitly gan blending for arbitrary stylized face generation. *Advances in Neural Information Processing Systems*, 34:29710–29722, 2021. 6

[28] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2085–2094, 2021. 1, 2, 3, 5

[29] Justin Pinkney. Toonify. *https://toonify.photos/*, 2020. 8

[30] Tiziano Portenier, Qiyang Hu, Attila Szabó, Siavash Arjomand Bigdeli, Paolo Favaro, and Matthias Zwicker. Faceshop: Deep sketch-based face image editing. In *ACM Transactions on Graphics (TOG)*, volume 37, page 13, New York, NY, USA, jul 2018. Association for Computing Machinery. 2, 3

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021. 2, 3

[32] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 2287–2296, 2021. 1, 2, 8

[33] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. In *ACM Transactions on Graphics (TOG)*, volume 42, pages 1–13. ACM New York, NY, 2022. 2

[34] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2111.09876*, 2022. 3

[35] 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 (CVPR)*, pages 9243–9252, 2020. 3, 8

[36] Pablo Stanley. Avataaars dataset. *https://www.avataaars.com/*, Accessed on 2023. 2

[37] Pablo Stanley. Openpeeps dataset. *https://www.openpeeps.com/*, Accessed on 2023. 1, 2, 5, 6

[38] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. In *ACM Transactions on Graphics (TOG)*, volume 40, pages 1–14. ACM New York, NY, USA, 2021. 3

[39] Sheng-Yu Wang, David Bau, and Jun-Yan Zhu. Rewriting geometric rules of a gan. In *ACM Transactions on Graphics (TOG)*, volume 41, pages 1–16. ACM New York, NY, USA, 2022. 3

[40] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 2256–2265, 2021. 3, 5

[41] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 168–184, 2018. 3

[42] Ceyuan Yang, Yujun Shen, Zhiyi Zhang, Yinghao Xu, Jiapeng Zhu, Zhirong Wu, and Bolei Zhou. One-shot generative domain adaptation. *arXiv preprint arXiv:2111.09876*, 2021. 3

[43] Shi Yang, Cao Nan, Ma Xiaojuan, Chen Siji, and Liu Pei. Emog: supporting the sketching of emotional expressions for storyboarding. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2020. 8

[44] Weidong Yin, Ziwei Liu, and Chen Change Loy. Instance-level facial attributes transfer with geometry-aware flow. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9111–9118, 2019. 2, 3

[45] 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 (ICCV)*, pages 2223–2232, 2017. 1

[46] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka. Barbershop: Gan-based image compositing using segmentation masks. *ACM Transactions on Graphics (TOG)*, 40(6), dec 2021. 3

[47] Peihao Zhu, Rameen Abdal, John Femiani, and Peter Wonka. Mind the gap: Domain gap control for single shot domain adaptation for generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2022. 3