Computer Vision and Image Understanding 207 (2021) 103205

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Image style disentangling for instance-level facial attribute transfer

Xuyang Guo [a,b], Meina Kan [a,*], Zhenliang He [a,b], Xingguang Song [c], Shiguang Shan [a]

[a] Key Lab of Intelligent Information Processing of Chinese Academy of Sciences, Institute of Computing Technology, Beijing 100190, China
[b] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, China
[c] Huawei Technologies Company Ltd., Shenzhen 518129, China

ARTICLE INFO

Communicated by Nikos Paragios

Keywords:
Instance-level facial attribute transfer
Image to image translation
Generative adversarial network
Weakly supervised style learning

ABSTRACT

Instance-level facial attribute transfer aims at transferring an attribute including its style from a source face to a target one. Existing studies have limitations on fidelity or correctness. To address this problem, we propose a weakly supervised style disentangling method embedded in Generative Adversarial Network (GAN) for accurate instance-level attribute transfer, using only binary attribute annotations. In our method, the whole attributes transfer process is designed as two steps for easier transfer, which first removes the original attribute or transfers it to a neutral state and then adds the attributes style disentangled from a source face. Moreover, a style disentangling module is proposed to extract the attribute style of an image used in the adding step. Our method aims for accurate attribute style transfer. However, it is also capable of semantic attribute editing as a special case, which is not achievable with existing instance-level attribute transfer methods. Comprehensive experiments on CelebA Dataset show that our method can transfer the style more precisely than existing methods, with an improvement of 39% in user study, 16.5% in accuracy, and about 3.3 in FID.

1. Introduction

Facial attribute manipulation is a challenging task in computer vision, which is widely used in many applications such as entertainment, automatic photoshop, face recognition, etc. Most existing studies (Choi et al., 2018; He et al., 2019; Romero et al., 2019; Xu et al., 2019; Upchurch et al., 2017; Zhao et al., 2018; Zhang et al., 2018; Lample et al., 2017; Larsen et al., 2016; Chen et al., 2018; Wu et al., 2019) focus on *semantic-level facial attribute editing*, that aims at adding or removing specific attributes, e.g. adding a mustache or removing bangs. These methods can get results with high fidelity, but they only care about whether an attribute is added or not and do not care about what the *style* of the attribute is. However, people may be more interested in adding an attribute with a specific style. For example, can we copy the bangs style from a movie star to our head? This kind of task is called *instance-level facial attribute transfer*, i.e. to transfer the attribute style from one image to another as shown in Fig. 1, which is more flexible and controllable than semantic-level editing.

Instance-level facial attribute transfer is more challenging because of lacking annotation — precise attribute style is extremely hard to be defined and annotated. Therefore, we resort to the more easily accessible binary attribute annotation (i.e. with or without an attribute, weaker supervision than the precise attribute style). Existing studies of instance-level facial attribute transfer are still preliminary. Several semantic-level editing methods (He et al., 2019; Romero et al., 2019; Xu et al., 2019) can achieve multi-mode editing of an attribute, i.e. adding an attribute with several choices of coarse styles, such as left bangs, right bangs, or thin bangs, but cannot transfer the attribute from one image to another. Some works are specialized for *specific attributes*, including expression (Ding et al., 2018), eyeglasses (Hu et al., 2020), makeup (Li et al., 2018; Chang et al., 2018), and global image style (Huang et al., 2018; Lee et al., 2020b; Yu et al., 2019; Choi et al., 2020). However, these methods are not applicable for transferring of various attributes.

Recently, several works (Zhou et al., 2017; Xiao et al., 2018a,b; Yin et al., 2019; Lee et al., 2020a) provide general solutions for instance-level attribute transfer. GeneGAN (Zhou et al., 2017) learns an encoder to disentangle the attribute with style from the attribute-unrelated information, then combines one's style and another's attribute-unrelated information to achieve the transfer. DNA-GAN (Xiao et al., 2018a) is a multi-attribute extension of GeneGAN. Inspired by GeneGAN and DNA-GAN, ELEGANT (Xiao et al., 2018b) achieves the transfer by exchanging attribute-related latent codes between two faces, which is capable of higher resolution editing with the more realistic results. A common problem with these methods is that they lack effective constraints on the variety of styles, resulting in a less accurate transfer. For more accurate transfer of attributes, GeoGAN (Yin et al., 2019) proposes a geometric approach to align the pose of faces and then copies the attribute-related part from the source image to the target

* Corresponding author.
E-mail addresses: xuyang.guo@vipl.ict.ac.cn (X. Guo), kanmeina@ict.ac.cn (M. Kan), zhenliang.he@vipl.ict.ac.cn (Z. He), songxingguang@huawei.com (X. Song), sgshan@ict.ac.cn (S. Shan).

https://doi.org/10.1016/j.cviu.2021.103205
Received 23 September 2020; Received in revised form 24 March 2021; Accepted 27 March 2021
Available online 30 March 2021

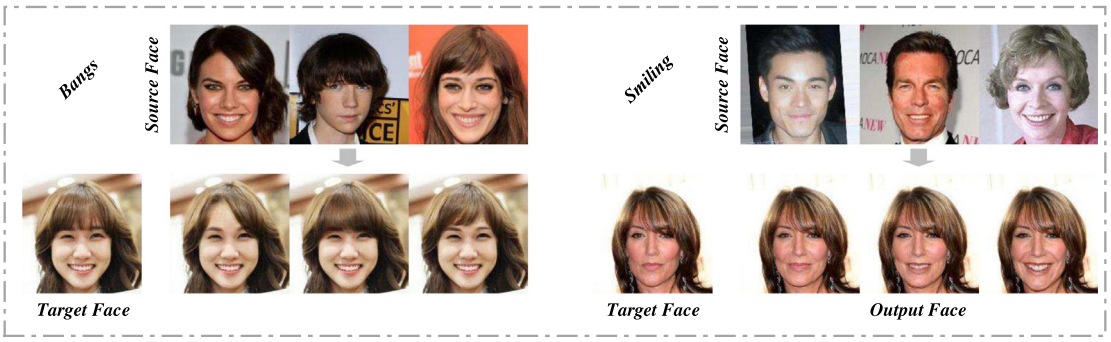


Fig. 1. Illustration of instance-level facial attribute transfer, that transfers style of *bangs* and *smiling* from several source faces to a target face respectively.

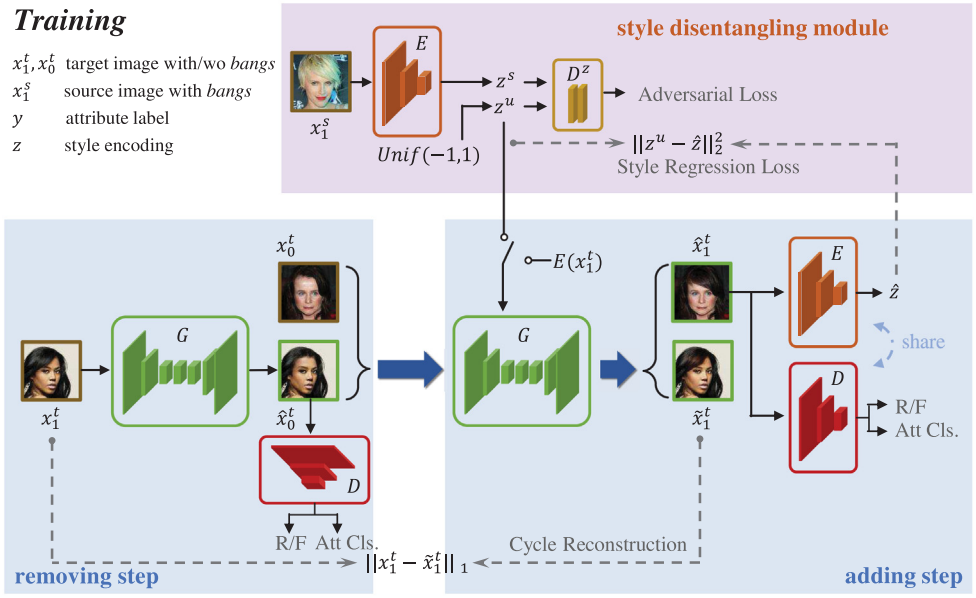


Fig. 2. An overview of our Style Disentangling GAN (STD-GAN). The whole process includes an attribute removing step and an attribute adding step. Style disentangling module is used to extract style information from a source image via an encoder E.

one. But it still cannot perform well for those attribute scattering in large areas such as *mouth open* and *hair color*. To accurately manipulate the attribute area, MaskGAN (Lee et al., 2020a) uses an attribute segmentation mask to enforce the target face to have the same attribute scope as the source one. However, it cannot deal with color or texture-related details. Besides, the annotation of attribute segmentation is expensive to acquire during training and testing.

In summary, the existing state-of-the-art instance-level transfer methods for general attributes are GeoGAN and ELEGANT. However, these methods have their limitations. GeoGAN can maintain the correct style for some attributes like *eyeglasses* through a geometric method, but it cannot perform well for attributes scattering in large areas, and the fidelity of edited images is not satisfactory. ELEGANT can generate relatively high-fidelity images but lacks the constraint of the correctness of style coding during its training. Therefore, a method that can ensure both the fidelity and correctness of the transferred attribute style is desired. Aiming for this, we propose a method called *STyle Disentangling GAN (STD-GAN)* which resolves the limitations of existing methods.

In our method, the transferring process is designed as two steps for easier instance-level attribute transfer: (1) a removing step that first removes the original attribute from the target image or transfers it to a neutral state; (2) an adding step that adds the attribute with style extracted from a source image. Moreover, a style disentangling module is proposed to disentangle the style code from face identity. The style code is used as input of a generator in the adding step to generate a new image with the same attribute style. The model is trained with the binary attribute annotation indicating with or without an attribute. Primarily, the proposed method STD-GAN aims for accurate instance-level attribute transfer. As a special case, STD-GAN degenerates to a method that is applicable for semantic attribute-editing when there is no source image for reference. To our best knowledge, our STD-GAN is the first one that can simultaneously be capable of instance-level attribute transfer and semantic attribute-editing.

Briefly, our contributions are summarized in three-folds:

1. We propose a novel style disentangling GAN to explicitly encode the style of each instance from a style disentangling module for precise instance-level facial attribute transfer. The method is learned with only binary attribute labels.
2. In a special case of no source image, STD-GAN degenerates to a method that is applicable for semantic attribute-editing, making our method capable of both instance-level attribute transfer and semantic attribute-editing.
3. Our method achieves superior performance compared to the state-of-the-art in many attributes in terms of visual comparison, user study, and quantitative evaluation.

2. Style disentangling GAN (STD-GAN)

For an attribute A, given a source image x_1^s and a target image x^t, our goal is to transfer the attribute style of the source image x_1^s

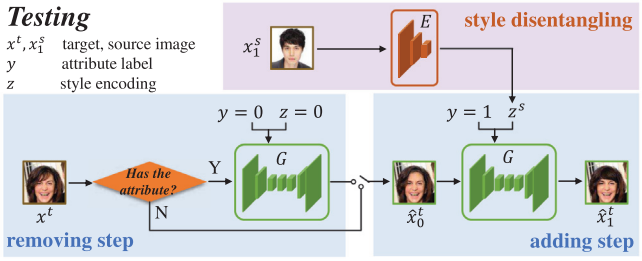


Fig. 3. Illustration of testing with our STD-GAN. Take *bangs* as an example, the testing phase aims at transferring the attribute style of source image x_1^s to the target image x^t. If x^t has the attribute, firstly remove the original attribute to get $\hat{x}_0^t$, otherwise $\hat{x}_0^t = x^t$. Then, $\hat{x}_0^t$ is fed into the generator with the style code z^s extracted from x_1^s as condition to get final image $\hat{x}_1^t$.

to the target x^t, where 1 denotes the image with the attribute A. The training data for this problem are images with only binary attribute annotations, denoted as $X = \{(x, y)\}$ where x is an image and $y \in \{0, 1\}$ is the corresponding attribute label. $y = 0$ indicates that x does not have the attribute A or has a neutral state (e.g. black for *hair color*, natural expression for *smiling*), and $y = 1$ indicates that x has the attribute A.

2.1. Formulation

Our STD-GAN for instance attribute transfer consists of two steps as shown in Fig. 3: (1) removing the original attribute from the target image, (2) transferring the attribute style from the source image to the target. Dividing the process into two steps makes each step focus on an easier task. We design a generator $G(x, y, z)$ for these two steps, where $y \in \{0, 1\}$ indicates removing or adding process, z indicates the style code of the attribute.

2.1.1. Removing step

Given a target image x^t, if it has the attribute A, this step removes the attribute or transfers it to a neutral state to avoid its interference with the attribute information from the source image, formulated as:

$$\hat{x}_0^t = \begin{cases} G(x^t, y = 0, z = 0) & \text{if } x^t \text{ has the attribute } A \\ x^t & \text{if } x^t \text{ does not have the attribute } A \end{cases} \tag{1}$$

where the second line means that if x^t does not have the attribute A, the removing step is just skipped.

2.1.2. Adding step

This step further adds the desired attribute to generate $\hat{x}_1^t$ which has the same attribute style as x_1^s:

$$\hat{x}_1^t = G(\hat{x}_0^t, y = 1, z^s), \quad \text{with } z^s = E(x^s), \tag{2}$$

where $y = 1$ indicates adding the attribute A, and $z^s \in \mathbb{R}^n$ is the attribute style extracted from the source image x^s via the encoder E. Through the removing and adding steps, the attribute style is transferred from x^s to x^t.

2.2. Objective of STD-GAN

The main modules to be optimized are the generator G and the style disentangling encoder E. The objective of them and the training process are presented below.

2.2.1. Objective of the generator G

The generator G aims at both removing and adding. For a target image x_1^t with the attribute, G can remove its attribute by

$$\hat{x}_0^t = G(x_1^t, y = 0, z = 0), \tag{3}$$

where $\hat{x}_0^t$ should *not have the attribute anymore, and look realistic*. Similarly, for those images without the attribute such as x_0^t, the attribute can be added by

$$\hat{x}_1^t = G(x_0^t, y = 1, z^u), \tag{4}$$

where z^u is a given style code sampled from a uniform distribution $Unif(-1, 1)$.

Like most adversarial generative methods (Odena et al., 2017; Choi et al., 2018; He et al., 2019), the above two objectives can be easily achieved by playing an adversarial game between the generator G and a discriminator D. This discriminator D is used for both realistic judgment and attribution classification. Specifically, $D(x)$ takes an image x as input and outputs two results for classification of real/fake and attribute respectively, denoted as D_{adv} and D_{cls}.

Fidelity Loss: To ensure the fidelity of those generated images from G, we adopt WGAN (Arjovsky et al., 2017) for the adversarial learning between the G and D as below:

$$\min_G \mathcal{L}_{adv}^g = -\mathbb{E}_{\hat{x}\sim p_g}[D_{adv}(\hat{x})], \tag{5}$$

$$\min_{\|D\|_L \leqslant 1} \mathcal{L}_{adv}^d = -\mathbb{E}_{x\sim p_r}[D_{adv}(x)] + \mathbb{E}_{\hat{x}\sim p_g}[D_{adv}(\hat{x})], \tag{6}$$

where p_g is the distribution of generated images and p_r is the distribution of real images, $\hat{x}$ denotes images generated from G including $\hat{x}_0^t$ and $\hat{x}_1^t$, x denotes real images including x_1^t and x_0^t. The discriminator makes efforts to distinguish the real images from the generated images, while the generator aims to generate real images that can fool the discriminator. Here, WGAN-GP (Gulrajani et al., 2017) is used for optimization.

Attribute Loss: To ensure the generated images to own the expected attribute, the attribute classification loss is exploited as below:

$$\min_D \mathcal{L}_{cls}^d = \mathbb{E}_{x\sim p_r}[-log D_{cls}(y^x|x)], \tag{7}$$

$$\min_G \mathcal{L}_{cls}^g = \mathbb{E}_{\hat{x}\sim p_{g|y}}[-log D_{cls}(y|\hat{x})], \tag{8}$$

where the $\mathcal{L}_{cls}^d$ aims to optimize an attribute classifier by using the real image x and its own attribute label $y^x \in \{0, 1\}$. $\mathcal{L}_{cls}^g$ aims to optimize the generator G to make the generated image $\hat{x}$ satisfy the given attribute via the judgment from the discriminator D. Here, $p_{g|y}$ means the editing result of G given $y \in \{0, 1\}$.

To well preserve attribute-irrelevant region, G is implemented as a residual architecture (Shen and Liu, 2017), i.e. $G(\cdot) = x + N(\cdot)$ where N is the network to output a residual image.

Style Loss: The above two losses in Eqs. (5) and (8) for G ensure that the transferred image $\hat{x}_1^t$ has the desired attribute, but the attribute style of the transferred image is not necessarily the same as that of the source image. Aiming for instance-level style transfer, a style regression objective is formulated as below:

$$\min_{G,E} \mathcal{L}_{mi} = \mathbb{E}_{z^u} \|z^u - \hat{z}\|_2^2 = \mathbb{E}_{z^u} \left\| z^u - E(G(x_0^t, y = 1, z^u)) \right\|_2^2, \tag{9}$$

where z^u is a style code randomly sampled from $Unif(-1, 1)$, and $\hat{z}$ is the style code of the transferred image $\hat{x}_1^t$. This style loss aims at enforcing the attribute style $\hat{z}$ of the transferred image to be the same as the given attribute style code z^u. Theoretically, sharing the same idea as InfoGAN (Chen et al., 2016), minimizing $\mathbb{E}_{z^u} \|z^u - \hat{z}\|_2^2$ is equivalent to maximizing the mutual information between $\hat{x}_1^t$ and z^u, which enables G to manipulate the style of an image by the style code z^u.

Moreover, for more precise control of style in the adding step, a cycle reconstruction objective is additionally introduced:

$$\min_{G,E} \mathcal{L}_{rec} = \mathbb{E}_{x_1^t} \left\| x_1^t - \tilde{x}_1^t \right\|_1 = \mathbb{E}_{x_1^t} \left\| x_1^t - G(\hat{x}_0^t, 1, E(x_1^t)) \right\|_1, \tag{10}$$

Algorithm 1: Training algorithm of STD-GAN

input : Images $\{x\}$ and attribute labels $\{y\} \in \{0, 1\}$;
output: Generator G and the style encoder E;
Initialize G, E, D, D^z, where E, D share network;
while *not converge* **do**
  Sample batch of images $x_1^t, x_0^t \in \{x\}$ with/without attribute, style code $z^u \sim Unif(-1, 1)$;
  Compute removing/adding/reconstruction results $\hat{x}_0^t, \hat{x}_1^t, \tilde{x}_1^t$ using Eqs. (3), (4) and (10);
  Compute overall objective $\mathcal{L}^g$ of G using Eq. (11);
  Update G according to $\mathcal{L}^g$ with E, D, D^z fixed;
  Sample and compute again as steps $3 \sim 4$;
  Compute overall objective $\mathcal{L}^d, \mathcal{L}^e$ of D, E using Eqs. (12) and (15);
  Update D, E according to $\mathcal{L}^d, \mathcal{L}^e$ with G, D^z fixed;
  Sample batch of images $x_1^s \in \{x\}$ with attribute, style code $z^u \sim Unif(-1, 1)$;
  Compute objective $\mathcal{L}_{unif}^{dz}$ using Eq. (14);
  Update D^z according to $\mathcal{L}_{unif}^{dz}$ with G, D, E fixed;
end

which means that an image could be recovered by its own style.

By summing up the above losses, the overall objective for the generator G is formulated as:

$$\min_G \mathcal{L}^g = \mathcal{L}_{adv}^g + \lambda_1 \mathcal{L}_{cls}^g + \lambda_2 \mathcal{L}_{mi} + \lambda_3 \mathcal{L}_{rec}, \tag{11}$$

where λ_1, λ_2 and λ_3 are the parameters to balance the losses.

Correspondingly, the objective of D is formulated as:

$$\min_D \mathcal{L}^d = \mathcal{L}_{adv}^d + \lambda_1 \mathcal{L}_{cls}^d. \tag{12}$$

2.2.2. Objective of the style encoder E

The style encoder E is a key part, which aims to extract the style information from an image, as shown at the top-right of Fig. 2.

Considering that the style of all instances is distinct, we enforce the style code z from all real instances to follow the uniform distribution $Unif(-1, 1)$ which has maximum entropy in a finite interval and thus makes them as different as possible. We apply an adversarial training with discriminator D^z to achieve this goal, formulated as below:

$$\min_E \mathcal{L}_{unif} = -\mathbb{E}_{z^s \sim p_E(x_1^s)}[D^z(z^s)], \tag{13}$$

$$\min_{\|D^z\|_L \leqslant 1} \mathcal{L}_{unif}^{dz} = -\mathbb{E}_{z^u \sim Unif(-1,1)}[D^z(z^u)] + \mathbb{E}_{z^s \sim p_{E(x_1^s)}}[D^z(z^s)]. \tag{14}$$

Besides, to make the style encoding related to the attribute rather than other information, the mutual information loss $\mathcal{L}_{mi}$ between style code and generated images in Eq. (9) is also used to optimize E. Moreover, the cycle reconstruction $\mathcal{L}_{rec}$ in Eq. (10) is also applied to improve the style correctness of the generated image. Therefore, the overall objective of the encoder E can be formulated as:

$$\min_E \mathcal{L}^e = \lambda_2 \mathcal{L}_{mi} + \lambda_3 \mathcal{L}_{rec} + \lambda_4 \mathcal{L}_{unif}, \tag{15}$$

where λ_2, λ_3, and λ_4 are parameters to balance the losses.

Moreover, following (Kaneko et al., 2017, 2018), the style encoder E shares the network with the attribute classifier D_{cls} except the output layer, since D_{cls} extracts attribute-related features for classification, so it would make E pay more attention to the given attribute.

2.2.3. Overall training

The four components in our STD-GAN including the generator G, the style encoder E, the discriminator D for image adversarial training, and D^z for uniform distribution training for style code, are optimized iteratively according to Eqs. (11), (12), (15) and (14) as shown in Algorithm 1.

2.3. Multi-attribute instance-level transfer

Although we use single attribute editing for example to illustrate the proposed method in Section 2.1, our STD-GAN is generally applicable for multi-attribute transfer by extending the attribute label y and style code z to $\{y_1, y_2, \ldots, y_k\}$ and $\{z_1, z_2, \ldots, z_k\}$, where k is the number of attributes, y_i and z_i indicates the ith attribute and its corresponding style code. In most scenarios, only some of the k attributes need to edit while the others remain unchanged. Considering this, inspired by STGAN (Liu et al., 2019), the range of input y_i of G is changed from $\{0, 1\}$ to $\{-1, 0, 1\}$, where -1 means removing attribute, 1 means adding attribute, while 0 means keeping the attribute unchanged whether or not the input image x has the attribute. When y_i is -1 or 0, style code z_i is always set as 0. Accordingly, the dimensionality of outputs from E and D_{cls} are expanded to support multiple attributes. The overall loss function is unchanged.

This generalized STD-GAN for multiple attributes not only disentangles the style of one attribute from the image but also decouples between multiple attributes to some extent.

For stable training, we first take turns to train the editing of each single attribute. When the training is stable, we simultaneously train the editing of all attributes.

2.4. Application to multiple types of editing

After optimization, the style code follows the uniform distribution. Benefit from this elaborate design, our method is also applicable for *style interpolation* and *semantic-level editing without any source image*, although it is primarily designed for instance-level transfer between faces.

Testing of instance-level attribute transfer: As shown in Fig. 4(a), given a source image x_1^s for reference, our STD-GAN can transfer its attribute style to a target image x_0^t, achieved as $\hat{x}_1^t = G(x_0^t, y = 1, E(x_1^s))$. This is the primary goal of this work.

Testing of semantic-level attribute editing: This task aims at adding an attribute to x_0^t without any source image as shown in Fig. 4(b), achieved by sampling a style code from $Unif(-1, 1)$:

$$\hat{x}_1^t = G(x_0^t, y = 1, z^u), \quad \text{with } z^u \sim Unif(-1, 1). \tag{16}$$

Testing of instance-level style interpolation: More interestingly, our method can achieve continuous style change between two images $x_1^{s_1}$ and $x_1^{s_2}$ as shown in Fig. 4(c), which is formulated as:

$$\hat{x}_1^t = G(x_0^t, y = 1, \alpha \cdot E(x_1^{s_1}) + (1 - \alpha) \cdot E(x_1^{s_2})), \tag{17}$$

where $0 \leqslant \alpha \leqslant 1$. The interpolation method can be extended to more than two styles. The result is shown in Section 3.4.

3. Experiments

3.1. Implementation details

We conduct experiments on CelebA dataset (Liu et al., 2015), which is commonly used for attribute editing. This dataset contains 202599 images, each of which is annotated with 40 binary attributes (with or without). We use the official protocol for training, validation, and test split. Besides, we exploit the official face alignment and then crop the 170×170 center part and resize it to 256×256. In the training process, the training images are horizontally flipped with a probability of 0.5 for data augmentation. In all experiments, seven attributes with clearer meaning of styles are selected for evaluation, including *bangs*, *eyeglasses*, *smiling*, *mustache*, *mouth slightly open*, *wearing lipstick* and *hair color* (black as negative, brown and blond as positive), which cover most attributes used in the existing works (they usually use 3–5 attributes for evaluation).

In the training phase, we adopt Adam optimizer (Kingma and Ba, 2015) ($\beta_1 = 0.5, \beta_2 = 0.999$ and the learning rate of 1×10^{-4}) with a

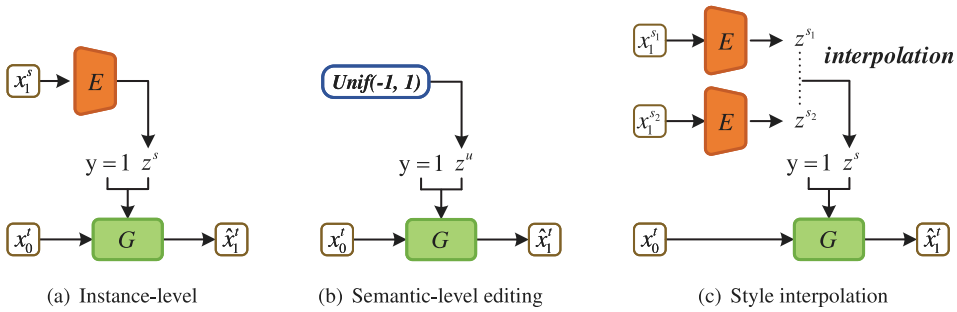


Fig. 4. Multiple types of editing tasks that our method is capable of.

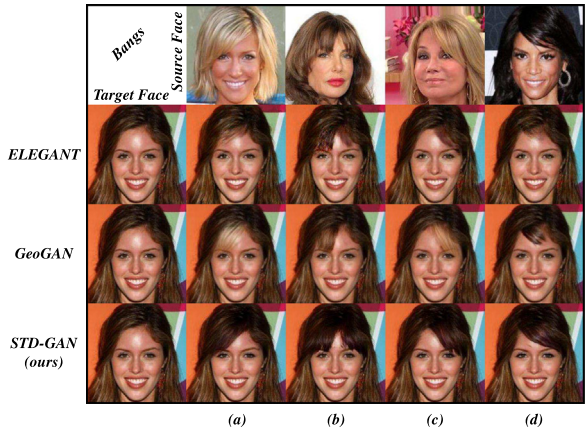


Fig. 5. Visual comparison on instance-level style transfer of *Bangs*.

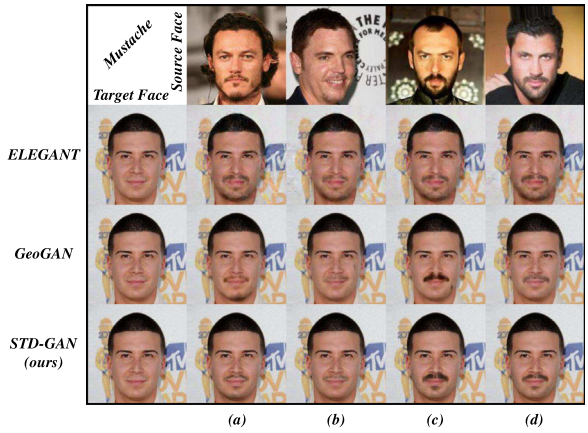


Fig. 6. Visual comparison on instance-level style transfer of *Mustache*.

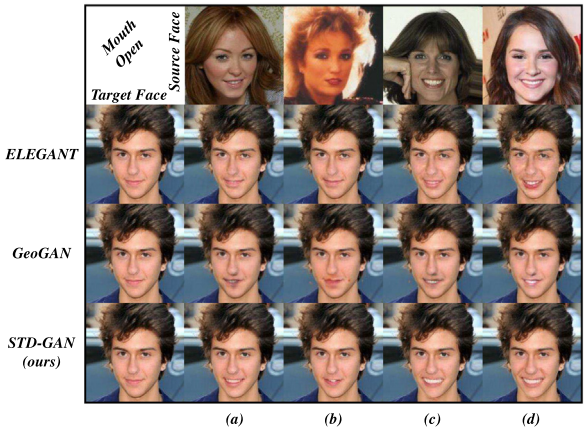


Fig. 7. Visual comparison on instance-level style transfer of *Mouth Open*.

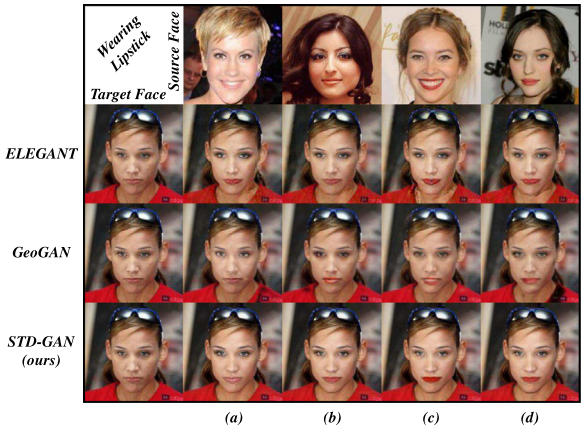


Fig. 8. Visual comparison on instance-level style transfer of *Wearing Lipstick*.

batch size of 8 images. The coefficients in Eqs. (11), (12) and (15) are set as: $\lambda_1 = 5$, $\lambda_2 = 5$, $\lambda_3 = 30$ and $\lambda_4 = 0.1$ to make the magnitudes of these losses have similar order. The dimension n of the style code z is experimentally set as 16.

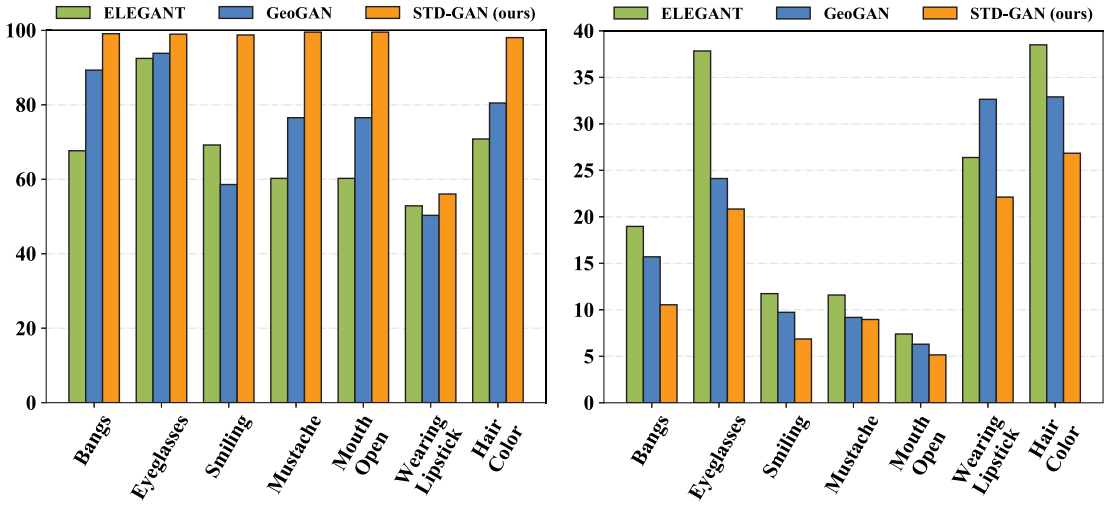
In the testing phase, similar to existing works, it is assumed to be known whether an image has an attribute or not. For practical applications, it can be simply predicted by an attribute classifier such as ResNet-18 (He et al., 2016), which can reach $92 \pm 2\%$ accuracy for most attributes. More carefully designed classifiers can also achieve higher accuracy.

The generator G follows a U-Net (Ronneberger et al., 2015) structure considering its success in semantic-level editing. Specifically, a three-layer down-sample block is used to extract the feature map (C128, W32, H32, "C" for channel, "W" for width, "H" for height) from the input image x (C3, W256, H256). For single attribute editing, y is a 1-dim vector, z is a 16-dim vector. Both of them are first copied to the spatial scale of the feature map (W32, H32) and are directly concatenated with the feature map at the channel-level (which means the feature map shape is C128+1+16, W32, H32). Then it will go through a six-layer residual block and a three-layer up-sampling block. For the editing of k attributes, the attribute label y is a k-dim vector, the style code z is a $16k$-dim vector. The discriminator D is designed as a PatchGAN (Zhu et al., 2017) network for better local discrimination with five down-sample layers.[1]

[1] More detailed network architecture and more results related to the following section can be found in the supplementary material.

Table 1
User study (higher is better) of style correctness and visual realism.

User study (%)		Bangs	Eyeglasses	Smiling	Mustache	Mouth open	Wearing lipstick	Hair color	Average
Style correctness ↑	ELEGANT	11.2	15.1	12.6	21.2	28.4	24.7	23.6	19.5
	GeoGAN	12.3	24.3	11.6	16.7	17.4	16.3	11.0	15.7
	STD-GAN (ours)	**76.5**	**60.6**	**75.8**	**62.1**	**54.2**	**59.0**	**65.4**	**64.8**
Visual realism ↑	ELEGANT	17.8	20.4	17.8	21.9	34.5	29.8	31.5	24.8
	GeoGAN	8.2	18.1	14.6	12.8	14.0	9.1	2.9	11.4
	STD-GAN (ours)	**74.0**	**61.5**	**67.6**	**65.3**	**51.5**	**61.1**	**65.6**	**63.8**



(a) Cls. Accuracy (%) (higher is better) (b) FID (lower is better)

Fig. 9. Quantitative comparison between different methods.

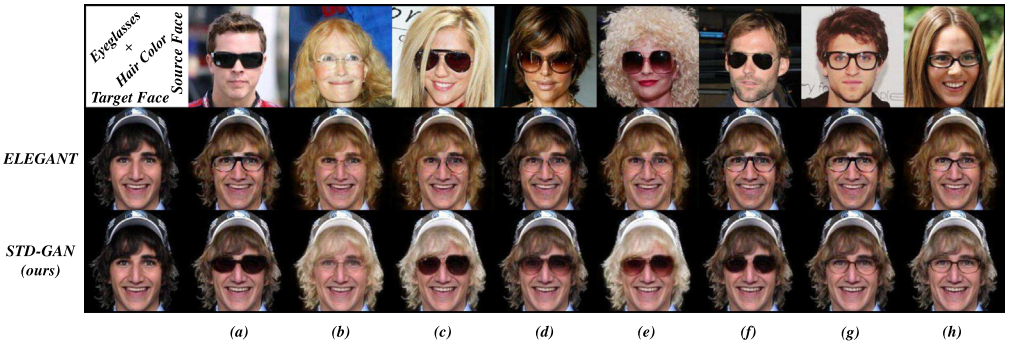


Fig. 10. Visual comparison on multi-attribute style transfer, i.e. simultaneously transfer styles of *Eyeglasses* and *Hair Color*.

3.2. Comparisons on instance-level transfer

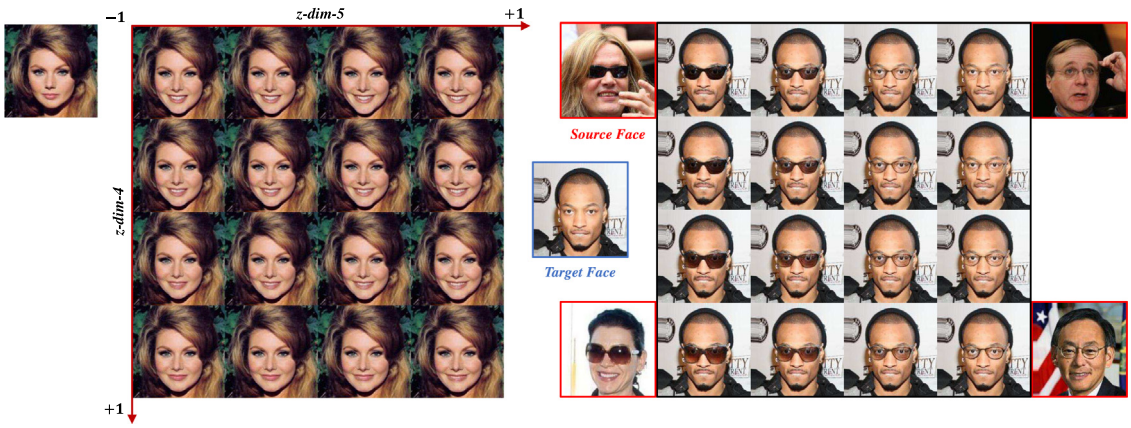
To investigate the effectiveness of our method, we compare it to other methods. Existing studies of instance-level facial attribute transfer are still preliminary. Some methods focus on specific attribute transfer, but cannot handle general attribute transfer. In summary, there are few existing methods for general attributes. We compare our method to the state-of-the-art general attribute transfer methods, including ELEGANT (Xiao et al., 2018b) and GeoGAN (Yin et al., 2019), in terms of visual comparison, user study, and quantitative comparison.

3.2.1. Visual comparison

Firstly, we visually compare our method to existing methods in Figs. 5–8 on various attributes. As can be seen, ELEGANT can generate images with the expected attribute, however, most styles are not correctly transferred. Furthermore, GeoGAN achieves a much better attribute style that benefits from the guidance of attribute segmentation. However, a few results of GeoGAN look unrealistic, such as the Bang's color in Fig. 5 is different from the rest hair region since GeoGAN only considers the shape of an attribute but neglects texture. In contrast, our STD-GAN achieves more favorable style transfer with correct style and high fidelity. Our method can even capture those subtle differences between styles, e.g. bangs in Figs. 5(c) and 5(d) are both right bangs but their width and height are slightly different according to their source styles. This illustrates that our style encoder can precisely capture the style information, and the generator can effectively transfer the style given a style code.

3.2.2. User study

For a comprehensive comparison, we perform a user study for human perception evaluation. Given a target image and a source image, users are required to choose the best output image by different methods. Two experiments are conducted. The first one is to evaluate the style correctness, i.e. choose the transferred image that has the most similar style as the source image. The second one is to choose the transferred image that looks most realistic without seeing the source



(a) Semantic-level editing of *smiling* (b) Style interpolation of *eyeglasses*

Fig. 11. Results of our STD-GAN on multiple types of attribute editing.

image. For each attribute, 300 edited images are randomly selected for evaluation, and each editing result is evaluated by 10 users.

As seen from the results in Table 1, ELEGANT has higher votes than GeoGAN on average. But in some attributes, such as bangs and eyeglasses, GeoGAN is better than ELEGANT on style correctness. This is because GeoGAN directly *copies* the attribute region from the source image to the target image, and thus it well preserves the style for some attributes but cannot ensure fidelity. However, ELEGANT *generates* the target image through the generation network instead of directly operating on pixels, so it can maintain visual reality. Beyond the expectation, more than 63% of the subjects choose our method as the best in terms of both style correctness and visual realism, about 39% more than ELEGANT and 49% than GeoGAN on average of all attributes. For each different attribute, 51%~76% of the subjects choose ours as the best, especially on *Bangs*, *Smiling*, *Mustache*, and *Hair Color*. This large improvement clearly shows the effectiveness of our elaborately designed image style disentangling.

3.2.3. Quantitative evaluation

We also quantitatively evaluate the attribute correctness and the fidelity of the transferred images. This experiment is conducted on 10000 randomly selected images from the testing set of CelebA. As the style correctness is impossible to evaluate quantitatively, we instead evaluate the semantic attribute classification accuracy for a rough comparison. Specifically, the attribute classification accuracy of generated images is examined via a ResNet-18 (He et al., 2016), which is trained as a binary classifier for each attribute on the CelebA training dataset. Besides, to evaluate the fidelity, Fréchet Inception Distance (FID) (Heusel et al., 2017) is employed to measure the similarity between generated images and real images.

Both results are shown in Fig. 9. For each attribute, both classification accuracy and FID of our method is better than other methods. In terms of attribute classification accuracy, our method achieves an average improvement of 16.5% for all attributes. The FID value is reduced by about 3.3 on average compared with the other methods on various attributes. It demonstrates that the editing of our method is more realistic, which is consistent with the visual comparison and user study.

3.3. Multi-attribute transfer

To investigate the effectiveness of simultaneously transferring the style of multiple attributes mentioned in Section 2.3, we compare our model to ELEGANT (Xiao et al., 2018b). GeoGAN (Yin et al., 2019) is not compared since it is not applicable for multi-attribute transfer. The Multi-attribute transfer is challenging for all existing methods and some methods are even not applicable for multi-attribute transfer since the instance styles of multiple attributes are explosive. As shown in Fig. 10, our method can transfer the styles of the two attributes accurately, but ELEGANT is hard to transfer both attributes very wells. From this, we infer that our method has better attribute disentangling ability. This indicates that our STD-GAN can well decouple the styles of different attributes, which is quite hard but a crucial task in the multi-attribute transfer.

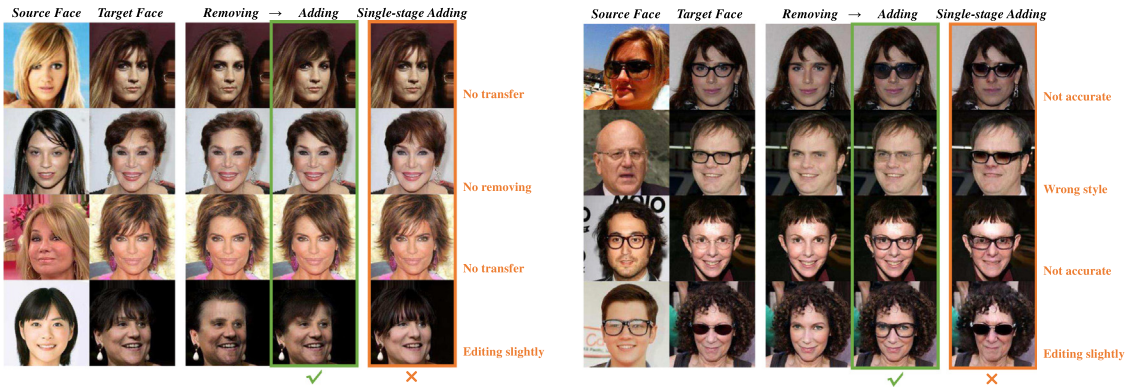
3.4. Multiple types of editing

As discussed in Section 2.4, our model is also applicable for semantic-level editing without any source image and style interpolation. Fig. 11(a) shows the result of semantic-level attribute editing of adding smiling without the source image. As in Eq. (16), we randomly sample a style code z from $Unif(-1, 1)$, and choose two dimensions to change from -1 to $+1$. As can be seen from Fig. 11(a), the 4th dimension controls the degree of mouth opening while the 5th dimension controls the degree of stretching of the mouth. This phenomenon demonstrates that different dimensions have a tendency to control different features of the style. Moreover, it should be noted that our method is applicable for both semantic attribute editing and instance-level attribute style transfer, while ELEGANT (Xiao et al., 2018b) and GeoGAN (Yin et al., 2019) can only be used for instance-level attribute style transfer. Fig. 11(b) shows style interpolation from four source images. The bilinearly interpolated style code is used to add eyeglasses for the target image as in Eq. (17). Notice that the style code only influences the attribute-related region, while other regions excluded the attribute is kept almost identical. These results show that our method is capable of semantic-level attribute transfer and style interpolation with a single model.

3.5. Ablation study

In the ablation study, we investigate the effect of the two steps editing design and all constraints related to style transfer.

Firstly, we investigate the effect of the design of two steps transfer. As a comparison, we use a single-step design, which directly transfers the attribute to whether the target image has the attribute or not. As shown in Fig. 12, the single-step design almost fails to transfer. We guess this is caused by the explosive variations from the original style to the given style. Concretely, taking n styles for an attribute, for example, the model of single-step design needs to fit n^2 situations, while our model with two steps only needs to handle $2n$ situations which is much easier to obtain favorable results as shown in Fig. 12. The comparison between the single-step design and two-steps design



(a) Bangs (b) Eyeglasses

Fig. 12. Ablation study of two steps and one step design. The 3rd and 4th columns are results of two-step style transfer design, while the 5th column shows results of single-step style transfer design.

Fig. 13. Visual ablation study of our STD-GAN on *Bangs* and *Eyeglasses*. Here, $full$ contains all part of our method. $full - RC$, $full - UN$, $full - MI$, and $full - UN - MI$ mean that the cycle reconstruction loss $\mathcal{L}_{rec}$ in Eq. (10), the uniform distribution adversarial training $\mathcal{L}_{unif}$ in Eq. (13), the mutual information loss $\mathcal{L}_{mi}$ in Eq. (9), and $\mathcal{L}_{unif} + \mathcal{L}_{mi}$ are removed from the full method respectively.

clearly demonstrates that our divide-and-conquer two-steps strategy is reasonable and effective.

Secondly, we investigate the effect of all constraints related to attribute style transfer, including: (1) the mutual information objective $\mathcal{L}_{mi}$ in Eq. (9), which makes the style code associated with the image style; (2) the cycle reconstruction objective $\mathcal{L}_{rec}$ in Eq. (10); (3) the adversarial training objective $\mathcal{L}_{unif}$ of uniform distribution in Eq. (13), which aims for finer style encoding. The results are shown in Fig. 13. As can be seen, the style correctness and the fidelity degrades after removing each loss, illustrating the necessity of each part in our method. Especially, as shown in Fig. 13(e), the style of the transferred image becomes inaccurate obviously when the mutual information term and the uniform distribution term are removed. It demonstrates that these two terms play an important role in style transfer, and verifies the effectiveness of the proposed Style Disentangling Module.

3.6. Failure cases and discussion

For most attributes, the editing requirements are compatible, which is the premise of attribute editing task or instance-level facial attribute transfer task. However, for some incompatible attributes such as adding bangs to bald people, there is a conflict between the realism and source attributes style preserving. Some failure cases caused by the conflict are shown in Fig. 14, where bangs are added to people wearing a hat and bald people. As can be seen, the edited result looks strange. Take adding bangs to bald man for example, in terms of realism, bangs should appear together with the hair, while in terms of correct transferring, the hair should not appear. Subjectively, our method is inclined to correctly retain the source attribute style and transfers the attribute "stiffly", and thus makes the editing results look strange.

Considering this conflict, it would be user-friendly if there is a way to control the balance between style accuracy and realism. This kind of control can be achieved by adjusting the weight coefficient of fidelity adversarial loss and style constraints during the training phase. However, in the testing phase, for most existing methods including ELEGANT (Xiao et al., 2018b), GeoGAN (Yin et al., 2019), and ours, there is no explicit consideration of style accuracy and realism, which makes it hard to control. In the future, we will investigate to establish an explicit module to balance between these two aspects during the generation process.

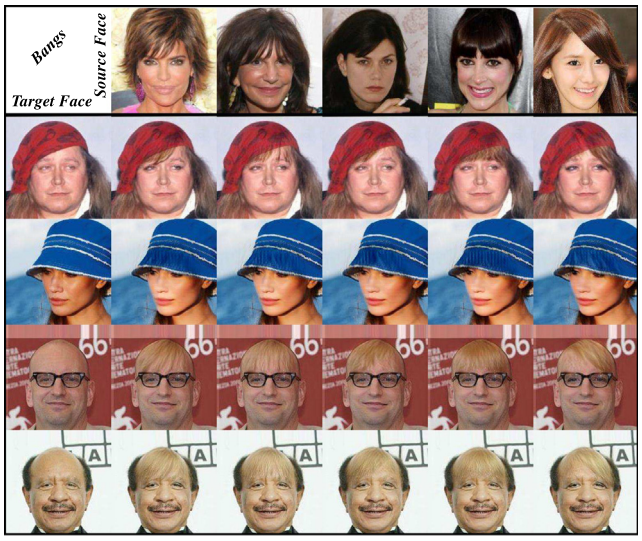


Fig. 14. Failure cases of our method on conflicting attributes, i.e., adding *bangs* to people wearing a hat and bald people.

4. Conclusion

In this work, we propose a novel Style Disentangling GAN (STD-GAN) to achieve precise instance-level facial attribute transfer, with only binary attribute annotations. The whole process consists of two steps, including a removing step to remove the original attribute and an adding step to add the attribute with another style. A style disentangling module is elaborately designed to extract style information from a source image. Our method is naturally extended to multi-attribute instance-level transfer. Moreover, it also capable of semantic-level attribute editing. Extensive experiments verify the effectiveness of our method.

CRediT authorship contribution statement

Xuyang Guo: Methodology, Software, Investigation, Writing - original draft, Visualization. **Meina Kan:** Methodology, Writing - review & editing, Supervision. **Zhenliang He:** Software, Validation, Writing - review & editing. **Xingguang Song:** Resources, Conceptualization. **Shiguang Shan:** Supervision, Conceptualization, Writing - review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is partially supported by National Key R&D Program of China (Grant No. 2017YFA0700800 and No. Y808401), and Natural Science Foundation of China (No. 61772496).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cviu.2021.103205.

References

Arjovsky, M., Chintala, S., Bottou, L., 2017. Wasserstein generative adversarial networks. In: International Conference on Machine Learning, pp. 214–223.

Chang, H., Lu, J., Yu, F., Finkelstein, A., 2018. Pairedcyclegan: Asymmetric style transfer for applying and removing makeup. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 40–48.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P., 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2172–2180.

Chen, Y.-C., Lin, H., Shu, M., Li, R., Tao, X., Shen, X., Ye, Y., Jia, J., 2018. Facelet-bank for fast portrait manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3541–3549.

Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., Choo, J., 2018. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 8789–8797.

Choi, Y., Uh, Y., Yoo, J., Ha, J.-W., 2020. Stargan v2: Diverse image synthesis for multiple domains. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 8188–8197.

Ding, H., Sricharan, K., Chellappa, R., 2018. Exprgan: Facial expression editing with controllable expression intensity. In: AAAI Conference on Artificial Intelligence, pp. 6781–6788.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C., 2017. Improved training of wasserstein gans. In: Advances in Neural Information Processing Systems. pp. 5767–5777.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

He, Z., Zuo, W., Kan, M., Shan, S., Chen, X., 2019. Attgan: Facial attribute editing by only changing what you want. IEEE Trans. Image Process. 28 (11), 5464–5478.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S., 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems. pp. 6626–6637.

Hu, B., Zheng, Z., Liu, P., Yang, W., Ren, M., 2020. Unsupervised eyeglasses removal in the wild. IEEE Trans. Cybern..

Huang, X., Liu, M.-Y., Belongie, S., Kautz, J., 2018. Multimodal unsupervised image-to-image translation. In: European Conference on Computer Vision, pp. 172–189.

Kaneko, T., Hiramatsu, K., Kashino, K., 2017. Generative attribute controller with conditional filtered generative adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 6089–6098.

Kaneko, T., Hiramatsu, K., Kashino, K., 2018. Generative adversarial image synthesis with decision tree latent controller. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 6606–6615.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization. In: International Conference on Learning Representations.

Lample, G., Zeghidour, N., Usunier, N., Bordes, A., Denoyer, L., Ranzato, M., 2017. Fader networks: Manipulating images by sliding attributes. In: Advances in Neural Information Processing Systems. pp. 5967–5976.

Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O., 2016. Autoencoding beyond pixels using a learned similarity metric. In: International Conference on Machine Learning, pp. 1558–1566.

Lee, C.-H., Liu, Z., Wu, L., Luo, P., 2020a. Maskgan: Towards diverse and interactive facial image manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5549–5558.

Lee, H.-Y., Tseng, H.-Y., Mao, Q., Huang, J.-B., Lu, Y.-D., Singh, M., Yang, M.-H., 2020b. Drit++: Diverse image-to-image translation via disentangled representations. Int. J. Comput. Vis. 1–16.

Li, T., Qian, R., Dong, C., Liu, S., Yan, Q., Zhu, W., Lin, L., Beautygan: Instance-level facial makeup transfer with deep generative adversarial network. In: ACM Multimedia Conference on Multimedia Conference, pp. 645–653.

Liu, M., Ding, Y., Xia, M., Liu, X., Ding, E., Zuo, W., Wen, S., 2019. STGAN: A unified selective transfer network for arbitrary image attribute editing. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3673–3682.

Liu, Z., Luo, P., Wang, X., Tang, X., 2015. Deep learning face attributes in the wild. In: IEEE International Conference on Computer Vision, pp. 3730–3738.

Odena, A., Olah, C., Shlens, J., 2017. Conditional image synthesis with auxiliary classifier gans. In: International Conference on Machine Learning, pp. 2642–2651.

Romero, A., Arbeláez, P., Van Gool, L., Timofte, R., 2019. SMIT: Stochastic multi-label image-to-image translation. In: IEEE International Conference on Computer Vision Workshops.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234–241.

Shen, W., Liu, R., 2017. Learning residual images for face attribute manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4030–4038.

Upchurch, P., Gardner, J., Pleiss, G., Pless, R., Snavely, N., Bala, K., Weinberger, K., 2017. Deep feature interpolation for image content changes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 7064–7073.

Wu, P.-W., Lin, Y.-J., Chang, C.-H., Chang, E.Y., Liao, S.-W., 2019. Relgan: Multi-domain image-to-image translation via relative attributes. In: IEEE International Conference on Computer Vision, pp. 5914–5922.

Xiao, T., Hong, J., Ma, J., 2018a. DNA-GAN: Learning disentangled representations from multi-attribute images. In: International Conference on Learning Representations Workshop.

Xiao, T., Hong, J., Ma, J., 2018b. Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. In: European Conference on Computer Vision, pp. 168–184.

Xu, W., Shawn, K., Wang, G., 2019. Toward learning a unified many-to-many mapping for diverse image translation. Pattern Recognit. 93, 570–580.

Yin, W., Liu, Z., Loy, C.C., 2019. Instance-level facial attributes transfer with geometry-aware flow. In: AAAI Conference on Artificial Intelligence, pp. 9111–9118.

Yu, X., Chen, Y., Liu, S., Li, T., Li, G., 2019. Multi-mapping image-to-image translation via learning disentanglement. In: Advances in Neural Information Processing System, pp. 2994–3004.

Zhang, J., Shu, Y., Xu, S., Cao, G., Zhong, F., Liu, M., Qin, X., 2018. Sparsely grouped multi-task generative adversarial networks for facial attribute manipulation. In: 2018 ACM Multimedia Conference on Multimedia Conference, pp. 392–401.

Zhao, B., Chang, B., Jie, Z., Sigal, L., 2018. Modular generative adversarial networks. In: European Conference on Computer Vision, pp. 150–165.

Zhou, S., Xiao, T., Yang, Y., Feng, D., He, Q., He, W., 2017. GeneGAN: Learning object transfiguration and attribute subspace from unpaired data. In: British Machine Vision Conference.

Zhu, J.-Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE International Conference on Computer Vision, pp. 2223–2232.

Xuyang Guo received the B.E. degree in software engineering from Wuhan University (WHU), Wuhan, China, in 2019. He is a Master candidate in Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, since 2019. His research interests include facial attribute editing, and generative adversarial networks.

Meina Kan received the Ph.D. degree in computer vision from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, under the supervision of Prof. X. Chen and Prof. S. Shan, in 2013. She is currently an Associate Professor with ICT, CAS. Her research mainly focuses on face recognition, transfer learning, multiview learning, and deep learning. Dr. Kan has served as the Co-Chair for the ICPR18 Workshop on Deep Learning for Pattern Recognition and the ACCV14 Workshop on Human Identification for Surveillance.

Zhenliang He received the B.E. degree from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2015. He is a Ph.D. candidate in Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, since 2015. His research interests include facial landmark detection, facial attribute editing, and generative adversarial networks.

Shiguang Shan received the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS) in 2004. Now, he is a Full Professor of ICT, CAS, and is the Deputy Director of the Key Laboratory of Intelligent Information Processing, CAS. His research interests include computer vision, pattern recognition, and machine learning. He especially focuses on face recognition related research topics and machine learning with little or weakly supervised data. He has served as the Area Chair for many international conferences, and as Associate Editor of several international journals.