IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 29, 2020, accepted February 12, 2020, date of publication February 17, 2020, date of current version March 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2974239

Self-Attention-Masking Semantic Decomposition and Segmentation for Facial Attribute Manipulation

XUAN XIA[1,2], FENGQI YU[1], NAN LI[2], YANSONG QU[1], JIAJIA ZHANG[3], AND CHENGGUANG ZHU[4]

[1]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences (University of Chinese Academy of Sciences), Shenzhen 518055, China
[2]Shenzhen Institute of Artificial Intelligence and Robotics for Society, Shenzhen 518055, China
[3]Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China
[4]Department of Instrument Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

Corresponding authors: Xuan Xia (xuan.xia@siat.ac.cn) and Fengqi Yu (fq.yu@siat.ac.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61806190, Grant 61806191, Grant U1613226, and Grant U1813216, in part by the China Postdoctoral Science Foundation under Grant 2018M633185, in part by the National key Research and Development Plan of China under Grant 61674162, Grant 2016YFC0105002, Grant 2018YFF01012500, and Grant 2018YFB2100900, in part by the Shenzhen Key Laboratory for RF Integrated Circuits, in part by the Shenzhen Shared Technology Service Center for Internet of Things, and in part by the Shenzhen Science and Technology Innovation Council under Grant JCYJ20160331192843950, Grant JCYJ20180305164616316, and Grant JCYJ20170410171923840.

ABSTRACT Many face attribute manipulation methods can only provide global attribute manipulation according to the attribute labels. In this paper, we propose a self-attention-masking semantic decomposition method which is able to learn an attribute attention mask for each attribute. User can adjust the strength and color of each attribute smoothly and more freely. We decouple the attention of different attributes and overcome the disadvantage of overlap between different attribute attention masks by an attention weighting module. Thanks to the attribute attention masks, our method allows manipulate facial attribute without generator after only once generation. Moreover, we can perform facial semantic segmentation without pixel level semantic labels. Experiments show that our method simultaneously improves the freedom of attribute manipulation and the authenticity of synthetic face. The mean intersection over union of semantic segmentation is over 65% for hair and skin. Our code is available at github.com/flyfeatherok/SAMSD.

INDEX TERMS Generative adversarial networks, semantic decomposition, semantic segmentation, facial attribute manipulation.

I. INTRODUCTION

Face attribute manipulation is an interesting but challenging task with many real-world vision applications. It has experienced significant improvements following the introduction of generative adversarial networks (GAN) [1] and enabling lots of functions such as facial expressions changing, eyeglasses adding, and styles (e.g. hair color, beautification/ de-beautification) transfer.

As mentioned in [2], facial attribute manipulation task can be roughly categorized into two types: semantic-level manipulation [3]–[6] and geometry-level manipulation [7], [8]. Early approaches such as StarGAN [3] and AttGAN [9] provide a kind of basic generator architecture and training strategy for semantic-level manipulation. However, they can only provide global attribute manipulation according to the attribute label and cannot be customized freely (e.g., you cannot adjust the hair color to green because there is no such attribute label). On the contrary, geometry-level manipulation methods have a higher degree of user freedom. User can guide the system to fix the image when the result is not as expected. But most of their training relies on expensive pixel level semantic labels.

It is highly desirable to adjust both strength and color of each face attribute smoothly at the same time. There are few solutions can do it. SCDFM [10] provides a solution since it divides a high-level attribute edit into multiple semantic components, where each works on one semantic region of a human face. It is the first attempt to learn semantic components from high-level attributes. However, SCDFM is

The associate editor coordinating the review of this manuscript and approving it for publication was Yongjie Li.

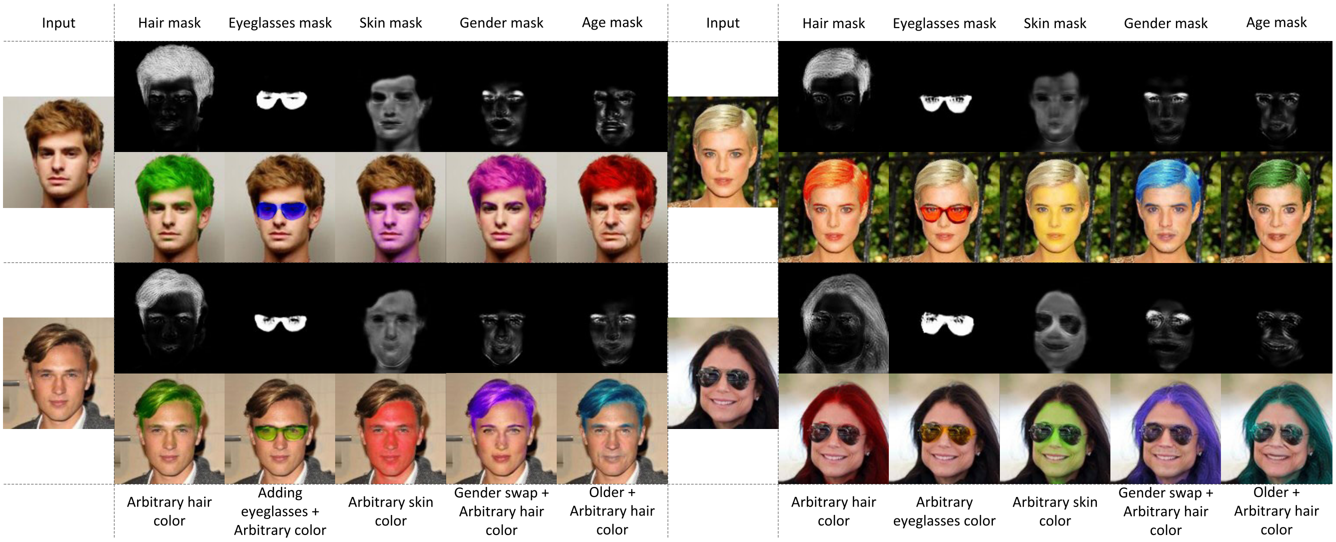

FIGURE 1. Our generator outputs single attention mask for each attribute (1st and 3th row), no matter whether the attribute is changed or not. Then we can adjust the color of attribute area arbitrarily (2nd and 4th row) without generator.

difficult to do multi-attribute training and need a pertained VGG network as the encoder.

In this paper, we propose a GAN based self-attention-masking semantic decomposition method which, unlike SCDFM, can generate an attribute attention mask (AAM) for each attribute and is fused into a general attention mask (GAM) for all attributes by an attention weighting module (AWM). Hence our method can manipulate the color and strength of single attribute more freely such as hair color, eyeglasses color, and gender swap strength. Meanwhile it will not interfere with the effect of other attribute manipulations. Moreover, the attention mask of single attribute gives us the opportunity to segment facial region automatically, without the supervision by semantic segmentation labels. Our method allows adjusting color and strength of different attributes, but what's more, allows to manipulate them freely even without generator after only once generation.

Figure 1 demonstrates some attribute manipulation examples by our method. The generator outputs a single attention mask for each attribute, shown in the first row, no matter whether the attribute is changed or not. Then we can adjust the color of attribute area arbitrarily only by the AAM, as shown in the second row. Please note that the AAM allows us to adjust not only the color of the attribute without generator, but also the strength of the attribute without generator, which will be described in a later section.

To summarize, our contributions are as follows:

1. We propose a self-attention-masking framework for face attribute manipulation, which is able to learn an attribute attention mask for each attribute semantic.

2. Our method can edit the color and strength of single attribute more freely, it benefits from the attention masks of semantic decomposition among different attribute. It is able to edit quickly without generator.

3. Our method can perform simple semantic segmentation of some facial areas automatically such as hair and skin, without semantic segmentation labels or any location labels. It is the first attempt for facial semantic segmentation only by image-level attribute labels to the best of our knowledge.

II. RELATED WORK

A. GAN BASED FACIAL ATTRIBUTE MANIPULATION

Several methods utilize GAN to build general face attribute manipulation frameworks since the success of GAN for image-to-image (I2I) translation [11]. For unpaired I2I translation tasks, CycleGAN [12] and its variants provide a method for evaluating image semantic consistency only by the images themselves. This makes it easier for face attribute manipulation without attribute disentangling in a deep space. Typical approaches such as StarGAN [3] and AttGAN [9] confirmed that only a pair of generator and discriminator is required for the face attribute manipulation, which can achieve remarkable translation effect. However, attribute label alone is not enough for accurate face attribute manipulation. There is still a lot of room for improvement.

Residual learning [6], [13] enables the network to learn the changing parts of the image while retaining other areas, which inspired the study of attention guidance. Then many scholars have noticed that the accuracy and freedom of attribute edit depend on the generator's attention guided by input information. With a similar training strategy and network structure to the reference [13], GANimation [14] proposes to use the attention mask to get the key areas for efficient attribute manipulation automatically, without affecting irrelevant areas and it worked out wonderfully. STGAN [15] proposes to use attribute vectors to guide the generator's attention. The generator in STGAN only reconstructs the image when the attribute vector is zero, then the generator can learn to distinguish between key areas and backgrounds. On the other hand, the strategy of supervision learning by semantic segmentation labels provides the capability of precise geometry-level manipulation. The attribute manipulation becomes very efficient since the generator can pay attention to the edited area directly through the semantic mask, such as SC-FEGAN [8] and MaskGAN [2]. However, their training is complex and the semantic annotation is expensive.

B. ATTRIBUTE SEMANTIC DECOMPOSITION

Although GANimation provides an attention mask for key attribute areas, it cannot be decomposed among attributes. Meanwhile, deep feature interpolation (or called latent space interpolation) [6], [16] was employed for face attribute manipulation. By shifting deep features of the query image with the attribute vectors in latent space, the semantic facial attributes can be updated accordingly. ELEGANT [6] even decouples the attribute in the latent space, but has to manipulate the attribute by target images. Based on this, Facelet [17] and SCDFM [10] provide two deep feature interpolation solutions without adversarial leaning and paired data. Facelets propose a Facelet-Bank framework that models face effects with respective middle-level convolutional layers. SCDFM divides a high-level attribute edit into multiple semantic components, where each works on one semantic region of a human face and users can make more fine adjustments. It allows adjusting edit strength of different components and manipulating edit effect on each component. It is the first attempt to learn semantic components from high-level attributes. However, SCDFM is difficult to do multi-attribute training since it has no control over the number of decompositions and its correspondence to attribute. Both Facelet and SCDFM need a pertained VGG network for training. This may restrict the scope of application.

C. WEAKLY SUPERVISED SEMANTIC SEGMENTATION

The attention mask of single attribute gives us the opportunity to segment facial region only by the image-level labels. Meanwhile most semantic segmentation methods rely on the pixel-level annotations, which require extremely expensive labeling efforts.

After FCN [18] and U-net [19] created the basic semantic segmentation network architecture under supervised learning, researchers have also strived to leverage weakly supervision instead such as multiple instance learning [20], EM algorithm [21] and constrained CNN [22], or semi-supervision by additionally using a few pixel-wise segmentation labels [23], [24]. Similar to this paper, some weakly supervised methods [25], [26] used attention masks and classification tags. They achieved an excellent level of semantic segmentation. However, the semantics of human face attribute labels overlap with each other on the face (e.g., ‘‘gender’’ and ‘‘age’’ almost share a same facial area), so it is difficult to obtain accurate attention mask simply by applying classification loss in the I2I translation task.

III. SELF-ATTENTION-MASKING SEMANTIC DECOMPOSITION

Give an origin image $I_o \in \mathbb{R}^{h\times w\times 3}$ and the corresponding attribute label $\boldsymbol{s}_o \in \mathbb{R}^{1\times c}$, where $h \times w$ is the size of I_o, and c is the category number of attribute label. We expect our model to generate a group of attribute attention masks $\boldsymbol{M} \in \mathbb{R}^{h\times w\times c}$ that can be used to control the strength of each attribute change, and freely synthesize an image I_t with target attributes $\boldsymbol{s}_t$ by $\boldsymbol{M}$ and a color mask $I_c \in \mathbb{R}^{h\times w\times 3}$.

A. OVERALL FRAMEWORK

For the same reasons mentioned in [14], we define the difference attribute vector $\boldsymbol{v}_s$ as the difference between target and source attribute labels that should be put into the generator,

$$\boldsymbol{v}_s = \boldsymbol{s}_t - \boldsymbol{s}_o, \tag{1}$$

where $\boldsymbol{s}_t$ is the target attribute label, and $\boldsymbol{s}_o$ is the source attribute label. Only the attributes to be changed should be considered, to prevent faulty manipulation.

In GANimation, the attention mask changes with the attribute labels if attention mask and color mask share the same generator (e.g., the attention mask upon the hair area is zeroed if the hair attributes are unchanged). However, for our purpose, the scope and intensity of attention must be decoupled in generator in order to stabilize the semantic segmentation results. Hence as shown in figure 2, color mask I_c and AATs $\boldsymbol{M}$ are generated by a color mask generator G_c and an attention mask G_a generator respectively. Our generator G consists of these two parts.

As shown in figure 2, the color mask generator G_c consists of several strided convolutional layers to down-sample the input, six adaptive residual blocks [27], and several convolutional layers for up-sampling. We equip the adaptive blocks with AdaIN [28] layers:

$$\mathrm{AdaIN}(\omega, \gamma, \beta) = \gamma\left(\frac{\omega - \mu(\omega)}{\sigma(\omega)}\right) + \beta, \tag{2}$$

where ω is the activation produced by the previous convolutional layer, μ and σ are channel-wise mean and standard deviation, γ and β are parameters generated by a 4-layer multilayer perceptron (MLP) from the attribute vector $\boldsymbol{v}_s$.

The attention mask generator G_a follows a basic U-net structure: several strided convolutional layers for down-sampling, several convolutional layers for up-sampling, and several skip connections between them. Two generators share the same down-sampling path for parameter saving.

Note that G_a has nothing to do with v_s, and then the attention of different attributes on a face will remain stable. However, G_a cannot automatically generate masks that correspond to attributes one to one without any guiding. Hence an attention weighting module (AWM) is proposed to guide AAMs generation and the synthesis of the GAM I_m, which will be introduced in the next section.

Finally, I_t is synthesized by I_o, I_c, and I_m,

$$I_t = I_m \cdot I_c + (1 - I_m) \cdot I_o, \tag{3}$$

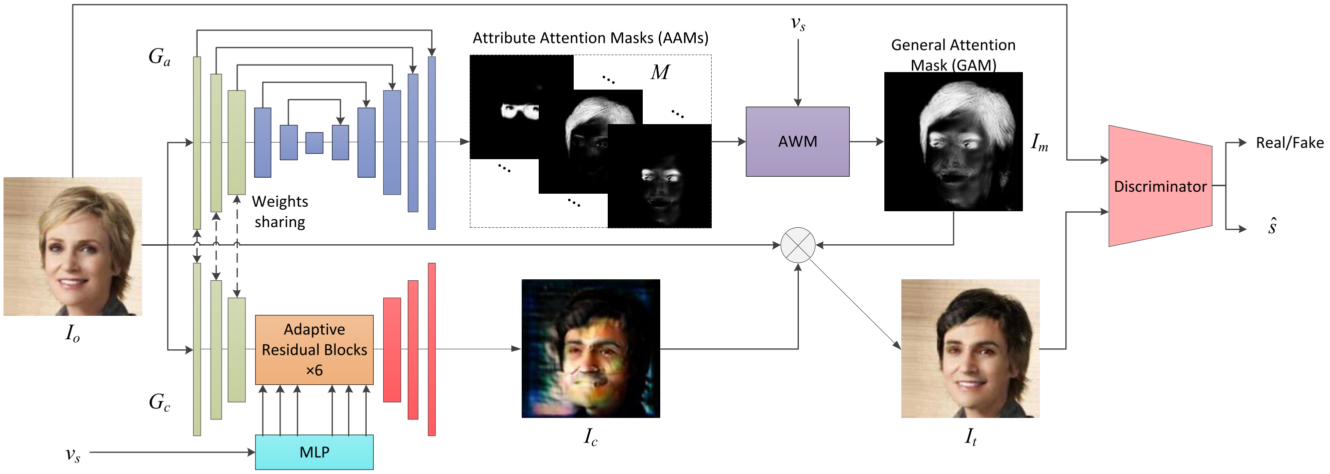

FIGURE 2. Overall framework of our model. Color mask I_c and AATs M are generated by a color mask generator G_c and an attention mask G_a generator respectively. Then AWM guide the synthesis of GAM I_m, and finally synthesize I_t.

where $I_m = AWM(G_a(I_o)) \in \{0, \ldots, 1\}^{h \times w}$ and $I_c = G_c(I_o|v_s) \in \mathbb{R}^{h \times w \times 3}$. In this way, the generator can focus exclusively on the pixels defining the facial attribute changes, leading to more realistic synthetic images. Meanwhile it retains the attention mask for each attribute, no matter whether the attribute has changed or not.

Similar to StarGAN, a discriminator D containing an attribute classifier is used to distinguish the true image I_o and the fake image I_t. Meanwhile the attribute classifier outputs the attributes estimation $\hat{\boldsymbol{s}}$ and ensures that I_t has the specified attributes $\boldsymbol{s}_t$. The specific parameters of our network structure are detailed in the appendix.

GANimation reports that attention masks can easily saturate to 1 without ‘‘total variation regularization’’. We found that this problem could be solved easily by adding the self-attention module [29] in the discriminator. This may be because the self-attention module in the discriminator is more efficient in passing the key information on the attribute region to the generator.

B. ATTENTION WEIGHTING MODULE

The synthesis of I_m and the generation of AAMs have two difficulties:

1. Generate masks of each attribute and decouple them.
2. Synthesize I_m without affecting the overlap region of AAMs.

For the former difficulty, we use the absolute value of $\boldsymbol{v}_s$ as the attribute strength indicator to update AAMs,

$$\boldsymbol{M} = |\boldsymbol{v}_s| \cdot \boldsymbol{M}, \tag{4}$$

According to (1), the corresponding value of the changed attribute in $\boldsymbol{v}_s \in$ [-1, 0) ∪ (0, 1], and the value of the unchanged attribute in $\boldsymbol{v}_s$ is 0. Therefore, $|\boldsymbol{v}_s|$ determines which masks are activated and the strength of activation. This forces G_a to learn to decompose the attention of different attributes to different masks. For example, the AAM corresponding to the hair color attribute must only pay attention to the hair area, because only this AAM is activated when only the hair color attribute changes, as shown in figure 3. However, the mask value will also decrease for small $\boldsymbol{v}_s$, lowering the strength of manipulation. This problem will be mitigated next by the attention weighting module.

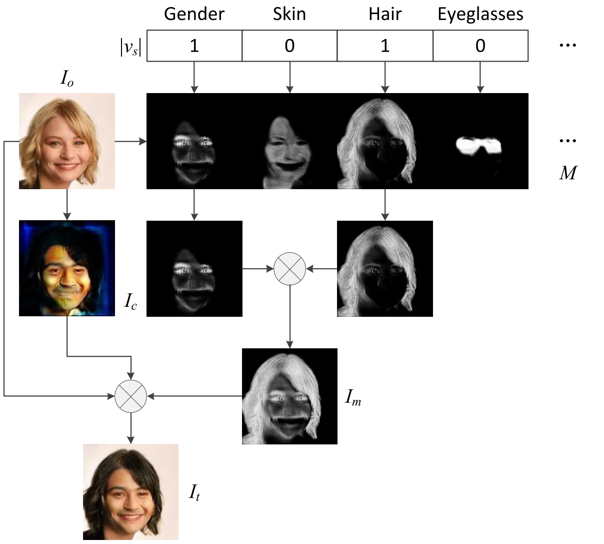

FIGURE 3. The basic process of AWM.

The values in I_m must between 0 and 1, hence I_m can’t be simply summed by AAMs. One plausible option is to take the maximum value on each pixel location in all AAMs. However, the maximization operation cannot effectively calculate the gradient in back propagation. Therefore, we use an attention weighting module for resolving the later difficulty:

$$I_m(m, n) = \frac{\sum_{i=1}^{c} \left[|\boldsymbol{v}_s|_i \, M_i(m, n) \right]^{\alpha}}{\sum_{i=1}^{c} \left[|\boldsymbol{v}_s|_i \, M_i(m, n) \right] + \varepsilon}, \tag{5}$$

where $|\boldsymbol{v}_s|_i$ is the i^{th} value of $|\boldsymbol{v}_s|$, $M_i(m, n)$ is the value of the i^{th} AAM on the pixel location (m, n), $\alpha \in [1, 2]$ is a scalar that controls attention weight, ε is a small constant value for prevent the division by 0.

The reason that $\alpha \in [1, 2]$ is as follows: On the one hand, the values of I_m must be between 0 and 1. Note that each $|\boldsymbol{v}_s|_i M_i(m, n) \in [0, 1]$, hence each $\left[|\boldsymbol{v}_s|_i M_i(m, n)\right]^{\alpha} \leq |\boldsymbol{v}_s|_i M_i(m, n)$ only if $\alpha \geq 1$. This guarantees the values of I_m must be between 0 and 1. On the other hand, in extreme cases, there are only some small values in the i^{th} AAM and 0 for the rest, then $I_m(m, n) = \left[|\boldsymbol{v}_s|_i M_i(m, n)\right]^{\alpha-1}$. Now α must smaller than 2, otherwise it will cause the value of I_m to be smaller, i.e., the attention will be weaker. On the contrary, even weak attention areas can be enhanced to ensure the attention strength if $\alpha < 2$. By this attention weighting module, the overlapping areas of different AAMs can be properly fused together, while non-overlapping areas are less affected if α is not too large. A large α exaggerates the attention difference among AAMs and makes the weak weaker. We found that $\alpha = 1.6$ and $\varepsilon = 0.01$ is appropriate in our experiments.

To summarize, AWM acts as an attribute switch, forcing the specified attention channel to generate the corresponding attribute attention mask. Meanwhile, the GAM outputted from AWM still contains the full attention of all the changed attributes. Other than that, the conduction of attention in the generator follows the same path as GANimation, that's why our framework works.

C. LOSS FUNCTIONS

1) CYCLE CONSISTENCY LOSS

The cycle consistency loss guarantees that translated images preserve the content of the input images. In this paper, it is defined as

$$\mathcal{L}_{cyc} = E_{I_o,s_o} \left\| I_o - G\left(G(I_o, \boldsymbol{v}_s), -\boldsymbol{v}_s\right)\right\|_1, \tag{6}$$

where $||\cdot||_1$ means L1 norm, G is the generator that contains G_c and G_a. $G(I_o, v_s)$ could be written according to (3) in more detail as

$$\begin{aligned} G(I_o, \boldsymbol{v}_s) = &\, AWM\ (G_a(I_o)) \cdot G_c(I_o, \boldsymbol{v}_s) \\ &+ [1 - AWM\ (G_a(I_o))] \cdot I_o, \end{aligned} \tag{7}$$

2) ATTRIBUTE CLASSIFICATION LOSS

This objective has two terms: a loss of real images used to optimize D, and a loss of fake images used to optimize G. The former is defined as

$$\mathcal{L}_{cls}^{D} = E_{I_o,s_o}\left[-\log D_{cls}(\boldsymbol{s}_o|I_o)\right], \tag{8}$$

where $D_{cls}(\boldsymbol{s}_o|I_o)$ represents a probability distribution over attribute labels computed by D. D learns how to classify facial attributes through this loss.

On the other hand, G tries to generate images that can be classified as the target attributes s_t. Hence the loss is defined as

$$\mathcal{L}_{cls}^{G} = E_{I_t,s_t}\left[-\log D_{cls}\left(\boldsymbol{s}_t|G(I_o, \boldsymbol{v}_s)\right)\right], \tag{9}$$

3) ADVERSARIAL LOSS

We adopt the Wasserstein GAN adversarial loss with gradient penalty [30], [31] as the adversarial loss to solve the problem of mode collapse. It is defined as

$$\begin{aligned} \mathcal{L}_{adv}^{D} = &\, E_{I_o}\left[D_{adv}(I_o)\right] - E_{I_t}\left[D_{adv}(I_t)\right] \\ &- \lambda_{gp} E_{\hat{I}}\left[\left(\left\|\nabla_{\hat{I}} D_{adv}(\hat{I})\right\|_2 - 1\right)^2\right], \end{aligned} \tag{10}$$

$$\mathcal{L}_{adv}^{G} = E_{I_o,s_o}\left[D_{adv}(G(I_o, \boldsymbol{v}_s))\right], \tag{11}$$

where $\hat{I}$ is sampled uniformly along a straight line between a pair of real and generated images. G generates a fake image I_t, while D_{adv} tries to distinguish between real and fake images by this loss. λ_{gp} is a hyper-parameter, we use $\lambda_{gp} = 10$ for all experiments.

4) FULL OBJECTIVE

The objective to optimize D and G are

$$\mathcal{L}_D = -\mathcal{L}_{adv}^{D} + \lambda_{cls}\mathcal{L}_{cls}^{D}, \tag{12}$$

$$\mathcal{L}_G = -\mathcal{L}_{adv}^{G} + \lambda_{cls}\mathcal{L}_{cls}^{G} + \lambda_{cyc}\mathcal{L}_{cyc}, \tag{13}$$

where λ_{cls} and λ_{cyc} are hyper-parameters. We use $\lambda_{cls} = 10$ and $\lambda_{cyc} = 10$ in all of our experiments.

IV. EXPERIMENTS

According to the characteristics of our method, the experiment is divided into the following three aspects:

1. Face attribute manipulation with generator

This section performs standard I2I translation tasks.

2. Face attribute manipulation without generator

Since our method can generate the AAM corresponding to each face attribute, we can further adjust the strength of the attribute transformation and the color of the attribute area by AAM without generator. Therefore, this part of the experiments will demonstrate the flexibility and effectiveness of our method.

3. Facial semantic segmentation

AAMs and prior knowledge can be used to further rough semantic segmentation of the face area. This section will show the results and performance evaluation of the facial semantic segmentation.

A. IMPLEMENTATION DETAILS

1) BASELINE MODELS

We choose state-of-the-art StarGAN, GANimation and STGAN as our baselines. The performances of some existing literature on I2I translation for two domains like DIAT [32] and CycleGAN or on facial attribute transfer like IcGAN [33] and [13] have been discussed in detail in [3] and [9]. StarGAN and AttGAN surpass them with significant margins. Therefore, we ignore them to save space.

In StarGAN, the attribute labels are combined with image by depth-wise concatenation, and the cycle consistent loss is used to preserve domain-unrelated contents. The generator of GANimation provides an attention mask for better preserve domain-unrelated contents. StarGAN and GANimation both use almost the same loss function. STGAN follows a basic U-net structure with selective transfer units and attribute vector. STGAN reduces the error of image reconstruction but almost doubles the number of parameters and the training time.

2) DATASETS

CelebA. [34] The CelebA dataset contains 202,599 face images of celebrities with 40 binary attributes. We use the 5-point landmarks to align all face images, then crop and resize them into 128×128 and 256×256. Just like StarGAN, we randomly select 2,000 images as test set and use remaining images for training data. We use the following attributes: gender (male/female), skin color (pale/not pale), hair color (black, blond, brown, gray), eyeglasses (with/without), smiling (with/without), and age (young/old).

CelebAMask-HQ. [2] CelebA-HQ [35] is a high-quality facial image dataset that consist of 30000 images picked from CelebA dataset. These images are processed with quality improvement to the size of 1024×1024. Based on CelebA-HQ, CelebAMask-HQ has 30000 semantic segmentation labels with the size of 512×512. Each label in the dataset has 19 classes such as ''hair'', ''skin'', ''nose'', ''eyes'', ''eyebrows'', ''ears'', ''mouth'', ''lip'', ''hat'', ''eyeglass'', ''earring'', ''necklace'', ''neck'', and ''cloth''.

3) TRAINING DETAILS

Our model is trained using Adam [36] with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The batch size is set to 16 for CelebA dataset. We flip the images horizontally with a probability of 0.5 for data augmentation. We perform one generator update after five discriminator updates and train our model with an initial learning rate of 0.0001 for the first 10 epochs and linearly decay the learning rate to 0 over the next 10 epochs (10000 iteration for one epoch). We use only one AAM for the four hair color attributes since their corresponding areas are the same. Therefore, the four hair color attribute vectors of $|v_s|$ after entering AWM are merged into one by summing and clipping. Training takes about 17 hours on a single NVIDIA RTX 2080Ti GPU.

B. FACIAL ATTRIBUTE MANIPULATION WITH GENERATOR

An ablation study of semantic decomposition is carried out. We trained a network without G_a and everything else remains the same. Just like GANimation, it output only one color mask and only one attention mask by a single generator without semantic decomposition. We train both networks with the same parameters, and observe the difference between their attention masks by using the same interpolation. Some ablation study results are shown in Fig. 4. We can observe that although it is difficult to tell the difference between generated images for the same interpolation by eye, their attention masks are different. Attention becomes more stable in our method. For example, with semantic decomposition, the attention in the red box changes only in strength as the interpolation changes (greater interpolation, stronger attention). On the contrary, without semantic decomposition, the attentions in the green box are unstable. The region of attention changes with the interpolation (e.g., there is no attention in the chin area when interpolation is 0.5). The ablation study has shown that our method contributes to the attention stability.

TABLE 1. User study results of performances evaluation (%).

Method	G	H	E	A	S	Total
StarGAN	13.80	12.54	9.34	10.10	6.57	10.47
GANimation	19.21	12.30	19.77	18.74	16.88	17.38
STGAN	36.77	34.48	37.92	34.52	41.83	37.10
Ours	30.22	40.68	32.97	36.64	34.72	35.05

Secondly, we compare our method with StarGAN, GANimation, and STGAN. We retrain all of them for the fair comparison. In the paper of GANimation, GANimation trained by action units (AUs). We do not use AUs for fairness. The qualitative results are shown in Fig. 5. It can be observed from Fig. 5 that some of the results of StarGAN show certain level of blur and artifact. And StarGAN cannot accurately reconstruct the details and colors of background. GANimation, STGAN, and our method have much better results. The results of STGAN look real but the background is still inevitably affected. By contrast, the background of GANimation and our method remains intact. However, some results of GANimation may lose details like StarGAN (e.g., the mole on the old man's face disappears).

However, our method still shows certain level of artifact such as the attributes of gender, pale skin and eyeglasses. We speculate that this may be because I_c lacks constraint in training, which is verified at some level by the better performance of STGAN.

To quantify the performances among different methods, we recruited six volunteers (5 male and 1 female) for user study as shown in Table 1. Each volunteer was asked to evaluate $50 \times 4 \times 5$ generated faces from 50 persons (half of these faces come from our own collection) with 128×128 size. Every person has five transformations: gender swap (G), hair color (H), eyeglasses adding (E), age swap (A), and smiling swap (S). Volunteers are asked which image is more realistic (images are randomly scrambled). We can draw some conclusions from Table 1: our method is better than StarGAN and GANimation, but worse than STGAN in gender swap, eyeglasses adding, and smiling swap. In general, our method has a performance close to STGAN in I2I translation task.

In particular, our method has great advantages in image reconstruction since $I_t = I_o$ when $\boldsymbol{v}_s = 0$. The peak signal to noise ratio (PSNR) and structural similarity (SSIM) of reconstructed image of StarGAN and STGAN are 22.80/0.819 and 31.67/0.948 respectively reported by [15]. By contrast, PSNR/SSIM of reconstructed image of our method is ∞/1.

However, as our mentioned early, our goal is the facial semantic decomposition and segmentation, but not to have better image-to-image translation. Hence the performance

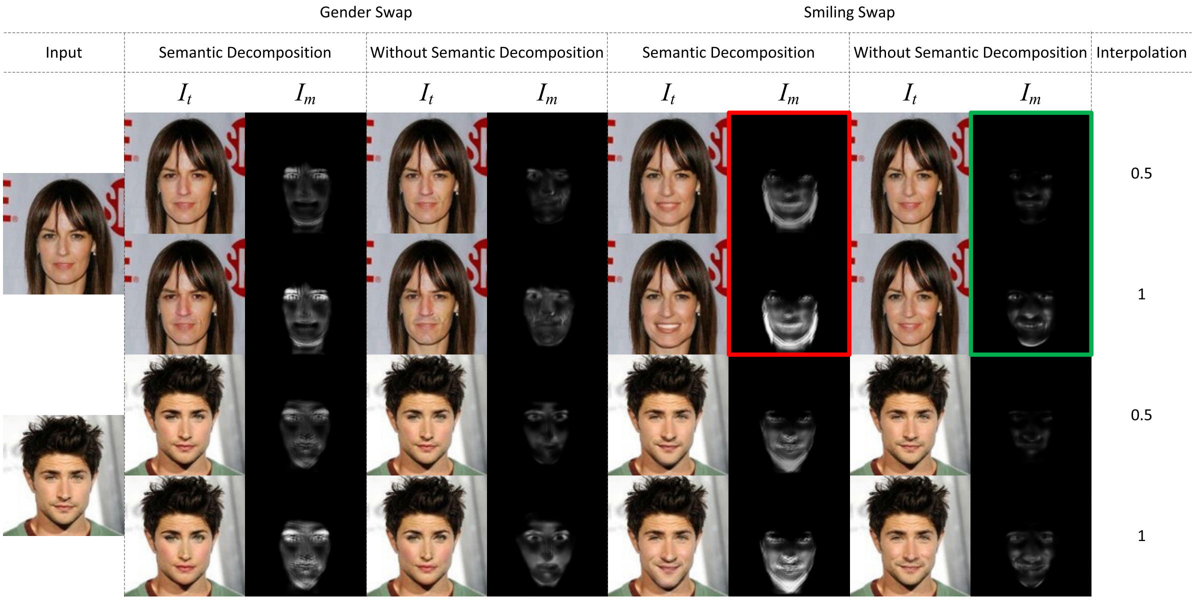

FIGURE 4. Some ablation study results of with/without semantic decomposition.

evaluation of our method for I2I translation is not the focus of this paper. The main advantages of our method are described in detail in the next two sections.

C. FACIAL ATTRIBUTE MANIPULATION WITHOUT GENERATOR

AAMs remain stable since G_a has nothing to do with $\boldsymbol{v}_s$, and each AAM overlays the area of the corresponding attribute. Hence, we can manipulate the color and strength of single attribute without generator after we get AAMs. Color manipulation can be achieved simply by adjusting the value of the pixel with AAM as

$$I_t' = [I_o + \Delta C \cdot M_c], \tag{14}$$

where, ΔC is color adjustment value, M_c is the AAM corresponding to the attribute you want to manipulate, $[\cdot]$ means clip the value to the effective color range.

Figure 6 illustrates the results of arbitrary manipulation of hair color. Hair color can be controlled at will through the hair mask and is no longer subject to attribute labels.

On the other hand, according to (3),

$$I_o = \frac{I_t - I_m \cdot I_c}{1 - I_m} = \frac{I_t - AWM(\boldsymbol{M}) \cdot I_c}{1 - AWM(\boldsymbol{M})}, \tag{15}$$

where I_t, I_c, and M are already known after the generation of I_t. Hence, we can reconstruct I_o without generator. What's more, we can adjust the reconstruction strength of any attribute by a strength factor $\boldsymbol{\rho}$:

$$I_f = \left[\frac{[I_t - AWM(\boldsymbol{\rho} \cdot \boldsymbol{M}) \cdot I_c]}{1 - AWM(\boldsymbol{\rho} \cdot \boldsymbol{M})}\right], \tag{16}$$

where $\boldsymbol{\rho} \in \{0, \ldots, 1\}^{1 \times c}$, the values in $\boldsymbol{\rho}$ determine the reconstruction strength of attributes. E.g., (16) is equivalent to (15) if all the values in $\boldsymbol{\rho}$ are 1, but only the hair color will turns back to the color in I_o if only the value of hair color in $\boldsymbol{\rho}$ is 1. Therefore, we can adjust the strength of the attribute changes even without the generator. This process can be called fading because I_f fades from the translated face I_t.

Figure 7 shows the results of qualitative comparison between interpolation and fading in different attribute strength manipulation. It makes a small difference in the effect of attribute strength manipulation whether the generator is used or not. I_c changes with $\boldsymbol{v}_s$ when manipulating with generator, but it is an invariant tensor in (16). Therefore, fading can provide a more linear changes for the attribute strength manipulation, attribute change is more obvious when $\rho = 0.2$ and 0.4. However, fading may not suitable for geometry-level manipulation due to the ghosting. For example, compared to the first row, the girl in the second row has a more pronounced double chin when $\rho = 0.4$ and 0.6.

Figure 8 demonstrates the process of attribute strength fading between age and gender. The overlapping areas of their masks cause them to be unable to adjust attributes independently without affecting one another. Interestingly, we can find out which areas are more important for which attributes. Eyebrows, for example, are more important to gender than age.

To quantify the difference between interpolation and fading, we recruited eight volunteers (5 male and 3 female) for user study. Each volunteer was asked to evaluate 70 × 2 × 4 generated faces from 70 persons with 128 × 128 size. Every person has four transformations: gender swap, paler skin, age swap, and smiling swap. There are two ways to do each transformation: 0.5 interpolation and half fading from the completely transformed face (0.5 for $|\boldsymbol{v}_s|$ and $\boldsymbol{\rho}$) respectively. Volunteers are asked two questions for interpolated face and

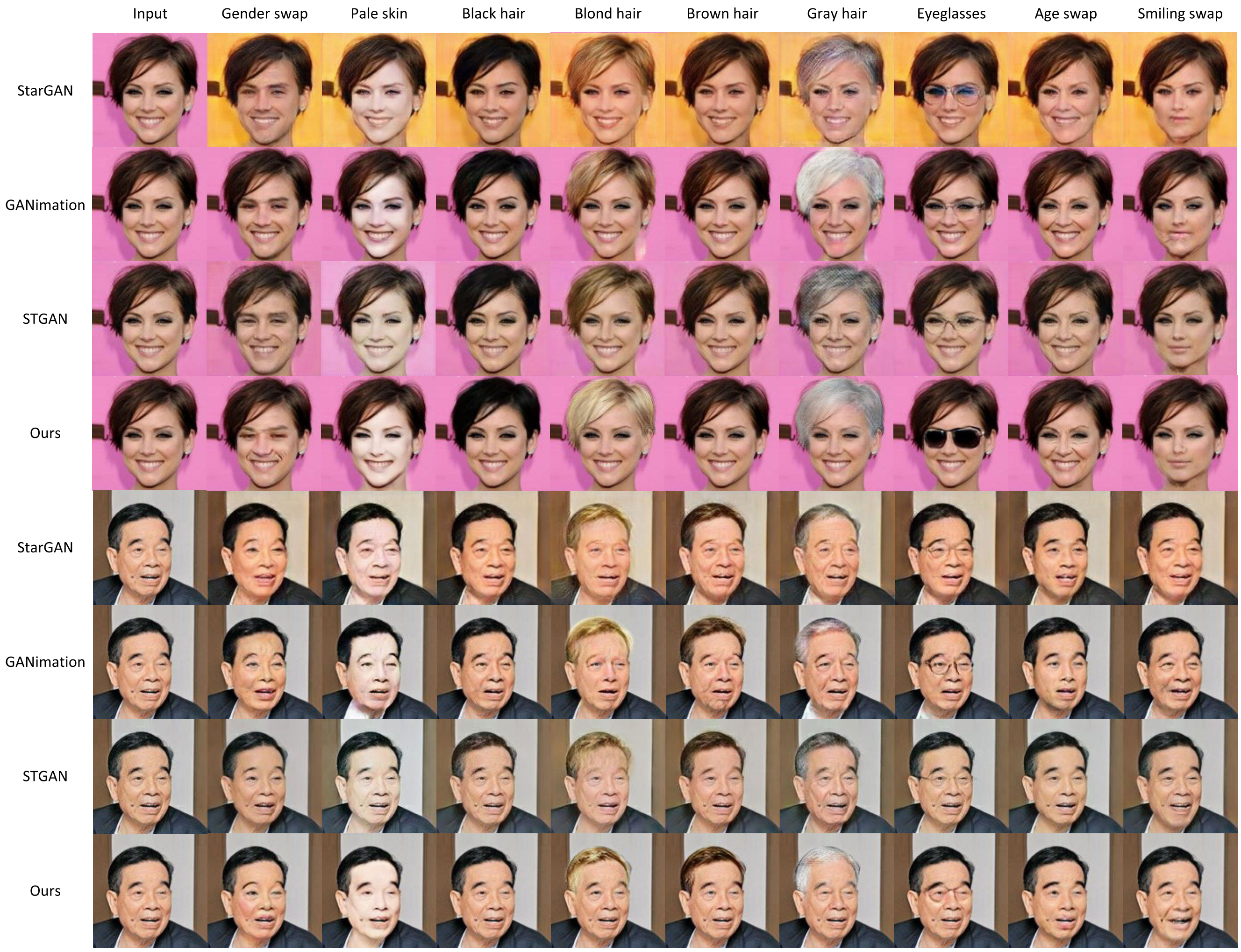

FIGURE 5. The qualitative results of facial attribute manipulation with generator.

faded face: which transformation is more obvious and which image is more realistic (images are randomly scrambled).

We can draw the following conclusions from table 2:

1) In general, the faded image has much more obvious attribute changes than the interpolated image. This proves that fading can provide more linear changes.

2) However, people tend to think that the images with small changes are more realistic. On the one hand, it is due to the lack of reality of the fake images, on the other hand, it may also be because people can speculate the results through hairstyles and so on (e.g., people tend to doubt the reality of men with long hair).

3) It is difficult for people to distinguish the obvious and realistic skin color, which shows that there is not much difference between the two methods in the result of color transformation.

D. FACIAL SEMANTIC SEGMENTATION

Theoretically, the attention mask of single attribute gives us the opportunity to segment facial region automatically, without the supervision by semantic segmentation labels. For example, the area corresponding to the hair color attribute can be used to segment the hair and the skin segmentation in the same way by skin color.

TABLE 2. User study results of interpolation and fading.

Attributes	Method	Obvious	Realistic
Gender swap	interpolation	13.80%	82.52%
	fading	86.20%	17.48%
Paler skin	interpolation	47.90%	56.08%
	fading	52.10%	43.92%
Age swap	interpolation	5.24%	86.20%
	fading	94.76%	13.80%
Smiling swap	interpolation	18.42%	75.60%
	fading	81.58%	24.40%
Total	interpolation	14.15%	80.11%
	fading	85.85%	19.89%

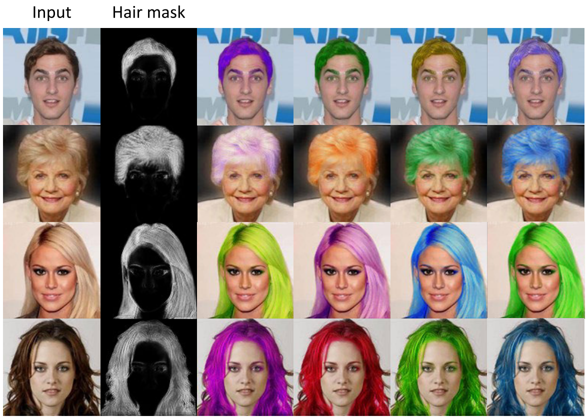

FIGURE 6. Results of hair color manipulation without generator.

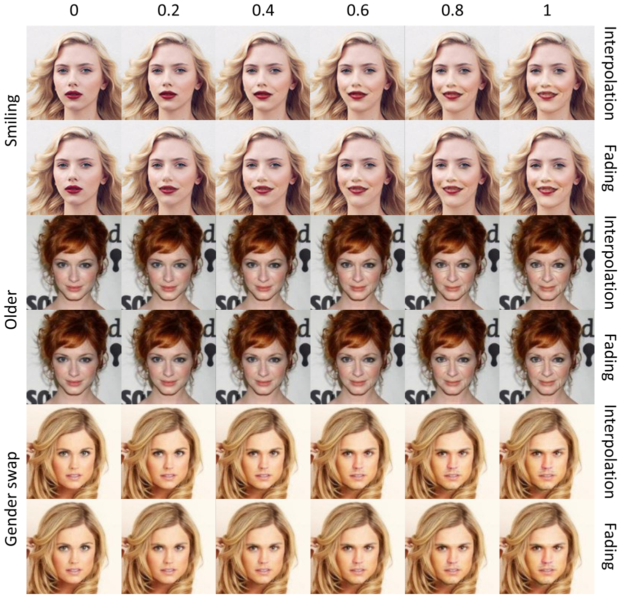

FIGURE 7. Results of attribute strength manipulation with/without generator.

However, the corresponding region of some attributes may contain unexpected regions (e.g., the AAM of hair includes the eyebrows due to they have a same color). On the other hand, some facial features have no corresponding mask such as mouth and eyes. Therefore, semantic segmentation results need to be processed by prior knowledge. Here are the logical rules based on prior knowledge in our method for calculating the face area:

1. Skin = Skin
2. Gender = Gender
3. Eyes = Eyeglasses - Skin
4. Hair = Hair - Skin - Gender - Eyes
5. Mouth = Smiling - Skin - Hair - Eyes

We first binarize each mask with a threshold of 25, and then according to the above rules, we get the semantic segmentation of each face area. Figure 9 shows some semantic segmentation results trained by CelebA-HQ with 256 × 256 size. Although there are still many holes in the image, our method has completed the correct semantic segmentation.

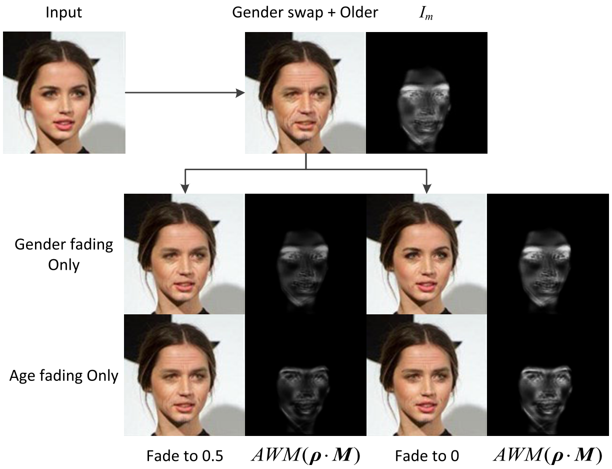
Input
Gender swap + Older
I_m
Gender fading Only
Age fading Only
Fade to 0.5
$AWM(\boldsymbol{\rho}\cdot\boldsymbol{M})$
Fade to 0
$AWM(\boldsymbol{\rho}\cdot\boldsymbol{M})$

FIGURE 8. Attribute strength fading between age and gender.

We use the semantic labels of CelebAMask-HQ as the benchmark to calculate the mIoU (mean intersection over union) with two sizes. The mIoU of each area is shown in Table 3. The "*" in table 2 means the model is trained by CelebA-HQ, otherwise trained by CelebA. We use deeper G_a and D for 256 × 256 size training. Some areas in CelebAMask-HQ are separated such as ears and neck, but our method identifies them all as skin. Hence in ground truth, we uniformly label them as skin.

As mentioned early, our method is the first attempt for facial semantic segmentation only by attribute labels. No other weakly supervised facial semantic segmentation method can be used for performance comparison. Existing weakly semantic segmentation methods such as [37] and [38] are based on class labels but not attribute labels. In these methods, the label indicates the existence of the object, e.g., in the training, [37] and [38] will output the semantic segmentation of the horse if "horse" in the image is labeled as 1, and will not output horse segmentation if "horse" in the image is labeled as 0.

However, attribute label does not indicate the existence of object but the attribute strength, e.g. "pale skin" is 0 doesn't mean there's no skin in the image. Therefore, when applying the existing weakly semantic segmentation methods directly, the methods in [37] and [38] will not output the semantic segmentation result of skin when "Pale skin" is 0. They can only output the skin segmentation when the skin is pale. Therefore, existing weakly semantic segmentation methods can only output segmentation results when the class label ground truth is 1. By the same reason, they cannot output semantic segmentation of the eyeglasses region for the people who do not wear glasses. On the contrary, our method outputs the eyeglasses mask even there is no eyeglasses. Hence, we can find the eyes segmentation by "Eyeglasses - Skin".

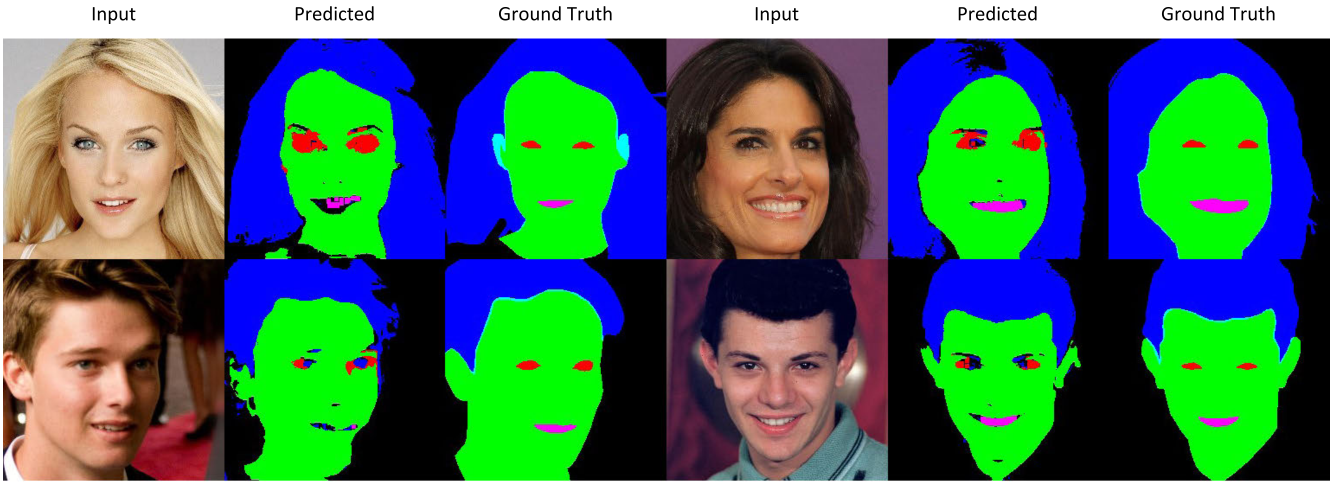

FIGURE 9. Some semantic segmentation results of hair, skin, eyes, and mouth (256 × 256 size).

TABLE 3. The mIoU comparison results (%).

Method	Size	Hair	Skin	Eyes	Mouth	Total
ours	128×128	66.71	79.78	31.96	16.52	48.74
ours	128×128*	65.00	67.26	13.23	14.93	40.10
U-net	128×128*	83.81	87.47	62.01	38.30	67.90
ours	256×256	66.47	76.63	23.68	19.66	46.61
ours	256×256*	66.18	77.44	15.07	23.86	45.64
U-net	256×256*	85.21	91.96	62.81	47.55	71.88

TABLE 4. Generator network architecture.

Part	Input → Output shape	Layer Information
Down-sampling for weights sharing	$(h, w, 3) \rightarrow (h, w, 64)$	conv-(N64, K7x7, S1, P3), IN, ReLU
	$(h, w, 64) \rightarrow (h/2, w/2, 128)$	conv-(N128, K4x4, S2, P1), IN, ReLU
	$(h/2, w/2, 128) \rightarrow (h/4, w/4, 256)$	conv-(N256, K4x4, S2, P1), IN, ReLU
Down-sampling for G_a	$(h/2^i, w/2^i, 256) \rightarrow (h/2^{i+1}, w/2^{i+1}, 256)$	conv-(N256, K4x4, S2, P1), IN, ReLU, $i = 2, 3, 4$
Adaptive residual block ×6 for G_c	$(h/4, w/4, 256) \rightarrow (h/4, w/4, 256)$	conv-(N256, K3x3, S1, P1), AdaIN, ReLU
MLP for G_c	$(1, 2^i \times L_v) \rightarrow (1, 2^{+1} \times L_v)$	Linear-(N$2^i \times L_v$), ReLU,i = 0, 1, 2, 3
	$(1, 16 \times L_v) \rightarrow (1, 512)$	Linear-(N512)
Up-sampling for G_a	$(h/2^{i+1}, w/2^{i+1}, 256) \rightarrow (h/2^i, w/2^i, 256)$	conv-(N256, K5x5, S1, P2), LN, ReLU, $i = 4$
	$(h/2^{i+1}, w/2^{i+1}, 512) \rightarrow (h/2^i, w/2^i, 256)$	conv-(N256, K5x5, S1, P2), LN, ReLU, $i = 3, 2$
	$(h/2^{i+1}, w/2^{i+1}, 512/2^{1-i}) \rightarrow (h/2^i, w/2^i, 128/2^{1-i})$	conv-(N512, K5x5, S1, P2), LN, ReLU, $i = 1, 0$
	$(h, w, 128) \rightarrow (h, w, L_v)$	conv-(NL_v, K7x7, S1, P3)
Up-sampling for G_c	$(h/4, w/4, 256) \rightarrow (h/2, w/2, 128)$	conv-(N128, K5x5, S1, P2), LN, ReLU
	$(h/2, w/2, 128) \rightarrow (h, w, 64)$	conv-(N64, K5x5, S1, P2), LN, ReLU
	$(h, w, 64) \rightarrow (h, w, 3)$	conv-(N3, K7x7, S1, P3)

It is unfair to compare a weakly supervised method with supervised ones. In spite of this, we trained a U-net [19] by CelebAMask-HQ for comparison (20,000 images as training set and 10,000 images as test set). We believe this should help reveal the gap between supervised learning and weakly supervised learning.

We can note that the segmentation of small area such as eyes and mouth by our method is difficult in all size images, meanwhile U-net has much higher accuracy. Therefore, there is still a big performance gap between weakly supervised methods and supervised ones. Interestingly, we found that the mIoU of the skin decreased when the ear was added to ground truth, possibly because CelebAMask-HQ marks the ear area completely, even though it is partially covered, such as the first image in Figure 8. This indicates that the semantic labels of CelebAMask-HQ may have potential defects.

V. CONCLUSION

In this paper, we propose a self-attention-masking semantic decomposition method, which is able to learn an attribute attention mask for each attribute. We decouple the attention of different attributes and overcome the disadvantage of overlap between different attribute attention masks by an attention weighting module. Our method allows manipulating facial attribute without generator after only once generation. User study shows that fading result is more obvious than interpolation result (over 80% for gender swap, age swap, and smiling swap). Moreover, the attention mask of single attribute can perform facial semantic segmentation without pixel level semantic labels, with mIoU over 65% for hair and skin.

Through the attention mask, we can segment the facial image semantically. At the same time, attention mask determines the authenticity of I2I translation. Therefore, the accuracy of this weakly supervised semantic segmentation may also determine the performance of I2I translation. Our future work will focus on improving this accuracy of semantic segmentation. On the other hand, we didn't train a model for

512×512 and 1024×1024 sizes since there is not enough memory for these sizes in one single GPU. We hope that in the future we will be able to achieve more streamlined network structure and larger size image processing.

APPENDIX

Table 4 and 5 show details about the network architecture. We use instance normalization (IN) [39] in all layers in G_a except the last output layer. In G_c, we use IN in all down-sampling layers except the weights sharing layers, and layer normalization (LN) [40] in all up-sampling layers except the output layer. We use nearest neighbor sampling before the convolution for up-sampling. For the discriminator network, we use Leaky ReLU with a negative slope of 0.02. A standard self-attention module is applied in the middle of discriminator. In tables, N is the number of output channels, K is kernel size, S is stride size, P is padding size, and L_v is the size of attribute vector.

TABLE 5. Discriminator network architecture.

Layer	Input → Output shape	Layer Information
Input Layer	$(h, w, 3) \rightarrow (h/2, w/2, 64)$	conv-(N64, K4x4, S2, P1), Leaky ReLU
Hidden Layer	$(h/2^i, w/^i, 64\times2^{i-1}) \rightarrow (h/2^{i+1}, w/^{i+1}, 64\times 2^i)$	conv-(N64×2^i, K4x4, S2, P1), Leaky ReLU, i=1,2
Self-attention module	$(h/8, w/8, 256) \rightarrow (h/8, w/8, 256)$	Standard self-attention module [29]
Hidden Layer	$(h/2^i, w/^i, 64\times2^{i-1}) \rightarrow (h/2^{i+1}, w/^{i+1}, 64\times 2^i)$	conv-(N64×2^i, K4x4, S2, P1), Leaky ReLU, i=3,4,5
Output Layer (D_{adv})	$(h/64, w/64, 2048) \rightarrow (h/64, w/64, 1)$	conv-(N1, K3x3, S1, P1)
Output Layer (D_{cls})	$(h/64, w/64, 2048) \rightarrow (1, 1, L_v)$	conv-(NL_v, K2x2, S1, P0)

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, and B. Xu, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[2] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, "MaskGAN: Towards diverse and interactive facial image manipulation," 2019, *arXiv:1907.11922*. [Online]. Available: http://arxiv.org/abs/1907.11922

[3] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain Image-to-Image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8789–8797.

[4] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 700–708.

[5] G. Lample, N. Zeghidour, N. Usunier, A. Bordes, and L. Denoyer, "Fader networks: Manipulating images by sliding attributes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5967–5976.

[6] T. Xiao, J. Hong, and J. Ma, "Elegant: Exchanging latent encodings with GAN for transferring multiple face attributes," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 168–184.

[7] W. Yin, Z. Liu, and C. C. Loy, "Instance-level facial attributes transfer with geometry-aware flow," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, Jul. 2019, pp. 9111–9118.

[8] Y. Jo and J. Park, "SC-FEGAN: Face editing generative adversarial network with user's sketch and color," 2019, *arXiv:1902.06838*. [Online]. Available: http://arxiv.org/abs/1902.06838

[9] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "AttGAN: Facial attribute editing by only changing what you want," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5464–5478, Nov. 2019.

[10] Y.-C. Chen, X. Shen, Z. Lin, X. Lu, I.-M. Pao, and J. Jia, "Semantic component decomposition for face attribute manipulation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9859–9867.

[11] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2223–2232.

[13] W. Shen and R. Liu, "Learning residual images for face attribute manipulation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4030–4038.

[14] A. A. A. Pumarola, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, "Ganimation: Anatomically-aware facial animation from a single image," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 818–833.

[15] M. Liu, Y. Ding, M. Xia, X. Liu, E. Ding, W. Zuo, and S. Wen, "STGAN: A unified selective transfer network for arbitrary image attribute editing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3673–3682.

[16] P. Upchurch, J. Gardner, K. Bala, R. Pless, N. Snavely, and K. Weinberger, "Deep feature interpolation for image content changes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 7064–7073.

[17] Y.-C. Chen, H. Lin, M. Shu, R. Li, X. Tao, Y. Ye, X. Shen, and J. Jia, "Facelet-bank for fast portrait manipulation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3541–3549.

[18] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[19] O. Ronneberger, P. Fischer, and T. Brox., "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[20] D. Pathak, E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional multi-class multiple instance learning," in *Proc. ICLR Workshop*, 2015, pp. 1–4.

[21] G. Papandreou, L.-C. Chen, K. Murphy, and A. L. Yuille, "Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1742–1750.

[22] D. Pathak, P. Krahenbuhl, and T. Darrell, "Constrained convolutional neural networks for weakly supervised segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1796–1804.

[23] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, "Simple does it: Weakly supervised instance and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 876–885.

[24] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, and T. S. Huang, "Revisiting dilated convolution: A simple approach for Weakly- and semi-supervised semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7268–7277.

[25] S. Mittal, M. Tatarchenko, and T. Brox, "Semi-supervised semantic segmentation with High- and low-level consistency," 2019, *arXiv:1908.05724*. [Online]. Available: http://arxiv.org/abs/1908.05724

[26] Y. Wei, J. Feng, X. Liang, M.-M. Cheng, Y. Zhao, and S. Yan, "Object region mining with adversarial erasing: A simple classification to semantic segmentation approach," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1568–1576.

[27] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[28] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.

[29] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," 2018, *arXiv:1805.08318*. [Online]. Available: http://arxiv.org/abs/1805.08318

[30] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[31] I. A. F. Gulrajani, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5767–5777.

[32] M. Li, W. Zuo, and D. Zhang, “Deep identity-aware transfer of facial attributes,” 2016, *arXiv:1610.05586*. [Online]. Available: http://arxiv.org/abs/1610.05586

[33] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, “Invertible conditional GANs for image editing,” 2016, *arXiv:1611.06355*. [Online]. Available: http://arxiv.org/abs/1611.06355

[34] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.

[35] T. Karras, T. Aila, S. Laine, and J. Lehtinen, “Progressive growing of GANs for improved quality, stability, and variation,” 2017, *arXiv:1710.10196*. [Online]. Available: http://arxiv.org/abs/1710.10196

[36] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[37] W. Zhang, S. Zeng, D. Wang, and X. Xue, “Weakly supervised semantic segmentation for social images,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015.

[38] Y. Wei, X. Liang, Y. Chen, X. Shen, M.-M. Cheng, J. Feng, Y. Zhao, and S. Yan, “STC: A simple to complex framework for weakly-supervised semantic segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2314–2320, Nov. 2017.

[39] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Instance normalization: The missing ingredient for fast stylization,” 2016, *arXiv:1607.08022*. [Online]. Available: http://arxiv.org/abs/1607.08022

[40] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” 2016, *arXiv:1607.06450*. [Online]. Available: http://arxiv.org/abs/1607.06450

XUAN XIA received the Ph.D. degree in instrument science and technology from Shanghai Jiao Tong University, in 2017. He currently holds a postdoctoral position at the Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences. He is also currently a Researcher with the Shenzhen Institute of Artificial Intelligence and Robotics for Society. His research areas include deep learning, pattern recognition, image processing, time-frequency analysis, navigation and positioning, and signal processing.

FENGQI YU received the Ph.D. degree from the Integrated Circuits and Systems Laboratory (ICSL), University of California at Los Angles (UCLA), Los Angles. In 2006, he joined the Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences, China, as a Full Professor and the Director of the Department of Integrated Electronics. His research and development interests include CMOS RF integrated circuit design, CMOS sensor design, wireless sensor networks, RFID, and the Internet of Things.

NAN LI received the B.E. degree from Hunan University, in 2011, and the Ph.D. degree in mechanical engineering from China Agricultural University, in 2017. From 2015 to 2016, he was a Visiting Scholar with the Department of Agricultural and Biological Engineering, University of Illinois at Urbana–Champaign. He is currently a Researcher with the Shenzhen Institute of Artificial Intelligence and Robotics for Society. His main research interests include machine vision and robotics.

YANSONG QU is currently pursuing the Ph.D. degree from Auburn University. He is currently an Intern Student with the Centre for Micro/Nano Systems and Bionic Medicine, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences. His research interests include image segmentation, salient object detection, and industry control.

JIAJIA ZHANG received the M.S. and Ph.D. degrees in computer sciences from the Harbin Institute of Technology, in 2009 and 2015, respectively. He held a postdoctoral position at Peking University, from 2015 to 2018. He is currently an Associate Research Fellow of the Harbin Institute of Technology (Shenzhen). His main research interests include artificial intelligence, computer game, intelligence strategy decision, and cyberspace security.

CHENGGUANG ZHU received the bachelor’s degree from Shenyang Ligong University, Shenyang, China, in 2010, and the master’s degree from the Institute of Seismology, China Earthquake Administration, Wuhan, China, in 2013. He is currently pursuing the Ph.D. degree with Shanghai Jiao Tong University. His current interests include error analysis, image processing, visual navigation, and relative pose estimation.