InjectionGAN: Unified Generative Adversarial Networks for Arbitrary Image Attribute Editing

CHEN DING, WEI KANG, JIAQI ZHU, AND SHUANGYAN DU

1College of Information Engineering, Fuyang Normal University, Fuyang 236041, China
2School of Computer and Information, Hefei University of Technology, Hefei 230601, China

Corresponding author: Wei Kang (kangwei0830@163.com)

This work was supported in part by the Natural Science Project of the Higher Education Institutions of Anhui Province under Grant KJ2019A0949 and Grant KJ2019A0589, in part by the Natural Science Foundation of Fuyang Normal University under Grant 2019FKJ01ZD, in part by the Teaching Research Project of the Higher Education Institutions of Anhui Province under Grant 2018jyxm0209, in part by the Anhui Provincial Natural Science Foundation under Grant 2008085QA16, in part by the Project of College of Information Engineering, Fuyang Normal University under Grant FXG20202ZY01 and 2018FXJK04.

ABSTRACT Existing image-to-image translation methods usually incorporate encoder-decoder and generative adversarial networks to generate images. The encoder compresses an entire image into a static representation using a sequence of convolution layers until a bottleneck, and then, the intermediate features are decoded to the target image. However, the existence of bottleneck layer in those approaches still has limitations in the sharpness of details, distinct image translation and identity preservation, since different domain translations may be related to the global or local region in the input image. To address these issues, we propose a new model, InjectionGAN, based on a novel generative adversarial network (GAN) for arbitrary attribute transfer. Specifically, conditional on the target domain label, an auto-encoder-like network with multiple linear transformation and refinement connections are trained to translate the input image into the target domain. The connections block better shuttle the low-level information in the encoder to the decoder, which helps to preserve the structural information while modify the appearance slightly at the pixel level through adversarial training. The results on two popular datasets suggest that InjectionGAN achieves a better performance.

INDEX TERMS Facial attribute transfer, GAN, image-to-image translation.

I. INTRODUCTION Image-to-image (I2I) translation is a task that aims to translate a given image from one domain to another. Many computer vision problems can be handled in this framework, e.g., style transfer [1], image colorization [2], image inpainting [3] and semantic segmentation [5]. Image attribute transfer has attracted great interest in the context of image translation. The promising results should not only contain the attributes of interest, but also keep the other attributes and background information. The previous works [7], [10] often tackle this problem by explicitly designing the loss function, while the target attributes are learned from the desired attributes by entering a pair of different images. Although such method can accurately capture the target attributes from the paired images, the paired images are impracticable to collect paired images with or without desirable attributes.

With the success of Variational Autoencoders (VAEs) [9] and Generative Adversarial Networks (GANs) [12] in diverse image processing tasks [6], [8], [13]–[16], GAN based methods [4], [11], [17]–[19] are springing up and used most on image attribute transfer. By implicitly defining the loss function, the discriminator and generator combat with each other, and eventually reach a state of equilibrium. Therefore, the GAN based method is more flexible in unpaired data. CycleGAN [4] propose a framework with inverse mapping function for unsupervised data, which using Cycle-Consistent Adversarial Network to perform attribute transfer as a cross-domain image-to-image translation task, which is able to transfer a single attribute from a source image to the target one. FaderNet [20] learns an attribute subspace by imposing adversarial constraint to enforce the independence between latent code and attributes, which can transfer a single attribute through toggling the two encoding parts in the learned subspace. However, these methods are only able to learn a single model for specific attribute transfer, which are inflexible for the multiple-attribute scenario.

For multiple-attribute transfer, IcGAN [21] can reconstruct and modify an input image of a face conditioned on arbitrary attributes by combining cGAN [22] with an attribute-independent encoder, and at the inference stage mapping a real image into a latent space and a conditional...
representation. However, IcGAN imposes attribute-independent constraint and normal distribution constraint on latent representation, resulting in low-resolution and blurry images. AttGAN [23] and StarGAN [24] are two models which tackle multi-domain attribute editing by taking target attribute vector as input to the transform model. AttGAN [25] performs facial attribute transfer based on the target attribute vector by concatenating attribute vector to image latent representations. In contrast, StarGAN [24] directly takes source image and target attribute vector as input. However, both of two methods lack the ability to model attributes effectively and cannot disentangle multiple attributes, thereby failing to control the procedure of attribute transfer. As shown in Fig. 2, when the given target attributes are unchanged, obvious visual changes can be observed in the reconstruction results of AttGAN and StarGAN.

In this work, we propose a generalized conditioning scheme to incorporate target attribute vector into the multiple-attribute transfer model. There are two key differences between our proposed approach and the existing multiple-attribute transfer methods. Firstly, we propose to apply the attribute vector as the conditioning operation in skip connection, but from the latent representation to guidance image as well. Secondly, we extend the existing feature-wise feature transformation in the skip connection to be spatially varying to adapt to different attribute features in the input image. We refer to our proposed approach as InjectionGAN, a deep conditional generative model that use target attribute as a guiding scheme. We validate the design of InjectionGAN through extensive experiments, including facial attribute transfer and season translation. We demonstrate that our method achieves competitive or better performance than the state-of-the-art. Through extensive ablation study, we also show that the proposed InjectionGAN is more effective than commonly used conditional schemes. In Fig. 1 we show the results of our single or multiple attribute transfer.

We make the following two contributions. First, we present the multiple-attribute feature transformation for generic target attribute vector guided image-to-image translation tasks. Compared to existing approaches that only allow the information flow from latent representation guidance to the source image, we show that incorporating target attribute vector information from the low-level to the high-level further help improve the performance of the end task. Second, we propose a spatially varying extension of feature-wise transformation to better capture local attribute features from the target attribute vector and the source image.

II. RELATED WORK

A. GENERATIVE ADVERSARIAL NETWORK

Generative Adversarial Network (GAN) is becoming a hot topic in recent years, and used for image generation tasks [25]–[28]. GAN comprises two components: a generator G and a discriminator D, which seek a Nash equilibrium within the adversarial training scheme. The aim of the generator is to capture the distribution of training samples and learns to generate new samples imitating the training ones, while the discriminator try to distinguish the generated samples from the training ones. By using the min-max game strategy for adversarial training, the difference between the distribution of generated images and the distribution of real data is minimized, thereby encouraging the generated results to be more realistic. There are many improvements based on the GAN model. DCGAN [29] refined the design of the network of generators and discriminators, which facilitated the application of GANs in many image generation tasks. cGAN [22] modifies GAN from unsupervised learning to semi-supervised learning by taking conditional variable as input to the generator and discriminator to generate image with desired properties. WGAN [30] utilize the Wasserstein distance and gradient penalty to improve stability of the optimization process.

B. IMAGE-TO-IMAGE TRANSLATION

The goal of image-to-image translation is to learn cross-domain mapping in supervised or unsupervised settings. Given pairs of data samples, pix2pix [2] presented a unified framework for learning image-to-image translation. To overcome the limitations of paired image-to-image translation, improved network architectures have been proposed. Specifically, CycleGAN [4] learns the Cycle-Consistent mappings between two image domains are enforced to explicitly reconstruct the source sample, which is translated into the target domain, allowing translation using unpaired data.. BicycleGAN [31] shows the mapping between the output and the latent representation, and pulls the latent distribution closer to the known one. Finally, a diverse output is obtained by sampling from the underlying distribution. In addition, UNIT [32] decouple generators by learning domain-specific encoders/decoders with shared latent space in two domains,
which shows remarkable results without paired data. However, the above models all try to learn the distribution between two domains, which limits them to deal with multiple domains at the same time. Recently, several image-to-image translation methods [33]–[35] successfully perform cross-domain retrieval as well as one-to-many translation with the usage of noise or labels.

C. FACIAL ATTRIBUTE TRANSFER

Facial attribute transfer actually is an unpaired multi-domain image-to-image translation task. Some methods implement the transfer of attributes by exchanging latent codes in the latent attribute space of the input image. ModularGAN [36] presents a modular architecture consisting of several reusable to connect specific attribute editing to arbitrary attribute editing, but it cost too much computation time. IcGAN [21] achieve the latent representation by sampling from a uniform distribution and therefore independent of the attributes. Then the attribute editing is performed by combining the latent representation of image and the given attributes. StarGAN [24] realize attribute transfer by designing a conditional attribute transfer network to learn attributes in a cyclic process. One of the novelties in StarGAN is that its discriminator D introduces a loss of attribute classification, allowing the original image to be reconstructed using a given original domain label. However, StarGAN uses a shallow encoder-decoder with residual blocks, which does not involve any latent representations, so its ability to change facial attributes is limited. AttGAN [23] applies a U-Net structure [37] to model the relationship between the latent representations and the attributes. It contains three loss functions at training: the attribute classification learning, the reconstruction and the adversarial constraint. However, AttGAN imposing the attribute-independent constraint from the latent representation, while our approach encodes the target attribute vector into different skip connection and latent representation to generate texture for face transfer.

D. CONDITIONAL NORMALIZATION SCHEME

Conditional normalization schemes have been an important method in modern deep neural networks. The most recent conditional normalization scheme, Feature-wise Linear Modulation (FiLM) [38], successfully infers information from external data and applies it as an affine transformation parameter to vision tasks. In this approach, the learned affine transformation is spatially invariant. There are some FiLM-like normalization, such as, conditional instance normalization (CIN) [39], which can be seen as a FiLM layer replacing a normalization layer. The activations of the feature representation is first normalized to zero mean and unit deviation. Then the learned activations are applied to the normalized feature representation to guidance style image. Another such approach is adaptive instance normalization (AdaIN) [40], it applies a strategy similar to CIN. However, the biggest difference between AdaIN and CIN is that AdaIN does not learn affine transformation parameters, but instead uses the mean and standard deviation of the guidance style image as scaling and shifting parameters, respectively.

Unlike existing conditional normalization schemes that allow guidance information flow only from the external data to the latent representation, we show that the proposed InjectionGAN brings considerable performance improvements. In addition, we have improved existing spatially invariant feature transformation methods to support spatially-varying function.

III. METHOD

In this work, we aim to perform image attribute transfer while preserving its own identity, and in addition to obeying the constraints specified by the given target attribute vectors. To tackle this problem, we propose InjectionGAN to incorporate the additional target attribute vector into the conditional generative model. We firstly start with the formulation and the implementation details of the proposed InjectionGAN, and then we present the network architecture of the whole model. Finally we introduce the loss function for better optimization of the model.

A. FORMULATION

Feature Transformation (FT) layer can be regarded as a variant of conditional normalization, which encodes the guidance information as an affine transformation. In an FT layer, we firstly perform features refine at skip connection layer by inferring an affine transformation on the normalized input features using scaling and shifting parameters computed from the features of the given target attribute vector. Then we use refinement block as contextual cues to select subtle features that help with attribute transfer. The framework of FT is illustrated in the top part of Fig. 2. The details of FT are shown below.

The input low-level feature $x$ is a batch of size $n \times c \times h \times w$, where $n$, $c$, $h$, $w$ respectively represent batch size, feature maps, height and width. It begins by learning the scaling and shifting parameters $\alpha$ and $\beta$, which are computed from the target attribute vector using a parameter generator. Then we use the learned affine transformation to modulate the normalization of low-level feature maps. Finally, the modulated normalization layer extracts salient attribute features and suppresses irrelevant attribute features through the refinement module. In Eqn. 1, we show this operation for an $l^{th}$ layer.

$$f(\alpha_{n,c,h,w}(d))X_{n,c,h,w}^i + \beta_{n,c,h,w}^i(d))$$

where $f(x)$ represents a refinement block, consisting of convolutions of several successive compression and reduction channels. $\hat{X}_{n,c,h,w}^i$ denotes the normalization step of each channel, we denote it as:

$$\hat{X}_{n,c,h,w}^i = \frac{X_{n,c,h,w}^i - \mu_c^i}{\sigma_c^i}$$

where $\mu$ and $\sigma$ are each channel of features, normalized using mean and variance of a batch over the channel.
The scaling \( \alpha_{n,c,h,w}(d) \) and shifting \( \text{beta}^i_{n,c,h,w}(d) \) of the FT layers are learned from target attribute vector for modulating the normalization layer. Our implementations of \( \alpha_{n,c,h,w}(d) \) and \( \text{beta}^i_{n,c,h,w}(d) \) is different from FiLM [40] layer. The affine transformation of the FiLM layers are vectors and are applied channel-wise. That is, this method ignores the spatial position on the feature map and uses the same affine transformation for all features. Such methods are reasonable when doing global texture changes, such as style transfer and visual reasoning. However, they may not be able to perform fine local image-to-image translation. In contrast, due to the deployment of our FT layers, we obtained a function of spatial-variation that can change local attribute details while paying attention to the overall texture of the image. Note that our implementations of affine transformation is simple. The scaling is first generated by a convolution layer. Then the shifting is produced by another convolution layer and a sigmoid operation. The reason we do this is to allow the affine parameters to control the generative model to better capture the constraints of the target attributes. In addition, in order to capture hard-to-find local fine attribute details, we added an additional refinement module. It discards details that are not related to attribute changes by compressing and restoring the number of channels, and then due to the addition of shortcut, retaining the original identity and unchanged attributes of the image. By combining these two strategies, our network can well complete different attribute transfer inference tasks.

The performance of FT module is attributed to its inherent characteristics. FT apply position dependent affine-transformations to the normalization parameters of the training network, and it allows to modify the shallow layer based on the spatially modified attribute information of the image without training a new network. In addition, we can obtain coarse-to-fine semantic information and control the effects of attribute modification at multiple levels of the network. Modification to a layer that is closer to the input tends to transform more global features. Our FT layer is deployed throughout the entire skip connection, so modifications to layers closer to the input tend to translate more global features.

### B. FRAMEWORK

The InjectionGAN consists of two modules, a discriminator \( D \) to distinguish between real and fake images and classify the real images to its corresponding domain, and a generator \( G \) generates a fake image using both the image and target attribute vector, see Fig.2. The architecture we use is similar to U-Net [37], the FT module is embedded in a symmetric
skip connection, and information is obtained from the target attribute vector to guide the information flow.

The network structure of $G$ has two branches $G_e$ and $G_d$. The encoder $G_e$ is a stack of five convolution layers with kernel size 4 and stride 2, while the decoder $G_d$ is a stack of five transposed convolution layers.

The discriminator $D$ is composed of three parts. The stack of convolutional layers serves as the shared layer of the next two parts.

The fully connected layer $D_{adv}$ consists of five convolutional layers to distinguish whether the image is a fake image or a real image. The classifier $D_{cls}$ has similar architecture to $D_{adv}$, but can predict attribute vectors.

C. Objective Function

For a given image $x$, the encoder $G_e$ firstly performs downsampling to obtain the abstract attributes and content of the image, the static representation is denoted as

$$r = G_e(x)$$

where $r = \{r_1, \cdots, r_5\}$. Then the FT module is deployed in all skip connections, using the target attribute vector as a guide to the generation of semantics. Let denote our FT block. Then the process of combining the FT module and the target attribute vector can be obtained by,

$$r'_f = F(v, r_e')$$

Note that we adopt four FT blocks, and directly pass $r_5$ to $G_d$. FT modules deployed at different layers do not share parameters because they can sense different semantic information at multiple scales.

Let $r_f = \{r'_1, \cdots, r'_d\}$. Thus, the edited result of the generated network can be given by,

$$y = G_d(r, r_f, v)$$

The full loss function includes the following losses: the adversarial loss that distinguishes the synthetic data distribution from the real distribution, the domain classification loss helps generate models to learn specific attributes of a given target label, and the image reconstruction loss, which guarantees the translation image Identity.

1) Adversarial Loss

To stabilize the generator, we employ Wasserstein GAN (WGAN) [30] and WGAN-GP [41] to distinguish the generated images from the real images. When we give the specified target attribute vector, the discriminator tries to distinguish between fake samples from the generator and real images, while G tries to fool the discriminator. The final loss function is optimized by the minimax game.

$$L_{adv}^G = -\mathbb{E}_x[\log D_{adv}(x)] + \mathbb{E}_x, v[\log D_{adv}(G(x, v))]$$

$$L_{adv}^D = -\mathbb{E}_x[\log D_{adv}(x)] + \mathbb{E}_x, v[\log D_{adv}(G(x, v))]$$

$$L_{adv}^{GP} = \mathbb{E}_x[\| \nabla_x D_{adv}(x) \|_2^2 - 1]^2$$

where $G(x, v)$ represents the fake samples, $\hat{x}$ denotes uniformly sampling along lines between paired real images and generated images.

2) Attribute Classification Loss

To train the generator to translate input image from source domain to the target domain according to the target attribute vector $v$, we add an auxiliary domain classifier $D_{adv}$ to the original discriminator $D$ which shares the convolution layers with $D_{adv}$. Therefore, the discriminator is trained not only to output a probability distribution over given input images and domain label, but also to distinguish the real and fake images. The domain classification loss is defined as:

$$L_{cls}^D = \mathbb{E}_{x, \hat{v}}[-\log D_{cls}(\hat{v}|x)]$$

$$L_{cls}^G = \mathbb{E}_{x, v}[-\log D_{cls}(v|G(x, v))]$$

3) Image Reconstruction Loss

The adversarial loss and domain classification loss enable to generate realistic facial textures with proper attributes. However, we cannot guarantee that the generated facial texture will retain the content of the input facial texture when the domain label is unchanged. The generator should learn to preserve the content of input images while changing only the domain related region of the image. To this end, we apply image reconstruction loss to the generator, defined as:

$$L_{rec}^G = \mathbb{E}_{x, v}[\| x - G(x, v) \|_1]$$

4) Total Loss

The complete objective function of the model we discussed above is:

$$L_D = L_D^D + \lambda_1 L_{cls}^D$$

$$L_G = L_G^{adv} + \lambda_2 L_{cls}^G + \lambda_3 L_{rec}^G$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are coefficients to balance the loss terms.

IV. Experiments

Our framework is optimized with Adam [42], where with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. We flip the images with a probability of 0.5 to augment the training dataset. As described in WGAN-GP [41], we perform one generator update after five discriminator updates. The batch size and training epoch time were about 64 and 200, respectively. We train the generator and discriminator in the first 100 epochs with an initial learning rate of 0.0001, and then linearly decay the learning rate to 0 in the next 100 epochs.

A. Facial Attribute Transfer

1) Dataset and Preprocessing

The Celeb Faces Attributes dataset [43] contains 202,599 face images and 40 attribute labels per face. To explicitly preserve the face ratio, we resize all aligned images from the $178 \times 178$
C. Ding et al.: InjectionGAN: Unified GANs for Arbitrary Image Attribute Editing

FIGURE 3. Comparisons with IcGAN [21], FaderNet [20], AttGAN [23], StarGAN [24] and our InjectionGAN on editing (inverting) specified attributes.

We leave out 1000 randomly sample images for validation and use all remaining validation and training images as training data. We mainly test thirty domains with the following attributes: Bald, Bangs, Black Hair, Blond Hair, Brown Hair, Bushy Eyebrows, Eyeglasses, Male, Mouth Slightly Open, Mustache, No Beard, Pale Skin and Young.

2) QUALITATIVE RESULTS

Fig.4 shows the single attribute transfer results generated by IcGAN [22], FaderNet [20], AttGAN [23] and StarGAN [24] and our proposed model. The first row is the input image. The following rows shows the translated images which changes the hair, changes the gender, changes the expression and

into 128 × 128.
the eyeglasses. It is worth noting that we also put the reconstructed image for comparison, which can better reflect the superiority of our method. From the figure, we can find that both IcGAN and FaderNet are able to perform image-to-image translation to each domain, they fail to well preserve the identity of the input image. In addition, their results are not visually plausible. For AttGAN and StarGAN, both of them can perform well on all the translation tasks, but some of their results are insufficiently modified. In some cases, such as changing the expression or changing the hair, results generated by AttGAN and StarGAN are low-resolution and blurry. In contrast, the images generated by our model are noticeable changes from the input image. For example, the hair region changes greatly when changing the bald. When changing the gender from male to female, the eye region and facial region are modified to be more woman-like without changing the hair.

More generated results of our proposed model is shown in Fig.5. We compare our model with AttGAN [23] and StarGAN [24] for multiple facial attribute transfer. Although the generated images of AttGAN and StarGAN seem have the correct attributes, they are not realistic. The main reason for the poor performance of AttGAN and StarGAN comes from their inability to effectively separate image attributes and background. We can see that InjectionGAN can successfully transform the input image from the source domain to multiple different domains and generate a realistic image while retaining the identity of the image. This is mainly because our FT layer slightly reduces the compactness of the network, which increases the freedom of the generator.
TABLE 1. Reconstruction quality of the comparison methods on facial attribute editing task.

| Method          | IoGAN | FaderNet | AttGAN | StarGAN | InjectionGAN |
|-----------------|-------|----------|--------|---------|--------------|
| PSNR/SSIM       | 35.28/0.940 | 30.62/0.908 | 24.07/0.841 | 22.82/0.821 | 31.37/0.949 |

3) QUALITATIVE EVALUATION

Table 1 and Fig. 4 shows qualitative evaluation on CelebA dataset [43]. As we can see in Table 1, our method obtains the best scores in both PSNR/SSIM results of reconstruction. In the case of PSNR, our method is 31.67, while AttGAN and StarGAN achieve 24.07 and 22.80 respectively. This clearly indicates that InjectionGAN can successfully generate realistic outputs using a single model. In addition, the high SSIM indicates that InjectionGAN effectively preserves the identity, achieving a competitive performance with FaderNet [22]. The better reconstruction results of FaderNet are mainly due to the fact that FaderNet model training can only handle one attribute one time.

Fig. 4 reports the attribute editing accuracy of generated images on the CelebA dataset. Higher value means the translated image is more likely in the target domain. The results show that our InjectionGAN has highest classification accuracy which is consistent with the qualitative resultst that we analyzed previously. This is mainly because that our proposed model can well keep the structure and identity of model in the image, and only change the domain related region. In addition, our method is 20% more accurate than other methods in the attributes of black hair, brown hair, beard, eyebrows and bald.

4) USER STUDY

We conducted user research to evaluate the image generation quality of AttGAN, StarGAN and InjectionGAN. We consider 11 attributes because hair color transfer between blonde, black and brown hair can be merged into hair color. For each attribute, 40 people were involved, and each person was given 20 questions. In each question, people are provided with source images from the test set and the edited results of AttGAN, StarGAN and InjectionGAN. For fair comparison, the results are shown in random order. The results are shown in Table 2. In all 11 tasks, the results of InjectionGAN are more attractive.

TABLE 2. Reconstruction quality of the comparison methods on facial attribute editing task.

| Method          | Bald | Bang | Eyebrow | Glasses | Hair Color | Male |
|-----------------|------|------|---------|---------|------------|------|
| AttGAN          | 12.76% | 34.28% | 10.64% | 30.04% | 11.32% | 15.68% |
| StarGAN         | 11.28% | 18.12% | 19.40% | 19.20% | 32.28% | 13.52% |
| InjectionGAN    | 75.96% | 47.60% | 69.36% | 50.76% | 58.20% | 70.80% |

B. SEASON TRANSITION

Our method can also be extended to other image-to-image translation tasks. By treating image styles as labels, we applied our method to a dataset published by CycleGAN [4] and performed conversion experiments between summer and winter. This dataset contains thousands of different summer and winter pictures. We follow the data processing strategy performed by CycleGAN and compare the results of the competition method with the published open source results of CycleGAN.

FIGURE 6. Exploration of AttGAN [23], StarGAN [24], CycleGAN [4] and our InjectionGAN on season translation.

It is worth noting that CycleGAN [4] requires a single generator to be trained for each pair of domains, while others including our proposed model can translate the input image to multiple domains using one model. As we can see in Fig. 6, the results of season are acceptable For CycleGAN, the results are not so good accompanied with artifacts and bluriness. In addition, it is difficult for AttGAN [23] and StarGAN [24] to deal with all the large variation at the same time when performing seasonal changes. Although our method contain some artifacts, but can still prove that our method is a potential framework which deserves more explorations and extensions.

V. ABLATION STUDY

To verify the effectiveness of the FT blocks in our model, we conduct the ablation studies with different variants of our model. First build a model which denotes as InjectionGAN-res that removes the affine parameters used for linear transformation learning from the FT module, and pass the normalized results directly to the refinement module. Then we add a convolutional layer to replace the refinement block in FT, denotes as InjectionGAN-conv. We report the classification accuracy of each variants in Fig 7. Unsurprisingly, the model without linear transformation has the worse performance the model with affine parameters. Interestingly, solely adding convolution even make the model pay more attention to the global texture structure, retaining more original information while reducing the accuracy of generated attributes, since the skip connection. Our InjectionGAN transit the feature maps that guided by target attribute vector to the decoder, which make the decoder pay more
attention to target image with correct attributes. By using two modules, the model gets the highest classification accuracy, which means performing the translation best.

VI. CONCLUSION

In this paper, we have presented a novel InjectionGAN for facial attribute transfer. In order to solve the problem that the use of discrete vectors by the generator results in poor diversity of migrated images, we introduce the FT module to establish the association between image attribute information and content information. The attribute latent variables and background latent variables are effectively separated, so every detail in the image can be easily modified and adjusted through these latent variables. In order to make the image passed by the attribute more accurate and real, our FT module is a conditional multi-scale module, which is beneficial to the generator to generate the image. In addition, we have added the refinement module to the FT module, which enables the model to generate detailed changes to imperceptible attributes. The experimental results on two datasets demonstrate the effectiveness of our method on both attribute transfer and image translation tasks.

Compared with the current method, the method in this paper clarifies the mapping relationship between the image domains and improves the visual effect and quality of the model migration image. However, there is still room for improvement in this paper. The method in this paper is to use attribute vectors as affine parameters to control image synthesis. How to generate target image attributes without the guidance of attribute tags and merge them with the original image content will be the next research work.

ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewers for providing constructive comments which enhanced the quality of their paper.

REFERENCES

[1] 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.

[2] 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.

[3] D. Yang, S. Hong, Y. Jang, T. Zhao, and H. Lee, “Diversity-sensitive conditional generative adversarial networks,” 2019, arXiv:1901.09024. [Online]. Available: http://arxiv.org/abs/1901.09024

[4] 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.

[5] 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.

[6] X. Liu, G. Yin, J. Shao, X. Wang, and H. Li, “Learning to predict layout-to-image conditional covariances for semantic image synthesis,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 568–578.

[7] L. A. Gatys, A. S. Ecker, and M. Bethge, “A neural algorithm of artistic style,” 2015, arXiv:1508.06576. [Online]. Available: https://arxiv.org/abs/1508.06576

[8] J. Lin, Y. Pang, Y. Xia, Z. Chen, and J. Luo, “TuGAN: Learning versatile image-to-image translation with two unpaired images,” 2020, arXiv:2004.04634. [Online]. Available: https://arxiv.org/abs/2004.04634

[9] Z. Zhang and Y. Peng, “Eyeglasses removal from facial image based on MVLVR,” in The Era of Interactive Media. New York, NY, USA: Springer, 2013.

[10] Z. Zhu, P. Luo, X. Wang, and X. Tang, “Recover canonical-view faces in the wild with deep neural networks,” 2014, arXiv:1404.3545. [Online]. Available: http://arxiv.org/abs/1404.3545

[11] P. Zhang, B. Zhang, D. Chen, L. Yuan, and F. Wen, “Cross-domain correspondence learning for exemplar-based image translation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2020, pp. 5143–5153.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2672–2680.

[13] W. Chen and J. Hays, “SketchyGAN: Towards diverse and realistic sketch to image synthesis,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 9416–9425.

[14] Y. Chen, Y.-K. Lai, and Y.-J. Liu, “CartoonGAN: Generative adversarial networks for photo cartoonization,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 2465–2474.

[15] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, “Photo-realistic single image super-resolution using a generative adversarial network,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jul. 2017, pp. 4681–4690.

[16] C. Li, C. Deng, N. Li, W. Liu, X. Gao, and D. Tao, “Self-supervised adversarial hashing networks for cross-modal retrieval,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 4242–4251.

[17] S. Zhou, T. Xiao, Y. Yang, D. Feng, Q. He, and W. He, “GeneGAN: Learning object transfiguration and object subspace from unpaired data,” in Proc. Brit. Mach. Vis. Conf., 2017, pp. 1–13.

[18] C. Wang, C. Wang, C. Xu, and D. Tao, “Tag disentangled generative adversarial networks for object image re-rendering,” in Proc. 26th Int. Joint Conf. Artif. Intell., Aug. 2017, pp. 2901–2907.

[19] X. Xiao, H. Hong, and J. Ma, “DNA-GAN: Learning disentangled representations from multi-attribute images,” 2018, arXiv:1711.05415v2. [Online]. Available: https://arxiv.org/abs/1711.05415v2

[20] G. Lampl, N. Zehidour, N. Usunier, A. Bordes, L. Denoyer, and M. A. Ranzato, “Fader networks: Manipulating images by sliding attributes,” in Proc. Adv. Neural Inf. Process. Syst. (NIPS), 2017, pp. 5967–5976.

[21] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, “Invertible conditional GANs for image editing,” in Proc. Comput. Vis. Pattern Recognit., 2016, pp. 1–9.

[22] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” 2014, arXiv:1411.1784. [Online]. Available: http://arxiv.org/abs/1411.1784

[23] 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.
[24] 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.

[25] N. Bodla, G. Hua, and R. Chellappa, “Semi-supervised FusedGAN for conditional image generation,” 2018, arXiv:1801.05551. [Online]. Available: http://arxiv.org/abs/1801.05551

[26] Y. Lu, Y. W. Tai, and C. K. Tang, “Attribute-guided face generation using conditional CycleGAN,” in Proc. Comput. Vis. Pattern Recognit., 2018, pp. 282–297.

[27] S. Ma, J. Fu, C. W. Chen, and T. Mei, “DA-GAN: Instance-level image translation by deep attention generative adversarial networks,” in Proc. CVPR, Jun. 2018, pp. 5657–5666.

[28] G. Zhang, M. Kan, S. Shan, and X. Chen, “Generative adversarial network with spatial attention for face attribute editing,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Sep. 2018, pp. 417–432.

[29] Q. Lian, L. Duan, F. Lv, and B. Gong, “Constructing self-motivated pyramid curriculums for cross-domain semantic segmentation: A non-adversarial approach,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 6758–6767.

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

[31] J. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, “Toward multimodal image-to-image translation,” in Proc. Neural Inf. Process. Syst., 2017, pp. 465–476.

[32] 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.

[33] A. Almahairi, S. Rajeshwar, A. Sordoni, P. Bachman, and A. Courville, “Augmented CycleGAN: Learning many-to-many mappings from unpaired data,” in Proc. Int. Conf. Mach. Learn., 2018, pp. 195–204.

[34] A. Pumarola, A. Agudo, A. M. Martinez, A. Sanfeliu, and F. Moreno-Noguer, “GANimation: Anatomically-aware facial animation from a single image,” in Proc. Eur. Conf. Comput. Vis. 2018, pp. 835–851.

[35] Y. Wang, J. van de Weijer, and L. Herranz, “Mix and match networks: Encoder-decoder alignment for zero-pair image translation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 5467–5476.

[36] B. Zhao, B. Chang, Z. Iee, and L. Sigal, “Modular generative adversarial networks,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 157–173.

[37] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241.

[38] E. Perez, F. Strub, H. D. Vries, V. Dumoulin, and A. Courville, “Film: Visual reasoning with a general conditioning layer,” in Proc. Nat. Conf. Artif. Intel., 2018, pp. 3942–3951.

[39] O. Nizan and A. Tal, “Breaking the cycle-colleagues are all you need,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2020, pp. 7860–7869.

[40] X. Huang and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Oct. 2017, pp. 1510–1519.

[41] I. Gulrajani, F. Ahmed, and M. Arjovsky, “Improved training of wasserstein GANs,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5767–5777.

[42] D. P. Kingma and B. Jimmy, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980. [Online]. Available: https://arxiv.org/abs/1412.6980

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