Multimedia Tools and Applications (2021) 80:4881–4902
https://doi.org/10.1007/s11042-020-09858-7

A novel attribute-based generation architecture for facial image editing

Defang Li[1,2] · Min Zhang[1,2] · Lifang Zhang[3] · Weifu Chen[1,2] · Guocan Feng[1,2]

Received: 21 September 2019 / Revised: 27 August 2020 / Accepted: 9 September 2020 /
Published online: 2 October 2020

Abstract

Facial image editing is one of the hot topics in recent years due to the great development in deep generative models. Current models are either based on variational autoencoder(VAE) or generative adversarial network(GAN). However, VAE-based models usually generate oversmooth images, while GAN-based-only models cannot randomly generate images with specific attributes and suffer from unstable training. To overcome these limitations, a novel attribute-disentangled generative model based on the combination of VAE and GAN is proposed for facial image editing by manipulating specific attributes and synthesizing facial images conditioned on the specified attributes. In the encoder-decoder architecture of the proposed model, the latent space mapped by the encoder is split into two subspaces: the attribute-irrelevant space and the attribute-relevant space. The attribute-irrelevant space characterizes the factors such as identity, position, background etc, which are expected to be kept unchanged during the editing. The attribute-relevant space is used to represent the attributes such as hair color, gender, age etc that we want to manipulate. We use the adversarial training scheme to train the model, where images generated by the proposed model are re-feeded to the encoder to ensure their distribution is close to the real data distribution in the attribute-irrelevant subspace while they can be correctly classified in the attribute-relevant subspace, without explicitly giving the discriminators such as in GANs. To evaluate the performance of the proposed model, quantitative and qualitative comparisons between the proposed model and other state-of-the-art algorithms were tesed on the CelebA dataset. The evaluation results show that the proposed model can effectively generate high-quality facial images with diverse specified attributes.

Keywords Adversarial Variational Autoencoders · Generative Adversarial Networks · Facial editing · Disentangled features

✉ Weifu Chen
chenwf26@mail.sysu.edu.cn

[1] School of Mathematics, Sun Yat-sen University, Guangzhou, China

[2] Guangdong Province Key Laboratory, Sun Yat-sen University, Guangzhou, China

[3] School of Computer Science and Technology, Dongguan University of Technology, Dongguan, China

1 Introduction

Facial image analysis, such as facial occlusions localization [52], 3D facial image analysis [37], forensic person identification [7] is an important topic in computer vision and pattern recognition. Recently, another facial analysis topic, image-to-image translation, has been developing very fast due to the great success of deep neural networks. The goal of image-to-image translation is to translate an image from one domain to another domain while maintaining some invariant or consistency property which is corresponding to specific tasks [20, 32, 46, 53, 60]. Facial image editing is one kind of image-to-image translation problems, in which we can manipulate the attributes of face images, i.e, with or without some kinds of attributes. Here, the term *attribute* represents some high-level feature of a facial image, e.g. expression, hair color, age and so on. We further denote the *attribute value* as a specific value of an attribute, e.g. neutral/smiling for expression or black/blond/brown for hair color or young/old for age, and *domain* as the images having the same attribute value. The key challenge of facial image editing is that the transformation is ill-posed and the training set is unpaired, that is, it is practically infeasible to collect images with arbitrarily specified attributes for each person. The problem has aroused a lot of interest [9, 17, 29, 39, 56, 57]. In particular, researchers tried to use deep generative models such as Bayesian inference [4, 14, 27, 41, 47], adversarial training [2, 13], variational autoencoders(VAEs) [27, 41] to solve this problem and make significant progress. Among those algorithms, variational autoencoders(VAEs) [27] and generative adversarial networks(GANs) [13] are the cornerstones.

To manipulate facial attributes given a facial image, many researches have been undertaken [1, 48]. Liu et al. [32] and Lu et al. [35] learn pair-wise generators and discriminators for every pair of image domains. Although these models handle well on the translation between two different domains, they are inefficient or ineffective in multiple-attribute editing. Some works learn to disentangle attribute representation in the latent space and use one block or one dimension of the latent vector to represent one attribute. Hence, feeding to the generator by swapping the corresponding components of two images from different domains is expected to generate image with swapped attributes [25, 55, 56, 59]. However, they are complained about the complex training pipelines and model structures. Larsen et al. [30], Radford et al. [40] and Upchurch et al. [51] compute the average direction vector of one attribute in the latent space over a pair of image sets with contrary attributes, which points from the average latent variable of the set with (without) that attribute to the average latent variable of the other set without (with) that attribute, then input image can be added or removed the specified attribute by adding its latent variable with the direction vector or subtracting the direction vector from its latent variable. Those models can add or remove attributes easily, but it is reported that the average direction vectors are not orthogonal and often contain highly correlated attributes which make the generation tend to transfer images with unwanted attributes. Bao et al. [3], Choi et al. [9], He et al. [17], Lample et al. [29], Perarnau et al. [39] and Yan et al. [57] extend CVAE [45] or CGAN [38] and inject different attribute labels to conduct the image generation.

Although these models success in generating new images, they still have one or several of the following limitations:

- generated images are in low resolution or with lots of artifacts [39, 57];
- label-paired images in different domains are needed to train the model [20, 53];
- multiple attribute editing is infeasible [24, 32, 60];

- the models combining VAEs and GANs, have more than three deep mappings, which increases the training complexity [9, 17];
- some models cannot generate facial images randomly with specified attributes [9, 17, 29, 39, 57].

Due to its nice manifold representation and stable training mechanism, VAE is theoretically well-founded and more stable. However, VAE trends to generate blurry images due to the limited representation ability of the inference distribution, the inherent over-regularization induced by the Kullback-Leibler divergence term and imperfect reconstruction error [10, 49]. GAN [13] introduces a generator and a discriminator for adversarial learning. When the network reaches the equilibrium, the fake (generated) images have the same distribution as the real images, which makes the generated images look more realistic. The emergence of GAN has attracted so many concerns that it makes other generative models obsolete and becomes one of the dominant approaches for generating images with surprising complexity and realism. One of the drawback of GAN is instability in training optimization and easily leads to the model-collapse problem [42].

The motivation of our model is quite straightforward, that is, we want to propose a novel model that inherits the advantages of VAE and GAN while abandons the disadvantages. In addition, we want the novel model can do multiple attribute facial image editing rather than we can only manipulate one attribute at a time. In order to achieve these goals, we develop a novel framework that incorporates the ideas of VAE and GAN. We first divide the latent space learned by the encoder into two independent subspaces, the attribute-irrelevant subspace and the attribute-relevant subspace. The attribute-irrelevant subspace is used to represent factors such as the identity, pose, illumination etc., while the attribute-relevant subspace represents the attributes such as the hair color, the hair style, gender, age and so forth that we can edit. Thus, each facial image can be represented in the latent space by an identification vector together with an attribute vector where each component corresponds to one attribute. During the editing, we want the identification vector keep unchanged, and only need to manipulate the attribute vector to generate the specified attribute images. We further show that without increasing the complexity of model or introducing additional deep mappings, the adversarial training can be potentially implemented via the encoder and the decoder. By viewing the KL divergence as a special form of reression and introducing classification loss on the attribute-relevant variables, the encoder can be treated not only as a discriminator for real and generated samples but also as a classifier for attributes. Subsequently, adversarial training is introduced into the latent space to align the generated data distribution to the real data distribution while the attributes of the generated images can be classified correctly. We have compared the proposed model with state-of-the-art algorithms for single-attribute and multiple-attribute facial editing. The quantitative and quantitative results show that proposed model can produce impressive and high-quality images. To summarize, the contributions of this paper include:

1. Based on the encoder-decoder architecture, the latent space of the proposed network is split into two independent subspaces, the attribute-irrelevant subspace and the attribute-relevant subspace;
2. Based on the combination of VAE and GAN, an attribute-disentangled generative model is proposed, which involves only two deep mappings: the encoder and the decoder;
3. Extensive experiments, including single-attribute and multiple-attribute facial image editing, were designed to evaluate the performance of the proposed model.

The remaining sections of the paper are organized as follows. Section 2 reviews the VAE and GAN as well as other related works. The proposed model is introduced in Section 3. Experimental results including quantitative and qualitative evaluation are presented in Section 4, followed by the conclusion in Section 5.

2 Related works

2.1 Variational autoencoders

Variational autoencoders(VAEs) [27, 41] were proposed to estimate flexible deep generative models by variational inference methods. A standard VAE consists of an encoder Enc and a decoder Dec. The encoder (also regarded as recognition model) maps an input sample x to a distribution over latent variable $z \sim Enc(x) = q_\phi(z|x)$. The decoder (also regarded as a generative model) maps from this latent space to a distribution over images $\tilde{x} \sim Dec(z) = p_\theta(x|z)$. VAE regularizes the encoder by imposing a prior over the latent distribution $p_\theta(z)$, which is typically chosen as the standard Gaussian distribution $N(0, I)$. The objective function of VAE is to maximize the evidence lower bound(ELBO) of log-likelihood $log\ p_\theta(x)$:

$$\begin{aligned}\mathcal{L}_{VAE} &= \mathbb{E}_{z\sim q_\phi(z|x)}[log\ p_\theta(x|z)] - D_{KL}(q_\phi(z|x) \parallel p_\theta(z)) \\ &\leqslant log\ p_\theta(x)\end{aligned} \tag{1}$$

where D_{KL} is the Kullback-Leibler divergence. The first term in Eq. 1 is a reconstruction error. If we assume that the decoder predicts a Gaussian distribution at each pixel, then it reduces to squared Euclidean error in the image space. The second term drives the recognition distribution towards the prior distribution which acts as a regularizer. Both $q_\phi(z|x)$ and $p_\theta(z)$ are commonly assumed to be Gaussian, in which case the KL divergence can be computed analytically. Assume that mean $\mu(x)$ and covariance $\sigma(x)$ of $q_\phi(z|x)$ are outputs of $Enc(x)$ for a given input x, then the KL divergence can be derived as follows [3]:

$$\mathcal{L}_{KL} = \tfrac{1}{2}(\mu(x)^T\mu(x) + sum(exp(\sigma(x)) - \sigma(x) - 1)) \tag{2}$$

In addition, a reparameterization of the recognition distribution in terms of auxiliary variables with fixed distributions is used so that the samples from the recognition model are a deterministic function of the inputs and auxiliary variables. That is, a latent sample z is drawn from $q_\phi(z|x)$ as follows: a random noise vector (also an auxiliary variable) $\epsilon \sim N(0, I)$, for example, then $z = g_\phi(\epsilon, x) = \mu(x) + \sigma(x) \odot \epsilon$, where $\mu(x)$ and $\sigma(x)$ are outputs of $Enc(x)$ for a given input x and $\odot$ signifies an element-wise product.

One of the major disadvantages of VAE is that, because of the injected noise and imperfect element-wise measures such as the squared error, the generated samples are often blurry.

2.2 Generative adversarial networks

Generative Adversarial Networks (GANs) [13] consist of a generator G and a discriminator D that compete in a two-player minimax game. The goal of GANs is to let the G learn a distribution $p_g(x)$ that matches the real data distribution $p_{data}(x)$ via an adversarial process. D tries to distinguish a real image x from a synthetic one $G(z)$, where z is a input noise variable sampled from a prior distribution $p_z(z)$, and G tries to synthesize realistic-looking images that can fool D. Concretely, D and G play the game with a value function $V(D, G)$:

$$\begin{aligned} \underset{G}{min}\,\underset{D}{max} V(D, G) &= \mathbb{E}_{x\sim p_{data}(x)}[log D(x)] \\ &+\mathbb{E}_{z\sim p_z(z)}[log(1 - D(G(z)))]. \end{aligned} \tag{3}$$

It is proved that this minimax game has a global optimum when the distribution p_g of the synthetic samples and the distribution p_{data} of the training samples are the same. Under mild conditions (e.g., G and D have enough capacity), p_g converges to p_{data}.

When trained on image dataset, GANs can produce visually sharp and compelling sample images. However, it is also complained about instabilities in optimization that leads to the problem of mode-collapse [42], which means that samples generated from GANs don't reflect the diversity of the underlying data distribution.

Many works have tried to improve the stability of training and the quality of generated images from different perspectives. DCGAN [40] which adopts deconvolutional and convolutional neural networks to implement G and D, respectively, is the first GAN model to learn to generate high resolution images in a single shot. Many GANs are at least loosely based on the DCGAN architecture. WGAN [2] and WGAN-GP [15] which use the Wasserstein distance instead of the Jensen-Shannon distance to form a new objective for training GANs, have provided a powerful theoretical proof and illustrated that they can make the GAN training process more stable.

2.3 Variants of combination of VAEs and GANs

Due to both VAEs and GANs having their own advantages and disadvantages, several recent works have looked for hybrid approaches to enable both sampling and inference like VAEs or autoencoders(AEs), while producing samples of quality comparable to GANs. Typically this is achieved by training an autoencoder(AE) jointly with one or more adversarial discriminators whose purpose is to improve the alignment of distributions in the latent space(AAE [36], IAN [6], AGE [50]), the data space(MRGAN [8], VAE/GAN [30]) or in the joint (product) latent-data space(BiGAN [11], ALI [12]). These algorithms have been demonstrated their power in generating excellent quantitative and visual results. However, while compounding autoencoding and adversarial training do improve VAEs and GANs, it is at the cost of adding complexity. In particular, these systems usually involve at least three [11, 12, 29, 30, 36] or four [17] deep mappings: an encoder for encoding representation, a decoder/generator for generating samples, a discriminator for discriminating real or generated samples and a classifier for classifying the attributes of samples.

By introducing additional conditionality, VAEs and GANs can also be trained to conduct conditional generation, e.g., CVAE [45] and CGAN [38]. CGAN [38] modified GAN from unsupervised learning into semi-supervised learning by feeding the conditional variable (e.g., the class label) into the data. CVAE-GAN [3] combines CVAE and CGAN for fine-grained category image generation.

AGE [50], which directly sets up an adversarial training between the encoder and the decoder of an AE and constrains the real and the generated data distribution to be the prior distribution in the latent space. IntroVAE [19] uses VAE to replace AE in AGE, and preserves the advantages of VAEs, such as stable training and nice latent manifold. Our work is partially inspired by AGE and IntroVAE, but the difference is that our model extends the latent structure of VAE and jointly trains with CGAN combining perceptual loss, which makes our model can not only achieve better reconstruction effect but can also synthesize photo-realistic images with specified attributes.

2.4 Image to image translation

The model pix2pix [20] was trained by combining cGAN with a L1 loss in a supervised manner, which means requiring paired training data. However, obtaining such data is usually expensive. DTN [46] presents a baseline formulation for unsupervised cross-domain image translation and trains an image-conditional generator including a pre-trained function as an encoder and enforces the translated image is close to the original image in the latent space. CoGAN [33] learns a joint distribution of images in two different domains by enforcing a weight-sharing constraint to the layers of a pair of GANs, each of which is responsible for synthesizing images in one domain. UNIT [32] extends CoGAN framework with VAEs and assumes that two different domains can be mapped to a shared-latent space. CycleGAN [60] and DiscoGAN [24] train two mapping functions that are inverse to each other between two image domains by employing the cycle consistency loss and two domain-specific discriminators to distinguish between the domains by employing the adversarial loss.

2.5 Facial attribute editing

The work disCVAE [57] learns the disentangled latent variable which is split into a foreground part and a background part by training with CVAE [45] to improve the generation quality and diversity. AD-VAE [16] trains VAEs by splitting the latent variables into different groups to learn a representation disentangled model. IcGAN [39] separately trains a cGAN [38] and an encoder which is the inverse of the mapping of the cGAN. DIAT [31] is presented as a deep identity-aware attribute transfer model to modify an attribute of a face image via adversarial learning. Shen and Liu [43] adopt the dual residual learning strategy to simultaneously train two generators for respectively adding and removing a specific attribute. To tackle the task of attribute transfer from an exemplar image with targeted attribute, Kim et al. [25], GeneGAN [59], DNA-GAN [55] and ELEGANT [56] encode a source image and an exemplar image to their respective latent variables and swap attribute-relevant latent code as representations of the “crossbreed” (residual) images to achieve (multiple) attribute transfer.

Recently, several works have been proposed for multiple facial attribute editing simultaneously with capability of high-quality image generation using one model by only training one time with images from different domains. Fader Networks [29] employs the adversarial learning on the latent representation of an autoencoder to learn attribute invariant representation. StarGAN [9] performs image-to-image translations for multiple domains using one single GAN model with a cycle consistency loss. AttGAN [17] uses an encoder-decoder architecture together with an attribute classifier and a discriminator and applies an attribute classification constraint to guarantee the generated images can be correctly changed with desired attributes. All these three models can handle multiple face attributes transfer and generate sharp images, but they are not able to generate facial images given by randomly specified attributes.

3 Latent space adversarial variational autoencoder

We denote the training dataset $\mathcal{D} = \left\{(x^i, y^i)\right\}$, which consists of m pairs (image, attribute), where x^i is the i-th image and $y^i = \{0, 1\}^n$ is the corresponding attribute vector of x^i with n dimensions. Each component y^i_k in y^i(we use the subscript k to refer to the k-th attribute) represents the k-th attribute value, which indicates whether x^i has certain attribute or not.

Our model is based on the encoder-decoder architecture. Concretely, the latent space mapped by the encoder is split into the attribute-irrelevant subspace $\mathcal{Z}$ (assume the prior distribution on this space is $p(z) = \mathcal{N}(0, I)$) and the attribute-relevant subspace $\mathcal{A}$ (denote the prior distribution on this space is $p(a)$). The former represents attribute-irrelevant factors, such as identity, position and background, etc. The latter represents attributes, such as hair color, gender, with or without glasses, etc. The decoder maps these subspaces together back to the data space. Denote $q_\phi(z|x)$ and $q_\phi(a|x)$ as the approximation posterior distributions and assume that $p(z)$ and $p(a)$ are independent, then the ELBO can be rewritten as:

$$\begin{aligned} log\ p_\theta(x) &\geqslant E_{q_\phi(a|x)q_\phi(z|x)}\left[log\ \frac{p_\theta(x,z,a)}{q_\phi(z|x)q_\phi(a|x)}\right] \\ &= E_{q_\phi(a|x)q_\phi(z|x)}\left[log\ \frac{p_\theta(x|a,z)p(a)p(z)}{q_\phi(z|x)q_\phi(a|x)}\right] \\ &= -D_{KL}(q_\phi(a|x) \parallel p(a)) - D_{KL}(q_\phi(z|x) \parallel p(z)) \\ &+E_{q_\phi(a|x),q_\phi(z|x)} log\ p_\theta(x|a,z) \\ &= \mathcal{L}_{ELBO}, \end{aligned} \tag{4}$$

where $q_\phi(z|x)$, $q_\phi(a|x)$ and $p_\theta(x|a,z)$ are assumed to be multivariate Gaussian distributions. Further, we choose a conditional distribution $p(a|y)$ as the prior of a instead, and $p(a_i|y_i) \sim N(y_i, \sigma)$, where y_i resfers to i-th binary attribute and σ is the standard deviation of $p(a|y)$. Thus the training objective can be rewritten as

$$\begin{aligned} \min_{\phi,\theta} -\mathcal{L}_{ELBO} &= \sum_i^n D_{KL}(q_\phi(a_i|x) \parallel p(a_i|y_i)) \\ &+\alpha D_{KL}(q_\phi(z|x) \parallel p(z)) \\ &-\beta E_{q_\phi(a|x),q_\phi(z|x)} log\ p_\theta(x|a,z), \end{aligned} \tag{5}$$

where α, β are hyper-parameters to control the relative importance of different terms. The training pipeline of our model consists of two training phases: reconstruction training phase and adversarial training phase.

In reconstruction training phase , since we specify the mean of prior $p(a|y)$ to be the binary attribute label y of the input image, the first term in Eq. 5 is discriminative. Hence $q_\phi(a|x)$ also can serve as a classifier for facial attributes, here we adopt binary cross entropy loss:

$$\mathcal{L}_{attr\_real} = -\sum_{i=1}^n y_i\, log\, a_i + (1-y_i)\, log\, (1-a_i), \tag{6}$$

where a_i indicates the predicted value for the i-th attribute. The third term in Eq. 5 is the reconstruction term. In order to overcome shortcomings of pixel-wise ℓ_2 loss, we use perceptual loss to measure the similarity between the input image and its reconstruction. Perceptual loss is widely used to measure the content difference between different images.

Denote $\Phi(x)^l$ is the l^{th} hidden layer with C^l channels and size of $W^l \times H^l$ when x is fed to the VGG-19 [44] pre-trained model Φ. Then, the feature perceptual loss for this layer between the input image x and its reconstruction image $\bar{x}$ is defined as

$$\mathcal{L}_{rec}^{\Phi,l} = \frac{1}{2C^l W^l H^l} \left\| \Phi(x^i)^l - \Phi(\bar{x}^i)^l \right\|_2^2. \tag{7}$$

Then the total feature perceptual loss is defined as a weighted feature perceptual loss based on some layers of Φ

$$\mathcal{L}_{rec} = \sum_{l} \omega_l \mathcal{L}_{rec}^{\Phi,l}, \tag{8}$$

where ω_l is the weight for the l^{th} hidden layer. We can rewrite loss function of the reconstruction training phase as

$$\mathcal{L}_{ir} = \mathcal{L}_{attr\_real} + \alpha\mathcal{L}_{kl\_real} + \beta\mathcal{L}_{rec}, \tag{9}$$

where $\mathcal{L}_{kl\_real}$ is the second term in Eq. 5 and α, β are hyper-parameters for balancing the losses.

In Adversarial training phase, as KL divergence statistics $D_{KL}(q_\phi(z \mid x) \parallel p(z))$ which computes a single number can serve as a special form of regression, the encoder Enc can be regarded as a discriminator and a classifier.In addition, as a generator, the decoder Dec can generate two types of different fake images:

(1) fake image $\tilde{x}$, which is generated by feeding Dec with random variables $\tilde{z}$ and $\tilde{y}$ drawn from $p(z)$ and $p(y)$,
(2) fake image $\hat{x}$, which is generated by feeding Dec with the attribute-irrelevant latent variable $\hat{z}$ of the input image x and another attribute latent variables $\hat{y}$ drawn from $p(y)$, that is different from the real attribute.

Hence, we end up with the following two objectives for adversarial training alternately:

– **For training the encoder** which is also a discriminator:

$$\begin{aligned} \min_{\phi} \mathcal{L}_{enc} = {} & \gamma_1\mathcal{L}_{kl\_real} + \gamma_2\mathcal{L}_{attr\_real} \\ & + max(0, m - \gamma_3\mathcal{L}_{kl\_fake}), \end{aligned} \tag{10}$$

where m is a positive margin and

$$\mathcal{L}_{kl\_fake} = \mathcal{L}_{kl\_\tilde{x}} + \mathcal{L}_{kl\_\hat{x}}, \tag{11}$$

$$\mathcal{L}_{kl\_\tilde{x}} = D_{KL}(q_\phi(z|\tilde{x})||p(z)), \tag{12}$$

$$\mathcal{L}_{kl\_\hat{x}} = D_{KL}(q_\phi(z|\hat{x})||p(z)), \tag{13}$$

where $\tilde{x}$ is the random generated sample from $p_\theta(x|\tilde{y}, \tilde{z})$, $\tilde{y}$ and $\tilde{z}$ are samples from $p(y)$ and $p(z)$, respectively, and $\hat{x}$ is the generated sample from $p_\theta(x|\hat{y}, \hat{z})$, $\hat{y}$ is randomly sampled from $p(y)$, $\hat{z}$ is the attribute-irrelevant latent variable of the input image.

– **For training the decoder** which is also a generator:

$$\min_{\theta} \mathcal{L}_{dec} = \gamma_4\mathcal{L}_{kl\_fake} + \gamma_5\mathcal{L}_{attr\_fake}, \tag{14}$$

where $\mathcal{L}_{kl\_fake}$ is the same as in Eq. 11 and

$$\mathcal{L}_{attr\_fake} = \mathcal{L}_{attr\_\tilde{x}} + \mathcal{L}_{attr\_\hat{x}}, \tag{15}$$

$$\mathcal{L}_{attr\_\tilde{x}} = -\sum_{i=1}^{L} \tilde{y}_i \, log \, \tilde{a}_i + (1 - \tilde{y}_i) \, log \, (1 - \tilde{a}_i), \tag{16}$$

$$\mathcal{L}_{attr\_\hat{x}} = -\sum_{i=1}^{L} \hat{y}_i \, log \, \hat{a}_i + (1 - \hat{y}_i) \, log \, (1 - \hat{a}_i), \tag{17}$$

where $\tilde{a}$ and $\hat{a}$ are the latent attribute variables of $\tilde{x}$ and $\hat{x}$, respectively.

The overall training loss function can be summarized as

$$\mathcal{L}_{total} = \mathcal{L}_{ir} + \mathcal{L}_{enc} + \mathcal{L}_{dec}, \tag{18}$$

where the exact forms of each term are presented in Eqs. 9, 10 and 14, respectively. Figure 1 summarizes the reconstruction training and the adversarial training phases of our model. The training procedure is presented in Algorithm 1. We name the proposed model Latent Space Adversarial Variational Autoencoder (LSA-VAE).

Algorithm 1 The training procedure of the proposed model.

```
Require:
$\phi \leftarrow$ initial $Enc$ network parameters.
$\theta \leftarrow$ initial $Dec$ network parameters.
N is the numbers of training epoches.
The prior, $p(z) = N(0, I)$.

for $i$ in range $(N)$ do
    // The reconstruction training phase
    Sample $\{x, y\} \sim p_{data}$ a batch from the training data
    $z \leftarrow$ Reparameterized$(Enc_{\phi,z}(x))$
    $a \leftarrow Enc_{\phi,a}(x)$
    $\mathcal{L}_{kl\_real} \leftarrow D_{KL}(q_{\phi}(z|x) \parallel p(z))$
    $\mathcal{L}_{attr\_real} \leftarrow$ Eq.6
    $\bar{x} \leftarrow Dec(z, a)$
    $\mathcal{L}_{rec} \leftarrow$ Eq.8
    $\theta \leftarrow \theta - \nabla_{\theta}(\beta\mathcal{L}_{rec})$
    $\phi \leftarrow \phi - \nabla_{\phi}(\mathcal{L}_{attr\_real} + \alpha\mathcal{L}_{KL\_real} + \beta\mathcal{L}_{rec})$
    // The adversarial training phase
    Sample $\{x, y\} \sim p_{data}$ a batch from the training data
    $\hat{z} \leftarrow$ Reparameterized$(Enc_{\phi,z}(x))$
    $\hat{a} \leftarrow Enc_{\phi,a}(x)$
    $\mathcal{L}_{kl\_real} \leftarrow D_{KL}(q_{\phi}(\hat{z}|x) \parallel p(z))$
    $\mathcal{L}_{attr\_real} \leftarrow$ Eq.6
    Sample $\hat{y}$ a batch of random attributes from training data
    $\hat{x} \leftarrow Dec(\hat{z}, \hat{y})$
    $\hat{a} \leftarrow Enc_{\phi}(\hat{x})$
    Sample $\tilde{z} \sim p(z)$ a batch of random noise
    Sample $\tilde{y}$ a batch of random attributes from training data
    $\tilde{x} \leftarrow Dec(\tilde{z}, \tilde{y})$
    $\tilde{a} \leftarrow Enc_{\phi}(\tilde{x})$
    Encoder loss $\underset{\phi}{min}\, \mathcal{L}_{enc} \leftarrow$ Eq.10
    $\phi \leftarrow \phi - \nabla_{\phi}\mathcal{L}_{enc}$
    Decoder loss $\underset{\theta}{min}\, \mathcal{L}_{dec} \leftarrow$ Eq.14
    $\theta \leftarrow \theta - \nabla_{\theta}\mathcal{L}_{dec}$
end for
```

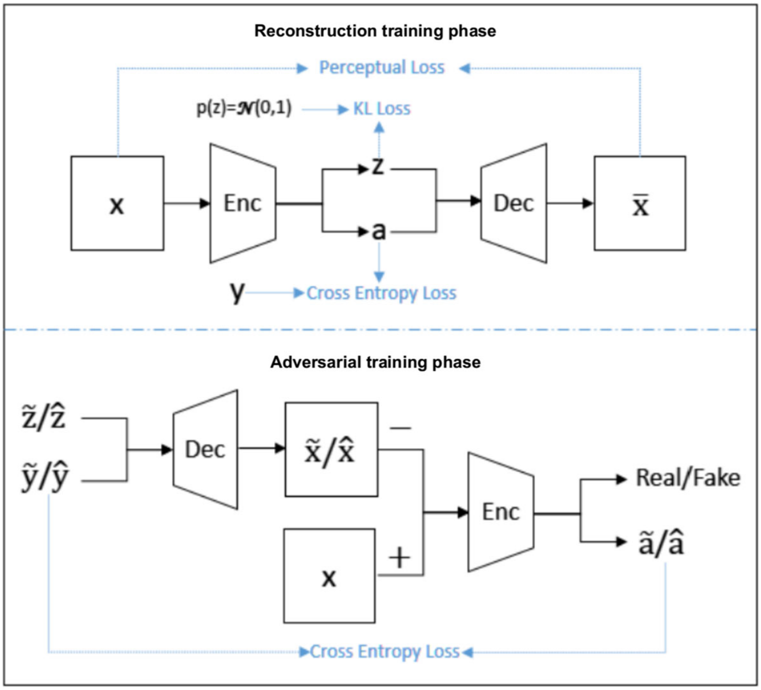

Fig. 1 Overview of our model. The meaning of each notation refers to the corresponding text part

4 Experiments

4.1 Dataset

We evaluated the proposed model on the CelebA dataset [34], which contains 202599 celebrity images and each of them is annotated with or without 40 binary attributes. We select 13 attributes, including “Bald”, “Bangs”, “Black Hair”, “Blond Hair”, “Brown Hair”, “Bushy Eyebrows”, “Eyeglasses”, “Male”, “Mouth Slightly Open”, “Mustache”, “No Beard”, “Pale Skin” and “Young”, due to that they are more distinctive in appearance. Officially, CelebA is separated into training set, validation set and testing set. We used the training set and validation set together to train our model while using the testing set for evaluation. In our experiment, all the images were cropped in the central 170×170 region and scaled down to 128×128 pixels and normalized to $[-1, 1]$.

4.2 Network architecture

The details of the network architectures of LSA-VAE are shown in Tables 1, 2 and 3. Both the encoder and the decoder networks are mainly based on residual blocks whose configuration is shown in Table 3. Like the other VAEs, mean $\mu(x)$ and covariance $\sigma(x)$ in Eq. 2 are output by the encoder of our model to compute the KL divergence loss and used for compute the attribute-irrelevant latent variable z. In addition, attribute-relevant latent variable a is output by another branch of the encoder. For the decoder, instead of standard zero-padding, we used replication padding, i.e., feature map of an input was padded with the replication of

Table 1 The architecture of the encoder in LSA-VAE

Encoder Layer	Output Size
Conv(64,5,1,2), BN, Leaky ReLU, AvgPool(2)	64 × 64 × 64
ResBlock(64,128), AvgPool(2)	128 × 32 × 32
ResBlock(128,256), AvgPool(2)	256 × 16 × 16
ResBlock(256,512), AvgPool(2)	512 × 8 × 8
ResBlock(512,512), AvgPool(2)	512 × 4 × 4
ResBlock(512,512)	512 × 4 × 4
Output Branch 1: FC(8192,512)	512
Output Branch 2: FC(8192,512)	512
Output Branch 3: FC(8192,1024) Leaky ReLU, FC(1024, 13)	13

the input boundary. We also used the nearest neighbor method by a scale of 2 to replace with fractional-strode convolutions for upsampling. The meanings of the notations in Tables 1, 2 and 3 are as follows: Conv(d,k,s,p) denotes the convolutional layer with d as the dimension, k as the kernel size, s as the stride and p as the padding, BN denotes the batch normalization, FC denotes a fully-connected layer, ResBlock(I,O) denotes a residual block with I and O as the numbers of input feature maps and output feature maps respectively, and AvgPool(w) denotes the average pooling with w as size of the window.

4.3 Training details

As illustrated in Algorithm 1, the reconstruction training phase and the adversarial training phase were trained iteratively using the ADAM optimizer [26] ($\beta_1 = 0.5$, $\beta_2 = 0.999$) with a batch size of 32 and a fixed learning rate of 0.0002. We set the dimensions of the attribute-irrelevant subspace and the attribute-relevant subspace to 512 and 13, respectively. The marginal m in Eq. 10 was set to 1500. We used a combination of relu1_1, relu2_1 and relu3_1 layer of VGG-19 pre-trained model to compute feature perceptual loss. Each ω_l was set to 0.5 in Eq. 8. α and β in Eq. 9 were set to 1 and 0.5, respectively. γ_1, γ_2, γ_3, γ_4 and γ_5 were set to 0.5, 0.5, 1, 1, 100, respectively. It is suggested to train the model with 10 epochs in the reconstruction training phase as initial weights before performing the adversarial training phase iteratively.

Our experiments were conducted on a computer with a GTX TitanX GPU of 12GB memory.

Table 2 The architecture of the decoder in LSA-VAE

Decoder Layer	Output Size
FC(525, 8192), ReLU	512 × 4 × 4
ResBlock(512,512)	512 × 8 × 8
ResBlock(512,512), Upsample	512 × 16 × 16
ResBlock(512,256), Upsample	256 × 32 × 32
ResBlock(256,128), Upsample	128 × 64× 64
ResBlock(128,64), Upsample	64 × 128 × 128
ResBlock(64,64)	64 × 128 × 128
Conv(64,5,1,2), BN, Leaky ReLU	3 × 128 × 128

Table 3 The structure of the ReBlock in LSA-VAE

Module	Structure
ResBlock(I,O)	Residual Block:Conv(I,3,1,1), BN, Leaky ReLU Conv(O,3,1,1), BN, Leaky ReLU

4.4 Baseline models

As our baseline models, we compared LSA-VAE with state-of-the-art algorithms including IcGAN [39], StarGAN [9] and AttGAN [17], which were reported that achieved the best performance for facial editing and were capable to manipulate images conditioned on multiple attributes with a single generator. The results were evaluated on quantitative comparison and qualitative comparison on single-attribute editing and multiple-attribute editing. For fair comparison, all the baselines were retrained on CelebA dataset by the authors' released codes using thirteen attributes mentioned above. We briefly introduce these models in following:

IcGAN combines an encoder with a cGAN model, where cGAN learns the mapping $G : \{z, c\} \rightarrow x$ that generates an image x conditioned on both the random noise z and the conditional representation c. Training on the random samples of z and c and their corresponding synthesized image x generated by the cGAN, an encoder learns the inverse mappings $E_z : x \rightarrow z$ and $E_c : x \rightarrow c$. This allows to synthesize images conditioned on arbitrary conditional representation.

StarGAN trains a single generator G that learns mappings among multiple domains and introduces an auxiliary classifier that allows a single discriminator to control multiple domains. That is, G translates an input image x into an output image y conditioned on the target domain label c, $G : \{x, c\} \rightarrow y$. The discriminator produces probability distributions over both sources and domain labels, $D : x \rightarrow \{D_{src}(x), D_{cls}(x)\}$, which means D can not only discriminate real or fake images, but also can classify domain labels. In addition, in order to preserve the content of its input images while changing only the domain-related part of the inputs, StarGAN applies a cycle consistency loss to the generator,$||x - G(G(x, c), c')||_1$, where c and c' are target label and original label, respectively.

AttGAN employs an encoder-decoder architecture and models the relation between the latent representation and the attributes, which are different from StarGAN. On the one hand, the encoder encodes input x_a into the latent representation z and the decoder receives z and original attribute a as input to be trained to reconstruct xa. On the other hand, the decoder receives z and target attribute b as inputs to be trained to generate target image $\hat{x}_b$, which is expected to be with target attributes while identity-preserving. AttGAN also applies the attribute-classification constraint on the generated image to guarantee the correct change of the attributes.

4.5 Qualitative analysis

4.5.1 Single facial attribute editing

In this section, we compared the proposed model with the baselines in terms of single facial attribute editing. We presented qualitative results in Figs. 2 and 3, where eight attributes were chosen to show the ability of these models on the task of editing single

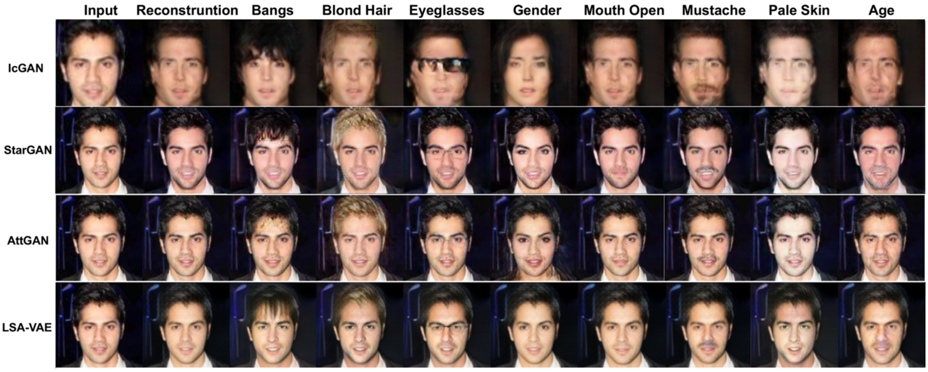

Fig. 2 Results of single facial attributes editing. The first and second columns from the left are ground truth and its reconstruction images, respectively. For each ground truth image, every row demonstrates results of single facial attributes editing by different models. The rows from top to bottom are generated by IcGAN, StarGAN, AttGAN and our model, respectively

attribute. As shown in the figures, IcGAN produced blurry images, and the reconstruction ability of IcGAN is very limited. StarGAN could generate sharper images than IcGAN, but the results of StarGAN contained some artifacts. In general, StarGAN, AttGAN and our model could edit attributes correctly. Both AttGAN and our model could reconstruct the original image much better than other algorithms and generate more realistic results. For "Pale Skin" attribute, StarGAN tended to generate less natural skin color. For "Bangs" attribute, the results of AttGAN seems to be less distinct. For global attribute like "Gender", both StarGAN and AttGAN tended to change male to female by putting makeup and wearing lipstick, while our model attempted to change eye shapes and facial lines.

In summary, the proposed model could manipulate the attribute correctly and generate images with high visual quality, and performed well on all the attribute testings, which is mainly due to the superior design of the architecture of the proposed network.

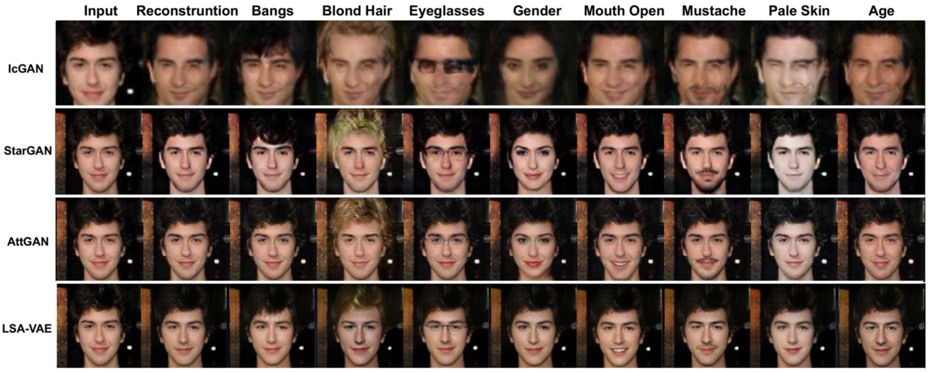

Fig. 3 Results of single facial attributes editing. The first and second columns from the left are ground truth and its reconstruction images, respectively. For each ground truth image, every row demonstrate results of single facial attributes editing by different models. The rows from top to bottom were generated by IcGAN, StarGAN, AttGAN and our model, respectively

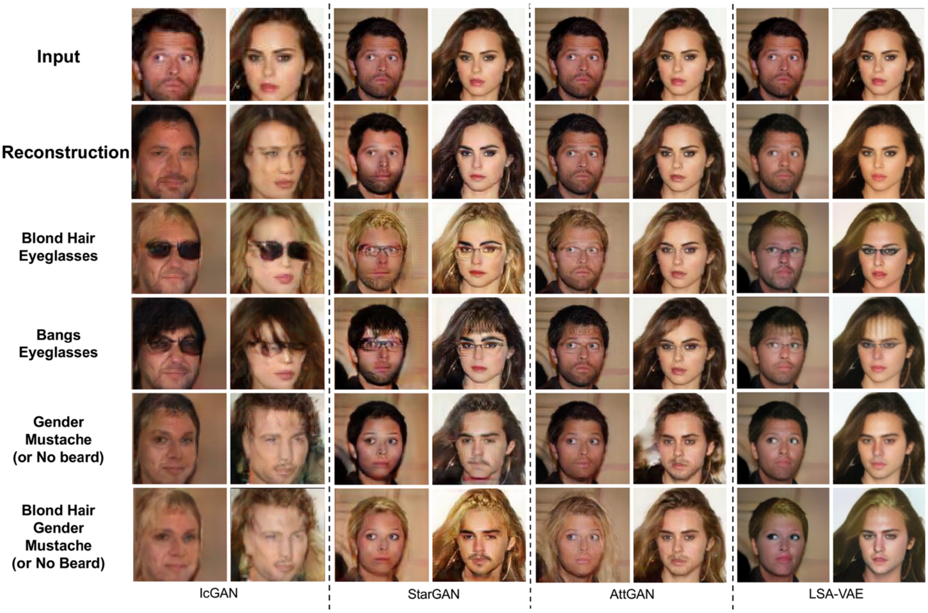

Fig. 4 Comparison results of IcGAN, StarGAN, AttGAN and LSA-VAE, with multiple-attribute facial editing. The first row from top is ground truth. Zoom in for better resolution

4.5.2 Multiple facial attribute editing

For more comprehensive comparison, we evaluated these four models in term of multiple-attribute facial editing. The comparison results are shown in Fig. 4. Similar to single-attribute editing, images generated by IcGAN were in distortion of facial details and seemed

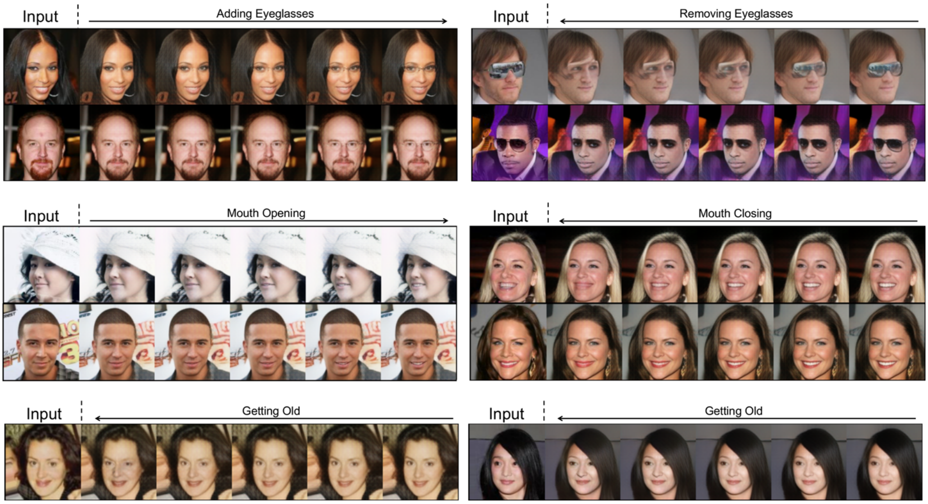

Fig. 5 Examples of attribute-conditioned image progression on adding or removing eyeglasses, opening or closing mouth and changing age, respectively. Zoom in for better resolution

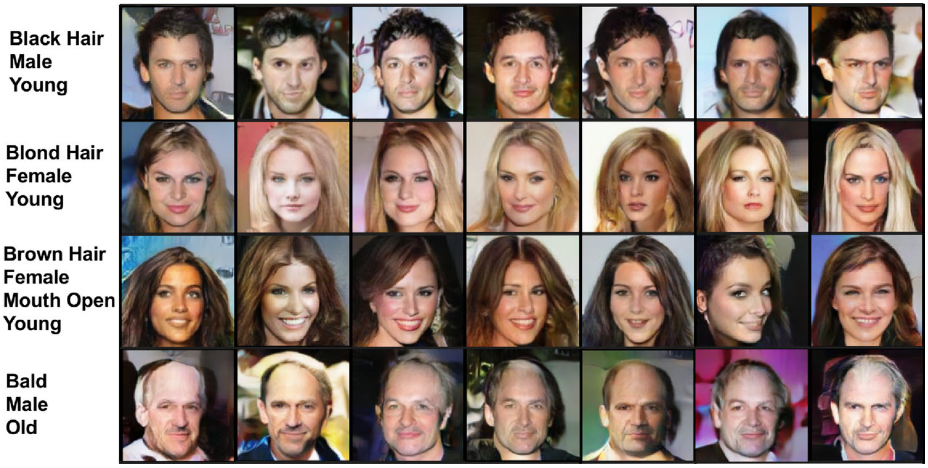

Fig. 6 Examples of attribute-conditional random image generation with our model. Images in each row were generated by feeding into the decoder with variables of noise and fixed attributes listed in the left

more blurry. For StarGAN, it can edit attributes accurately. However, some of its results look unnatural and it didn't perform as well as AttGAN and our model in terms of identity preserving. As for AttGAN, when we tried to manipulate the gender and the hair color simultaneously (see row 5, column 6 and row 6, column 5 in Fig. 4), it also tended to change the hair style such as making it short hair when changing female to male with mustache, or making it long hair when changing male to female with blond hair. By contrast, our model can generate more natural and realistic images and handle the task of multiple-attribute facial editing much better.

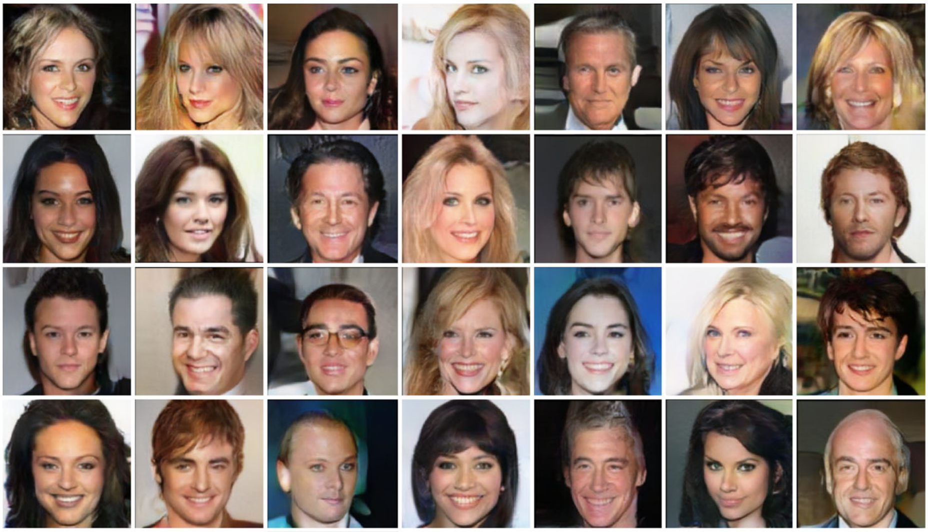

Fig. 7 More examples of attribute-conditional random image generation. All of them were synthesized by random attributes. Zoom in for better resolution

Table 4 Quantitative comparison results of the baseline models and the proposed model, evaluated by PSNR, SSIM, LPIPS and FID. The number of the parameters of each models is also listed in the last column

Model	PSNR	SSIM	LPIPS	FID	# of params
IcGAN	15.19	0.44	0.17	42.3	67.9M
StarGAN	22.47	0.78	0.092	30.13	53.6M
AttGAN	31.67	0.93	0.024	16.41	88.1M
LSA-VAE	26.28	0.83	0.055	30.02	47.9M

4.5.3 Attribute-conditioned image progression

Although our model was trained with discrete binary attribute values (0 or 1), we found that it is compatible with continuous attribute value in the testing phase and can generate a progression process of attribute intensity. In order to demonstrate this attributed-condition progression, we manipulated the value of one dimension in the attribute variable by modifying it from 0 to 1 smoothly and keeping all other latent variables fixed. As we can see in Fig. 5, samples generated by progression are visually consistent with attribute description. By changing value of attribute like "Mouth Open" or "Eyeglasses" and "Age", respectively, attribute intensity was strengthened or weakened smoothly, and other visual appearances irrelevant to the attribute of interest remained unchanged. In particular, the identity-related visual appearance was well preserved.

4.6 Random image generation

Comparing with those baselines, our model is also capable of synthesizing diverse and realistic facial images given specific attributes. With this unique property, our model can not only generate realistic facial images from random noise as recent works [5, 21–23], but it can also control the attributes that we want the images to possess.

To examine this important property of our model, we evaluated it on the task of attribute-conditional random image generation. To synthesize facial images with specific attributes, we generated these samples in Fig. 6 through the following process: firstly, noise variables were randomly sampled from unit isotropic Gaussian distribution; secondly, the noise variables and specific attribute variables with values 1 or 0 were fed into the generator of LSA-VAE. As shown in Fig. 6, the four groups of facial images, synthesized with four

Table 5 Network architecture of the encoder in LSA-VAE-CNN, which is an invariant of LSA-VAE based on convolutional networks

Encoder Layer	Output Size
Conv(32,4,2,1), BN, Leaky ReLU	$32 \times 64 \times 64$
Conv(64,4,2,1), BN, Leaky ReLU	$64 \times 32 \times 32$
Conv(128,4,2,1), BN, Leaky ReLU	$128 \times 16 \times 16$
Conv(256,4,2,1), BN, Leaky ReLU	$256 \times 8 \times 8$
Conv(256,4,2,1), BN, Leaky ReLU	$512 \times 4 \times 4$
Output Branch 1: FC(8192,512)	512
Output Branch 2: FC(8192,512)	512
Output Branch 3: FC(8192, 13)	13

Table 6 Network architecture of the decoder in LSA-VAE-CNN, which is an invariant of LSA-VAE based on convolutional networks

Decoder Layer	Output Size
FC(140, 8192), LeakyReLU	512 × 4 × 4
Upsample, Conv(256,3,1,0), BN, ReLU	256 × 8 × 8
Upsample, Conv(128,3,1,0), BN, ReLU	128× 16× 16
Upsample, Conv(64,3,1,0), BN, ReLU	64× 32× 32
Upsample, Conv(32,3,1,0), BN, ReLU	32× 64× 64
Upsample, Conv(3,3,1,0)	3× 128× 128

groups of attributes which are ‘Black Hair/ Male/ Young’, ‘Blond Hair/ Female/ Young’, ‘Brown Hair/ Female/ Mouth Open/ Young’ and ‘Bale/ Male/ Old’, look photo-realistic with vivid texture. In each group, the faces were different with several view points. Those results imply that the proposed model is capable of generating facial images with randomly specified attributes without retraining the model using label-wise samples.

More results of random image generation are shown in Fig. 7 and all of them were synthesized by random attributes sampled from the thirteen attributes. For instance, the image in second row and fifth column was generated by feeding attributes of “Black Hair”, “Bushy Eyebrows”, “Mouth Open”, “Pale Skin”, “Female” and “Young”.

4.7 Quantitative analysis

In this section, we performed two kinds of quantitative analysis on similarity between source images and their reconstructed images and quality of generated images. For fair comparison, 10,000 images from testing set were randomly selected as the source image set, and their reconstructed images and transformed images with 13 attributes which were generated by each model were formed as the reconstruction set and the transformation set. We evaluated the abilities of the models using four metrics.

Identity preserving is an important factor in facial image editing. We evaluated the similarity between source images and the reconstructed images for each model. Three metrics were adopted, which are Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [54] and Learned Perceptual Image Patch Similarity (LPIPS) [58]. For PSNR and SSIM, the higher value means more similar between source image and its reconstructed image. For LPIPS, which evaluates similarity on deep features of images by feeding them to the pre-trained network, such as VGG [44] or AlexNet [28], the lower value means the more similar. To evaluate the quality of generated images, we used Fréchet Inception distance (FID) [18]. FID calculates the Fréchet distance also known as Wasserstein-2 distance between the source images and the generated images in the feature space of Inception

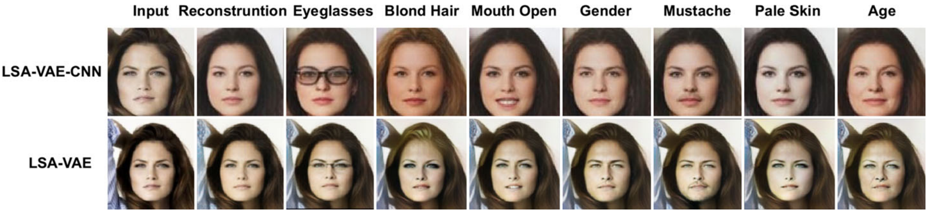

Fig. 8 Comparison between LSA-VAE-CNN and LSA-VAE on the task of single-attribute facial editing

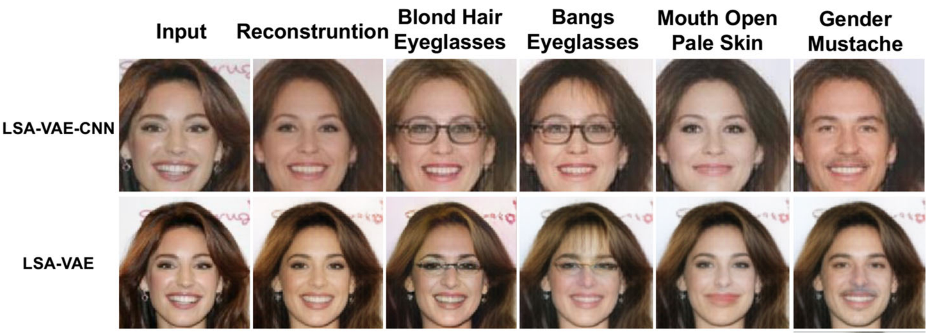

Fig. 9 Comparison between LSA-VAE-CNN and LSA-VAE on the task of multiple-attribute facial editing

Net. FID has been shown to be consistent with human judgement and robust to noise. Lower FID value means the distribution of the generated images is of closer distance to the distribution of the source images. In addition, the number of parameters of each model is also a key point needed to be compared with. A model with more parameters means it needs more time and memory to train the model. Thus, it is useful to compare the number of parameters of each model (in unit of a million).

All these comparison results are shown in Table 4. From which it can be seen that AttGAN achieved the best scores in terms of PSNR, SSIM, LPIPS and FID, and our model took the second place. IcGAN produced the poorest evaluation results, which further confirms the conclusion observed from Figs. 2, 3 and 4. Note that although evaluation vales of our model were a little weaker than those of AttGAN, the number of parameters of the proposed model is much fewer than the number of AttGAN, which implies that our model is more efficient and less complex.

4.8 Ablation study

To study the function of each part in the proposed model, we did the ablation study in this section. We established two different variants of our model: one based on convolutional

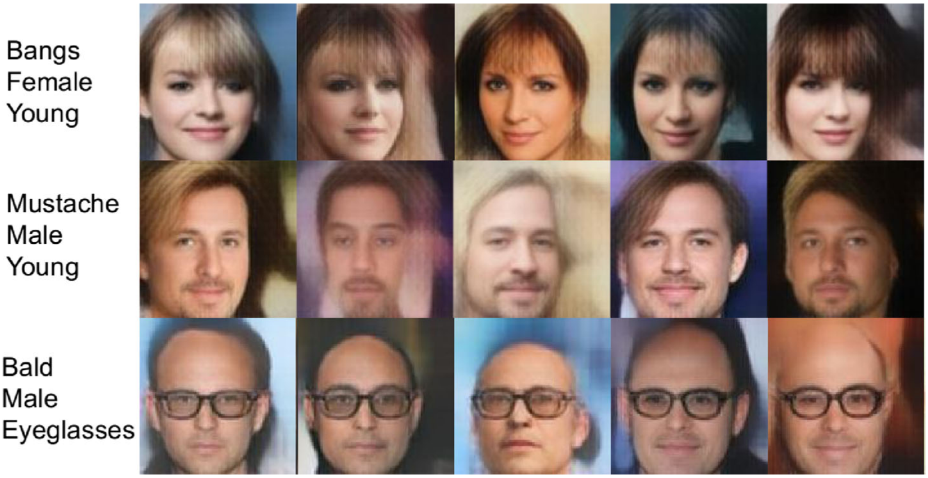

Fig. 10 Examples of attribute-conditional random image generation based on LSA-VAE-CNN

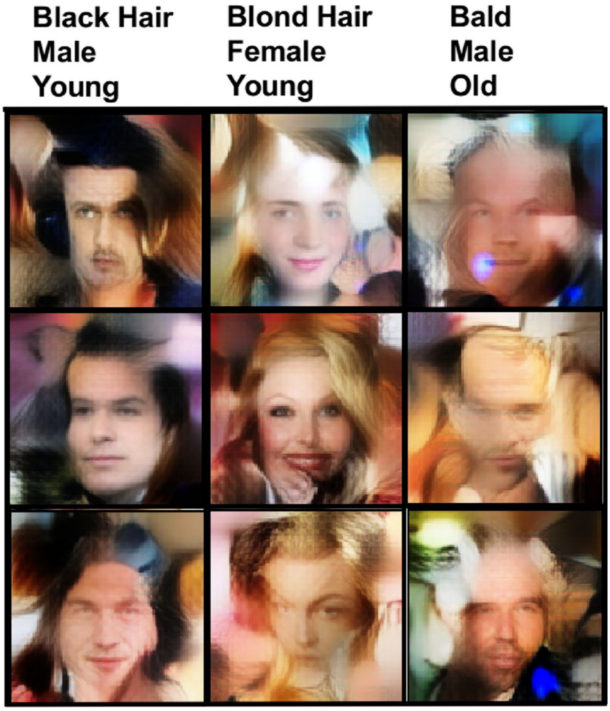

Fig. 11 Examples of attribute-conditional random image generation with LSA-VAE without performing the adversarial training phase

network and one without performing the adversarial training phase, which were named LSA-VAE-CNN and LSA-VAE-part, respectively.

The network architectures of LSA-VAE-CNN are shown in Tables 5 and 6. Both encoder and decoder network were based on deep convolutional neural network. Since the only difference between LSA-VAE and LSA-VAE-CNN is the network architecture, we can analyze the effect of different network architectures on the quality of generated images. In addition, due to the different network structure to that of LSA-VAE, different parameter settings were adopted for training LSA-VAE-CNN. Specifically, LSA-VAE-CNN was trained by the ADAM optimizer [26] ($\beta_1 = 0.5$, $\beta_2 = 0.999$) with the learning rate of 0.0005. The marginal m in Eq. 10 was set to 10. α and β in Eq. 9 were set to 0.5 and 100, respectively. γ_1, γ_2, γ_3, γ_4 and γ_5 were set to 0.01,100, 1, 100, 10, respectively. We performed comparison experiments between LSA-VAE-CNN and LSA-VAE on the task of single-attribute editing and multiple-attribute editing. The results are shown in Figs. 8 and 9. As we can see, LSA-VAE-CNN could change image attribute accurately, but generated more blurry images than LSA-VAE, and the images generated by LSA-VAE-CNN lack details of skin and hair texture. Moreover, LSA-VAE-CNN could not preserve the facial identity like LSA-VAE. We also performed random images generation with LSA-VAE-CNN to look into whether it can synthesize realistic images given specific attributes. As shown in Fig. 10, LSA-VAE-CNN can synthesize acceptable facial images, but the oversmooth face and checker texture in hair area make them not look like real faces. In a word, LSA-VAE can generate much sharper and realistic images than LSA-VAE-CNN. We suggest the reason is that, the skip connection in residual block is helpful to enhance image quality of editing result.

LSA-VAE-part employs the same network architecture as mentioned in Tables 1 and 2, but only performs the reconstruction training phase without the adversarial training. In this case, we investigated the role of adversarial training in random image generation. As shown in Fig. 11, the images generated by LSA-VAE-part with specific attributes are blurry and distorted seriously, comparing with results in Figs. 6 and 7, which demonstrates that the adversarial training plays an important role in the success of image synthesis.

5 Conclusion

This paper proposed a novel attribute-disentangled generative model for facial image editing conditioned on arbitrarily specified attributes by combining the advantages of variational autoencoders and generative adversarial networks. In the proposed model, the latent space is split into two independent subspaces. By introducing the adversarial training strategy on the latent space, the generated data distribution is trained to approach the real data distribution in the latent space and meanwhile the generated images are used to trained the encoder like a discriminator. We evaluated our model by attribute manipulation and random images generation experiments. The experimental results demonstrated our proposed model could learn attribute-disentangled representations of facial images and generate face images with rich details and high visual quality.

Acknowledgments This research was funded by Natural Science Foundation of China under grants numbers 61673018, 61272338, 61703443 and Guangzhou Science and Technology Founding Committee under grant No. 201707010222 and Guangdong Province Key Laboratory of Computer Science.

References

1. Akhtar Z, Dasgupta D, Banerjee B (2019) Face authenticity: An overview of face manipulation generation, detection and recognition. In: Nutan College of Engineering & Research, International Conference on Communication and Information Processing (ICCIP)
2. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp 214–223
3. Bao J, Chen D, Wen F, Li H, Hua G (2017) Cvae-gan: fine-grained image generation through asymmetric training. In: Proceedings of the IEEE International Conference on Computer Vision, pp 2745–2754
4. Bengio Y, éric ThibodeauLaufer, Alain G, Yosinski J (2013) Deep generative stochastic networks trainable by backprop. Computer Science 2:226–234
5. Brock A, Donahue J, Simonyan K (2019) Large scale GAN training for high fidelity natural image synthesis. In: International Conference on Learning Representations. https://openreview.net/forum?id=B1xsqj09Fm
6. Brock A, Lim T, Ritchie JM, Weston N (2016) Neural photo editing with introspective adversarial networks. arXiv:1609.07093
7. Charlier P, Froesch P, Huynh-Charlier I, Fort A, Hurel A, Jullien F (2014) Use of 3d surface scanning to match facial shapes against altered exhumed remains in a context of forensic individual identification. Forensic Science, Medicine, and Pathology 10(4):654–661
8. Che T, Li Y, Jacob AP, Bengio Y, Li W (2016) Mode regularized generative adversarial networks. arXiv:1612.02136
9. Choi Y, Choi M, Kim M, Ha J-W, Kim S, Choo J (2018) Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8789–8797
10. Dai B, Wipf D (2019) Diagnosing and enhancing VAE models. In: International Conference on Learning Representations. https://openreview.net/forum?id=B1e0X3C9tQ

11. Donahue J, Krähenbühl P, Darrell T (2016) Adversarial feature learning. arXiv:1605.09782
12. Dumoulin V, Belghazi I, Poole B, Lamb A, Arjovsky M, Mastropietro O, Courville A (2016) Adversarially learned inference. arXiv:1606.00704
13. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
14. Gregor K, Danihelka I, Graves A, Rezende DJ, Wierstra D (2015) Draw: a recurrent neural network for image generation
15. Gulrajani I, Ahmed F, Arjovsky M, Dumoulin V, Courville AC (2017) Improved training of wasserstein gans. In: Advances in neural information processing systems, pp 5767–5777
16. Guo Q, Zhu C, Xia Z, Wang Z, Liu Y (2017) Attribute-controlled face photo synthesis from simple line drawing. arXiv:1702.02805
17. He Z, Zuo W, Kan M, Shan S, Chen X (2019) Attgan: Facial attribute editing by only changing what you want. IEEE Trans Image Process 28(11):5464–5478
18. Heusel M, Ramsauer H, Unterthiner T, Nessler B, Hochreiter S (2017) Gans trained by a two timescale update rule converge to a local nash equilibrium. In: Advances in Neural Information Processing Systems, pp 6626–6637
19. Huang H, He R, Sun Z, Tan T et al (2018) Introvae: Introspective variational autoencoders for photographic image synthesis. In: Advances in neural information processing systems, pp 52–63
20. Isola P, Zhu J-Y, Zhou T, Efros AA (2017) Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1125–1134
21. Karras T, Aila T, Laine S, Lehtinen J (2018) Progressive growing of GANs for improved quality, stability, and variation. In: International Conference on Learning Representations. https://openreview.net/forum?id=Hk99zCeAb
22. Karras T, Laine S, Aila T (2019) A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4401–4410
23. Karras T, Laine S, Aittala M, Hellsten J, Lehtinen J, Aila T (2019) Analyzing and improving the image quality of stylegan. arXiv:1912.04958
24. Kim T, Cha M, Kim H, Lee JK, Kim J (2017) Learning to discover cross-domain relations with generative adversarial networks, JMLR. org. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp 1857–1865
25. Kim T, Kim B, Cha M, Kim J (2017) Unsupervised visual attribute transfer with reconfigurable generative adversarial networks Computer Vision and Pattern Recognition
26. Kingma DP, Ba JL (2015) Adam: A method for stochastic optimization, international conference on learning representations
27. Kingma DP, Welling M (2014) Auto-encoding variational bayes. In: international conference on learning representations
28. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
29. Lample G, Zeghidour N, Usunier N, Bordes A, Denoyer L, Ranzato M (2017) Fader networks: Manipulating images by sliding attributes. In: Advances in Neural Information Processing Systems, pp 5967–5976
30. Larsen ABL, Sønderby SK, Larochelle H, Winther O (2016) Autoencoding beyond pixels using a learned similarity metric International Conference on Machine Learning, pp 1558–1566
31. Li M, Zuo W, Zhang D (2016) Deep identity-aware transfer of facial attributes. arXiv:1610.05586
32. Liu M, Breuel TM, Kautz J (2017) Unsupervised image-to-image translation networks
33. Liu M, Tuzel O (2016) Coupled generative adversarial networks
34. Liu Z, Luo P, Wang X, Tang X (2016) Deep learning face attributes in the wild. In: IEEE International Conference on Computer Vision, pp 3730–3738
35. Lu Y, Tai Y-W, Tang C-K (2018) Attribute-guided face generation using conditional cyclegan. In: Proceedings of the European conference on computer vision (ECCV), pp 282–297
36. Makhzani A, Shlens J, Jaitly N, Goodfellow I, Frey B (2015) Adversarial autoencoders, Computer Science
37. Marcolin F, Vezzetti E (2017) Novel descriptors for geometrical 3d face analysis. Multimedia Tools and Applications 76:13805–13834
38. Mirza M, Osindero S (2014) Conditional generative adversarial nets
39. Perarnau G, van de Weijer J, Raducanu B, Álvarez JM (2016) Invertible Conditional GANs for image editing. In: NIPS Workshop on Adversarial Training

40. Radford A, Metz L, Chintala S (2016) Unsupervised representation learning with deep convolutional generative adversarial networks. In: International Conference on Learning Representations
41. Rezende DJ, Mohamed S, Wierstra D (2014) Stochastic backpropagation and approximate inference in deep generative models. In: international conference on machine learning, pp 1278–1286
42. Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X (2016) Improved techniques for training gans. In: Advances in neural information processing systems, pp 2234–2242
43. Shen W, Liu R (2017) Learning residual images for face attribute manipulation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 1225–1233
44. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition, Computer Science
45. Sohn K, Lee H, Yan X (2015) Learning structured output representation using deep conditional generative models. In: Advances in neural information processing systems, pp 3483–3491
46. Taigman Y, Polyak A, Wolf L (2017) Unsupervised cross-domain image generation, international conference on learning representations
47. Tang Y, Salakhutdinov R (2013) Learning stochastic feedforward neural networks. In: International Conference on Neural Information Processing Systems, pp 530–538
48. Tolosana R, Vera-Rodriguez R, Fierrez J, Morales A, Ortega-Garcia J (2020) Deepfakes and beyond: A survey of face manipulation and fake detection. arXiv:2001.00179
49. Tolstikhin I, Bousquet O, Gelly S, Schoelkopf B (2018) Wasserstein auto-encoders. In: International Conference on Learning Representations. https://openreview.net/forum?id=HkL7n1-0b
50. Ulyanov D, Vedaldi A, Lempitsky V (2018) It takes (only) two: Adversarial generator-encoder networks. In: Thirty-Second AAAI Conference on Artificial Intelligence
51. Upchurch P, Gardner JR, Pleiss G, Pless R, Snavely N, Bala K, Weinberger KQ (2017) Deep feature interpolation for image content changes
52. Vezzetti E, Tornincasa-Luca S, Federica Marcolin U, Dagnes N (2018) 3d geometry-based automatic landmark localization in presence of facial occlusions. Multimedia Tools and Applications 77: 14177–14205
53. Wang T-C, Liu M-Y, Zhu J-Y, Tao A, Kautz J, Catanzaro B (2018) High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8798–8807
54. Wang Z, Bovik AC, Sheikh HR, Simoncelli EP (2004) Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing 13(4):600–612
55. Xiao T, Hong J, Ma J (2018) Dna-gan: Learning disentangled representations from multi-attribute images. In: International Conference on Learning Representations, Workshop
56. Xiao T, Hong J, Ma J (2018) Elegant: Exchanging latent encodings with gan for transferring multiple face attributes. In: Proceedings of the European conference on computer vision (ECCV), pp 168–184
57. Yan X, Yang J, Sohn K, Lee H (2016) Attribute2image: Conditional image generation from visual attributes, Springer International Publishing
58. Zhang R, Isola P, Efros AA, Shechtman E, Wang O (2018) The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR
59. Zhou S, Xiao T, Yang Y, Feng D, He Q, He W (2017) Genegan: Learning object transfiguration and attribute subspace from unpaired data. In: Proceedings of the British Machine Vision Conference (BMVC). arXiv:1705.04932
60. Zhu J-Y, Park T, Isola P, Efros AA (2017) Unpaired image-to-image translation using cycle-consistent adversarial networkss. In: Computer Vision (ICCV), 2017 IEEE International Conference on