Neurocomputing xxx (xxxx) xxx

Contents lists available at ScienceDirect

Neurocomputing

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

Two birds with one stone: Transforming and generating facial images with iterative GAN

Dan Ma[a], Bin Liu[b], Zhao Kang[a], Jiayu Zhou[c], Jianke Zhu[d], Zenglin Xu[a,*]

[a] SMILE Lab, School of Computer Science & Engineering, University of Electronic Science and Technology of China, Chengdu, China
[b] The Center of Statistical Research, School of Statistics, Southwestern University of Finance and Economics, Chengdu, China
[c] Michigan State Universtiy, East Lansing, MI, United States
[d] Zhejiang University, 38 Zheda Road, Hangzhou, China

ARTICLE INFO

Article history:
Received 28 March 2018
Revised 10 September 2018
Accepted 12 October 2018
Available online xxx

Keywords:
Image transformation
Image generation
Perceptual loss
GAN

ABSTRACT

Generating high fidelity identity-preserving faces with different facial attributes has a wide range of applications. Although a number of generative models have been developed to tackle this problem, there is still much room for further improvement. In particular, the current solutions usually ignore the perceptual information of images, which we argue that it benefits the output of a high-quality image while preserving the identity information, especially in facial attributes learning area. To this end, we propose to train GAN iteratively via regularizing the min-max process with an integrated loss, which includes not only the per-pixel loss but also the perceptual loss. In contrast to the existing methods only deal with either image generation or transformation, our proposed iterative architecture can achieve both of them. Experiments on the multi-label facial dataset CelebA demonstrate that the proposed model has excellent performance on recognizing multiple attributes, generating a high-quality image, and transforming image with controllable attributes.

1. Introduction

Image generation [1–3] and image transformation [4–6] are two important topics in computer vision. A popular way of image generation is to learn a complex function that maps a latent vector onto a generated realistic image. By contrast, image transformation refers to translating a given image into a new image with modifications on desired attributes or style. Both of them have wide applications in practice. For example, facial composite, which is a graphical reconstruction of an eyewitness's memory of a face [7], can assist police to identify a suspect. In most situations, police need to search a suspect with only one picture of the front view. To improve the success rate, it is very necessary to generate more pictures of the target person with different poses or expressions. Therefore, face generation and transformation have been extensively studied.

Benefiting from the successes of the deep learning, image generation and transformation have seen significant advances in recent years [8,9]. With deep architectures, image generation or transformation can be modeled in more flexible ways than traditional approaches. For example, the conditional PixelCNN [9] was developed to generate an image based on the PixelCNN (a generative model that predict the image distribution pixel by pixel). The generation process of this model can be conditioned on visible tags or latent codes from other networks. However, the quality of generated images and convergence speed need improvement[1]. In [1] and [10], the Variational Auto-encoders (VAE) [11] was proposed to generate natural images. Recently, Generative adversarial networks (GAN) [12] has been utilized to generate natural images [13] or transform images [4,6,14,15] with conditional settings [16].

The existing approaches can be applied to face generation or face transformation respectively, however, there are several disadvantages of doing so. First, face generation and face transformation are closely connected with a joint distribution of facial attributes while the current models are usually proposed to achieve them separately (face generation [10,17] or transformation [14,15,18]), which may limit the prediction performance. Second, learning facial attributes has been ignored by existing methods of face generation and transformation, which might deteriorate the quality of facial images. Third, most of the existing conditional deep models

* Corresponding author.
E-mail address: zlxu@uestc.edu.cn (Z. Xu).

[1] https://github.com/openai/pixel-cnn

https://doi.org/10.1016/j.neucom.2018.10.093
0925-2312/

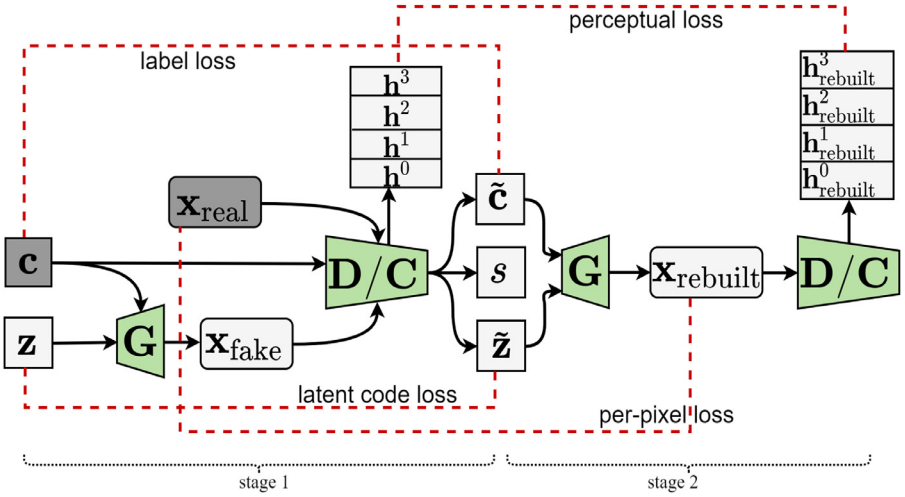


Fig. 1. The architecture of our proposed model. Training: the model begin with an original GAN process, the generator **G** takes a random noise vector **z** and label **c** as inputs. It outputs a generated face image $\mathbf{x}_{\text{fake}}$. The discriminator **D** receives both $\mathbf{x}_{\text{fake}}$ and a real image $\mathbf{x}_{\text{real}}$, and outputs the probability distribution over possible image sources. An auxiliary classifier **C**, which shares layers (without the last layer) with **D**, predicts a label $\tilde{\mathbf{c}}$ and outputs reconstructed noise code $\tilde{\mathbf{z}}$. Meanwhile, **D** outputs s (indicates whether the image is fake or real). At the second stage, **G** rebuilds the $\mathbf{x}_{\text{real}}$ by generating a $\mathbf{x}_{\text{rebuild}}$ with $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$. Then **D** iteratively receives $\mathbf{x}_{\text{rebuild}}$ and updates the hidden layers (from $\mathbf{h}$ to $\mathbf{h}_{\text{rebuild}}$). During testing, we can deal with both face generation and transformation. By sampling a random vector **z** and a desired facial attribute description, a new face can be generated by **G**. For face transformation, we feed the discriminator **D** a real image $\mathbf{x}_{\text{real}}$ together with its attribute labels **c**, then we get a noise representation $\tilde{\mathbf{z}}$ and label vector $\tilde{\mathbf{c}}$. The original image can be reconstructed with $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$ by feeding them back to the generator **G**. The new image generated by **G** is named $\mathbf{x}_{\text{rebuild}}$. Alternatively, we can modify the content of $\tilde{\mathbf{c}}$ before inputting **G**. According to the modification of labels, the corresponding attributes of the reconstructed image will be transformed.

did not consider to preserve the facial identity during the face transformation [19] or generation [10];

To this end, we propose an iterative GAN with an auxiliary classifier in this paper, which can not only generate high fidelity face images with controlled input attributes, but also integrate face generation and transformation by learning a joint distribution of facial attributes. We argue that the strong coupling between face generation and transformation should benefit each other. And the iterative GAN can learn and even manipulate multiple facial attributes, which not only help to improve the image quality but also satisfy the practical need of editing facial attributes at the same time. In addition, in order to preserve the facial identity, we regularize the iterative GAN by the perceptual loss in addition to the pixel loss. A quantity metric was proposed to measure the face identity in this paper.

To train the proposed model, we adopt a two-stage approach as shown in Fig. 1. In the first stage, we train a *discriminator* **D**, a *generator* **G**, and a *classifier* **C** by minimizing adversarial losses [12] and the label losses as in [3]. In the second stage, **G** and **D**/**C** are iteratively trained with an integrated loss function, which includes a perceptual component [20] between **D**'s hidden layers in stage 1 and stage 2, a latent code loss between the input noise **z** and the output noise $\tilde{\mathbf{z}}$, and a pixel loss between the input real facial images and their corresponding rebuilt version.

In the proposed model, the generator **G** not only generates a high-quality facial image according to the input attribute (single or multiple) but also translates an input facial image with desired attribute modifications. The fidelity of output images can be highly preserved due to the iterative optimization of the proposed integrated loss. To evaluate our model, we design experiments from three perspectives, including the necessity of the integrated loss, the quality of generated natural face images with specified attributes, and the performance of face generation. Experiments on the benchmark CelebA dataset [21] have indicated the promising performance of the proposed model in face generation and face transformation.

2. Related work

2.1. Facial attributes recognition

Object Recognition method has been researched for a long while as an active topic [22–24] especially for human recognition, which takes attributes of a face as the major reference [25,26]. Such attributes include but not limited to ***natural looks*** like Arched_Eyebrows, Big_Lips, Double_Chin, Male, etc. Besides, some ***artificial attributes*** also contribute to this identification job, like Glasses, Heavy_Makeup, Wave_Hair. Even some ***expression*** like Smiling, Angry, Sad can be labeled as a kind of facial attributes to improve identification. For example, Devi et al. analyze the complex relationship among these multitudinous attributes to categorize a new image [27]. The early works on automatic expression recognition can be traced back to the early nineties [28]. Most of them [25,29,30] tried to verify the facial attributes based on the methods of HOG(Histogram of Oriented Gradient) [31] and SVM (Support Vector Machine) [32]. Recently, the development of the deep learning flourished the expression recognition [33] and made a success of performing face attributes classification based on CNN (the Convolutional Neural Networks) [21]. The authors of [21] even devised a dataset celebA that includes more than 200k face images (each with 40 attributes labels) to train a large deep neural network. celebA is widely used in facial attributes researches. In this paper, we test iterative GAN with it. FaceNet [34] is another very important work on face attributes recognition that proposed by Google recently. They map the face images to a Euclidean space, thus the distance between any two images that calculated in the new coordinate system shows how similar they are. The training process is based on the simple heuristic knowledge that face images from the same person are closer to each other than faces from different persons. The FaceNet system provides a compare function to measure the similarity between a pair of images. We prefer to utilize this function to measure the facial identity-preserving in quantity.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

2.2. Conditioned image generation

Image generation is an very popular and classic topic in computer vision. The vision community has already taken significant steps on the image generation especially with the development of deep learning.

The conditional models [9,16,35] enable easier controlling of the image generation process. In [9], the authors presented an image generation model based on PixelCNN under conditional control. However, it is still not satisfying on image quality and efficiency of convergence. In the last three years, generating images with Variational Auto-encoders (VAE) [11] and GAN [12] have been investigated. A recurrent VAE was proposed in [1] to model every stage of picture generation. Yan et al. [10] used two attributes conditioned VAEs to capture foreground and background of a facial image, respectively. In addition, the sequential model [1,13] has attracted lots of attention recently. The recurrent VAE [1] mimic the process of human drawing but it can only be applied to low-resolution images; in [13], a cascade Laplacian pyramid model was proposed to generate an image from low resolution to a final full resolution gradually.

GAN [12] has been applied to lots of works of image generation with conditional setting since Mirza and Osindero [16] and Gauthier [36]. In [13], a Laplacian pyramid framework was adopted to generate a natural image with each level of the pyramid trained with GAN. In [37], the S^2GAN was proposed to divide the image generation process into structure and style generation steps corresponding to two dependent GAN. The Auxiliary classifier GAN (ACGAN) in [3] tries to regularize traditional conditional GAN with label consistency. To be more easy to control the conditional tag, the ACGAN extended the original conditional GAN with an auxiliary classifier **C**.

2.3. Image Transformation

Image transformation is a well-established area in computer vision which could be concluded as "where a system receives some input image and transforms it into an output image [20]". According to this generalized concept, a large amount of image processing tasks belong to this area such as images denoise [38], generate images from blueprint or outline sketch [19], alternate a common picture to an artwork maybe with the style of Vincent van Gogh or Claude Monet [6,39], image inpainting [38], the last but not the least, change the appointed facial attributes of a person in the image.

Most works of image transformation are based on pixel-to-pixel loss. Like us, [39] transforms images from one style to another with an integrated loss (feature loss and style reconstruction loss) through CNN. Though the result is inspiring, this model is very cost on extracting features on pre-trained model. Recently, with the rapid progress of generative adversarial nets, the quality of output transformed images get better. The existing models of image transforming based on GAN fall into two groups. Manipulating images over the natural image manifold with conditional GAN belongs to the first category. In [4], the authors defined user-controlled operations to allow visual image editing. The source image is arbitrary, especially could lie on a low-dimensional manifold of image attributes, such as the color or shape. In a similar way, Zhang et al. [14] assume that the age attribute of face images lies on a high-dimensional manifold. By stepping along the manifold, this model can obtain face images with different age attributes from youth to old. The remaining works consist of the second group. In [19], the transformation model was built on the conditional setting by regularizing the traditional conditional GAN [16] model with an image mapping loss. Some works consider deploying GAN models for each of the related image domains (e.g. input domain and output domain), more than one set of adversarial nets then cooperate and restrict with each other to generate high-quality results [5,6]. In [5], a set of aligned image pairs are required to transfer the source image of a dressed person to product photo model. In contrast, the Cycle GAN [6] can learn to translate an image from a source domain to a target domain in the absence of paired samples. This is a very meaningful advance because the paired training data will not be available in many scenarios. Moreover, transformation over multi-domain has been researched by [18], instead of using different models for each pair of image domains, StarGAN extends the architecture of GANs to fit training attributes from several datasets, makes the transformation more flexible. The StarGAN achieves state_of_art results of image transformation area, in Section 5.3.4, we perform a series of experiments to compare the result of transformation with proposed model and StarGAN.

2.4. Perceptual Losses

A traditional way to minimize the inconsistency between two images is optimizing the pixel-wise loss [40,41]. However, the pixel-wise loss is inadequate for measuring the variations of images, since the differences computed in pixel element space is not able to reflect the otherness in visual perspective. We could easily and elaborately produce two images which are absolutely different observed in human eyes but with minimal pixel-wise loss and vice verse. Moreover, using single pixel-wise loss often tend to generate blurrier results with the ignorance of visual perception information [42]. In contrast, the perceptual loss explores the discrepancy between high-dimensional representations of images extracted from a well-trained CNN (for example VGG-16 [43]) can overcome this problem. By narrowing the discrepancy between ground-truth image and output image from a high-feature-level perspective, the main visual information is well-reserved after the transformation. Recently, the perceptual loss has attracted much attention in image transformation area [20,39,44–46]. By employing the perceptual information, [39] perfectly created the artistic images of high perceptual quality that approaching the capabilities of human; [47] highlighted the applying of perceptual loss to generate super-resolution images with GAN, the result is amazing and state-of-art; [44] proposed a generalized model called perceptual adversarial networks (PAN) which could transform images from sketch to colored, semantic labels to ground-truth, rainy day to derainy day etc. Zhu et al. [4] focused on designing a user-controlled or visual way of image manipulation utilize the perceptual similarity between images that defined over a low dimensional manifold to visually edit images. An interesting topic in [46] aims to transform low-dose CT images to standard normal-dose CT images, the author also applies perceptual loss among VGG-19 model layers instead of MSE(Mean Squared Error) to measure the distance between the ground truth and faked feature maps.

A convenient way of calculating the perceptual loss between a ground-truth face $\mathbf{x}_{\text{ground-truth}}$ and a transformed face $\mathbf{x}_{\text{rebuild}}$ is to input them to a Convolutional Neural Networks (such as the VGG-16 [43] or the GoogLeNet [48]) then sum the differences of their representations from each hidden layers [20] of the network. As shown in Fig. 2 [20], both of the $\mathbf{x}_{\text{ground-truth}}$ and $\mathbf{x}_{\text{rebuild}}$ was passed to the pretrained network as inputs. The differences between their corresponding latent feature matrices $\mathbf{h}^i$ and $\mathbf{h}^i_{\text{rebuild}}$ can be calculated ($L_i = \|\mathbf{h}^i_{\text{rebuild}} - \mathbf{h}^i\|$).

The final perceptual loss can be represented as follows:

$$L_{per} = \sum_{i=1} \alpha_i \|\mathbf{h}^i_{\text{rebuild}} - \mathbf{h}^i\|$$

Where $\mathbf{h}^i_{\text{rebuild}}, \mathbf{h}^i$ are the feature matrices output from each convolutional layers, α_i are the parameters to balance the affections from each layers.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

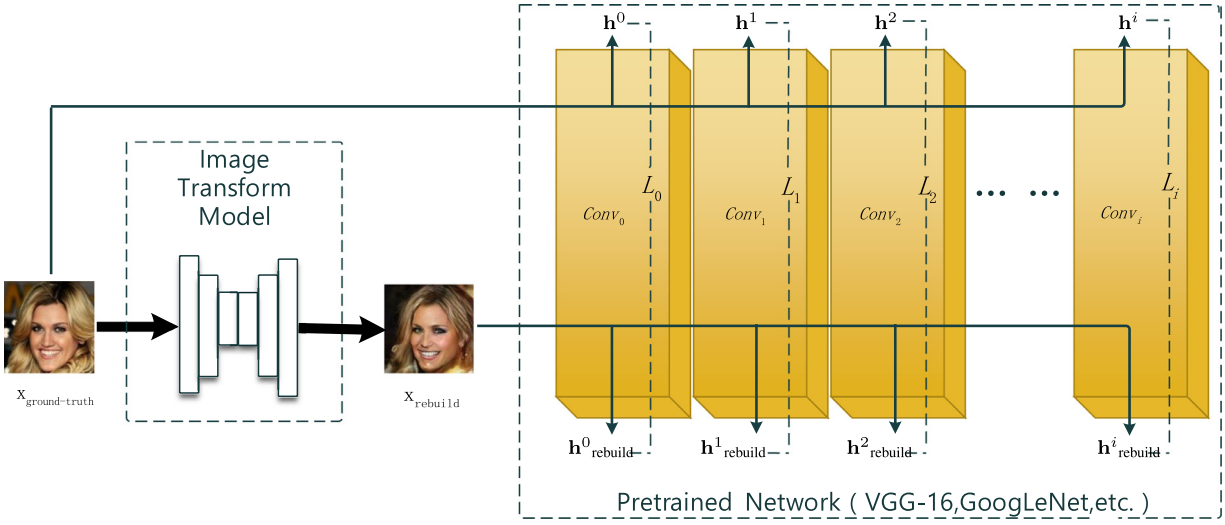


Fig. 2. Demonstration of how to train an image transformation model [20] with the perceptual loss. A pretrained network (right yellow part) plays a role in extracting feature information of both ground-truth and fake image through several convolutional layers. For each layer, the semantic differences L_i between the two images is calculated. Finally, by reducing the total differences collected through all layers, the image transform model (left part in dashed box) is optimized to rebuild the ground-truth image.

For facial image transformation, perceptual information shows its superiority in preserving the facial identity. To take advantages of this property of the perceptual loss, in our proposed model, we leverage it to keep the consistency of personal identity between two face images. In particular, we choose to replace the popular pretrained Network with the discriminator network to decrease the complexity. We will discuss this issue in detail later.

3. Proposed model

We first describe the proposed model with the integrated loss, then explain each component of the integrated loss.

3.1. Problem overview

To our knowledge, most of the existed methods related to image processing are introduced for one single goal, such as facial attributes recognition, generation or transformation. Our main purpose is to develop a multi-function model that is capable of managing these tasks altogether, end-to-end.

- ***Facial attributes recognition***
  By feeding a source image **x** (it doesn't matter if it is real or fake) to the **D**, the classifier **C** would output the probability of concerned facial attributes of **x**.
- ***Face Generation***
  By sampling a random vector **z** and a desired facial attributes description, a new face can be generated by **G**.
- ***Face Transformation***
  For face transformation, we feed the discriminator **D** a real image $\mathbf{x}_{\text{real}}$, then we get a noise representation $\tilde{\mathbf{z}}$ and label vector $\tilde{\mathbf{c}}$. The original image can be reconstructed with $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$ by feeding them back to the generator **G**. The new image generated by **G** is named $\mathbf{x}_{\text{rebuild}}$. Alternatively, we can modify the content of $\tilde{\mathbf{c}}$ recognized by classifier **C** before we input it into **G**. According to the modification of labels, the corresponding attributes of the reconstructed image then transformed.

To this end, We design a variant of GAN with iterative training pipeline as shown in Fig. 1, which is regularized by a combination of loss functions, each of which has its own essential purpose.

In specific, the proposed iterative GAN includes a generator **G**, a discriminator **D**, and a classifier **C**. The discriminator **D** and generator **G** along with classifier **C** can be re-tuned with three kinds of specialized losses. In particular, the perceptual losses captured by the hidden layers of discriminator will be back propagated to feed the network with the semantic information. In addition, the difference between the noise code detached by the classifier **C** and the original noise will further tune the networks. Last but not least, we let the generator **G** to rebuild the image $\mathbf{x}_{\text{rebuild}}$, then the error between $\mathbf{x}_{\text{rebuild}}$ and the original image $\mathbf{x}_{\text{real}}$ will also feedback. We believe this iterative training process learns the facial attributes well and the experiments demonstrate that aforementioned three loss functions play indispensable roles in generating natural images.

The object of iterative GAN is to obtain the optimal **G, D**, and **C** by solving the following optimization problem:

$$\arg\min_{\mathbf{G}} \max_{\mathbf{D},\mathbf{C}} L_{\text{ACGAN}}(\mathbf{G},\mathbf{D},\mathbf{C}) + L_{\text{inte}}(\mathbf{G},\mathbf{D},\mathbf{C}) \tag{1}$$

where L_{ACGAN} is the ACGAN [3] loss term to guarantee the excellent ability of classification, L_{inte} is the integrated loss term which contributes to the maintenance of source image's features. We appoint no balance parameters for the two loss terms because L_{inte} is a conic combination of another three losses (we will introduce them later). The following part will introduce the definitions of the two losses in detail.

3.2. ACGAN Loss

GAN is not only helpful in data generation, some classification areas also use GAN to finish the discrimination task. [49] introduced a GAN model to distinguish the telecom fraud case. And to produce higher quality images with classification, the ACGAN [3] extended the original adversarial loss [12] to a combination of the adversarial and a label consistency loss. Thus, the label attributes are learned during the adversarial training process. The label consistency loss makes a more strict constraint over the training of the min-max process, which results in producing higher quality samples of the generating or transforming. Inspired by the ACGAN, we link the classifier **C** into the proposed model as an auxiliary decoder. But for the purpose of recognizing multi-labels, we modify the representation of output labels into an n-dimensional vector, where n is the number of concerned labels. The ACGAN loss function is as follows:

$$L_{\text{ACGAN}}(\mathbf{G},\mathbf{D},\mathbf{C}) = L_{\text{adv}} + L_{\text{label}}$$

where the L_{adv} is the min-max loss defined in the original GAN [12,16] (Section 3.2.1), L_{label} is the label consistency loss [3] of classifier **C** (see Section 3.2.2).

3.2.1. Adversarial loss

As a generative model, GANs consists of two neural networks [12], the generative network **G** which chases the goal of learning distribution of the real dataset to synthesize fake images and the discriminative network **D** which endeavor to predict the source of input images. The conflicting purposes forced **G** and **D** to play a min-max game, and regulated the balance of the adversarial system.

In standard GAN training, the generator **G** takes a random noise variable $\mathbf{z}$ as input and generates a fake image $\mathbf{x}_{fake} = \mathbf{G}(\mathbf{z})$. In opposite, the discriminator **D** takes both the synthesized and native images as inputs and predicts the data sources. We follow the form of adversarial loss in ACGAN [3], which increases the input of **G** with additional conditioned information $\mathbf{c}$ (the attributes labels in the proposed model). The generated image hence depends on both the prior noise data $\mathbf{z}$ and the label information $\mathbf{c}$, this allows for reasonable flexibility to combine the representation, $\mathbf{x}_{fake} = \mathbf{G}(\mathbf{z}, \mathbf{c})$. Notice that unlike the CGANs [16], in our model, the input of **D** remains in the primary pattern without any conditioning.

During the training process, the discriminator **D** is forced to maximize the likelihood it assigns the correct data source, and the generator **G** performs oppositely to fool the **D** as following:

$$L_{adv}(\mathbf{G}, \mathbf{D}) = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\log \mathbf{D}(\mathbf{x})] + \mathbb{E}_{\mathbf{z},\mathbf{c}\sim p(\mathbf{z},\mathbf{c})}[\log(1 - \mathbf{D}(\mathbf{G}(\mathbf{z}|\mathbf{c})))]$$

3.2.2. The consistency of data labels

The label loss function of the classifier **C** is as following:

$$L_{label} = \mathbb{E}_{\mathbf{x}\in\{\mathbf{x}_{real},\mathbf{x}_{fake}\}}[\log \mathbf{C}(\mathbf{D}(\mathbf{x}))]$$

Either for the task of customized images generation or appointed attributes transformation, proper label prediction is necessary to resolve the probability distribution over the attributes of samples. We take the successful experiences of ACGAN [3] for reference to keep the consistency of data labels for each real or generated sample. During the training process, the real images $\mathbf{x}_{real}$ as well as the fake ones $\mathbf{x}_{fake} = \mathbf{G}(\mathbf{z}, \mathbf{c})$ are all fed to the discriminator **D**, the share layer output from **D** then is passed to classifier **C** to get the predicted labels $\tilde{\mathbf{c}} = \log(\mathbf{c} - \mathbf{C}(\mathbf{D}(\mathbf{x})))$. The loss of this predicted labels $\tilde{\mathbf{c}}$ and the actual labels $\mathbf{c}$ then is propagated back to optimize the **G, D**, and **C**.

3.3. Integrated loss

The ACGAN loss in Eq. (1) keeps the generated images by the iterative GAN lively. Additionally, to rebuild the information of the input image, we introduce the integrated loss that combines the per-pixel loss, the perceptual loss, and the latent code loss with parameter $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3) \in \mathbb{R}^3$,

$$L_{inte} = \lambda_1 L_{per} + \lambda_2 L_{pix} + \lambda_3 L_{\mathbf{z}}, \quad \lambda_i \geq 0.$$

The conic coefficients $\boldsymbol{\lambda}$ also suggests that we do not need to set a trade-off parameter in Eq. (1). We study the necessity of the three components by reconstruction experiments as shown in Section 5.3.2. These experiments suggest that combining three loss terms together instead of using only one of them clearly strengthens the training process and improves the quality of reconstructed face image. During the whole training process, we set $\lambda_1 = 2.0$, $\lambda_2 = 0.5$ and $\lambda_3 = 1.0$.

We then introduce three components of the integrated loss as following:

3.3.1. Per-pixel Loss

Per-pixel loss [8,40] is a straightforward way to pixel-wisely measure the difference between two images, the real face $\mathbf{x}_{real}$ and the fake face $\mathbf{x}_{rebuild}$ as follows:

$$L_{pix} = \mathbb{E}[\|\mathbf{x}_{real} - \mathbf{x}_{rebuild}\|]$$

where $\mathbf{x}_{rebuild} = \mathbf{G}(\tilde{\mathbf{z}}, \tilde{\mathbf{c}})$ is the generator **G** that reconstructs the real image $\mathbf{x}_{real}$ based on predicted values $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$. The per-pixel loss forces the source image and destination image as closer as possible within the pixel space. Though it may fail to capture the semantic information of an image (a tiny and invisible diversity in human-eyes may lead to huge per-pixel loss, vice versa), we still think it is a very important measure for image reconstruction that should not be ignored.

The process of rebuilding $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$ is demonstrated in Fig. 1. Given a real image $\mathbf{x}_{real}$, the discriminator **D** extracts a hidden map $\mathbf{h}^3_{real}$ with its 4*th* convolution layer. Then $\mathbf{h}^3_{real}$ is linked to two different full connected layers and they output a 1024-dimension share layer (a layer shared with C) and a scalar (the data source indicator s) respectively. The classifier **C** also has two full-connected layers. It receives the 1024-dimension share layer from **D** as an input and outputs the predicted noise $\tilde{\mathbf{z}}$ and the predicted label $\tilde{\mathbf{c}}$ with its two full-connected layers as shown in Fig. 3:

$$\tilde{\mathbf{z}}, \tilde{\mathbf{c}} = \mathbf{C}(\mathbf{D}(\mathbf{x}_{real})).$$

3.3.2. Perceptual loss

Traditionally, per-pixel loss [8,11,40] is very efficient and popular in reconstructing image. However, the pixel-based loss is not a robust measure since it cannot capture the semantic difference between two images [20]. For example, some unignorable defects, such as blurred results (lack of high-frequency information), artifacts (lack of perceptual information) [44], often exist in the output images reconstructed via the per-pixel loss. To balance these side effects of the per-pixel loss, we feed the training process with the perceptual loss [20,44,45] between $\mathbf{x}_{real}$ and $\mathbf{x}_{rebuild}$. We argue that this perceptual loss captures the discrepancy between two images in semantic space, so it focuses on the global structural differences between images rather than local pixel differences(pixel loss) or tell the entire data distribution over the image dataset(which is GAN loss's job).

To reduce the model complexity, we calculate the perceptual loss on all convolutional layers of the discriminator **D** rather than on third-part pre-trained networks like VGG [43], GoogLeNet [48]. Let $\mathbf{h}^i$ and $\mathbf{h}^i_{rebuild}$ be the feature maps extracted from the ith layer of **D** with real image and fake image respectively, we obtain feature difference for each convolutional layer, then we sum up all these difference from all convolutional layers rather than for a few of them to avoid missing information extracted from some layers (which is a common accepted method introduced in [20,44,45]), the perceptual loss L_{per} between the original image and the fake one is defined as follows:

$$L_{per} = \mathbb{E}[\|\mathbf{h}^i_{rebuild} - \mathbf{h}^i\|].$$

According to [44], We simply employ L1 norm to measure the L_{per} distance. The minimization of the L_{per} forces the perceptual information in the fake face to be consistent with the original face.

3.3.3. Latent code loss

The intuitive idea to rebuild the source images is that we assume that the latent code of face attributes lie on a manifold $\mathcal{M}$ and faces can be generated by sampling the latent code on different directions along the $\mathcal{M}$.

In the train process, the generator **G** takes a random latent code $\mathbf{z}$ and label $\mathbf{c}$ as an input and outputs the fake face $\mathbf{x}_{fake}$. Then the min-max game forces the discriminator **D** to discriminate between

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

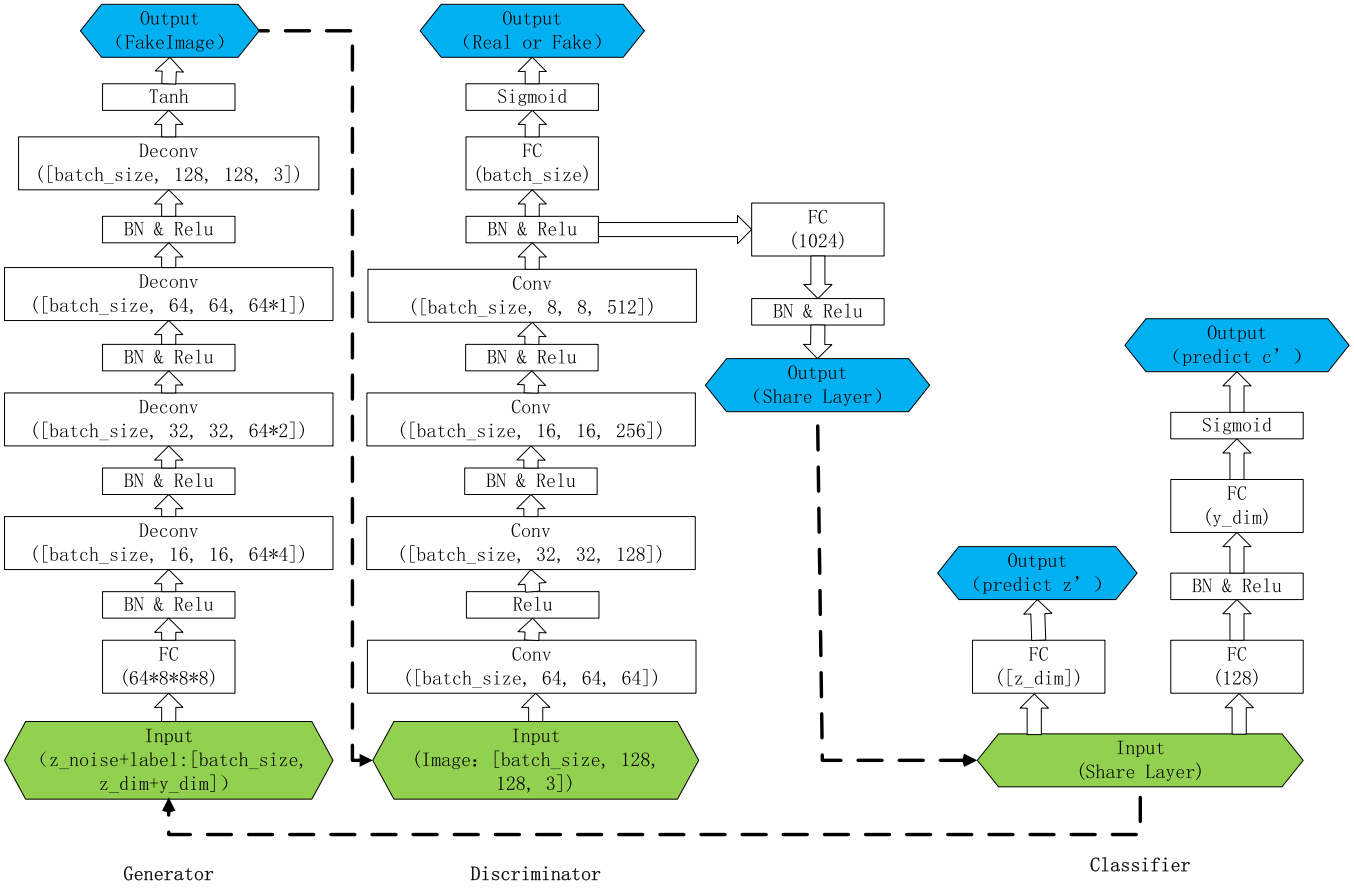


Fig. 3. The overview of the network architecture of iterative GAN.

$\mathbf{x}_{\text{fake}}$ and the real image $\mathbf{x}_{\text{real}}$. Meanwhile, the auxiliary classifier **C**, which shares layers (without the last layer) with **D**, detach a reconstructed latent code $\tilde{\mathbf{z}}$. At the end of the min-max game, both $\mathbf{z}$ and $\tilde{\mathbf{z}}$ should share a same location on the $\mathcal{M}$ because they are extracted from a same image. Hence, we construct a loss $\tilde{\mathbf{z}}$ and the random $\mathbf{z}$ with L1 norm to regularize the process of image generation, i.e.,

$$L_{\mathbf{z}} = \mathbb{E}[\|\tilde{\mathbf{z}} - \mathbf{z}\|]$$

In this way, the latent code $\tilde{\mathbf{z}}$ detached by the classifier **C** will be aligned with $\mathbf{z}$.

4. Network architecture

Iterative GAN includes three neural networks. The generator consists a fully-connected layer with 8584 neurons, four deconvolutional layers and each has 256, 128, 64 and 3 channels with filer size 5*5. All filter strides of the generator **G** are set as 2*2. After processed by the first fully-connected layer, the input noise $\mathbf{z}$ with 100 dimensions is projected and reshaped to [Batch_size, 5, 5, 512], the following 4 deconvolutional layers then transpose the tensor to [Batch_size, 16, 16, 256], [Batch_size, 32, 32, 128], [Batch_size, 64, 64, 64], [Batch_size, 128, 128, 3], respectively. The tensor output from the last layer is activated by the *tanh* function.

The discriminator is organized almost in the reverse way of the generator. It includes 4 convolutional layers with filter size 5*5 and stride size 2*2. The training images with shape [Batch_size, 128, 128, 3] then transpose through the following 4 convolutional layers and receive a tensor with shape [Batch_size, 5, 5, 512]. The discriminator needs to output two results, a shared layer for the classifier and probability that indicates if the input image is a fake one. We flat the tensor and pass it to classifier as input for the former purpose. To get the reality of image, we make an extra full connect layer which output [Batch_size, 1] shape tensor with *sigmoid* active function.

The classifier receives the shared layer from the discriminator which contains enough feature information. We build two full-connected layers in the classifier, one for the predicted noise $\tilde{\mathbf{z}}$ and the other for the output predicted labels $\tilde{\mathbf{c}}$. And we use the *tanh* and *sigmoid* function to squeeze the value $\tilde{\mathbf{z}}$ to $(-1,1)$, $\tilde{\mathbf{c}}$ to $(0, 1)$, respectively. And the Fig. 3 shows how we organize the networks of iterative GAN.

4.1. Optimization

Following the DCGAN [50], we adopt the Adam optimizer [51] to train proposed iterative GAN. The learning rate α is set to 0.0002 and β_1 is set to 0.5, β_2 is 0.999 (same settings as DCGAN). To avoid the imbalanced optimization between the 2 competitors G and D, which happened commonly during GANs's training and caused the vanishment of gradients, we set 2 more parameters to control the update times of D and G in each iteration. While the loss of D is overwhelmingly higher than G, we increase the update times of D in this iteration, and vice versa. By using this trick, the stability of training process is apparently approved.

4.2. Statistics for training and testing

We introduce the time costs of training and testing in this section. The proposed iterative GAN model was trained on an Intel Core i5-7500 CPU@3.4 GH_z with 4 cores and NVIDIA 1070Ti GPU.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

Table 1
Analysis of time consumption for training.

	epoch	Image Size	Cost time (s)
Training	10	128*128	106,230 s
	20	128*128	23,9460 s
	50	128*128	600,030 s

Table 2
Analysis of time consumption for testing.

	Image num	Input image size	Output image size	Cost time (s)
Rebuild	64	128*128	128*128	3.2256
Generate	64	\	128*128	2.1558

The training process goes through 50 epochs and shows the final results. The Analysis of the time consumption (including both training and forward propagation process) are shown in Table 1 and Table 2, respectively.

5. Experiments

We perform our experiment for multiple tasks to verify the capability of iterative GAN model: recognition of facial attributes, face images reconstruction, face transformation, and face generation with controllable attributes.

5.1. Dataset

We run the iterative GAN model on celebA dataset [21] which is based on celebFace+ [52]. celebA is a large-scale face image dataset contains more than 200k samples of celebrities. Each face contains 40 familiar attributes, such as ***Bags_Under_Eyes, Bald, Bangs, Blond_Hair***, etc. Owing to the rich annotations per image, celebA has been widely applied to face visual work like face attribute recognition, face detection, and landmark (or facial part) localization. We take advantage of the rich attribute annotations and train each label in a supervised learning approach.

We split the whole dataset into 2 subsets: 185000 images of them are randomly selected as training data, the remaining 15000 samples are used to evaluate the results of the experiment as the test set.

We crop the original images of size 178*218 into 178*178, then further resize them to 128*128 as the input samples. The size of the output (generated) images are as well as the inputs.

5.2. The metric of face identity

Given a face image, whatever rebuilding it or transforming it with customized attributes, we have to preserve the similarity (or identity) between the input and output faces. It is very important for human face operation because the maintenance of the primary features belong to the same person is vital during the process of face transformation. Usually, the visual effect of face identity-preserving can only be observed via naked eyes.

Besides visual observation, in this paper, we choose FaceNet [34] to define the diversity between a pair of images. In particular, FaceNet accepts two images as input and output a score which reveal the similarity. A lower score indicates that the two images are more similar, and vice versa. In other words, FaceNet provides us a candidate metric on the face identity evaluation. We take it as an important reference in the following related experiments.

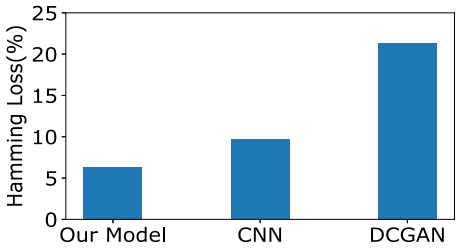


Fig. 4. The hamming loss of attribute prediction.

5.3. Results

5.3.1. Recognition of facial attributes (multi-labels)

Learning face attributes is fundamental to face generation and transformation. Previous work learned to control single attribute [17] or multi-category attributes [3] through a *softmax* function for a given input image. However, the natural face images are always associated with multiple labels. To the best of our knowledge, recognizing and controling the multi-label attributes for a given facial image are among most challenging issues in the community. In our framework, the classifier **C** accepts a 1024 dimensions shared vector that outputted by the discriminator **D**, then **C** squash it into 128 dimensions by a full connection. To output the multiple labels, we just need to let the classifier **C** to compress the 128 dimension median vector into d dimensions (as shown in Fig. 3, d is the dimension of label vector, $d = 40$ in this paper). By linking the d dimensions vector to d *sigmoid* mappings, the classifier **C** output the predictions of attribute labels finally.

We feed the images in test set to the classifier **C** and calculate the hamming loss for the multi-label prediction. The statistic of hamming loss is over all the 40 labels associated with each test samples. Two methods are selected as baselines. The first one is native DCGAN with L2-SVM classifier which is reported superior to k-means and k-NN [50]. The other is a Convolutional Neural Network. We train both referenced models on celebA. For the DCGAN, the discriminator extracts the features and feeds it to the linear L2-SVM with Euclidean distance for unsupervised classification. Mean while, the CNN model outputs the predicted labels directly after standard process of convolution training.

Fig. 4 illustrates the hamming loss of the three algorithms. It is clear to see that the iterative GAN significantly outperforms DCGAN + L2-SVM and CNN. We speculate that the proposed joint architecture on both face generation and transformation regularized by the integrated loss make the facial attribute learning of iterative GAN is much easier than the baselines.

Besides the hamming loss statistics of hamming loss, we also visualize part of the results in Table 3. Row 2, 3 and 4 in Table 3 illustrate three examples in the test set. Row 2 and 3 are the successful cases, while row 4 shows a failed case on predicting ***Heavy-MakeUp*** and ***Male***.

5.3.2. Reconstruction

In this experiment, we reconstruct given target faces in 4 different settings (per-pixel_loss,**z**_loss, **z**_loss + per-pixel_loss, and the integrated Loss) separately. This experiment proves that the iterative GAN provided with integrated loss has a strong ability to deal with the duties mentioned above and archive the similar or better results than previous works.

By feeding the noise vector $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$, the generator **G** can reconstruct the original input image with its attributes preserved ($\mathbf{x}_{\text{rebuild}}$ in Fig. 1). We will evaluate the contributions of integrated loss in this experiment of face reconstruction. In detail, we run 4 experiments by regularizing the ACGAN Loss with: only per-pixel loss, latent code loss(**z**_loss), latent code loss(**z**_loss) + per-

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

Table 3
Demonstrate the example of classification result of iterative GAN. The listed ground-truth tags of target image are expressed by two integer 1 and −1. Row 1 and 2 show the exactly correct prediction examples. Row 3 demonstrates the miss-classification example: the classifier failed to determine the attribute Heavy_makeup and Male of the face which expressed in black font.

Target Image	Attribute	Truth	Prediction
	Bald	−1	0.0029
	Bangs	−1	0.0007
	BlackHair	1	0.8225
	BlondeHair	−1	0.2986
	EyeGlass	−1	0.0142
	Male	1	0.8669
	Nobeard	1	0.7255
	Smiling	1	0.9526
	WaveHair	1	0.6279
	Young	1	0.6206
	Attractive	1	0.7758
	Bald	−1	0.1826
	Male	−1	0.00269
	Smiling	1	0.7352
	HeavyMakeUp	1	0.5729
	Wearinghat	1	0.7699
	Young	1	0.8015
	Attractive	−1	0.4629
	Bald	−1	0.6397
	EyeGlass	1	0.8214
	HeaveMakeUp	**−1**	**0.7566**
	Male	**1**	**0.3547**

Fig. 5. Comparison of rebuilding images through different losses. The first column shows the original nature-image. The 2nd and 3rd columns are images rebuilt with only per-pixel loss or **z**_loss (latent code). Column 4 shows the effect of **z**_loss + pixel_loss effect. The last column shows the final effect of the integrated loss. The FaceNet scores below each image in 2nd to 5th columns reveals the distance from the target image. Images rebuild from integrated loss(the last column) get the smallest score(expressed in black font).

pixel loss, and latent code loss(**z**_loss) + per-pixel loss + perceptual loss.

The comparisons among the results of the 4 experiments are shown in Fig. 5. The First column displays the original images to be rebuilt. The remains are corresponding to the 4 groups of experiments mentioned before.

From column 2 to column 5, we have three visual observations:

- latent code loss (**z**_loss) prefers to preserve image quality (images reconstructed with per-pixel loss in column 2 are blurrier

Fig. 6. Four examples (male, glasses, bangs, and bald as shown in the four rows) of face transformation with a single attribute changing. For each example (row), we display five illustrations. For instance, the first row shows the results of controlling the label of the male. The odd columns of row 1 are the given faces, the corresponding even columns display the faces with male reversed. From the 1st and 5th instances (columns 1,2 and 9,10), we clearly see that the mustache disappeared from male to female.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

than images reconstructed with only the latent code loss in the 3rd column) because the calculation of per-pixel loss over the whole pixel set flattens the image;
- images reconstructed with per-pixel loss, latent code loss, latent code loss + per-pixel loss all failed on preserving the face identity;
- the integrated loss benefits the effects of its three components that reconstruct the original faces with high quality and identity-preserving.

FaceNet also calculates an identity-preserving score for each rebuild face as shown in Fig. 5 (from column 2 to column 5). A smaller score indicates a closer relationship between two images. The scores from column 2 to column 5 demonstrate that the faces reconstructed by the integrated loss preserve better facial identity than the faces rebuild with other losses (column 2 to 4) in most of the cases. In other words, the integrated loss not only has an advantage on produce high-quality image but also can make a good facial identity-preserving. These experiments prove that the iterative GAN provided with integrated loss has a strong ability to deal with the tasks mentioned above and archives similar or better results than previous work.

5.3.3. Face transformation with controllable attributes

Based on our framework, we feed the discriminator **D** a real image without its attribute labels **c**, then we get a noise

Fig. 7. Demonstrations of rebuilding the target images and reversing (modifying) some attributes of the original face simultaneously. The first column shows the target face. The rebuilt faces are shown in the 2nd column with all attributes unchanged. Then we reverse the 3 single labels successively from 3rd column to the 7th column. For example, the target face has the attribute 'Bangs' (column 3), we reverse the corresponding label 'Bangs' to eliminate this attribute and keep others fixed. The last 2 columns show the combination of attributes modification.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

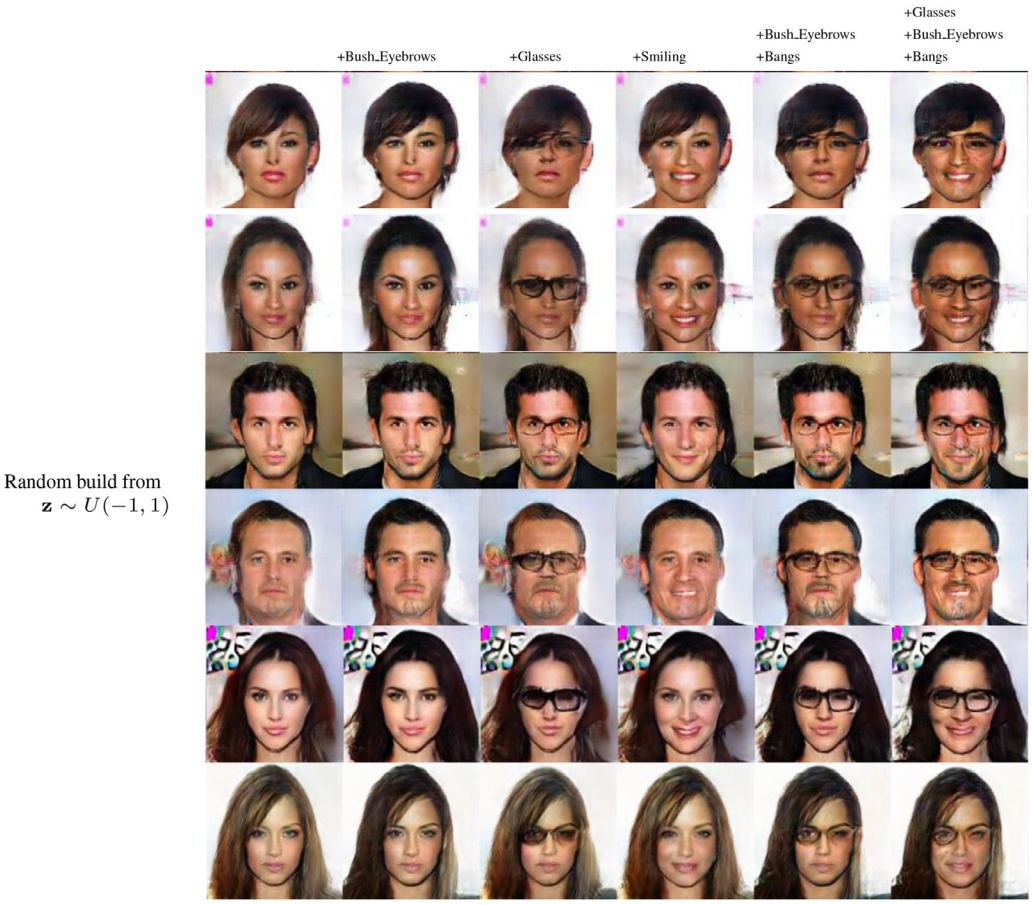


Fig. 8. Demonstration of building faces. The 1st column is faces build from random noise **z** which is sampled from the Uniform distribution. The 2nd to 4th columns show the standard faces created with the noise vector **z** plus single label such as Bush_Eyebrows, Glasses, Smiling. The left 2 columns are the examples of manipulating multiple attributes.

representation $\tilde{\mathbf{z}}$ and label vector $\tilde{\mathbf{c}}$ from **C** as output. We can reconstruct the original image with $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{c}}$ as we did in Section 5.3.2. Alternatively, we can transform the original image into another one by customizing its attributes in $\tilde{\mathbf{c}}$. By modifying part or even all labels of $\tilde{\mathbf{c}}$, the corresponding attributes of the reconstructed image will be transformed.

In this section, we study the performance of image transformation of our model. We begin the experiments with controlling a single attribute. That is, to modify one of the attributes of the images on the test set. Fig. 6 shows the part of the results of the transformation on the test set. The four rows of Fig. 6 illustrate four different attributes ***Male, Eye_Glasses, Bangs, Bald*** have been changed. The odd columns display the original images, and even columns illustrate the transformed images. We observe that the transformed images preserve high fidelity and their old attributes.

Finally, we extend it to the attribute manipulation from the single case to the multi-label scenario. Fig. 7 exhibits the results manipulating multiple attributes. The first column is the target faces. Faces in column 2 are the corresponding ones that reconstructed by iterative GAN (no attributes have been modified). The remaining 5 columns display the face transformation (column 3,4,5: single attribute; column 6,7: multiple attributes). For these transformed faces, we observe that both the image quality and face identity preserving are well satisfied. As we see, for the multi-label case, we only modified 3 attributes in the test. Actually, we have tried to manipulate more attributes (4–6) one time while the image quality drastically decreased. There is a lot of things to improve and we left it as the future work.

5.3.4. Compare with existing method of image transformation

Image transformation puts the emphases on finding a way to map the original image to an output image which subjects to another different domain. Editing the image attributes of a person is a special topic in this area. One of the popular methods attracted lots of attention recently is the CycleGAN [6]. The key point of CycleGAN is that it builds upon the power of the PIX2PIX [9] architecture, with discrete, unpaired collections of training images. Besides, another recent famous work in this area is StarGAN [18] which focuses on transforming facial attributes from multi-domain dataset, and obtains the high-quality results that is state_of_art.

In this experiment, we compare CycleGAN and StarGAN with iterative GAN on face transformation. We randomly select three facial attributes the ***Bangs,Glasses***, and ***Bald*** for testing.

For CycleGAN, we split the training dataset into 2 groups for each of the three attributes. For example, to train CycleGAN with the ***Bangs***, we divide the images into 2 sets, faces with bangs belong to domain 1 and the ones without bangs belong to domain 2. According to the results shown in Fig. 9(*a*), we found that CycleGAN is insensitive to the geometry transformation though it did a good job in catching some different features between two domains like color. As we know, CycleGAN is good at transforming the style of an image, for example, translate a horse image to zebra one [6]. For the test of human faces, however, it fails to recognize and manipulate the three facial attributes ***Bangs,Glasses***, and ***Bald*** as shown in column 2 of Fig. As for the results of StarGAN, it makes quite excellent transformation through attributes and preserve the

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

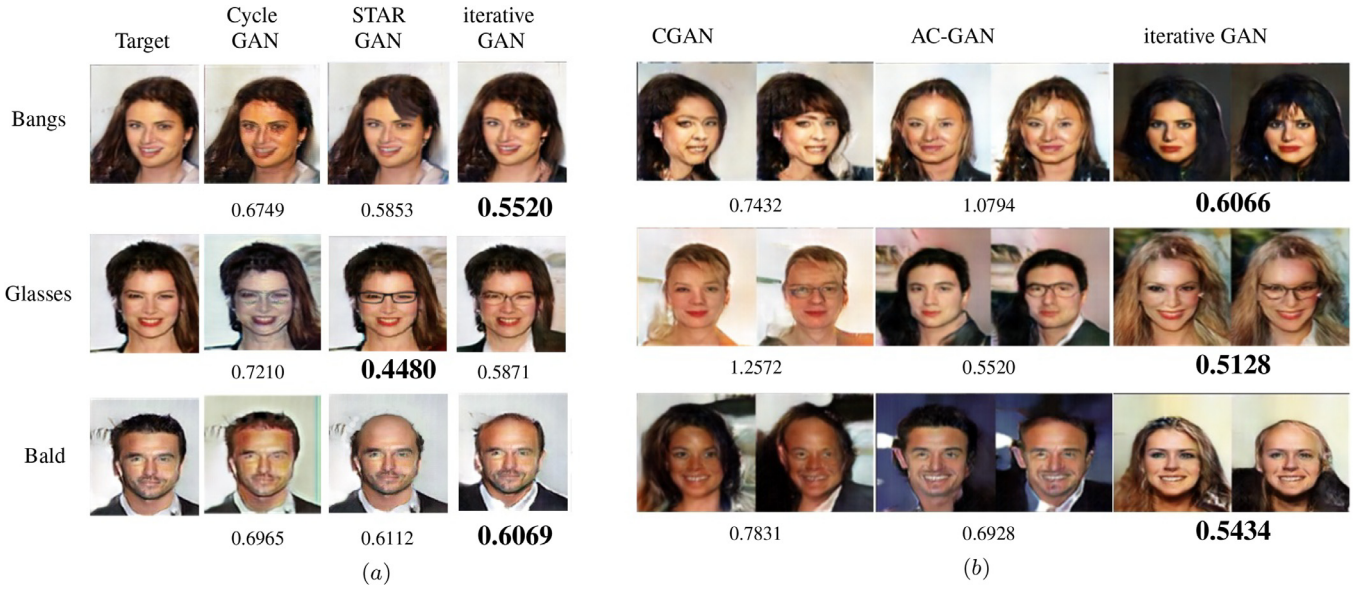


Fig. 9. Comparing with iterative GAN and other GANs on face generation and transformation. Part (*a*) shows the results of transforming given images to another with facial attributes changed. When Cycle GAN is performed, the quality of output is so poor since Cycle GAN seems to insensitive to the subtle changes of facial attributes and even failed in some labels(see the image in row 3 column 2). Part *b* compared the capability of generating an image with iterative GAN and CGAN, AC-GAN. The result shows that our model can produce the desired image with comparable or even better quality than AC-GAN, and far better than native CGAN. More importantly, according to the FaceNet scores(below each output image), it seems clear that the proposed iterative architecture has an advantage of preserving facial identity.

native facial attributes well. (9(*a*)). Compared with CycleGAN, the iterative GAN achieves better results in transforming the same face with one attribute changed and others preserved.Besides, we can recognize that for the same attribute to be transformed, the StarGAN and iterative GAN output almost the same quality image(with different pixel contents) in visual effect, and makes preservation in the same level indicated by FaceNet scores.

5.3.5. Face generation with controllable attributes

Different from above, we can also generate a new face with a random **z** sampling from a given distribution and an artificial attribute description **c** (labels). The generator **G** accepts **z** and **c** as the inputs and fabricates a fictitious facial image $\mathbf{x}_{\text{fake}}$ as the output. Of course, we can customize an image by modifying the corresponding attribute descriptions in **c**. For example, the police would like to get a suspect's portrait by the witness's description **"He is around 40 years old bald man with arched eyebrows and big nose"**. Fig. 8 illustrates the results of generating fictitious facial images with random noise and descriptions. We sample **z** from the Uniform distribution. The first column displays the images generated with **z** and initial descriptions. The remaining columns demonstrate the facial images generated with modified attributes (single or multiple modifications).

5.3.6. Compare with existing method of face generation

To examine the ability to generate realistic facial images of the proposed model, we compare the results of face generation of the proposed model with two baselines, the CGAN [16] and ACGAN [3], respectively. These three models can all generate images with a conditioned attributes description. For each of them, we begin the experiment by generating random facial images as illustrated in the 1st, 3rd, and 5th column of Fig. 9, part (*b*), respectively. The column 2, 4, and 6 display the generated images with the 3 attributes (***Bangs, Glasses, Bald*** modified for CGAN, ACGAN, and our model. It is clear to see that the face quality of our model is better than CGAN and ACGAN. And most importantly, in contrast with the failure of preserving face identity (see Fig. 9(*b*), the intersections between column 3, 4 and row 1 of ACGAN, column 1, 2 and row 2, 3 of CGAN), our model can always perform the best in face identity preserving.

It is clear to see that the face quality of ACGAN and our model is much better than CGAN. And most importantly, in contrast with the failure of preserving face identity (see the intersections between column 3, 4 and row 1, 2 of ACGAN), our model can always make a good face identity-preserving.

In summary, extensive experimental results indicate that our method is capable of: (1) Recognizing facial attribute; (2) Generating high-quality face images with multiple controllable attributes; (3) transforming an input face into an output one with desired attributes changing; (4)preserving the facial identity during face generation and transformation.

6. Conclusion

We propose an iterative GAN to perform face generation and transformation jointly by utilizing the strong dependency between the face generation and transformation. To preserve facial identity, an integrated loss including both the per-pixel loss and the perceptual loss is introduced in addition to the traditional adversarial loss. Experiments on a real-world face dataset demonstrate the advantages of the proposed model on both generating high-quality images and transforming image with controllable attributes.

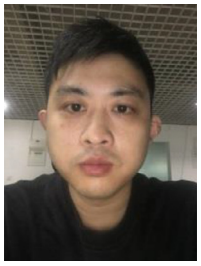
Acknowledgments

This work was partially supported by the Natural Science Foundation of China (61572111, G05QNQR004), the National High Technology Research and Development Program of China (863 Program) (No. 2015AA015408), a 985 Project of UESTC (No. A1098531023601041) and three Fundamental Research Fund for the Central Universities of China (Nos. A03017023701012, JBK120509, and JBK140507),

References

[1] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, D. Wierstra, Draw: a recurrent neural network for image generation, Proceedings of the International Conference on Machine Learning (2015) 1462–1471.

[2] C. Wang, C. Wang, C. Xu, D. Tao, Tag disentangled generative adversarial network for object image re-rendering, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, 2017, pp. 2901–2907.
[3] A. Odena, C. Olah, J. Shlens, Conditional image synthesis with auxiliary classifier GANs, Proceedings of the International Conference on Machine Learning (2017) 2642–2651.
[4] J. Zhu, P. Krahenbuhl, E. Shechtman, A.A. Efros, Generative visual manipulation on the natural image manifold, Proceedings of the European Conference on Computer vision (2016) 597–613.
[5] D. Yoo, N. Kim, S. Park, A.S. Paek, I.S. Kweon, Pixel-level domain transfer, Proceedings of the European Conference on Computer Vision (2016) 517–532.
[6] J. Zhu, T. Park, P. Isola, A.A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, Proceedings of the International Conference on Computer Vision (2017) 2242–2251.
[7] D. Mcquistonsurrett, L.D. Topp, R.S. Malpass, Use of facial composite systems in us law enforcement agencies, Psychol. Crime Law 12 (5) (2006) 505–517.
[8] C. Dong, C.C. Loy, K. He, X. Tang, Learning a deep convolutional network for image super-resolution, European conference on computer vision, 2014, pp. 184–199.
[9] A.V. Den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, K. Kavukcuoglu, Conditional image generation with pixelCNN decoders, Proceedings of the Neural Information Processing systems (2016) 4790–4798.
[10] X. Yan, J. Yang, K. Sohn, H. Lee, Attribute2image: conditional image generation from visual attributes, Proceedings of the European Conference on Computer Vision (2015) 776–791.
[11] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, Proceedings of the International Conference on Learning Representations (2014).
[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Advances in neural information processing systems, 2014, pp. 2672–2680.
[13] E.L. Denton, S. Chintala, A. Szlam, R.D. Fergus, Deep generative image models using a Laplacian pyramid of adversarial networks, Proceedings of the Neural Information Processing Systems (2015) 1486–1494.
[14] Z. Zhang, Y. Song, H. Qi, Age progression/regression by conditional adversarial autoencoder, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5810–5818.
[15] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, D. Samaras, Neural face editing with intrinsic image disentangling, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5541–5550.
[16] M. Mirza, S. Osindero, Conditional generative adversarial nets, 2014. arXiv preprint arXiv: 1411.1784.
[17] Z. Li, Y. Luo, in: Generate identity-preserving faces by generative adversarial networks, 2017. arXiv preprint arXiv: 1706.03227.
[18] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, J. Choo, Stargan: unified generative adversarial networks for multi-domain image-to-image translation, Proceedings of the Computer Vision and Pattern Recognition (2018) 8789–8797.
[19] P. Isola, J. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, Proceedings of the Computer Vision and Pattern Recognition (2017) 5967–5976.
[20] J. Johnson, A. Alahi, L. Feifei, Perceptual losses for real-time style transfer and super-resolution, Proceedings of the European Conference on Computer Vision (2016) 694–711.
[21] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of International Conference on Computer Vision (ICCV), 2015, pp. 3730–3738.
[22] A. Farhadi, I. Endres, D. Hoiem, D. Forsyth, Describing objects by their attributes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009. CVPR 2009, 2009, pp. 1778–1785.
[23] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, S. Belongie, Visual recognition with humans in the loop, European Conference on Computer Vision, 2010, pp. 438–451.
[24] M.-E. Nilsback, A. Zisserman, Automated flower classification over a large number of classes, 2008 Sixth Indian Conference on Computer Vision, Graphics \& Image Processing, 2008, pp. 722–729.
[25] N. Kumar, A.C. Berg, P.N. Belhumeur, S.K. Nayar, Attribute and simile classifiers for face verification, 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 365–372.
[26] N. Cherniavsky, I. Laptev, J. Sivic, A. Zisserman, Semi-supervised learning of facial attributes in video, European Conference on Computer Vision, 2010, pp. 43–56.
[27] D. Parikh, K. Grauman, Relative attributes, 2011 International Conference on Computer Vision, 2011, pp. 503–510.
[28] V. Bettadapura, in: Face expression recognition and analysis: the state of the art, 2012. arXiv preprint arXiv: 1203.6722.
[29] T. Berg, P. Belhumeur, Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 955–962.
[30] L. Bourdev, S. Maji, J. Malik, Describing people: A poselet-based approach to attribute classification, 2011 International Conference on Computer Vision, 2011, pp. 1543–1550.
[31] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, international Conference on computer vision \& Pattern Recognition (CVPR'05), 1, 2005, pp. 886–893.
[32] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297.
[33] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, L.D. Bourdev, Panda: pose aligned networks for deep attribute modeling, in: Proceedings of the Computer Vision and Pattern Recognition, 2014, pp. 1637–1644.
[34] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: a unified embedding for face recognition and clustering, Proceedings of the Computer Vision and Pattern Recognition (2015) 815–823.
[35] D.P. Kingma, D.J. Rezende, S. Mohamed, M. Welling, Semi-supervised learning with deep generative models, Adv. Neural Inf. Process. Syst. 4 (2014) 3581–3589.
[36] J. Gauthier, Conditional generative adversarial nets for convolutional face generation, Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester 2014 (5) (2014) 2.
[37] X. Wang, A. Gupta, Generative image modeling using style and structure adversarial networks, Proceedings of the European Conference on Computer zvision (2016) 318–335.
[38] J. Xie, L. Xu, E. Chen, Image denoising and inpainting with deep neural networks, Advances in neural information processing systems, 2012, pp. 341–349.
[39] L.A. Gatys, A.S. Ecker, M. Bethge, A neural algorithm of artistic style, Proceedings of the Nature Communications (2015).
[40] M. Tatarchenko, A. Dosovitskiy, T. Brox, Multi-view 3D models from single images with a convolutional network, Proceedings of the European Conference on Computer Vision (2016) 322–337.
[41] R. Zhang, P. Isola, A.A. Efros, Colorful image colorization, Proceedings of the European Conference on Computer Vision (2016) 649–666.
[42] A.B.L. Larsen, S.K. Sonderby, H. Larochelle, O. Winther, Autoencoding beyond pixels using a learned similarity metric, Proceedings of the International Conference on Machine Learning (2016) 1558–1566.
[43] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, Proceedings of the International Conference on Learning Representations (2015).
[44] C. Wang, C. Xu, C. Wang, D. Tao, Perceptual adversarial networks for image–to-image transformation, IEEE Transactions on Image Processing 27 (8) (2018) 4066–4079.
[45] A. Dosovitskiy, T. Brox, Generating images with perceptual similarity metrics based on deep networks, Proceedings of the Neural Information Processing Systems (2016) 658–666.
[46] Q. Yang, P. Yan, Y. Zhang, H. Yu, Y. Shi, X. Mou, M.K. Kalra, Y. Zhang, L. Sun, G. Wang, Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss, IEEE Trans. Med. Imaging 37 (6) (2018) 1348–1357.
[47] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A.P. Aitken, A. Tejani, J. Totz, Z. Wang, et al., Photo-realistic single image super-resolution using a generative adversarial network, Proceedings of the Computer Vision and pattern Recognition (2016) 4681–4690.
[48] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S.E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, Proceedings of the Computer Vision and Pattern Recognition (2015) 1–9.
[49] Y. Zheng, X. Zhou, W. Sheng, Y. Xue, S. Chen, Generative adversarial network based telecom fraud detection at the receiving bank, Neural Netw. 102 (2018) 78–86.
[50] A. Radford, L. Metz, S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, Proceedings of the International Conference on Learning Representations (2016).
[51] D. Kingma, J. Ba, Adam: a method for stochastic optimization, in: Proceedings of the International Conference on Learning Representation, 2015.
[52] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, Proceedings of the Neural Information Processing Systems (2014) 1988–1996.



Dan Ma received the M.Sc. degree in the School of University of Electronic Science & Technology of China, in 2005. He is currently a Ph.D. student in the School of Computer Science and Engineering, University of Electronic Science and Technology of China, China. His research interests are machine learning, data mining,especially interested in GAN.

Bin Liu is an Assistant Professor with the Department of Statistics, Southwestern University of Finance and Economics, Chengdu, China. He received the Ph.D. degree in computer science from the University of Electronic Science and Technology of China. His research interests include machine learning and data mining.He was the recipient of the Best Student Paper runner up Award at the 8th Asian Conference on Machine Learning and the best paper candidate at the 33rd IEEE Global Communications Conference.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093

Zhao Kang obtained his M.S. degree in theoretical physics from Sichuan University, China. He got his Ph.D. degree in 05/2017, in Southern Illinois University Carbondale. Now he isan assistant professor in University of Electronic Science and Technology of China. His major research area is machine learning.

Jiayu Zhou received the Ph.D. degree in computer science from Arizona State University, Tempe, AZ,USA, in 2014. He is an Assistant Professor with the Department of Computer Science and Engineering at Michigan State University, East Lansing, MI, USA. His research interests include large scale machine learningand data mining, and biomedical informatics. Prof. Zho served as Technical Program Committee Member of premier conferences such as NIPS, ICML, and SIGKDD. He was the recipient of the Best Student Paper Award at the 2014 IEEE international conference on data mining (ICDM) and the Best Student Paper Award at the 2016 International Symposium on Biomedical Imaging (ISBI).

Jianke Zhu received the Ph.D. degree in computer science from Zhejiang University in China. He is currently a associate professor in Zhejiang University.His research interests include computer vision and machine learning.

Zenglin Xu received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong. He is currently a full professor in University of Electronic Science & Technology of China. He has been working at Michigan State University, Cluster of Excellence at Saarland University and Max Planck Institute for Informatics, and later Purdue University. Dr. Xu's research interests include machine learning and its applications in information retrieval, health informatics, and social network analysis. He has been elected in the 2013's China Youth 1000-talent Program. He is the recipient of the outstanding student paper honorable mention of AAAI 2015, the best student paper runner up of ACML 2016, and the 2016 young researcher award from APNNS.

Please cite this article as: D. Ma, B. Liu and Z. Kang et al., Two birds with one stone: Transforming and generating facial images with iterative GAN, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.093