Disentangled Feature Networks for Facial Portrait and Caricature Generation

Kaihao Zhang, Wenhan Luo, Lin Ma, Wenqi Ren, and Hongdong Li

***Abstract*—Facial portrait is an artistic form which draws faces by emphasizing discriminative or prominent parts of faces via various kinds of drawing tools. However, the complex interplay between the different facial factors, such as facial parts, background, and drawing styles, and the significant domain gap between natural facial images and their portrait counterparts makes the task challenging. In this paper, a flexible four-stream Disentangled Feature Networks (*DFN*) is proposed to learn disentangled feature representation of different facial factors and generate plausible portraits with reasonable exaggerations and richness in style. Four factors are encoded as embedding features, and combined to reconstruct facial portraits. Meanwhile, to make the process fully automatic (without manually specifying either portrait style or exaggerating form), we propose a new Adversarial Portrait Mapping Module (*APMM*) to map noise to the embedding feature space, as proxies for portrait style and exaggerating. Thanks to the proposed *DFN* and *APMM*, we are able to manipulate the portrait style and facial geometric structures to generate a large number of portraits. Extensive experiments on two public datasets show that our proposed methods can generate a diverse set of artistic portraits.**

***Index Terms*—Facial portraits, facial caricature, four-stream disentangled feature networks, adversarial portrait mapping modules.**

I. Introduction

A facial portrait can be defined as an art form by which certain striking characteristics are represented through artistic drawings like sketching or pencil strokes. Though portraits are created based on natural facial images, they are more creative. The exaggerated features of **a** person applied in portraits give them the ability to express certain emotions or become a lovely profile image. It has become an increasingly active research topic in the field of computer vision, as it plays an important role in entertainment.

There have been a few studies to synthesize facial portrait. Early studies require professional skills to produce expressive results [1], [2], [3], [4]. For example, Akleman *et al.* [2] propose a technique of interactive 2D deformation to produce portraits with extreme exaggerations. The simplex primitive defining local coordinates from user is used to estimate a translation vector in 2D space. Similarly, portraits with exaggerated expression is deformed from the black-and-white facial illustrations with an interactive technique in [4]. Later, some automatic or semi-automatic systems are developed [5], [6], [7]. Specially, Liang *et al.* [5] generate the portraits via two steps including exaggeration and texture style transferring. They use a prototype-based method to exaggerate facial expression via focusing on distinctive features of faces. However, they are typically restricted to be applicable to a special drawing style, such as sketch, a special cartoon or outline tools. Based on computer softwares, artists can also create realistic portraits through warping the natural facial photos and transferring the drawing style [8]. However, the lacking of vivid details makes their results unappealing.

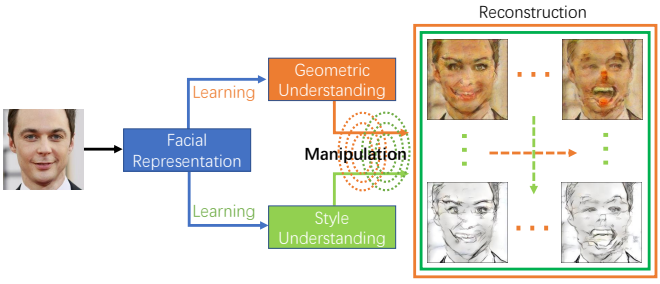


Fig. 1. **Integrating Geometric and Styles Understanding towards Facial Portraits Generation**. Natural facial images are put into networks to learn two kinds of information to represent geometric and style information. Then both of them are modified and then reconstruct facial portraits images. Along with the yellow dotted lines, we can generate portraits with different expression. Along with the green dotted lines, portraits with different style information are generated. Inside the *APMM*, two style and landmarks mapping modules are proposed to map the two kinds of information from noise. In this way, *APMM* can automatic generate different kinds of portrait for people to choose. Best viewed in color.

In recent years, deep learning methods have been successfully used in image-to-image translation [9], [10], [11], [12], [13], [14] and facial analysis [15], [16], [17]. Naively, one can adopt such a kind of methods for transformation from natural image and portrait image. However, most of these methods cannot handle the nature that most natural facial images and portrait samples are unpaired. Even though some pair images are of the same identity, they exhibit different poses or expressions [18]. Because of this, the typical GAN based methods such as Pix2Pix [19] and CycleGAN [20] are difficult to generate plausible **portraits** with both reasonable emotion changes and artistic styles.

Though GAN based methods and auto-encoder may not be able to directly applicable to generating portraits, their success in image generation inspires us to approach this problem based on them. In this paper, a disentangled feature network, *DFN*, of four streams based on conditional generative adversarial

K. Zhang and H. Li are with the College of Engineering and Computer Science, the Australian National University, Canberra, ACT, Australia. E-mail: {kaihao.zhang@anu.edu.au; hongdong.li@anu.edu.au}.

W. Luo is with the Tencent, Shenzhen 518057, China. E-mail: whluo.china@gmail.com

L. Ma is with Meituan, Beijing 100000, China. E-mail: forest.linma@gmail.com

W. Ren is with State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, China. E-mail: rwq.renwenqi@gmail.com.

networks [21] is proposed to generate portraits with reasonable emotion changes and artistic styles. In the proposed approach, an input facial image is disentangled into three streams which include face landmarks, prominent facial organs, and face textures, and then combined to reconstruct the facial image. Similar to traditional conditional GAN models which have an additional condition to guide GAN to generate images, the condition to control drawing styles is extracted from another image, *i.e.*, a given portraits, as the fourth stream. The changing of facial parts is controlled by landmarks which are extracted from the input facial image in the training stage. From the perspective of flexibility, one commonly requires more control over the generation of portraits (*e.g.*, changes of facial parts and drawing styles). Our proposed *DFN* framework thus enables to control them via changing the landmarks or providing different stylizing images.

Furthermore, in order to generate diverse portraits more freely without specifying stylizing image and landmark configurations, two adversarial portraits mapping module, *APMM*, are proposed to map Gaussian noise to the space of drawing styles or factors which control the emotion of exaggeration. Different from the traditional auto-encoder framework which has one encoder and one decoder, the proposed *APMM* has one encoder, one generator, one decoder and one discriminator. As mentioned above, the artistic style comes from an additional portrait image. The proposed *DFN* framework extracts the style information contained in this image. After that, the style information is encoded into embedding features, which are then fed into the decoder to generate the input style. To get rid of the stylizing image, Gaussian noise is input to the generator to produce features of the same size as the embedding features above. Embedding features from style image and Gaussian noise are labeled as "real" and "fake", respectively, which are then sent into the discriminator. When the generator is able to fool the discriminator so that it fails to distinguish fake embedding from real one, it is endowed the ability of mapping Gaussian noise into latent space of styles. The fake embedding features are input into the decoder to produce style which can be sent into the *DFN* framework to generate portraits. Likewise, we can also use *APMM* module to create landmark configuration to control the changes of emotion. Based on *APMM*, we finally develop two switches in the *DFN* framework to control whether the generation is automatic or semi-automatic. As Figure 2 shows, when the style or exaggeration configurations are specified by given images, it is semi-automatic. When the configurations are from *APMM*, it turns to be automatic. Figure 1 shows the portraits generated by our proposed method.

We have conducted extensive experiments on public datasets. Ablation study is also carried out to investigate effectiveness of different modules in the approach. We also compare with existing state-of-the-art methods to verify our proposed method.

Our main contributions are three-fold.

- We develop a flexible four-stream disentangled feature networks, DFN, with each stream specifically controlling one factor of portraits in the process of generation. Through altering different landmarks and style, our DFN framework is able to generate plausible portraits with reasonable changes of facial parts.
- We propose an *APMM* module which can map Gaussian noise into latent space of drawing style and motion changes, which could help *DFN* generating portraits with different kinds of style and various motion automatically.
- Extensive experimental results on public datasets demonstrate the effectiveness of the proposed approach, which can generate abundant plausible portraits compared with state-of-the-art methods.

The rest of the paper is organized as follows. Related literature is discussed in Section II. Section III presents the proposed approach. Experimental study results are reported in Section IV. In the end, Section V draws conclusions of this paper.

II. Related Work

As our work is related to the specific tasks of portrait generation, style transfer, and generative adversarial networks, we discuss the existing studies as follows.

A. Portrait Generation

Nowadays, many works have been proposed to transfer natural facial images to artistic styles like portrait, cartoon or caricature. The main difference between them is portraits and cartoons pay more attention to various drawing styles than emotion changes, while the caricatures focus on emotion exaggeration. Roughly, these methods can be classified into two categories: graphic based methods and deep learning based methods. Early studies mainly focus on graphic based solutions. Some deformation systems for users with expert knowledge and experienced artists are proposed to manipulate photos interactively [2], [3], [4]. Brennan *et al.* [7] present an idea of EDFM which defines hand-craft rules to automatically modify emotion. Base on this idea, 2D [6], [22], [23] and 3D model [24], [8] are proposed to manipulate facial parts. Liang *et al.* [5] propose a method to learn rules from paired images directly based on partial least squares. The deformation capability of them is usually limited because only the representation space is shape modeled.

Some works try to use warping methods to enhance the spatial variability of neural networks. They first predict transformation parameters based on spatial transformer networks [25], [26] and then warp to obtain the final images. These methods cannot handle local warping because there are not enough global transformation parameters. Others aim to use a dense deformation field [27], which have to predict all the vertices during the warping process.

Deep learning methods have been successfully applied to computer vision, which pushes researchers to make attempts to generate artistic images based on convolutional neural networks. Cao *et al.* [28] propose a framework based on CycleGAN to model facial information and then exaggerate then via warping methods. Li *et al.* [29] introduce a weakly paired training setting to train their proposed CariGAN model. Shi *et al.* [30] propose a GAN-based framework based on warping method to automaticaly generate caricature. Liu *et*

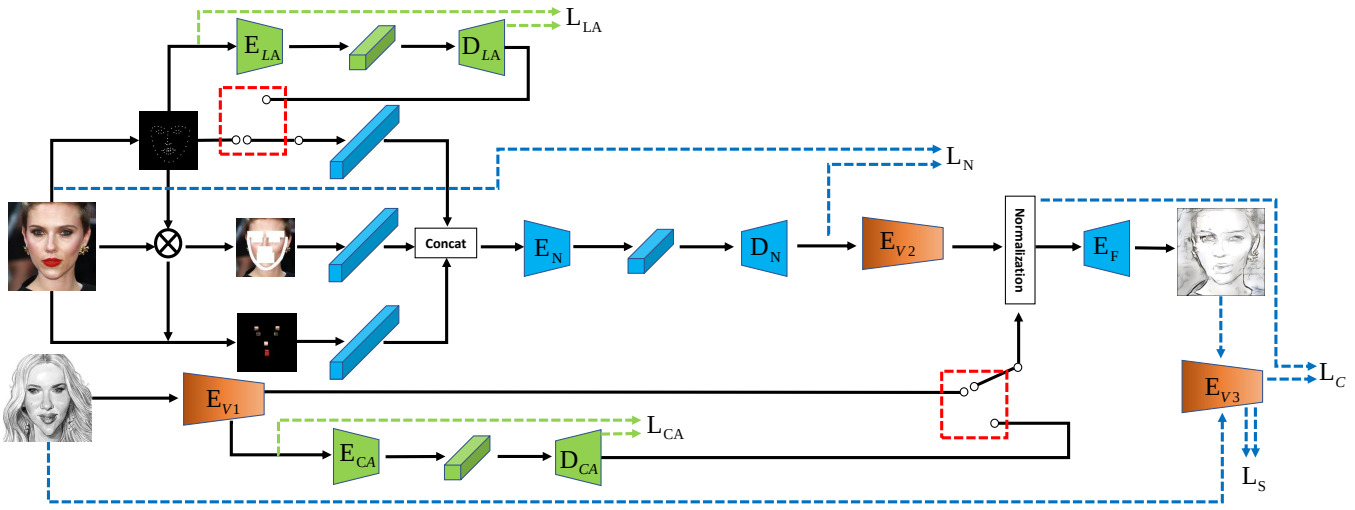


Fig. 2. **Disentangled Feature Network (DFN)**. The framework is composed of four streams: landmarks, face texture, prominent face organs, and artistic style. The first three streams are from natural facial image. Facial landmarks are input into CNN layers to obtain embedding features. Based on the facial landmarks and the given natural facial image, facial texture and important facial organs are obtained, which are also input into CNN layers to derive embedding features. These three kinds of features are concatenated and forwarded to an autoencoder architecture to reconstruct the input natural face image. The stylization cue comes from another caricature image. E_{V1}, E_{V2} are utilized to extract features of style and content information, which are normalized and input into E_F to create the final caricature. The black line is the direction of forward flow. The green and blue components are trained and their loss functions are calculated according to the corresponding dashed lines. The orange components are fixed. The green components are trained for the Adversarial Portrait Mapping Module (APMM) in Section III-C. The red bounding boxes show two switches, choosing how the style and landmark streams are configured in the deployment. One can specify style and landmarks by providing a caricature image and a specific configuration of landmarks, to generate caricature in a semi-automatic way. One can also switch to connect to the green components (D_{LA} and D_{CA}) using style and landmark embedding mapped from noise by APMM, to create in an automatic manner.

al. generate cartoon images from sketch based on using conditional generative adversarial networks. In order to change emotion, Wu *et al.* [31] introduce a landmark assisted CycleGAN to generate lovely facial images.

B. Style Transfer

Early methods of style transfer typically rely on low-level statistics and often include histogram specification on non-parametric sampling [32], [33]. Given the power of NN-based methods to extract features, Gatys *et al.* [34] for the first time propose a general method that uses a CNN method to transfer the style from the style image to any image automatically by matching features in convolutional layers. However, this method is an iterative methods, which requires rounds of iteration to apply the back propagated gradient to the original content image. This iterative procedure is time-consuming, up to several minutes for one image. To this end, Johnson *et al.* propose a convolutional network for the image style transfer network which requires only a single forward pass for the style transfer task by employing the perceptual loss based on a VGG-like network. This single pass network dramatically accelerates the task of style transfer. Then Gatys *et al.* [35] introduce a new method to control the color, the scale and the spatial location. There is still one limitation of these studies, *i.e.*, for a specific style (given by a style image), a network is specifically trained for this style, which cannot be used to stylize image to another style. Huang and Belongie [36] thus propose to develop a network based on instance normalization to transfer image with arbitrary style. In addition, several methods have been proposed by other researchers to improve the performance [37], [38], [39]. Li *et al.* [37] develop a method to enforce local pattern in the deep feature space based on Markov Random Field. The application of style transfer has also extend to videos. Reduer *et al.* [38] impose temporal constraints to improve the performance of video style transfer. However, this approach is also iterative-style one which is time-consuming. In light of this, Huang *et al.* [40] propose to learn a network which stylizes the video frame by frame with single forward pass. To ensure the spatio-temporal consistency, a constraint that, the motion in terms of optical flow between two continuous frames before stylization and after the stylization should be identical, is enforced in the training of the network. In inference, real-time performance is achieved with a single forward pass through the network and also the spatio-temporal consistency is ensured. Similarly, nearly real-time performance is also achieved by Chen *et al.* in [41]. Both short-term and long-term consistency are enforced in their approach.

C. Generative Adversarial Networks

The seminal work [42] introduce the generative adversarial networks (GAN) to the community for the generative task. In the framework GAN, there are typical two nets, *i.e.*, one is the generator and the other one is the discriminator. The generator tries to produce samples which approximate the distribution of real data samples. The discriminator distinguishes between the real samples and the fake samples produced by the generator. With the minmax game between the generator and the discriminator, a equilibrium is achieved and the derived generator is optimal for the targeted generative task. As a following work,

the conditional GAN [21], [43], [44] is proposed to generate various samples with different conditions.

The paradigm of GAN has inspires many applications [19], [45], [46], [47]. For example, Isola *et al.* [19] propose a Pix2Pix framework to achieve the tasks of photo-label, photo-to-map, and sketch-to-photo, under the supervision of pair images. Zhu *et al.* [45] introduce a BicycleGAN to ensure that the model has the ability to create different outputs. However, these methods require paired samples from two different domains, which are expensive to obtain. To this end, CycleGan [20] is proposed by Zhu *et al.* to alleviate the requirement of paired data. Samples are transformed from one domain to the other domain, and then transformed back to its original domain, and a cycle consistency loss is minimized to make sure a sample is close to its cycle-transformed version. Several studies have also been conducted following this strategy, such as DualGAN [12], DiscoGAN [10], UNIT [11] and DTN [48]. Another strategy popularly utilized in GAN is the coarse-to-fine one. Multiple stages are composed in the methods like StackGAN [49] by Zhang *et al.* and LAPGAN [50] by Denton *et al.*, to refine the generative results from the previous stage. GAN has also been employed for the video related generation task such as video prediction [46], [51], [15], video deblurring [52]. These studies achieve surprisingly good performance for their specific generation tasks, which motivates us to exploit Generative Adversarial Networks for portrait generation.

III. Approach

Our goal is to generate plausible portraits with reasonable emotion manipulation and styles. To achieve this goal, we disentangle different facial factors of facial images. These factors control the facial emotion and drawing style. Through manipulating these factors, we can generate facial portraits with freedom. Figure 2 shows the process of generation. As this figure shows, four factors, facial landmarks, textures, salient face organs, and artistic style are disentangled. In order to generate portraits with rich drawing styles and various patterns, a module called Adversarial Portrait Mapping Model, *APMM*, is proposed to map noise to embedding features, which can be utilized to replace the real features of style or exaggeration. In this section, we first introduce the method of disentangled natural image reconstruction, then discuss how to transfer drawing style. We represent the module of *APMM*, followed by how the generation is conducted in deployment.

A. Disentangled Natural Image Reconstruction

We propose a four-stream network to disentangle facial factors including the facial landmarks, facial textures, face organs, and drawing style. Among these streams, the first three are utilized in the reconstruction of a given natural face image, which is shown as the upper part (till the E_{V2} component) of Figure 2.

Suitable training data is important for increasing the performance of the framework. To train our model, we obtain the 68 landmark points of a given natural facial image. We take them as the input of the first stream. Based on the landmarks, we wipe away five nuclear parts including eyes, eyebrows, nose, mouth and cheeks, to obtain the stream of facial textures. Then, we create an image which contains only prominent organs of eyes, eyebrows, nose and mouth, as the stream of facial organs. These three kinds of streams, as shown in Figure 3, are input into CNN layers to extract features. We concatenate the extracted features and input them into an encoder-decoder structure to reconstruct the given natural facial image. Apart from these components, which are shown in blue, the landmarks are input into an additional U-Net model, E_{LA} and D_{LA}, to reconstruct themselves. This U-Net structure is trained in this stage, but it is prepared for *APMM* in Section III-C.

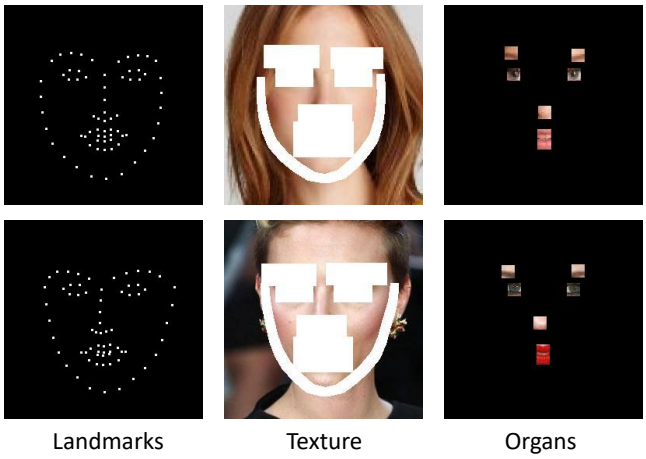


Fig. 3. **The input of three different streams**. We first detect 68 landmarks, shown in the left. Then we extract different parts of natural face image and obtain the middle and right images. The landmarks control the various of emotion, and the texture and face organs serve as important reference for generation.

Network Architecture. Inspired by recent image reconstruction studies, our reconstruction network is based on encoder-decoder architecture like the U-Net [53]. The input to the upper part of our framework is a RGB image of size $244 \times 244 \times 3$. A binary landmark image describing the structure of facial salient points is obtained based on the input image. By the landmark image, an incomplete RGB image with some removal regions, and a RGB image which includes only some important face organs are derived. These two images along with the binary landmark image serve as individual input to three streams of convolutional layers to extract features. The extracted features are then concatenated as a whole and input into a U-Net network. All convolutional layers use convolution-BatchNorm-ReLu block with kernels of 3×3. Skip connections [54], [55] are applied between the convolutional and deconvolutional layers.

Loss Functions. The natural image reconstruction process is optimized with both content loss and adversarial loss. The objective of the adversarial game is to reconstruct an image as realistic as possible to fool a discriminator. Thus the adversarial loss function can be defined as,

$$\begin{aligned}\mathcal{L}_a(E_N, D_N) =& \mathbb{E}_{x,y}[logD(x,y)]+ \\ & \mathbb{E}_{x,l,k}[log(1-D(x, D_N(E_N(x,l,k))))],\end{aligned} \tag{1}$$

where x, l and k are the input face texture, landmarks and facial organ images. y is a real (and complete) image. The U-Net architecture which includes D_N and E_N is applied to generate a realistic facial image, while an additional discriminator D is utilized to distinguish real images from fake images. Namely, D_N and E_N try to minimize the loss function against an adversary D which tries to maximize it.

In order to push the U-Net architecture to generate images which are close to the ground truth. We use L1 distance, denoted as,

$$\mathcal{L}_{L1}(E_N, D_N) = ||y - D_N(E_N(x, l, k))||. \tag{2}$$

The final loss in this part is,

$$\mathcal{L}_N = \min_{D_N, E_N} \max_{D} (\mathcal{L}_a(E_N, D_N) + \alpha \mathcal{L}_{L1}(E_N, D_N)). \tag{3}$$

B. Drawing Style Transfer

The bottom stream in Figure 2 shows the pipeline of our method for drawing style transfer. Our style transfer network takes a drawing style image s and the reconstructed output c of the previous section as input, and generates an output portraits which contains the content of c and the drawing style of s.

In this part, we also use the encoder-decoder architecture, in which the encoder is a fixed VGG-19 [56] model. After the content and drawing style images are encoded in latent space by the encoder, we feed them to an Instance Normalization layer which is used to align the channel-wise mean and variance of content feature to match style features. The traditional BN layer is originally proposed by [57] to accelerate training of CNN models. Radford *et al.* [58] find that it is also useful to generate images,

$$BN(c, s) = \lambda(\frac{c - \mu(c)}{\sigma(c)}) + \beta. \tag{4}$$

where BN normalizes the mean and standard deviation for input. λ and β are parameters learned from data.

In this word, in order to better combine the content and style input to generate portraits, the process of computing is employed as,

$$IN(c, s) = \sigma(s)(\frac{c - \mu(c)}{\sigma(c)}) + \mu(s). \tag{5}$$

As such, the normalized content features are scaled with $\sigma(s)$ and shifted with $\mu(s)$. Similar to [59], the $\sigma(s)$ and $\mu(s)$ are computed across spatial dimensions for each channel.

A decoder E_F is built to generate portraits based on the output of IN. This model is trained based on the loss function computed by a pre-trained VGG-19 [56] model E_{V3}, as

$$\mathcal{L} = \mathcal{L}_c + \beta \mathcal{L}_s, \tag{6}$$

where β is a style loss weight, which is utilized to balance the content loss $\mathcal{L}_c$ and drawing style loss $\mathcal{L}_s$. We use the Euclidean distance between the input and generated images to calculate content loss, which is denoted as

$$\mathcal{L}_c = ||E_{V3}(E_F(IN(c, s))) - IN(c, s)||. \tag{7}$$

The style loss [60] can be calculated as,

$$\begin{aligned}\mathcal{L}_s = \sum_{i=1}^{L} &||\mu(E^i_{V3}(E_F(IN(c, s))) - \mu(E^i_{V3}(E_F(s))))||_2 \\ &+ ||\sigma(E^i_{V3}(E_F(IN(c, s))) - \sigma(E^i_{V3}(E_F(s))))||_2,\end{aligned} \tag{8}$$

where i indexes layers in the E_{V3} model to compute the style loss. $E_F(s)$ represents the feature extracted from the style image s by the decoder E_F. In this paper, $relu1\_1$, $relu2\_1$, $relu3\_1$, $relu4\_1$ layers are employed in our practice. We use the style loss to control the style of generated caricatures. Specially, we use a pre-trained VGG-19 [56] to extract features from different layers. c and s mean an input batch and the index of style images.

Similar to the top part of DFN, a U-Net architecture, E_{CA} and D_{CA}, is trained to map features of drawing style to latent space and then transfer them back to the space of input. These green parts are also trained in this stage, but prepared for *APMM* in Section III-C. In summary, We use the disentangled reconstruction to modify the facial organs. In order to generated related realistic images. We use the GAN framework to update the parameters of the disentangled feature network. Features captured from the input facial images are encoded by E_N and then reconstructed to new facial images by D_N. We use L1 loss function to train our network even the L2 loss function achieve similar results in my experiments.

C. Adversarial Portrait Mapping Module

In order to generate caricature with more freedom, we propose an adversarial portrait mapping module, *APMM*. The schematic is shown in Figure 4, which produces learned features from sole noise. Images can be represented in a feature embedding space, thus we can use the learned features to replace the semantic style or emotion encoding during the generation process. Some previous works [49], [61] have made attempts to train another CNN based on the feature representations extracted from a pre-trained model. Our proposed *APMM* is inspired by this concept. We firstly train encoder-decoder models to embed features to latent feature space and transfer them back to the input space. Then a generator is trained to map Gaussian noise to a feature embedding space. We label these mapped features as "fake" and label the features extracted from input images as "real". The discriminator is used to distinguish the real features from fake ones. When the competition between the generator and the discriminator reaches equilibrium, the mapped features from noise is expected to be alternative of stylization or motion exaggeration from specified images. This endows us more freedom to generate caricatures providing a single natural face image.

The training process is shown in Algorithm 1. Both the style mapping network and the landmark mapping network are trained with two steps (steps (1)(3) for style and (2)(4) for landmarks). For each network, the first step is to train the encoder-decoder structure, shown in the green part in Figure 4, to reconstruct the style input or the landmarks. The second step is conducted to train a generator and a discriminator, shown in

Algorithm 1 The training procedure of *APMM*

(1) Pre-train E_{CA} and D_{CA}

- Sample n artistic facial images as input y from training set.
- Extract drawing style features $E_{V1}(y)$ through a pre-trained VGG-19 model.
- Calculate the reconstruction $D_{CA}(E_{CA}(E_{V1}(y)))$.
- Back propagate loss $\mathcal{L}_{CA}$.

(2) Pre-train E_{LA} and D_{LA}

- Sample n landmark images as input l from training set.
- Calculate the reconstruction $D_{LA}(E_{LA}(l))$.
- Back propagate loss $\mathcal{L}_{LA}$.

(3) Train the Generator with parameter θ_G and Discriminator with parameter θ_D for style

for # of iterations **do**

Update D:

- Sample n real style features $\mathbf{S}_{real}$ as a batch from the training set.
- Synthesize n fake style features $\mathbf{S}_{fake}$ by G.
- $\theta_D := \theta_D + \rho_D \nabla_{\theta_D} \frac{1}{n} \sum_{i=1}^{n} (\log D(\mathbf{S}^i_{real}) + log(1 - D(\mathbf{S}^i_{fake})))$

Update G:

- Sample n real style features $\mathbf{S}_{real}$ as a batch from the training set.
- Synthesize n fake style features $\mathbf{S}_{fake}$ by G.
- $\theta_G := \theta_G - \rho_G \nabla_{\theta_G} \frac{1}{n} \sum_{i=1}^{n} (\left\|\mathbf{S}^i_{real} - \mathbf{S}^i_{fake}\right\|_2^2 + \alpha_G \cdot \log(1 - D(\mathbf{S}^i_{fake})) + \gamma_G \cdot P(\mathbf{S}^i_{fake}) \log(Q(\mathbf{S}^i_{fake})))$

end for

(4) Train the Generator with parameter θ_G and Discriminator with parameter θ_D for landmarks

for # of iterations **do**

Update D:

- Sample n real landmark features $\mathbf{L}_{real}$ as a batch from the training set.
- Synthesize n fake landmark features $\mathbf{L}_{fake}$ by G.
- $\theta_D := \theta_D + \rho_D \nabla_{\theta_D} \frac{1}{n} \sum_{i=1}^{n} (\log D(\mathbf{L}^i_{real}) + log(1 - D(\mathbf{L}^i_{fake})))$

Update G:

- Sample n real landmark features $\mathbf{L}_{real}$ as a batch from the training set.
- Synthesize n fake landmark features $\mathbf{L}_{fake}$ by G.
- $\theta_G := \theta_G - \rho_G \nabla_{\theta_G} \frac{1}{n} \sum_{i=1}^{n} (\left\|\mathbf{L}^i_{real} - \mathbf{L}^i_{fake}\right\|_2^2 + \alpha_G \cdot \log(1 - D(\mathbf{L}^i_{fake})) + \gamma_G \cdot P(\mathbf{L}^i_{fake}) \log(Q(\mathbf{L}^i_{fake})))$

end for

the yellow part in Figure 4, for style or landmarks respectively. P and Q in the algorithm represent probability distributions.

The loss functions, $\mathcal{L}_{CA}$ and $\mathcal{L}_{LA}$ for style and landmarks reconstruction respectively, are defined as

$$\mathcal{L}_{CA} = ||D_{CA}(E_{CA}(E_{V1}(y))) - E_{V1}(y)||_2^2, \tag{9}$$

$$\mathcal{L}_{LA} = ||D_{LA}(E_{LA}(l))) - l||_2^2, \tag{10}$$

where y and l are the input style image and the landmark configurations image. E_{V1} is a fixed VGG-19 model, E_{CA} and D_{CA}, E_{LA} and D_{LA} are the encoders and decoders of U-Net structure for style and landmark mapping, respectively.

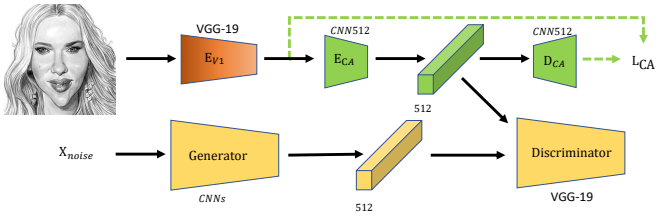
After the encoder-decoder network is trained, the generators and discriminators for style and landmark mapping can be trained thereafter. For the style mapping network, n real style features are sampled as a batch from the training set and input into the generator to synthesize fake style features S_{fake}. After adversarial learning between the generator and discriminator, our generator can create creative style. Likewise, we then train the other generator and discriminator for landmark mapping.

D. Deployment of Portraits Generation

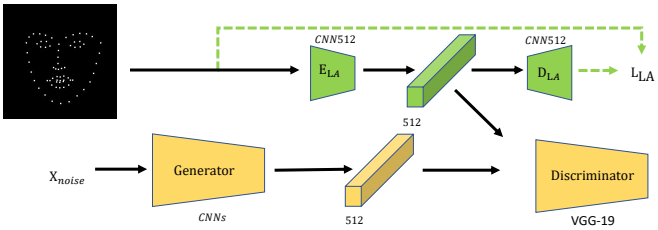
In testing, the generation can be deployed in either semi-automatic manner or automatic manner. As shown in Figure 2, given a natural face image, we can specify an additional portrait image and a landmark configuration as the style stream and the landmark stream to generate a portrait. This is semi-automatic. We can also replace the feature embedding output by E_{LA} and E_{CA} with the mapped results from noise by the generators in *APMM*. This is automatic as we do not need to provide style and landmarks specification.

IV. Experiments

To verify our model, we test our approach on a public dataset [18], which is introduced in Section IV-A. Implementation details are given in Section IV-B. We conduct an ablation study to investigate the effectiveness of different modules of our approach in Section IV-C. Then, we report the comparison with state-of-the-art methods in Section IV-D. Finally, we show exemplar qualitative results.



(a) Style mapping networks



(b) Landmarks mapping networks

Fig. 4. **Style and landmarks mapping modules**. It is trained with two steps. In the first step, the encoder and decoder are pre-trained in the proposed *DFN*. In the second step, generators and discriminators are trained. Taking style mapping as an example, the drawing style information is input into an encoder (E_{CA}) to produce embedding features. A noise sample is passed through a generator to produce feature representation, of the same size as the above embedding features. The discriminator is used only at the training stage to push the generator to map noise to representation which can hardly be discriminated from the real embedding features. E_{V1} is a fixed VGG network. The training of generator and discriminator for landmark embedding is carried out in the same way.

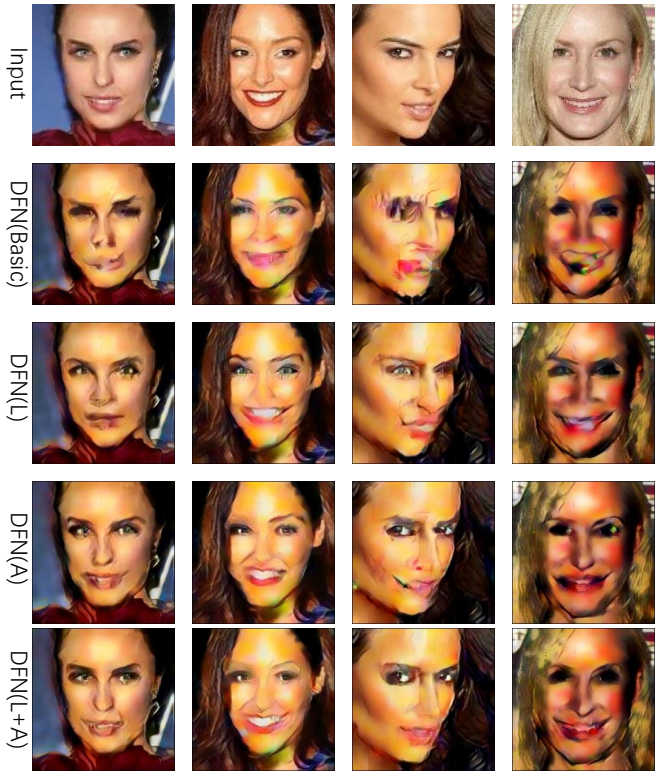


Fig. 5. **Ablation study results**. Comparisons between our proposed *DFN* model with its different variants. Rows from top to bottom present input facial images and their corresponding caricatures generated by *DFN*(Basic), *DFN*(L), *DFN*(A) and *DFN*(L+A), respectively.

A. Dataset

As the largest caricature dataset, the WebCaricature [18] contains $6,042$ caricatures and $5,974$ natural facial images from 252 identities. As discussed above, the main difference between portrait and caricature is that the former focuses on representing natural faces with various artistic styles, while the latter pays main attention to emotion exaggeration. Because the inputs of artistic images and photos are not required to be of the same identity, the proposed *DFN* learns the ability of understanding artistic information from Huo *et al.* [18] dataset, while the geometric information is learned from both **Huo** *et al.* and CelebA [63] datasets. The training set of **Huo** *et al.* dataset includes images of 200 identities, and the images from the rest 52 identities are used for testing. The CelebA dataset contains more than 200K celebrity images. There are many state-of-the-art facial landmark detectors. In this paper, we directly use a popular facial tool [64] to detect facial landmark. This is a public C++ Library, which can directly be used to detect. In practice, it also can be replaced by any latest landmark detectors. These images are then aligned based on the locations of eyes through similarity transformation and resized to 256×256. To augment the dataset, 244×244 patches are cropped at random locations of a facial image and randomly mirrored.

B. Implementation Details

During training, model weights are initialized by sampling from a normal distribution with zero mean and a standard deviation of 0.01. We update all weights with a mini-batch of size 2 in each iteration. The momentum and weight decay is set as 0.9 and 0.005, respectively. The model is trained with an annealing learning rate scheme, starting with 10^{-4}, and decreasing to 10^{-5} after convergence. α and β are set as 0.01 and 10, respectively. We use the PyTorch framework run on a PC with a Tesla M40 GPU.

C. Ablation Study

As our framework includes different components, in this section, we conduct experiments to investigate their effectiveness in generating portraits. Specially, we develop the following variants for the ablation study.

- ***DFN*(Basic)** is a traditional conditional GAN consisting of convolutional, pooling, and BN layers. The architecture is a two-stream version of Figure 2. One stream is the facial texture, which is shown in the center of Figure 3, and the other stream is the stylization from a given portrait image.
- ***DFN*(L)** is a variant of *DFN*. It is a three-stream architecture. Apart from the portrait image stream, it additionally includes landmarks and facial texture as another two streams, which are shown in the left and center of Figure 3, respectively.
- ***DFN*(A)** is also a variant of *DFN*. Different from *DFN*(L), this model replaces landmarks with the stream of facial organs. Thus, two streams of facial textures and organs, shown in the center and right of Figure 3

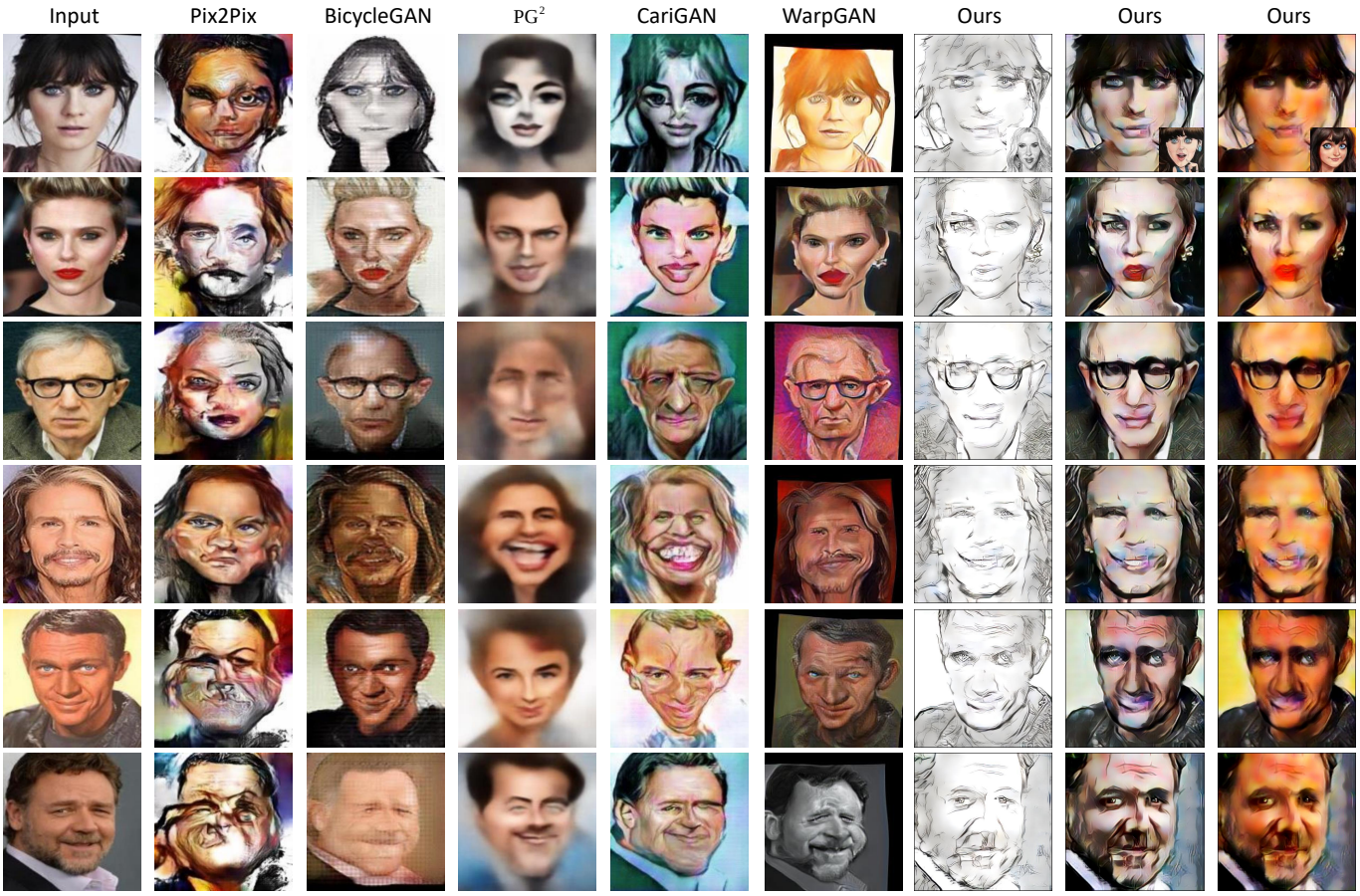


Fig. 6. **Comparison with existing state-of-the-art methods.** The leftmost column shows various input facial images. Columns to the right present caricature results generated by Pix2Pix [19], BicyleGAN [45], PG^2 [62], CariGAN [29], WarpGAN [30] and our proposed method with stylization from three different portraits (shown in the bottom-right corner of images in the first row).

respectively, along with the portrait stream are included in this counterpart.

- ***DFN*(L+A)** represents the whole framework of *DFN*. It is a four-stream architecture, which includes the landmarks, facial textures, facial organs, and portraits.

Figure 5 shows exemplar qualitative results of our *DFN* framework, *i.e.*, *DFN*(L+A) with four streams, and its variants, which are trained in two-stream, *i.e.*, *DFN*(Basic), and three-stream, *i.e.*, *DFN*(L) and *DFN*(A), respectively. Our proposed full model not only maintains the information of original images, but also generates visual-pleasing portraits. The landmarks can control the location of facial parts, which are important for the exaggeration during generation. The facial organ branch provides essential information for caricature generation. Through disentangling different factors, our proposed *DFN* can be trained based on unpaired images and endows users opportunities to create portraits freely.

Another reasonable baseline might be a network with the full non-disentangled facial image as input. However, the above ablation study has demonstrated that the quality of the generated images downgrades, when the parts of landmark and the organ are removed. This sufficiently verifies the importance of them. Moreover, the existing methods compared in the following sections can be treated as such baseline methods. So it is not necessary to compare this baseline in the ablation study here.

D. Comparison with Existing Methods

In this section, we show the comparison between different deep learning methods in Figure 6. Results of all methods are based on implementations provided by authors [29]. These methods include pixel-wise image translation approaches such as Pix2Pix [19], and ones creating diverse images from an identical image such as BicycleGAN [45]. In addition, a method proposed to generate human pose called PG^2 is compared. CariGAN [29] is a method proposed for facial caricature generation. WarpGAN [30] is a method which is able to create caricature images given face image preserving identity. It also provides customization of exaggeration extent and the visual styles. We also provide the results of our proposed *DFN* model based on fixed landmark configurations with different drawing styles.

As Figure 6 shows, different methods generate different kinds of caricatures or portraits, which cater to the preferences of different people. The proposed method provides a new approach to generate portraits, which is flexible and able to generate caricatures with the guidance of an input style.

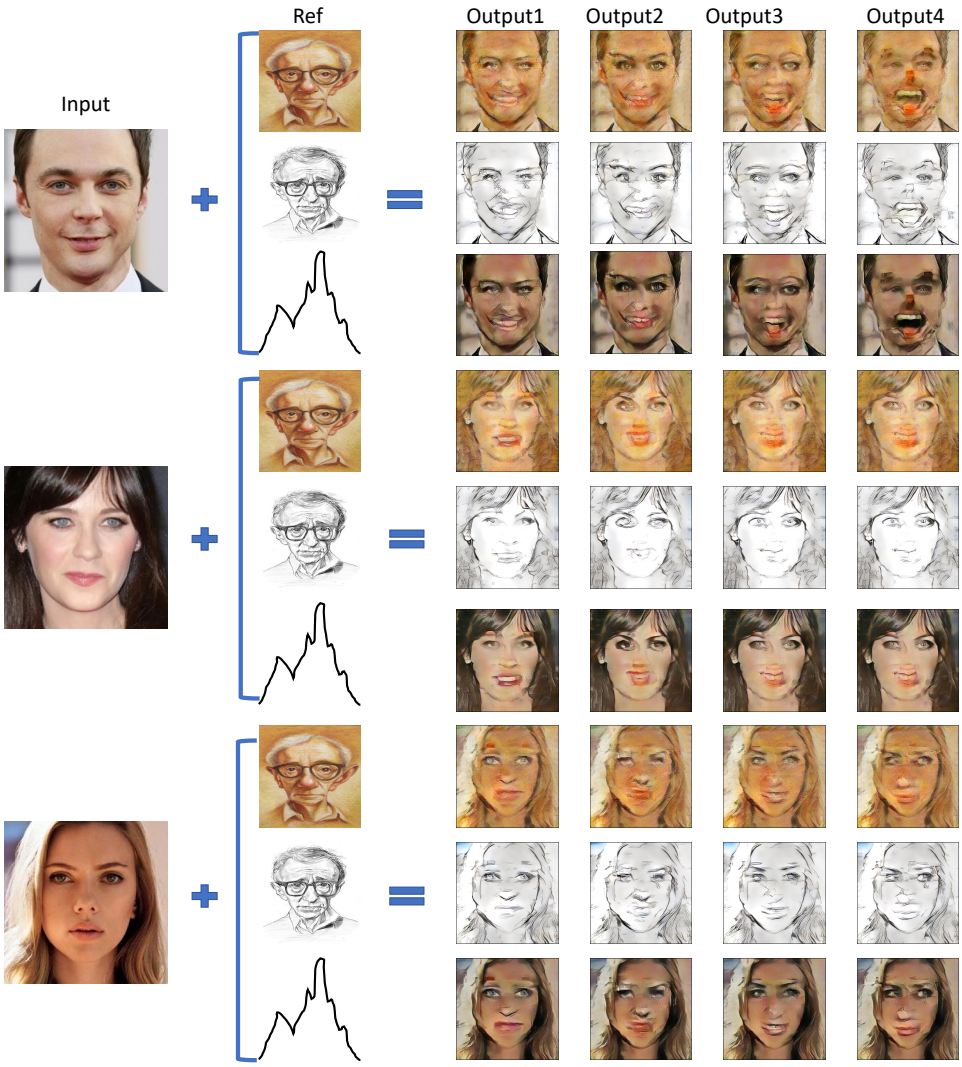


Fig. 7. **Examples of portraits generated based on reference or noise.** Three input faces are given in the leftmost column. The following portraits are generated by our proposed method. To the right, every three columns present results for one face in the leftmost column. Specifically, the first two columns shows examples generated with specified style images shown in the bottom-right corner. The third column in every three columns present portraits generated with stylization mapped from noise by *APMM*. For emotion, the portraits for the first two faces (column 2rd to 7th) are generated by specifying landmark configurations, while the remaining three columns for the third face are produced with random emotion mapped from noise by *APMM*. Our method generates portraits with plausible emotions.

Meanwhile, given that portraits are different from natural images, some generated portraits may suffer from artifacts.

One might observe that the results of our proposed method tend to lose tops of some heads. This is because in our method we use the face detection code, which is prone to ignore the top of head, consequently resulting in the lost of tops of some heads.

Figure 7 shows the portraits generated based on various landmark configurations and drawing styles. The leftmost column shows three individual exemplar faces. To the right, every three columns present the generation results of one face. In every three columns, the first two columns show results by providing two different style images (shown in the bottom-right corner in the first row). The third column in every three columns presents results with style solely from noise mapped by *APMM*. For the faces in the 2rd to 7th columns, emotion is specified by providing landmark configurations. For the last face in the 8th to 10th columns, emotion is provided by embedding mapped from noise by *APMM*. Our method is capable of generating visual-pleasing portraits based on reference or noise.

In summary, the small organs in our framework can provide reference for producing portrait images. As Figure 6 shows, the generated portrait images still keep the characters of the input nature versions like the color of eye and mouth. On the other hand, it avoids restrictions on the generated portraits via modifying the landmarks and styles. For example, the expression of smile can be exaggerated. In addition, the matching between portraits and natural facial images is an interesting task, which deserves in-depth study in future.

V. Conclusion

We have proposed a flexible disentangled feature network, textitDFN, to learn rich representation of different facial

factors and generate plausible portraits with reasonable emotion. By the obtained disentangled feature, we can control portrait generation by setting different landmarks and drawing styles. We have also proposed an automatic portrait generation procedure, *APMM*, which frees from the burden of setting the above parameters. It maps noise to style and landmark embedding features, improving even layperson's user experience. Experiments have demonstrated that our framework is able to generate various portraits without manual intervention.

ACKNOWLEDGMENT

This work is funded in part by the ARC Centre of Excellence for Robotics Vision (CE140100016), ARC-Discovery (DP 190102261) and ARC-LIEF (190100080) grants, as well as a research grant from Baidu on autonomous driving. The authors gratefully acknowledge the GPUs donated by NVIDIA Corporation. We thank all anonymous reviewers and editors for their constructive comments.

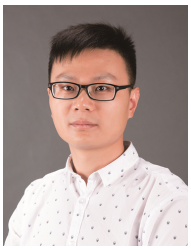
REFERENCES

[1] E. Akleman, "Making caricatures with morphing," in *ACM SIGGRAPH*, 1997.
[2] E. Akleman, J. Palmer, and R. Logan, "Making extreme caricatures with a new interactive 2d deformation technique with simplicial complexes," in *Proceedings of Visual*, 2000.
[3] H. Chen, N.-N. Zheng, L. Liang, Y. Li, Y.-Q. Xu, and H.-Y. Shum, "Pictoon: a personalized image-based cartoon system," in *Proceedings of the ACM International Conference on Multimedia*, 2002.
[4] B. Gooch, E. Reinhard, and A. Gooch, "Human facial illustrations: Creation and psychophysical evaluation," *ACM Transactions on Graphics*, 2004.
[5] L. Liang, H. Chen, Y.-Q. Xu, and H.-Y. Shum, "Example-based caricature generation with exaggeration," in *Pacific Conference on Computer Graphics and Application*, 2002.
[6] Z. Mo, J. P. Lewis, and U. Neumann, "Improved automatic caricature by feature normalization and exaggeration.," in *Siggraph Sketches*, p. 57, 2004.
[7] S. E. Brennan, "Caricature generator: The dynamic exaggeration of faces by computer," *Leonardo*, 2007.
[8] T. Lewiner, T. Vieira, D. Martínez, A. Peixoto, V. Mello, and L. Velho, "Interactive 3d caricature from harmonic exaggeration," *Computers & Graphics*, 2011.
[9] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proceedings of the European Conference on Computer Vision*, 2018.
[10] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," in *Proceedings of the International Conference on Machine Learning*, 2017.
[11] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," in *Advances in Neural Information Processing Systems*, 2017.
[12] Z. Yi, H. Zhang, P. Tan, and M. Gong, "Dualgan: Unsupervised dual learning for image-to-image translation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
[13] Y. Liu, W. Chen, L. Liu, and M. S. Lew, "Swapgan: A multistage generative approach for person-to-person fashion style transfer," *IEEE Transactions on Multimedia*, 2019.
[14] K. Zhang, W. Luo, W. Ren, J. Wang, F. Zhao, L. Ma, and H. Li, "Beyond monocular deraining: Stereo image deraining via semantic understanding," in *European Conference on Computer Vision*, 2020.
[15] W. Xiong, W. Luo, L. Ma, W. Liu, and J. Luo, "Learning to generate time-lapse videos using multi-stage dynamic generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
[16] K. Zhang, Y. Huang, Y. Du, and L. Wang, "Facial expression recognition based on deep evolutional spatial-temporal networks," *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4193–4203, 2017.
[17] K. Zhang, Y. Huang, C. Song, H. Wu, and L. Wang, "Kinship verification with deep convolutional neural networks," in *The British Machine Vision Conference*, 2020.
[18] J. Huo, W. Li, Y. Shi, Y. Gao, and H. Yin, "Webcaricature: a benchmark for caricature face recognition," in *The British Machine Vision Conference*, 2018.
[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.
[20] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
[21] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *Advances in Neural Information Processing Systems*, 2014.
[22] J. Liu, Y. Chen, and W. Gao, "Mapping learning in eigenspace for harmonious caricature generation," in *Proceedings of the ACM international conference on Multimedia*, 2006.
[23] C.-C. Tseng and J.-J. J. Lien, "Synthesis of exaggerative caricature with inter and intra correlations," in *Asian Conference on Computer Vision*, 2007.
[24] F. Yang, L. Bourdev, E. Shechtman, J. Wang, and D. Metaxas, "Facial expression editing in video using a temporally-smooth factorization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2012.
[25] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," in *Advances in Neural Information Processing Systems*, 2015.
[26] C.-H. Lin, E. Yumer, O. Wang, E. Shechtman, and S. Lucey, "St-gan: Spatial transformer generative adversarial networks for image compositing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
[27] Y. Ganin, D. Kononenko, D. Sungatullina, and V. Lempitsky, "Deepwarp: Photorealistic image resynthesis for gaze manipulation," in *European Conference on Computer Vision*, 2016.
[28] K. Cao, J. Liao, and L. Yuan, "Carigans: unpaired photo-to-caricature translation," in *SIGGRAPH Asia*, 2018.
[29] W. Li, W. Xiong, H. Liao, J. Huo, Y. Gao, and J. Luo, "Carigan: Caricature generation through weakly paired adversarial learning," *arXiv preprint arXiv:1811.00445*, 2018.
[30] Y. Shi, D. Deb, and A. K. Jain, "Warpgan: Automatic caricature generation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.
[31] R. Wu, X. Gu, X. Tao, X. Shen, Y.-W. Tai, and J. Jia, "Landmark assisted cyclegan for cartoon face generation," *arXiv preprint arXiv:1907.01424*, 2019.
[32] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *International Conference on Computer Graphics and Interactive Techniques*, 2001.
[33] D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis," in *The IEEE International Conference on Image Processing*, p. 3648, 1995.
[34] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
[35] L. A. Gatys, A. S. Ecker, M. Bethge, A. Hertzmann, and E. Shechtman, "Controlling perceptual factors in neural style transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
[36] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
[37] C. Li and M. Wand, "Combining markov random fields and convolutional neural networks for image synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2479–2486, 2016.
[38] M. Ruder, A. Dosovitskiy, and T. Brox, "Artistic style transfer for videos," in *German Conference on Pattern Recognition*, 2016.
[39] Y. Li, N. Wang, J. Liu, and X. Hou, "Demystifying neural style transfer," *International Joint Conference on Artificial Intelligence*, 2017.
[40] H. Huang, H. Wang, W. Luo, L. Ma, W. Jiang, X. Zhu, Z. Li, and W. Liu, "Real-time neural style transfer for videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
[41] D. Chen, J. Liao, L. Yuan, N. Yu, and G. Hua, "Coherent online video style transfer," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
[42] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014.

[43] Y. Guo, Q. Chen, J. Chen, Q. Wu, Q. Shi, and M. Tan, "Auto-embedding generative adversarial networks for high resolution image synthesis," *IEEE Transactions on Multimedia*, 2019.
[44] X. Guo, R. Nie, J. Cao, D. Zhou, L. Mei, and K. He, "Fusegan: Learning to fuse multi-focus image via conditional generative adversarial network," *IEEE Transactions on Multimedia*, 2019.
[45] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darrell, A. A. Efros, O. Wang, and E. Shechtman, "Toward multimodal image-to-image translation," in *Advances in Neural Information Processing Systems*, 2017.
[46] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *Advances In Neural Information Processing Systems*, 2016.
[47] K. Zhang, W. Luo, Y. Zhong, L. Ma, B. Stenger, W. Liu, and H. Li, "Deblurring by realistic blurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.
[48] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised cross-domain image generation," 2017.
[49] H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. N. Metaxas, "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017.
[50] E. L. Denton, S. Chintala, R. Fergus, *et al.*, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Advances in Neural Information Processing Systems*, 2015.
[51] C. Vondrick and A. Torralba, "Generating the future with adversarial transformers," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
[52] K. Zhang, W. Luo, Y. Zhong, L. Ma, W. Liu, and H. Li, "Adversarial spatio-temporal learning for video deblurring," *IEEE Transactions on Image Processing*, 2018.
[53] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015.
[54] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2016.
[55] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proceeding of European Conference on Computer Vision*, 2016.
[56] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *The International Conference on Learning Representations*, 2015.
[57] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning*, 2015.
[58] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *The International Conference on Learning Representations*, 2015.
[59] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.
[60] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*, 2016.
[61] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in *The International Conference on Learning Representations*, 2015.
[62] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose guided person image generation," in *Advances in Neural Information Processing Systems*, 2017.
[63] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015.
[64] "Dlib C++ Library:." http://dlib.net/.

Kaihao Zhang is currently pursuing the Ph.D. degree with the College of Engineering and Computer Science, The Australian National University, Canberra, ACT, Australia. He worked at the Center for Research on Intelligent Perception and Computing, National Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China for two years and the Tencent AI Laboratory, Shenzhen, China for two years. His research interests focus on image enhancement and face analysis with deep learning.

Wenhan Luo is currently working as a senior researcher in the Tencent AI Lab, China. His research interests include several topics in computer vision and machine learning, such as motion analysis (especially object tracking), image/video quality restoration, reinforcement learning. Before joining Tencent, he received the Ph.D. degree from Imperial College London, UK, 2016, M.E. degree from Institute of Automation, Chinese Academy of Sciences, China, 2012 and B.E. degree from Huazhong University of Science and Technology, China, 2009.



Lin Ma (M'13) received the B.E. and M.E. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 2006 and 2008, respectively, and the Ph.D. degree from the Department of Electronic Engineering, The Chinese University of Hong Kong, in 2013. He was a Researcher with the Huawei Noah'Ark Laboratory, Hong Kong, from 2013 to 2016. He is currently a Principal Researcher with the Tencent AI Laboratory, Shenzhen, China. His current research interests lie in the areas of computer vision, multimodal deep learning, specifically for image and language, image/video understanding, and quality assessment.

Dr. Ma received the Best Paper Award from the Pacific-Rim Conference on Multimedia in 2008. He was a recipient of the Microsoft Research Asia Fellowship in 2011. He was a finalist in HKIS Young Scientist Award in engineering science in 2012.

Wenqi Ren is an Associate Professor in Institute of Information Engineering, Chinese Academy of Sciences, China. He received his Ph.D. degree from Tianjin University, Tianjin, China, in 2017. During 2015 to 2016, he was supported by China Scholarship Council and working with Prof. Ming-Husan Yang as a joint-training Ph.D. student in the Electrical Engineering and Computer Science Department, at the University of California at Merced. He received Tencent Rhino Bird Elite Graduate Program Scholarship in 2017, MSRA Star Track Program in 2018. His research interests include image processing and related high-level vision problems.

Hongdong Li is currently a Professor with the Computer Vision Group of ANU (Australian National University). He is also a Chief Investigator for the Australia ARC Centre of Excellence for Robotic Vision (ACRV). His research interests include 3D vision reconstruction, structure from motion, multi-view geometry, as well as applications of optimization methods in computer vision. Prior to 2010, he was with NICTA Canberra Labs working on the "Australia Bionic Eyes" project. He is an Associate Editor for IEEE T-PAMI, and served as Area Chair in recent year ICCV, ECCV and CVPR. He was a Program Chair for ACRA 2015 - Australia Conference on Robotics and Automation, and a Program Co-Chair for ACCV 2018 - Asian Conference on Computer Vision. He won a number of prestigious best paper awards in computer vision and pattern recognition, and was the receipt for the CVPR Best Paper Award in 2012 and the Marr Prize Honorable Mention in 2017.