

Facial Synthesis From Visual Attributes via Sketch Using Multiscale Generators

Xing Di¹, *Student Member, IEEE*, and Vishal M. Patel¹, *Senior Member, IEEE*

Abstract—Automatic synthesis of faces from visual attributes is an important problem in computer vision and has wide applications in law enforcement and entertainment. With the advent of deep generative convolutional neural networks (CNNs), attempts have been made to synthesize face images from attributes and text descriptions. In this paper, we take a different approach, where we formulate the original problem as a stage-wise learning problem. We first synthesize the facial sketch corresponding to the visual attributes and then we generate the face image based on the synthesized sketch. The proposed framework, is based on a combination of two different Generative Adversarial Networks (GANs) – (1) a sketch generator network which synthesizes realistic sketch from the input attributes, and (2) a face generator network which synthesizes facial images from the synthesized sketch images with the help of facial attributes. Extensive experiments and comparison with recent methods are performed to verify the effectiveness of the proposed attribute-based two-stage face synthesis method.

Index Terms—Face synthesis, visual attributes, generative adversarial networks.

I. INTRODUCTION

FACIAL attributes are descriptions or labels that can be given to a face by describing its appearance [1]. In the biometrics community, attributes are also referred to as soft-biometrics [2]. Various methods have been developed in the literature for predicting facial attributes from images [3]–[9]. In this work, we aim to tackle the inverse problem of synthesizing faces from their corresponding attributes (see Fig. 1). Visual description-based facial synthesis has many applications in law enforcement and entertainment. For example, visual attributes are commonly used in law enforcement to assist in identifying suspects involved in a crime when no facial image of the suspect is available at the crime scene. This is commonly done by constructing a composite or forensic sketch of the person based on the visual attributes.

Reconstructing an image from attributes or text descriptions is an extremely challenging problem because the model is required to learn the mapping from a semantic abstract space

Manuscript received July 23, 2019; revised October 30, 2019 and December 16, 2019; accepted December 17, 2019. Date of publication December 21, 2019; date of current version January 24, 2020. This work was supported by the Office of the Director of National Intelligence through Intelligence Advanced Research Projects Activity under Contract 2019-19022600002. This article was recommended for publication by Associate Editor T. Boulton upon evaluation of the reviewers' comments. (Corresponding author: Xing Di.)

The authors are with the Whiting School of Engineering, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: xing.di@jhu.edu; vpatel36@jhu.edu).

Digital Object Identifier 10.1109/TBIOM.2019.2961926

2637-6407 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.



Fig. 1. Attribute prediction vs. face synthesis from attributes. (a) Attribute prediction: given a face image, the goal is to predict the corresponding attributes. (b) Face synthesis from attributes: given a list of facial attributes, the goal is to generate a face image that satisfies these attributes.

to a complex RGB image space. This task requires the generated images to be not only realistic but also semantically consistent, i.e., the generated face images should preserve the facial structure as well as the content described in attributes. Several recent works have attempted to solve this problem by using recently introduced CNN-based generative models such as conditional variational auto-encoder (CVAE) [10]–[12] and generative adversarial network (GAN) [13]–[18]. For instance, Yan *et al.* [11] proposed a disentangled CVAE-based method for attribute-conditioned image generation. In a different approach, Reed *et al.* [14] introduced a GAN-based method for synthesizing images from detailed text descriptions. Similarly, Zhang *et al.* [15] proposed the StackGAN method for synthesizing photo-realistic images from text.

It is well-known that CVAE-based methods often generate blurry images due to the injected noise and imperfect element-wise squared error measure used in training [19]. In contrast, GAN-based methods have shown to generate high-quality images [13]. In order to synthesize photo-realistic facial images, rather than directly generating an image from attributes, we first synthesize a sketch image corresponding to the attributes and then generate the facial image from the synthesized sketch. Our approach is motivated by the way forensic sketch artists render the composite sketches of an unknown subject using a number of individually described parts and attributes. Our approach is also inspired by the recent works [10], [20]–[23] that have shown the effectiveness of stage-wise training.

In particular, the proposed framework consists of two stages (see Fig. 2) – sketch generator network and face generator network. Given a noise vector sampled from the normal distribution and sketch attributes, the sketch generator learns to synthesize sketch images. In the second stage, given the synthesized sketch from the first stage, a different generator

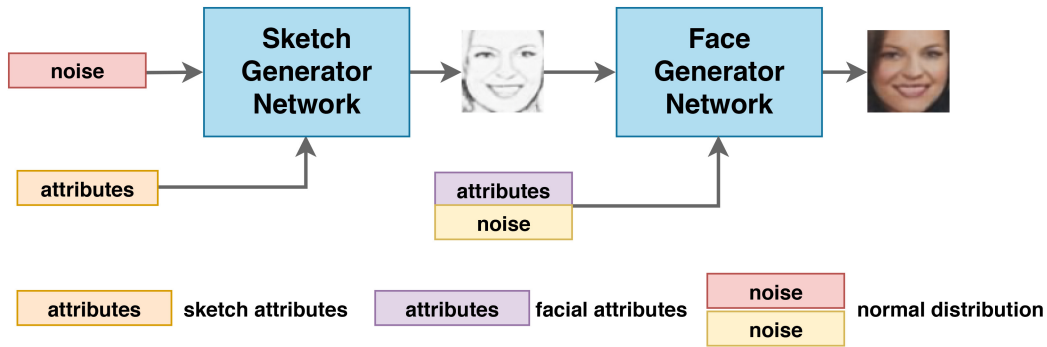


Fig. 2. An overview of the proposed synthesis method. Given a noise vector sampled from the normal distribution, the Sketch Generator Network synthesizes sketch image conditioned on the sketch attributes. The synthesized sketch is then given as an input to the Face Generator Network, which outputs the high-quality face images conditioned on the facial attributes.

network is trained to synthesize high-quality face images with the help of attributes. In particular, the attribute augmentation module is adapted from StackGAN [15] in both sketch and face generator networks. This module aims to increase the generation diversity by adding redundant information from the standard Gaussian noise. In experiments we observed that, due to the sparsity of visual attributes, the input attribute values become all zero when the input batch of attributes are all the same. In order to overcome this “attribute vanishing issue”, we replace the batch normalization layers with the conditional batch normalization layers [24]. We refer to this module as the attribute augmentation module.

To summarize, this paper makes the following contributions:

- We formulate the attribute-to-face generation problem as a stage-wise learning problem, i.e., attribute-to-sketch, and sketch-to-face. The synthesis networks are based on multi-scale generators with an attribute augmentation module for synthesizing photo-realistic images.
- For the face generator network, we propose a novel visual attribute conditioned sketch-to-face synthesis network. The network is composed of an attribute augmentation module and a UNet [25] shape translation network. With the help of the attribute-augmentation module, the training stability is improved and the generators are able to synthesize diverse set of realistic face/sketch images.
- Extensive experiments are conducted to demonstrate the effectiveness of the proposed image synthesis method. Furthermore, an ablation study is conducted to demonstrate the improvements obtained by different stages of our framework.

Rest of the paper is organized as follows. In Section II, we review a few related works. Details of the proposed facial composite synthesis from visual attribute method are given in Section III. Experimental results are presented in Section IV, and finally, Section V concludes the paper with a brief summary.

II. BACKGROUND AND RELATED WORK

Recent advances in deep learning have led to the development of various deep generative models for the problem of text-image synthesis and image-to-image translation [26], [12], [13], [27], [28], [10], [29], [30], [31], [32],

[33], [34], [35], [36], [37]. Among them, variational auto-encoder (VAE) [12], [27], [29], generative adversarial network (GAN) [13], [28], [32]–[35], [37], and Autoregression [26] are the most widely used approaches.

VAEs [12], [27] are powerful generative models that use deep networks to describe distribution of observed and latent variables. A VAE model consists of two parts, with one network encoding a data sample to a latent representation and the other network decoding latent representation back to data space. VAE regularizes the encoder by imposing a prior over the latent distribution. Conditional VAE (CVAE) [10], [11] is an extension of VAE that models latent variables and data, both conditioned on side information such as a part or label of the image. For example, Yan *et al.* [11] proposed a decomposing conditional VAE (disCVAE) model to synthesize facial image from visual attributes. They took the assumption that a face image could be decomposed into two parts: foreground and background. By taking this assumption, the disCVAE model is able to generate plausible face images with corresponding attributes. However, due to the imperfect element-wise square error measurements, the VAE model usually generates blurry images [19].

GANs [13] are another class of generative models that are used to synthesize realistic images by effectively learning the distribution of training images [38]–[40]. The goal of GAN is to train a generator G , to produce samples from training distribution such that the synthesized samples are indistinguishable from actual distribution by the discriminator, D . Conditional GAN is another variant where the generator is conditioned on additional variables such as discrete labels, text or images. The objective function of a conditional GAN is defined as follows

$$L_{cGAN}(G, D) = E_{\mathbf{x}, \mathbf{y} \sim P_{data}(\mathbf{x}, \mathbf{y})} [\log D(\mathbf{x}, \mathbf{y})] + E_{\mathbf{x} \sim P_{data}(\mathbf{x}), \mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(\mathbf{x}, G(\mathbf{x}, \mathbf{z})))]$$

where \mathbf{z} the input noise, \mathbf{y} the output image, and \mathbf{x} the observed image, are sampled from distribution $P_{data}(\mathbf{x}, \mathbf{y})$ and they are distinguished by the discriminator, D . While for the generated fake $G(\mathbf{x}, \mathbf{z})$ sampled from distributions $\mathbf{x} \sim P_{data}(\mathbf{x})$, $\mathbf{z} \sim p_z(\mathbf{z})$ would like to fool D .

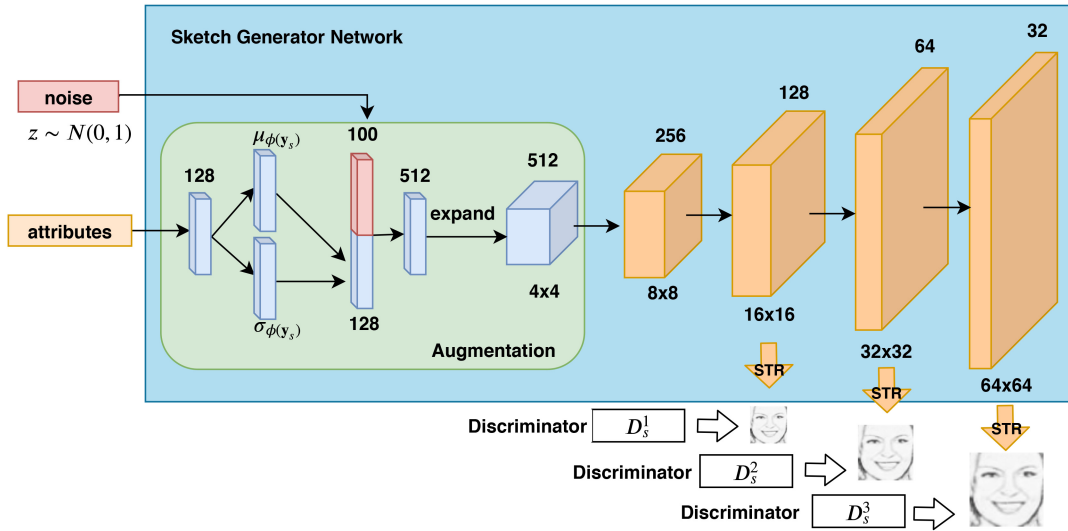


Fig. 3. The sketch generator network architecture. The sketch attributes are first augmented by the attribute augmentation module, in which a new latent attribute variable is re-sampled from the estimated latent distribution ($\mu_{\phi(y_s)}$ and $\sigma_{\phi(y_s)}$) and concatenated with a noise vector. Then, the remaining up-sample modules (orange) aim to generate a series of multi-scale sketches with the augmented sketch attributes.

Based on these generative models, two common problems have been widely studied by researchers: image-to-image synthesis and text-to-image synthesis.

Image-to-image synthesis: One important motivation behind the image-to-image synthesis problem is to bridge the gap between different image domains. Image-to-image translation models are often built based on the common networks like UNet [25] and FCN [41]. Isola *et al.* [42] proposed conditional GANs [43] for several tasks such as labels to street scenes, labels to facades, image colorization, etc. In another variant, Zhu *et al.* [44] proposed CycleGAN that learns image-to-image translation in an unsupervised fashion. Similarly, Yi *et al.* [45] developed an unsupervised dual image-to-image translation model.

Text-to-image synthesis: Isola *et al.* [42] proposed Conditional GANs [43] for several tasks such as labels to street scenes, labels to facades, image colorization, etc. Built on this, Reed *et al.* [14] proposed a conditional GAN network to generate images conditioned on the text description. Several text-to-image synthesis works have been proposed in the literature that make use of the multi-scale information [15]–[17], [30], [46]–[48]. Zhang *et al.* [15] proposed a two-stage stacked GAN (StackGAN) method which achieves the state-of-the-art image synthesis results. More recently this work was extended in [16] by using additional losses and better fine-tuning procedures. Xu *et al.* [18] proposed an attention-driven method to improve the synthesis results. Zhang *et al.* [17] (HDGAN) adopted a multi-adversarial loss to improve the synthesis by leveraging more effective image and text information at multi-scale layers.

III. PROPOSED METHOD

In this section, we provide details of the proposed GAN-based attribute to face synthesis method, which consists of two components: sketch generator and face generator. Note that the training phase of our method requires ground truth attributes and the corresponding sketch and face images. Furthermore,

the attributes are divided into two separate groups - one corresponding to texture and the other corresponding to color. Since sketch contains no color information, we use only the texture attributes in the first component (i.e., sketch generator) as indicated in Fig. 2.

In order to explore the multi-scale information during training, inspired by the previous works [15]–[17], [49], we adopt the idea of hierarchically-integrated multiple discriminators at different layers in our generators. The sketch/face generator network learns the training data distribution from low-resolution to high-resolution. This also helps in improving the training stability of the overall network [50].

A. Stage 1: Attribute-to-Sketch

An overview of the sketch generator network architecture is shown in Fig. 3. Given the sketch attribute vector \mathbf{y}_s , the goal of the sketch generator network G_s is to produce multi-scale sketch outputs as follows

$$G_s(\mathbf{z}_s, \mathbf{y}_s) = \{\hat{\mathbf{x}}_s^1, \hat{\mathbf{x}}_s^2, \dots, \hat{\mathbf{x}}_s^m\} \triangleq \hat{\mathbf{X}}_s, \quad (2)$$

where \mathbf{z}_s is the noise vector sampled from a normal Gaussian distribution, $\{\hat{\mathbf{x}}_s^1, \hat{\mathbf{x}}_s^2, \dots, \hat{\mathbf{x}}_s^m\}$ are the synthesized sketch images with gradually growing resolutions, and $\hat{\mathbf{x}}_s^m$ is the final output with the highest resolution. In order to explore the multi-scale information at different image resolutions, a set of distinct discriminators $D_s = \{D_s^1, \dots, D_s^m\}$ are implemented for each $\hat{\mathbf{x}}_s^i$, $i = 1, 2, 3, \dots, m$. An example of 3-scale generator architecture is shown in Fig. 3. It can be observed that the output sketch images are generated from the feature maps with certain resolutions (width \times height) from different layers of the network.

The generator network consists of three modules: the attribute augmentation module (AA), the up-sample module (UP), and the stretching module (STR). The STR module consists of two 1×1 convolution layers followed by a Tanh layer, which aims to convert the feature map into a 3-channel output image. The UP module consists of an up-sampling

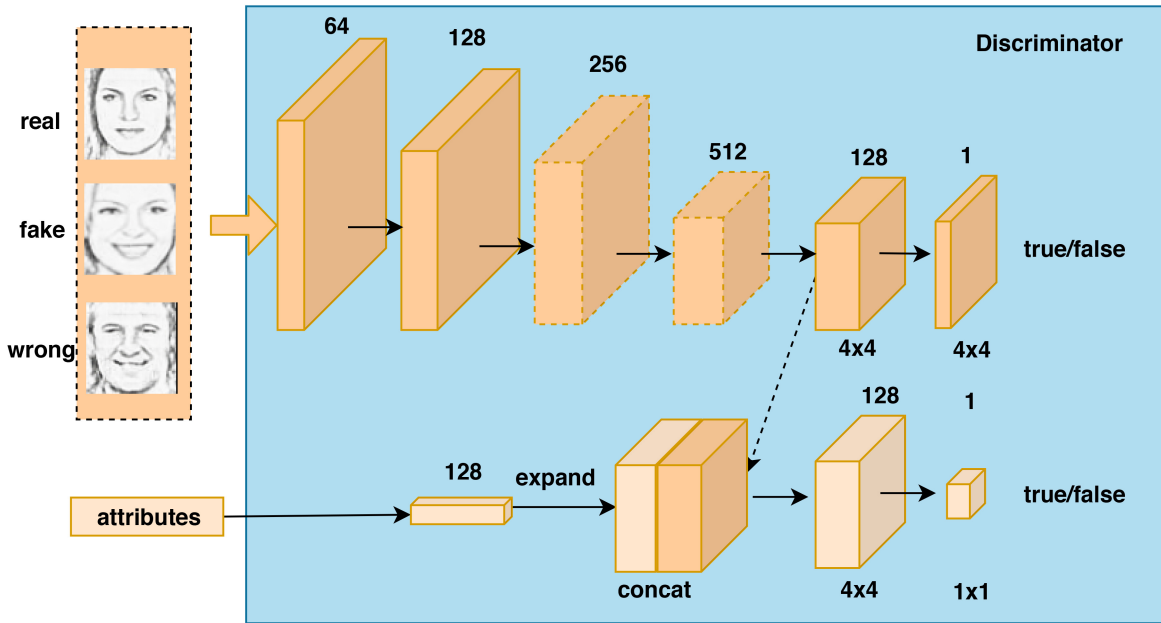


Fig. 4. Sketch discriminator at 64×64 resolution scale: given a sketch attribute vector, the discriminator is trained using the triplets: (i) real-sketch and real-sketch-attributes, (ii) synthesized-sketch and real-sketch-attribute, (iii) wrong-sketch (real sketch but mismatching attributes) and same real-sketch-attributes. Note that the convolutional layers with dashed line are removed when training at lower-resolution discriminator.

layer followed by convolutional, batch normalization, and ReLU layers. Between each UP module, there is an additional residual block (Res) module [51], [52].

The AA module consists of a series of fully-connected neural networks which aim to learn a latent representation of the given visual attribute vector \mathbf{y} . During training, we randomly sample a latent variable $\hat{\mathbf{y}}$ from an independent Gaussian distribution $\mathcal{N}(\mu_{\phi(\mathbf{y})}, \sigma_{\phi(\mathbf{y})})$, where the mean $\mu_{\phi(\mathbf{y})}$ and the diagonal covariance matrix $\sigma_{\phi(\mathbf{y})}$ are learned as the functions of visual attributes \mathbf{y} . In order to avoid over-fitting, the following KL-divergence regularization term is added during training between the augmented visual attribute distribution and the standard Gaussian distribution

$$\mathcal{L}_{aug} = \mathcal{D}_{KL}(\mathcal{N}(\mu_{\phi(\mathbf{y})}, \sigma_{\phi(\mathbf{y})}) \parallel \mathcal{N}(0, \mathcal{I})), \quad (3)$$

where $\mathcal{N}(0, \mathcal{I})$ is the normal Gaussian distribution [12], [15], [29]. Different from previous works [15], [16], we replace the traditional batch normalization layers with the conditional batch normalization [24] in order to overcome the attribute vanishing problem.

As shown in Fig. 3, the overall sketch generator network architecture is as follows: AA(512)-UP(256)-Res(256)-UP(128)-Res(128)-UP(64)-Res(64)-UP(32), where the number in round bracket indicates the output channel of feature maps. As shown in Fig. 3, the three stretching (STR) modules convert the feature maps into 3-channel output sketch images at different resolutions.

Discriminator and Training Loss: The proposed sketch generator produces multi-scale resolution synthesized sketch images. In order to leverage the hierarchical property of the network, a set of discriminators $D_s = \{D_s^1, \dots, D_s^m\}$ with similar architectures are designed for each scale. For a particular scale, the sketch discriminator is developed as shown in Fig. 4. In order to learn the discrimination in both image content

and semantics, we adopt the triplet matching training strategy [14], [16], [17], [53]. Specifically, given sketch attributes, the discriminator is trained by using the following triplets: (i) real-sketch and real-sketch-attributes, (ii) synthesized-sketch and real-sketch-attributes, and (iii) wrong-sketch (real sketch but mismatching attributes) and same real-sketch-attributes. As shown in Fig. 4, two kinds of errors are used to train the discriminator. They correspond to (i) real/face sketch images, and (ii) sketch images and attributes.

The architecture of the proposed sketch discriminator for 64×64 resolution is shown in Fig. 4. This architecture can be easily adapted for other resolution scales by adding/removing appropriate the convolutional layers. As shown in Fig. 4, two branches with different losses are used to train the discriminator at a certain resolution scale. One consists of a series of down-sampling convolutional layers (with filter-size 4, stride 2 and padding size 1) to produce a 4×4 probability map and classify each location as true or false. The other branch first embeds the sketch attributes to a $128 \times 4 \times 4$ feature map and concatenates it with the feature maps from the first branch. Another two 1×1 convolutional layers are used to fuse the concatenated feature maps to produce 4×4 probability maps for classification. This branch aims to distinguish whether the semantics in sketch images match the sketch attribute or not, through the feedback loss from another 4×4 probability map.

The overall adversarial loss used to train the network is defined as follows:

$$\begin{aligned} \mathcal{L}_{SDis} &= \min_{G_s} \max_{D_s} V(G_s, D_s, \mathbf{X}_s, \mathbf{y}_s, \mathbf{z}_s) \\ &= \sum_{i=1}^m \min_{G_s} \max_{D_s^i} (\mathcal{L}_{s_{real}}^i + \mathcal{L}_{s_{fake}}^i + \mathcal{L}_{s_{wrong}}^i), \\ \mathcal{L}_{s_{real}}^i &= \mathbb{E}_{\mathbf{x}_s^i \sim P_{data}(\mathbf{x}_s^i)} [\log D_s^i(\mathbf{x}_s^i)], \\ &\quad + \mathbb{E}_{\mathbf{x}_s^i \sim P_{data}(\mathbf{x}_s^i), \mathbf{y}_s \sim P_{data}(\mathbf{y}_s)} [\log D_s^i(\mathbf{x}_s^i, \mathbf{y}_s)], \end{aligned}$$

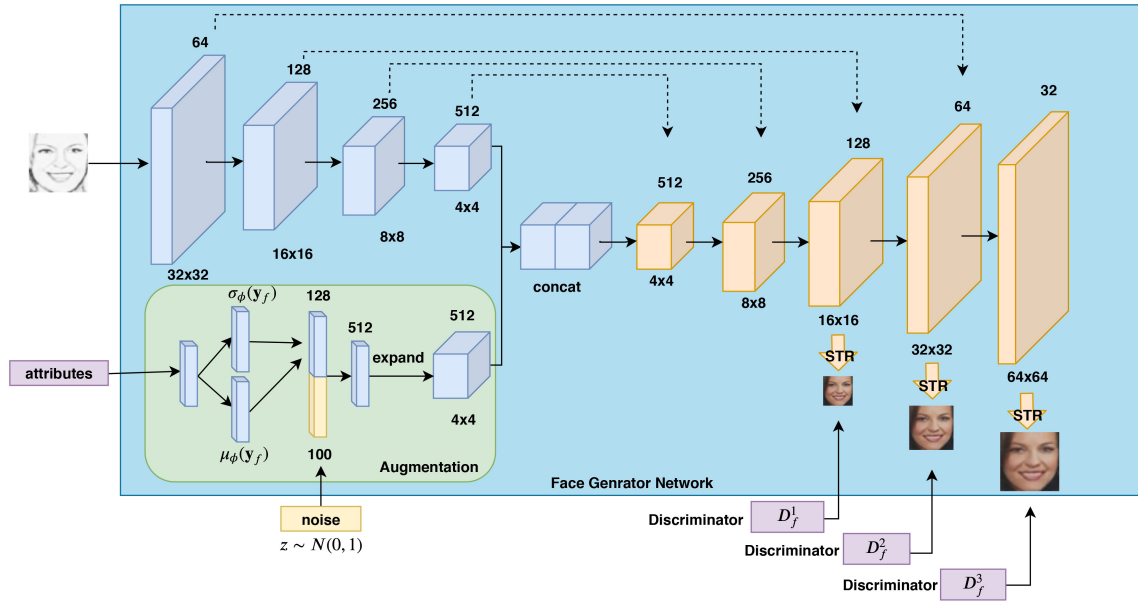


Fig. 5. The architecture of Face Generator Network. The facial attributes are first embedded by the attribute augmentation module, similar to the one used in stage 1. The synthesized sketch image is also embedded by a sequence of down-sample convolutional layers. These two feature maps are then fused by concatenation. Finally, the fused feature maps are used by the up-sample module to synthesize multi-scale face images.

$$\begin{aligned} \mathcal{L}_{S_{wrong}}^i &= \mathbb{E}_{\mathbf{x}_s^i \sim P_{data}(\mathbf{x}_s^i), \mathbf{y}_s \sim P_{data}(\mathbf{y}_s)} [\log(1 - D_s^i(\mathbf{x}_s^i, \mathbf{y}_s))], \\ \mathcal{L}_{S_{fake}}^i &= \mathbb{E}_{\hat{\mathbf{x}}_s^i \sim P_{G_s}(\mathbf{y}_s, \mathbf{z}_s)} [\log(1 - D_s^i(\hat{\mathbf{x}}_s^i))] \\ &\quad + \mathbb{E}_{\hat{\mathbf{x}}_s^i \sim P_{G_s}(\mathbf{y}_s, \mathbf{z}_s), \mathbf{y}_s \sim P_{data}(\mathbf{y}_s)} [\log(1 - D_s^i(\hat{\mathbf{x}}_s^i, \mathbf{y}_s))], \end{aligned} \quad (4)$$

where $\hat{\mathbf{x}}_s^i \sim P_{G_s}(\mathbf{y}_s, \mathbf{z}_s)$ stands for the synthesized (fake) sketch image sampled from the sketch generator at scale i , $\mathbf{x}_s^i \sim P_{data}(\mathbf{x}_s^i)$ stands for the real sketch image sampled from the sketch image data distribution at scale i , $\hat{\mathbf{x}}_s^i$ is the attribute-mismatching sketch image sample at scale i , and \mathbf{y}_s is the sketch attribute vector. The total objective loss function is given as follows

$$\mathcal{L}_{S_{total}} = \sum_{i=1}^m \min_{G_s} \max_{D_s^i} (\mathcal{L}_{S_{real}}^i + \mathcal{L}_{S_{fake}}^i + \mathcal{L}_{S_{wrong}}^i) + \lambda_s \mathcal{L}_{S_{aug}}, \quad (5)$$

where the hyperparameter λ_s is set equal to 0.01 in our experiments, $\mathcal{L}_{S_{aug}}$ is the KL-divergence regularization in the AA module with sketch attribute \mathbf{y}_s and noise \mathbf{z}_s as inputs.

B. Stage 2: Sketch-to-Face

Given the synthesized sketches $\hat{\mathbf{X}}_s$ and the facial attributes \mathbf{y}_f , the face generator network G_f aims to produce multi-scale outputs as follows

$$G_f(\hat{\mathbf{X}}_s; \mathbf{z}, \mathbf{y}_f) = \{\hat{\mathbf{x}}_f^1, \hat{\mathbf{x}}_f^2, \dots, \hat{\mathbf{x}}_f^m\} \triangleq \hat{\mathbf{X}}_f, \quad (6)$$

where \mathbf{z} is noise sampled from a normal Gaussian distribution and $\hat{\mathbf{X}}_f$ are the synthesized facial images with gradually growing resolutions. Similar to the sketch generation network, a set of distinct discriminators are designed for each scale. The overall objective is given as follows:

$$G_f^*, D_f^* = \arg \min_{G_f} \max_{D_f} V(G_f, D_f; \hat{\mathbf{X}}_s, \mathbf{y}_f, \mathbf{z}), \quad (7)$$

where $D_f = \{\mathbb{D}_1, \dots, \mathbb{D}_m\}$ and $\mathbf{X}_f = \{\mathbf{x}_f^1, \dots, \mathbf{x}_f^m\}$ denote real training images at multiple scales $1, \dots, m$. In order to preserve the geometric structure of the synthesized sketch from the attribute-to-sketch stage, we adopt the skip-connection architecture from UNet related works [25], [53], [54]. By using skip-connections, the feature maps from the encoding network are concatenated with the feature maps in the decoding network. This way, the geometric structure of the learned sketch image is inherited in the synthesized facial image. The proposed method is trained end-to-end. The lower-resolution outputs fully utilize the top-down knowledge from the discriminators at higher resolutions. Therefore, the synthesized images from different resolutions preserve the geometric structure, which improves the training stability and synthesis quality.

The architecture of the face generator network is shown in Fig. 5. The generator consists of four modules: the AA module, the down-sample module (DO), the UP module, and the STR module. As before, the STR aims convert the feature map into a 3-channel output image. It consists of two 1×1 convolutional (conv) layers with one Tanh layer. The UP module consists of an up-sampling layer followed by conv-BN-ReLU layers and an additional residual block [51], [52] is between each UP module. The DO module consists of a series of conv-BN-ReLU layers. The overall face generator network architecture consists of the following components DO(64)-DO(128)-DO(256)-DO(512)-AA(512)-UP(512)-UP(256)-UP(128)-UP(64)-UP(32), where the number in round brackets indicate the output channel of feature maps. As shown in Fig. 5, the three stretching (STR) modules convert the feature maps into 3-channel output face images at different resolutions.

Discriminator and Training Loss: In the sketch-to-face stage, we use the same architecture of the discriminator as



Fig. 6. Sketch images sampled from the LFW and the CelebA datasets are shown in row 1 and row 2 respectively.

was used in stage 1. We input the triplets as facial images instead of sketch images and replace the sketch attributes by the facial attributes. Furthermore, the training loss function is also the same as the one used in the attribute-to-sketch stage.

C. Testing

Fig. 2 shows the testing phase of the proposed method. A sketch attribute vector \mathbf{y}_s and \mathbf{z}_s sampled from a normal Gaussian distribution are first passed through the sketch generator network G_s to produce a sketch image. Then the synthesized sketch image with the highest resolution, attribute vector \mathbf{y}_f and another noise vector \mathbf{z}_f are passed through the face generator network to synthesize a face image. In other words, our method takes noise and attribute vectors as inputs and generates high-quality face images via sketch images.

IV. EXPERIMENTAL RESULTS

In this section, experimental settings and evaluation of the proposed method are discussed in detail. Results are compared with several related generative models: disCVAE [11], GAN-INT-CLS [14], StackGAN [15], Attribute2Sketch2Face [55], StackGAN++ [16] and HDGAN [17]. The entire network in Fig. 2 is trained end-to-end using Pytorch. When training, the learning rate for the generator and the discriminator in the first stage is set equal to 0.0002, while the learning rate in the second stage is set equal to 0.0001.

We conduct experiments using two publicly available datasets: CelebA [3], and deep funneled LFW [56]. The CelebA database contains about 202,599 face images, 10,177 different identities and 40 binary attributes for each face image. The deep funneled LFW database contains about 13,233 images, 5,749 different identities and 40 binary attributes for each face image which are from the LFWA dataset [3].

Note that the training part of our network requires original face images and the corresponding sketch images as well as the corresponding list of visual attributes. The CelebA and the deep funneled LFW datasets consist of both the original images and the corresponding attributes. To generate the missing sketch images in the CelebA and the deep funneled LFW datasets, we use a public pencil-sketch synthesis method¹ to generate the sketch images from the face images. Fig. 6 shows some sample generated sketch images from the CelebA and the deep funneled LFW datasets.

¹<http://www.askaswiss.com/2016/01/how-to-create-pencil-sketch-opencv-python.html>

TABLE I
LIST OF FINE-GRAINED TEXTURE AND COLOR ATTRIBUTES

Texture	5_o_Clock_Shadow, Arched_Eyebrows, Bags_Under_Eyes, Bald, Bangs, Big_Lips, Big_Nose, Bushy_Eyebrows, Chubby, Eyeglasses, Male, Mouth_Slightly_Open, Narrow_Eyes, No_Beard, Oval_Face, Smiling, Young
Color	Black_Hair, Blond_Hair, Brown_Hair, Gray_Hair, Pale_Skin, Rosy_Cheeks

The MTCNN method [57] was used to detect and crop faces from the original images. The detected faces were scaled to the size of 64×64 . Since many attributes from the original list of 40 attributes were not significantly informative, we selected 23 most useful attributes for our problem. Furthermore, the selected attributes were further divided into 17 texture and 6 color attributes as shown in Table I. During experiments, the texture attributes were used to train the sketch generator network while all 23 attributes were used to train the face generator network.

A. CelebA Dataset Results

The CelebA dataset [3] consists of 162,770 training samples, 19,867 validation samples and 19,962 test samples. We combine the training and validation splits together to train our models. After detection and alignment, we obtain 182,468 samples which we use for training our proposed model. During training, we use the batch size of 40. The ADAM algorithm [58] with learning rate of 0.0002 is used to train the network. In total, 20 training epoch are used during training and the initial learning rate is frozen for the first 10 epochs. For the next 10 epochs, we let it drop by 0.1 of the initial value after every epoch. The latent feature dimension for sketch/facial attribute is set equal to 128. The noise vector dimension is set equal to 100. Three scales (16×16 , 32×32 , and 64×64) are used in our multi-scale network.

Sample image generation results corresponding to different methods from the CelebA are shown in Fig. 7. For fair comparison with those stage-wise training algorithm, we adopt StackGAN [15] network to two scale resolution 32×32 and 64×64 . Moreover, we adopt StackGAN++ [16] HDGAN [17] in the same resolution scales: 16×16 , 32×32 , and 64×64 . Note that these results are obtained by inputting a certain attribute vector along with random noise. As can be seen from this figure, GAN-INT-CLS and StackGAN methods easily meet the modal collapse issue in this problem. During training, the generator learns to generate a limited number (1 or 2) of image samples corresponding to a certain list of attributes. This synthesized results are good enough to fool the discriminator. Thus, the generator and discriminator networks do not get optimized properly. The disCVAE method is able to reconstruct the images without model collapse but they are blurry due to the imperfect L_2 measure in the Gaussian distribution loss. In addition, some of the attributes are difficult to see in the reconstructions corresponding to the disCVAE method, such as the hair color. This is because of the imperfect latent embedding by the variational bound limitation in VAE. Also, the other Attribute2Sketch2Face is able to generate realistic results, but the image quality is slightly inferior. The recent

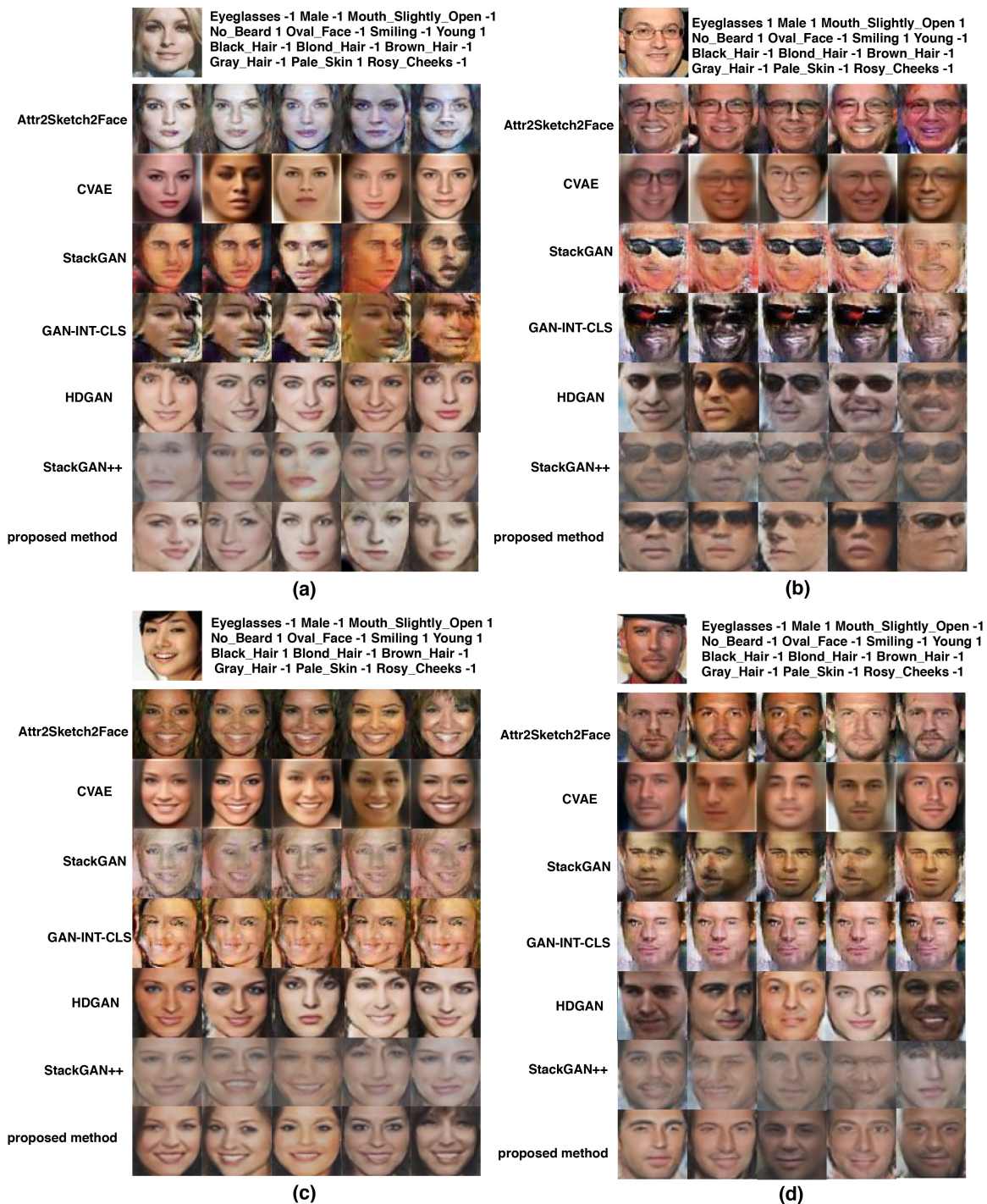


Fig. 7. Image generation results on the CelebaA dataset. First row of each sub-figure shows the reference image and its corresponding attributes. The images generated by different methods are shown in different rows.

state-of-art text-to-image synthesis approaches (stackGAN++ and HDGAN) generate plausible facial images from visual attributes. However, the generated facial images do not always preserve the corresponding attributes very well. Compared with all the baselines, the proposed method not only generates realistic facial images but also preserves the attributes better than the others. We believe that this is mainly due to the way we approach the attribute-to-face synthesis approach by decomposing it into two problems, attribute-to-sketch

and sketch-to-face. By factoring the original problem into two separate problems, the model at each stage learns better conditional data distribution. Furthermore, the use of multi-scale generators in the proposed GANs also help in improving the performance of our method.

B. LFW Dataset Results

Images in the LFWA dataset come from the LFW dataset [56], [59], and the corresponding attributes come

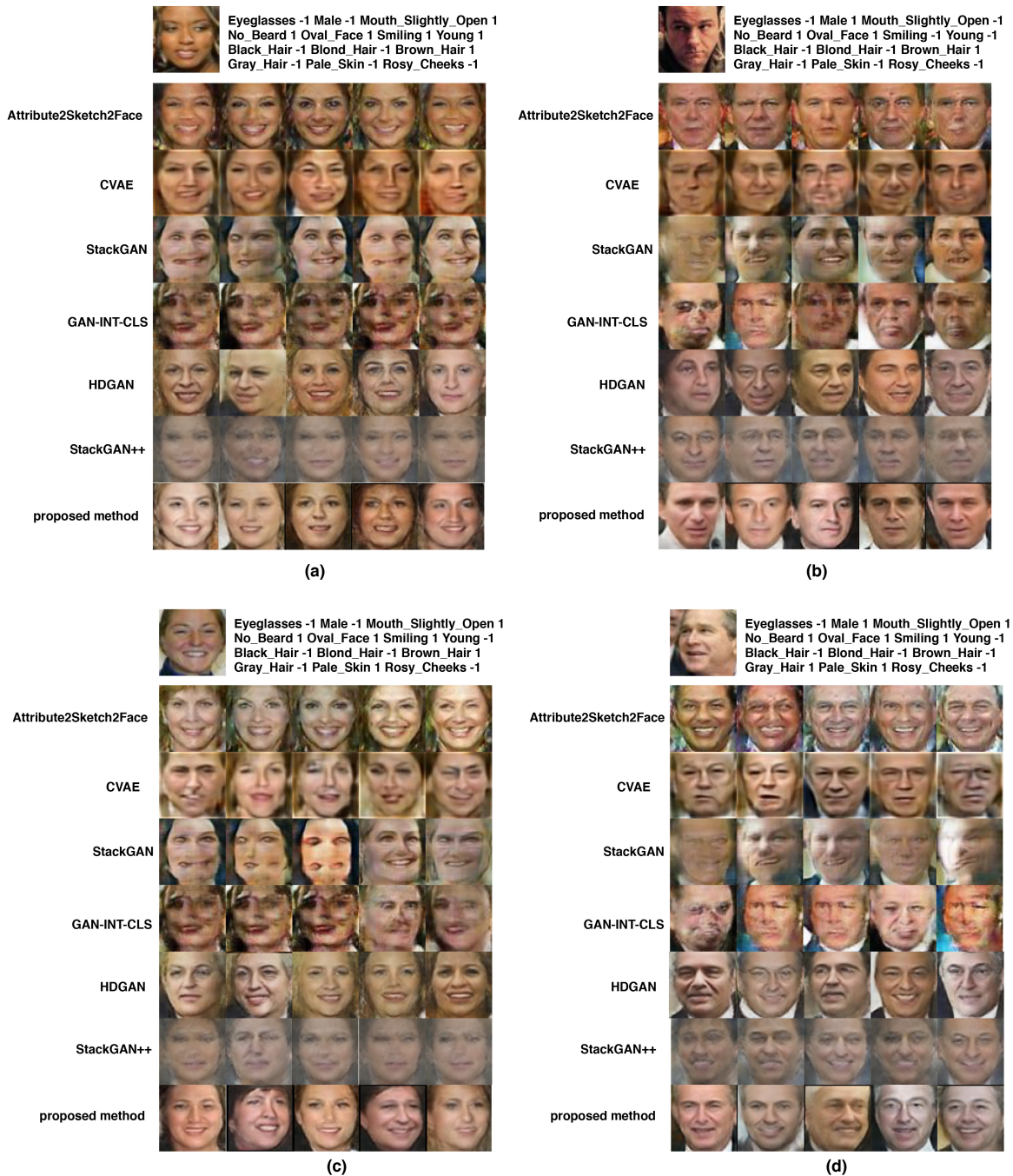


Fig. 8. Image generation results on the LFWA dataset. First row of each sub-figure shows the reference image and its corresponding attributes. The images generated by different methods are shown in different rows.

from [3]. This dataset contains the same 40 binary attributes as in the CelebA dataset. After pre-processing, the training and testing subsets contain 6,263 and 6,880 samples, respectively. We use all the training splits to train our model. The ADAM algorithm [58] with learning rate of 0.0002 is used for both generators and discriminators. The initial learning rate is frozen in the first 100 epochs and is then dropped by 0.01 for the remaining 100 epochs. All the other experimental settings are the same as the ones used in with CelebA dataset.

Sample results corresponding to different methods on the LFWA dataset are shown in Fig. 8. For fair comparison,

the multi-scale resolution settings are the same as used with the experiments on the CelebA dataset. In particular, we use 32×32 and 64×64 resolution scales for StackGAN [15] training and 16×16 , 32×32 and 64×64 multiple resolution scales for HDGAN [17] and StackGAN++ [16] as well as our proposed method. The disCVAE method produces reconstructions which are blurry. Previous conditional GAN-based approaches such as GAN-INT-CLS [14] and StackGAN [15] also produce poor quality results due to the model collapse during training. Recent StackGAN++ and HDGAN works generate plausible facial images (HDGAN is better at the color diversity). The previous work Attribute2Sketch2Face, which

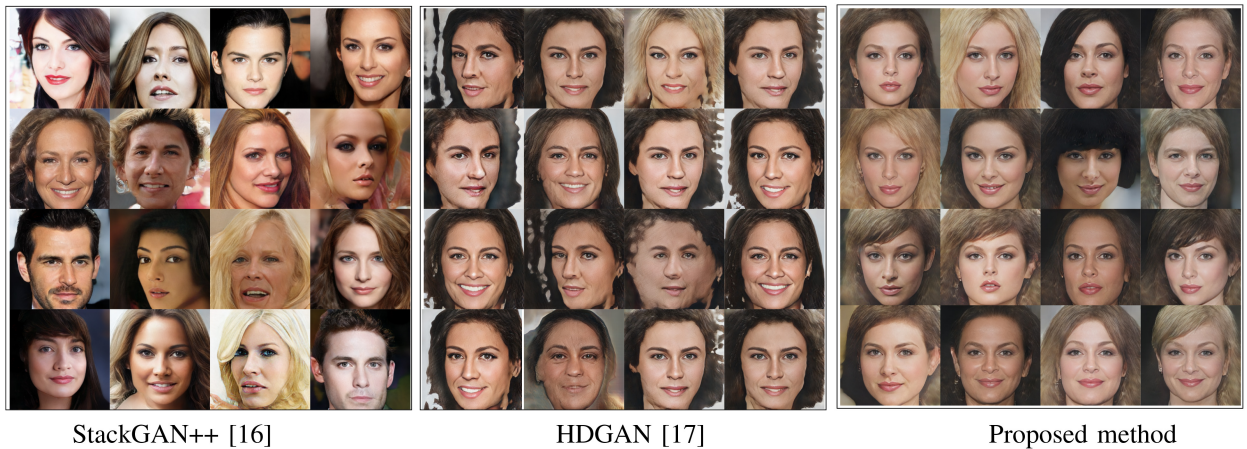


Fig. 9. Image synthesis results on 256×256 resolution. The attributes used to generate these images are: Eyeglasses -1 Male -1 Mouth_Slightly_Open -1 No_Beard 1 Oval_Face -1 Smiling -1 Young 1 Black_Hair -1 Blond_Hair -1 Brown_Hair -1 Gray_Hair -1 Pale_Skin 1 Rosy_Cheeks -1.

is a combination of CVAE and GAN, is also able to generate facial images with corresponding attributes. However, the proposed method is able to reconstruct high-quality attribute-preserved face images better than the previous approaches.

C. CelebA-HQ Dataset Results

In order to demonstrate how our proposed method works on high-resolution images, we also conduct an experiment using a recent proposed CelebA-HQ dataset. The CelebA-HQ dataset [50] is a high-quality version of the CelebA dataset, which consists of 30,000 images with 1024×1024 resolution. Due to GPU and memory limitations, we conduct experiments on 256×256 resolution images and compare the performance with StackGAN++ [16] and HDGAN [17]. The reason why we chose these two baselines is due to their capability to deal with high resolution images. Sample results are shown in Figure 9.

For fair comparison, we set the number of resolution scale $s = 3$ for all methods. In order to adopt our method to the high-resolution dataset, we follow the strategy that removing/adding the number of UP/DO block (as defined in Section III) in the generator and the discriminator. In particular, we set the STR modules at resolution 64×64 , 128×128 and 256×256 respectively. In experiments, the batch-size is set equal to 16 for our proposed method, which is smaller than StackGAN++ and HDGAN, which are set equal to 24, due to the GPU memory limitations. Also, when training on this dataset, we train the sketch generator first and then use the pre-trained model for training the face generator.

As can be seen from Figure 9, our proposed method can synthesize photo-realistic images on high-resolution images as well. Moreover, when we compare the attributes from the synthesized images with the given attributes, we can observe that our method preserves the attributes better than the other methods. Quantitative comparisons in terms of the FID scores also show that the proposed method performs favorably compared to StackGAN++ and HDGAN. In addition, comparison of our method in Table II with only a single scale shows the significance of our multi-scale network.

TABLE II
QUANTITATIVE RESULTS (FID SCORES) CORRESPONDING TO DIFFERENT METHODS ON THE CELEBA-HQ DATASET

Methods	FID score
HDGAN [17]	114.912
StackGAN++ [16]	35.988
Single-scale (proposed method)	37.381
Proposed method	30.566

D. Face Synthesis

In this section, we show the image synthesis capability of our network by manipulating the input attribute and noise vectors. Note that, the testing phase of our network takes attribute vector and noise as inputs and produces synthesized face as the output. In the first set of experiments with image synthesis, we keep the random noise vector frozen and change the weight of a particular attribute as follows: $[-1, -0.1, 0.1, 0.4, 0.7, 1]$. The corresponding results on the CelebA dataset are shown in Fig. 10. From this figure, we can see that when we give higher weights to a certain attribute, the corresponding appearance changes. For example, one can synthesize an image with a different gender by changing the weights corresponding to the gender attribute as shown in Fig. 10(a). Each row shows the progression of gender change as the attribute weights are changed from -1 to 1 as described above. Similarly, figures (b), (c) and (d) show the synthesis results when a neutral face image is transformed into a smily face image, skin tones are changed to pale skin tone, and hair colors are changed to black, respectively. It is interesting to see that when the attribute weights other than the gender attribute are changed, the identity of the person does not change too much.

In the second set of experiments, we keep the input attribute vector frozen but now change the noise vector by inputting different realizations of the standard Gaussian. Sample results corresponding to this experiment are shown in Fig. 11 using the CelebA. Each column shows how the output changes as we change the noise vector. Different subjects are shown in different rows. It is interesting to note that, as we change the noise vector, attributes stay the same while the identity changes.

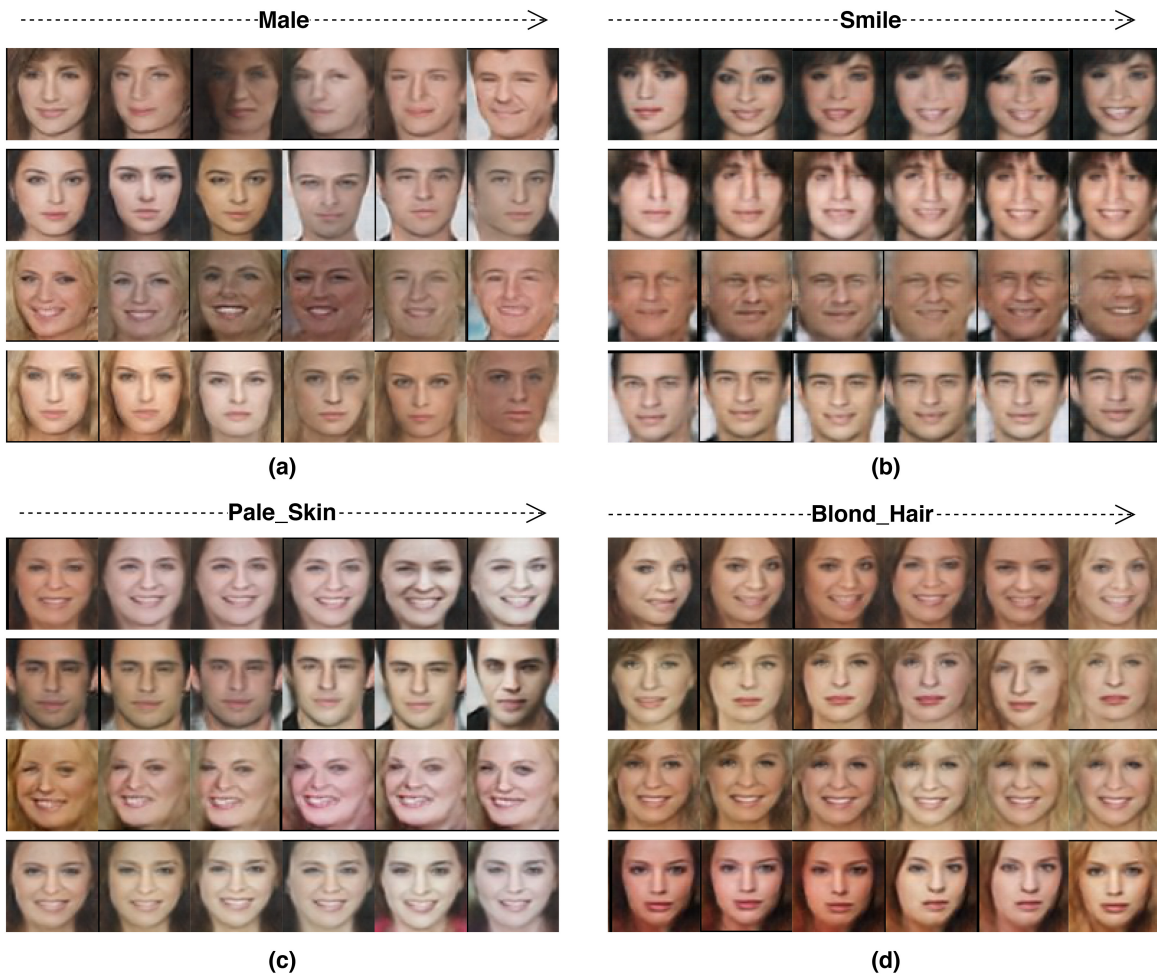


Fig. 10. Facial image progressive synthesis on CelebA when attributes are changed. These progressive changes are based on one certain attribute manipulating while the others are keep frozen. (a) Male. (b) Smile. (c) Original skin tone to pale skin tone. (d) Original hair color to black hair color.

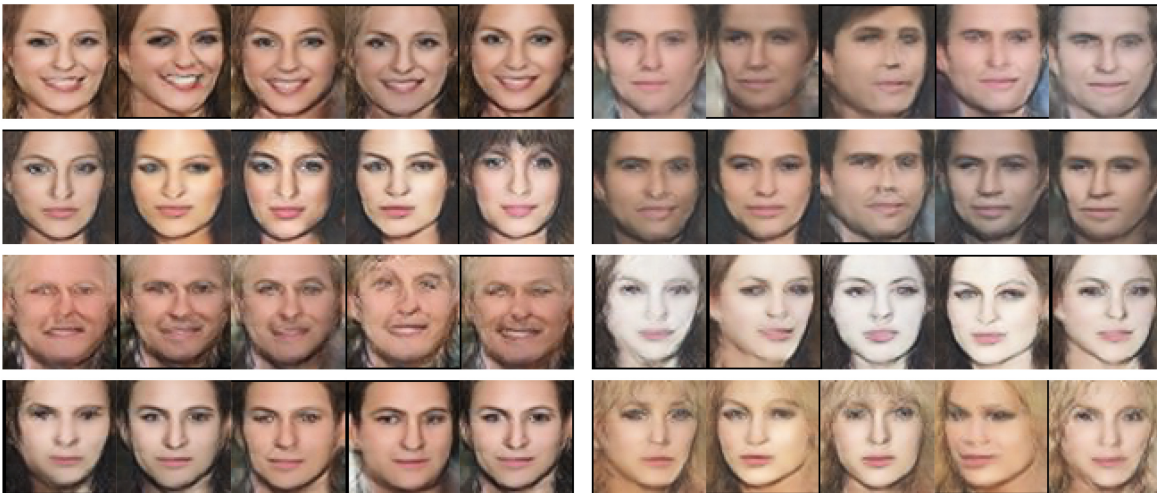


Fig. 11. Facial image synthesis sampled when attributes are kept frozen while the noise vector is changed. Note that the identity, pose, or facial shape changes as we vary the noise vector but the attributes stay the same on the synthesized images.

This can be clearly seen by comparing the synthesized results in each row.

E. Quantitative Results

In addition to the qualitative results presented in Fig. 7, 8, we present quantitative comparisons in Table III. Since the

ground-truth images corresponding to the noise-generated images are not available, we choose the quantitative criterion based on the Fréchet Inception Distance (FID) [60], [61] and Attribute L_2 -norm. The FID is a measure of similarity between two datasets of images. It was shown to correlate well with human judgment of visual quality and is most often used to

TABLE III
QUANTITATIVE RESULTS CORRESPONDING TO DIFFERENT METHODS. THE FID SCORE AND ATTRIBUTE L_2 MEASURE ARE USED TO COMPARE THE PERFORMANCE OF DIFFERENT METHODS

Baselines	LFW		CelebA	
	FID Score	Attribute L_2	FID Score	Attribute L_2
GAN-INT-CLS [14]	85.811	0.093 ± 0.027	92.793	0.104 ± 0.024
disCVAE [11]	103.855	0.086 ± 0.019	91.012	0.080 ± 0.042
StackGAN [15]	70.379	0.085 ± 0.029	63.816	0.091 ± 0.021
StackGAN++ [16]	50.360	0.059 ± 0.026	49.889	0.061 ± 0.026
HDGAN [17]	48.930	0.053 ± 0.023	43.206	0.056 ± 0.020
Attribute2Sketch2Face [55]	60.487	0.059 ± 0.034	58.896	0.067 ± 0.022
Attribute2Sketch2Face-v2 (proposed method)	43.712	0.048 ± 0.020	33.497	0.051 ± 0.019

evaluate the quality of samples generated by GANs. Attribute L_2 -norm is used to compare the quality of attributes corresponding to different images. We extract the attributes from the synthesized images as well as the reference image using the MOON attribute prediction method [7]. Once the attributes are extracted, we simply take the L_2 -norm of the difference between the attributes as follows

$$\text{Attribute } L_2 = \|\hat{a}_{ref} - \hat{a}_{synth}\|_2, \quad (8)$$

where \hat{a}_{ref} and \hat{a}_{synth} are the 23 extracted attributes from the reference image and the synthesized image, respectively. Note that lower values of the FID score and the Attribute L_2 measure imply the better performance. The quantitative results corresponding to different methods on the CelebA and LFW datasets are shown in Table III. Results are evaluated on the test splits of the corresponding dataset and the average performance along with the standard deviation are reported in Table III.

As can be seen from this table, the proposed method produces the lowest FID scores implying that the images generated by our method are more realistic than the ones generated by other methods. Furthermore, our method produces the lowest Attribute L_2 scores. This implies that our method is able to generate attribute-preserved images better than the other compared methods. This can be clearly seen by comparing the images synthesized by different methods in Fig. 7 and Fig. 8.

V. CONCLUSION

We presented a novel deep generative framework for reconstructing face images from visual attributes. Our method makes use of an intermediate representation to generate photo realistic images. The training part of our method consists of two models: Sketch Generator Network and Face Generator Network. Multi-scale hierarchical network architectures are proposed for each generator networks. Various experiments on three publicly available datasets show the significance of the proposed synthesis framework. In addition, an ablation study was conducted to show the importance of different components of our network. Various experiments showed that the proposed method is able to generate high-quality images and achieves significant improvements over the state-of-the-art methods.

One of the limitations of this work is that the synthesized images do not preserve the identity. In the future, we will develop methods that can synthesize identity-preserving images from visual attributes. These images can then be used to augment the datasets for face recognition [62], [63].

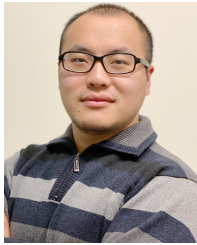
ACKNOWLEDGMENT

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

REFERENCES

- [1] N. Kumar, A. Berg, P. N. Belhumeur, and S. Nayar, "Describable visual attributes for face verification and image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 1962–1977, Oct. 2011.
- [2] A. Dantcheva, P. Elia, and A. Ross, "What else does your biometric data reveal? A survey on soft biometrics," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 3, pp. 441–467, Mar. 2016.
- [3] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3730–3738.
- [4] N. Kumar, P. N. Belhumeur, and S. K. Nayar, "FaceTracer: A search engine for large collections of images with faces," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 340–353.
- [5] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. D. Bourdev, "PANDA: Pose aligned networks for deep attribute modeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 1637–1644.
- [6] R. Ranjan, V. M. Patel, and R. Chellappa, "HyperFace: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 1, pp. 121–135, Jan. 2019.
- [7] E. M. Rudd, M. Günther, and T. E. Boulton, "Moon: A mixed objective optimization network for the recognition of facial attributes," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 19–35.
- [8] Y. Lu, A. Kumar, S. Zhai, Y. Cheng, T. Javidi, and R. Feris, "Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5334–5343.
- [9] M. Günther, A. Rozsa, and T. E. Boulton, "AFACT: Alignment-free facial attribute classification technique," in *Proc. IEEE Int. Joint Conf. Biometr. (IJCB)*, 2017, pp. 90–99.
- [10] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3483–3491.
- [11] X. Yan, J. Yang, K. Sohn, and H. Lee, "Attribute2Image: Conditional image generation from visual attributes," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 776–791.
- [12] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Represent.*, 2014.
- [13] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
- [14] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proc. 33rd Int. Conf. Mach. Learn. (ICML)*, vol. 48, 2016, pp. 1060–1069. [Online]. Available: <http://dl.acm.org/citation.cfm?id=3045390.3045503>
- [15] H. Zhang *et al.*, "StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 5907–5915.
- [16] H. Zhang *et al.*, "Stackgan++: Realistic image synthesis with stacked generative adversarial networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1947–1962, Aug. 2018.

- [17] Z. Zhang, Y. Xie, and L. Yang, "Photographic text-to-image synthesis with a hierarchically-nested adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6199–6208.
- [18] T. Xu *et al.*, "AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1316–1324.
- [19] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, "CVAE-GAN: Fine-grained image generation through asymmetric training," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2764–2773.
- [20] X. Wang and A. Gupta, "Generative image modeling using style and structure adversarial networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 318–335.
- [21] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, "Decomposing motion and content for natural video sequence prediction," *arXiv preprint arXiv:1706.08033*, 2017.
- [22] Q. Chen and V. Koltun, "Photographic image synthesis with cascaded refinement networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1520–1529.
- [23] J. Walker, K. Marino, A. Gupta, and M. Hebert, "The pose knows: Video forecasting by generating pose futures," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3352–3361.
- [24] H. De Vries, F. Strub, J. Mary, H. Larochelle, O. Pietquin, and A. C. Courville, "Modulating early visual processing by language," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6594–6604.
- [25] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Intervent.*, 2015, pp. 234–241.
- [26] H. Larochelle and I. Murray, "The neural autoregressive distribution estimator," in *Proc. 14th Int. Conf. Artif. Intell. Stat.*, 2011, pp. 29–37.
- [27] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1278–1286.
- [28] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.
- [29] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1558–1566.
- [30] E. L. Denton, S. Chintala, A. Szlam, and R. Fergus, "Deep generative image models using a Laplacian pyramid of adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1486–1494.
- [31] A. Dosovitskiy, J. T. Springenberg, M. Tatarchenko, and T. Brox, "Learning to generate chairs, tables and cars with convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 692–705, Apr. 2017.
- [32] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.
- [33] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–25.
- [34] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–25.
- [35] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–13.
- [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, 2014.
- [37] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 2642–2651.
- [38] L. Song, Z. Lu, R. He, Z. Sun, and T. Tan, "Geometry guided adversarial facial expression synthesis," in *Proc. ACM Multimedia Conf. Multimedia Conf.*, 2018, pp. 627–635.
- [39] Z. Lu, T. Hu, L. Song, Z. Zhang, and R. He, "Conditional expression synthesis with face parsing transformation," in *Proc. ACM Multimedia Conf. Multimedia Conf.*, 2018, pp. 1083–1091.
- [40] R. Huang, S. Zhang, T. Li, and R. He, "Beyond face rotation: Global and local perception GAN for photorealistic and identity preserving frontal view synthesis," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2439–2448.
- [41] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
- [42] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 5967–5976.
- [43] M. Mirza and S. Osindero, "Conditional generative adversarial nets," in *Proc. Comput. Vis. Pattern Recognit.*, 2014.
- [44] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.
- [45] Z. Yi, H. Zhang, P. Tan, and M. Gong, "DualGAN: Unsupervised dual learning for image-to-image translation," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 2868–2876.
- [46] P. Li, Y. Hu, R. He, and Z. Sun, "Global and local consistent wavelet-domain age synthesis," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 11, pp. 2943–2957, Nov. 2019.
- [47] I. Korschunova, W. Shi, J. Dambre, and L. Theis, "Fast face-swap using convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3677–3685.
- [48] S. Banerjee, W. J. Scheirer, K. W. Bowyer, and P. J. Flynn, "On hallucinating context and background pixels from a face mask using multi-scale GANs," *arXiv preprint arXiv:1811.07104*, 2018.
- [49] L. Wang, V. Sindagi, and V. Patel, "High-quality facial photo-sketch synthesis using multi-adversarial networks," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, 2018, pp. 83–90.
- [50] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–26. [Online]. Available: <https://openreview.net/forum?id=Hk992CeAb>
- [51] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 630–645.
- [52] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [53] X. Di, H. Zhang, and V. M. Patel, "Polarimetric thermal to visible face verification via attribute preserved synthesis," in *Proc. IEEE Int. Conf. Biometr. Theory Appl. Syst. (BTAS)*, 2018, pp. 1–10.
- [54] X. Di, V. A. Sindagi, and V. M. Patel, "GP-GAN: Gender preserving GAN for synthesizing faces from landmarks," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, 2018, pp. 1079–1084.
- [55] X. Di and V. M. Patel, "Face synthesis from visual attributes via sketch using conditional VAEs and GANs," *arXiv preprint arXiv:1801.00077*, 2017.
- [56] G. B. Huang, M. A. Mattar, H. Lee, and E. G. Learned-Miller, "Learning to align from scratch," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 773–781.
- [57] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.
- [58] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015.
- [59] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Univ. Massachusetts, Amherst, MA, USA, Rep. 07-49, Oct. 2007.
- [60] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6626–6637.
- [61] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, "Are GANs created equal? A large-scale study," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 700–709.
- [62] X. Wu, R. He, Z. Sun, and T. Tan, "A light CNN for deep face representation with noisy labels," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2884–2896, Nov. 2018.
- [63] J.-C. Chen, V. M. Patel, and R. Chellappa, "Unconstrained face verification using deep CNN features," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.



Xing Di (Student Member, IEEE) received the M.E. degree in electrical engineering from the Stevens Institute of Technology, Hoboken, NJ, USA, in 2015, and the first Ph.D. degree from the Department of Electrical and Computer Engineering (ECE), Rutgers University. He is currently pursuing the second Ph.D. degree with the Department of ECE, Johns Hopkins University. His current research interests include machine learning, computer vision, and image process with applications in biometrics. He has received the Best Student Paper Award

at IAPR ICPR 2018 and the ECE Student Development Award from Rutgers University. He also serves as the Journal Reviewer of the IEEE TRANSACTIONS ON IMAGE PROCESSING and PR, and a Conference Reviewer of BTAS and ICB.



Vishal M. Patel (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Maryland, College Park, MD, USA, in 2010. He was an A. Walter Tyson Assistant Professor with the Department of Electrical and Computer Engineering (ECE), Rutgers University and a member of the research faculty with the University of Maryland Institute for Advanced Computer Studies. He is an Assistant Professor with the Department of ECE, Johns Hopkins University. His current research interests include computer

vision, image processing, and pattern recognition with applications in biometrics and imaging. He has received a number of awards, including the 2016 ONR Young Investigator Award, the 2016 Jimmy Lin Award for Invention, the A. Walter Tyson Assistant Professorship Award, the Best Paper Awards at IEEE AVSS 2017 and 2019, the Best Paper Award at IEEE BTAS 2015, the Honorable Mention Paper Award at IAPR ICB 2018, two Best Student Paper Awards at IAPR ICPR 2018, and the Best Poster Awards at BTAS 2015 and 2016. He is an Associate Editor of *IEEE Signal Processing Magazine*, IEEE BIOMETRICS COMPENDIUM, and serves on the Information Forensics and Security Technical Committee of the IEEE Signal Processing Society. He is a member of Eta Kappa Nu, Pi Mu Epsilon, and Phi Beta Kappa.