High Resolution Face Completion with Multiple Controllable Attributes via Fully End-to-End Progressive Generative Adversarial Networks

ZEYUAN CHEN, North Carolina State University
SHAOLIANG NIE, North Carolina State University
TIANFU WU, North Carolina State University
CHRISTOPHER G. HEALEY, North Carolina State University

Fig. 1. Face completion results of our method on CelebA-HQ [Karras et al. 2017]. Images in the left most column of each group are masked with gray color, while the rest are synthesized faces. *Top*: our approach can complete face images at high resolution (1024×1024). *Bottom*: the attributes of completed faces can be controlled by conditional vectors. Attributes ["Male", "Smiling"] are used in this example. The conditional vectors of column two to five are [0, 0], [1, 0], [0, 1], and [1, 1] in which "1" denotes the generated images have the particular attribute while "0" denotes not. Images are at 512×512 resolution. All images best viewed enlarged.

We present a deep learning approach for high resolution face completion with multiple controllable attributes (e.g., male and smiling) under arbitrary masks. Face completion entails understanding both structural meaningfulness and appearance consistency locally and globally to fill in "holes" whose content do not appear elsewhere in an input image. It is a challenging task with the difficulty level increasing significantly with respect to high resolution, the complexity of "holes" and the controllable attributes of filled-in fragments. Our system addresses the challenges by learning a fully end-to-end framework that trains generative adversarial networks (GANs) progressively from low resolution to high resolution with conditional vectors encoding controllable attributes.

We design novel network architectures to exploit information across multiple scales effectively and efficiently. We introduce new loss functions encouraging sharp completion. We show that our system can complete faces with large structural and appearance variations using a single feed-forward pass of computation with mean inference time of 0.007 seconds for images at 1024×1024 resolution. We also perform a pilot human study that shows our approach outperforms state-of-the-art face completion methods in terms of rank analysis. The code will be released upon publication.

CCS Concepts: • **Computing methodologies** → **Neural networks**; **Image processing**;

Additional Key Words and Phrases: GAN, Deep Learning, Face Completion

1 INTRODUCTION

Making things complete and more satisfying is always fascinating people in many creative ways. Image completion is a technique to replace target regions, either missing or unwanted, of images with synthetic content so that the completed images look natural, realistic and appealing. The capability of seeing the unseen or realizing imagination has broad applications in visual content editing. Image completion can be divided in two categories: generic scene image completion and specific object image completion (e.g., human faces). Due to the well-known compositionality and reusability of

Authors' addresses: Zeyuan Chen, North Carolina State University; Shaoliang Nie, North Carolina State University; Tianfu Wu, North Carolina State University; Christopher G. Healey, North Carolina State University.

visual patterns [Geman et al. 2002], target regions in the former usually have a high chance of finding similar patterns in either the surrounding context of the same image or images in an external image dataset subject to the context. Target regions in the latter are more specific, especially when large portions of essential parts of an object are missing (e.g., facial parts in Figure 1). So, the completion entails fine-grained understanding of the semantics, structures and appearance of images, and thus is a more challenging task. Face images have become one of the most popular source of images collected in people's daily lives and transmitted on social networks. We focus on human face completion in this paper.

Two broad frameworks have been proposed in the literature of image completion: data similarity driven methods and data distribution based generative methods. In the first paradigm, texture synthesis or patch matching are usually used [Barnes et al. 2009; Criminisi et al. 2003; Darabi et al. 2012; Efros and Leung 1999; Huang et al. 2014; Komodakis 2006; Kwatra et al. 2003; Wexler et al. 2007; Wilczkowiak et al. 2005]. Textures or patches are generated by finding similar exemplars in the known contexts and then stitched together to fill in the "holes". An alternative is the data-driven method [Hays and Efros 2007], which searches a large image database for plausible patches based on context similarity. These methods are often utilized for generic scene image completion and their limitations are obvious. They are bound to fail when no similar exemplars can be found in either the context or the external dataset, and thus are not applicable to face completion (as well as other objects) as pointed out in [Iizuka et al. 2017; Yeh et al. 2017]. Instead of seeking similar exemplars, the second paradigm learns the underlying distribution governing the data generation with respect to the context. Much progress [Denton et al. 2016; Iizuka et al. 2017; Li et al. 2017; Pathak et al. 2016; Yang et al. 2016; Yeh et al. 2017] has been made since the recent resurgence of deep convolutonal neural networks (CNNs) [Krizhevsky et al. 2012; LeCun et al. 1989], especially the generative adversarial network (GAN) [Goodfellow et al. 2014].

We adopt the data distribution based generative method in this paper and address three important issues. *First*, previous methods are only able to complete faces at low resolutions (e.g. 128×128 [Li et al. 2017] and 176×216 [Iizuka et al. 2017]). *Second*, most approaches cannot control the attributes of the synthesized content. Previous works focus on generating realistic content. However, users may want to complete the missing parts with certain properties (e.g. smiling or not). *Third*, most existing approaches require post processing or complex inference process. Generally, these methods [Iizuka et al. 2017; Li et al. 2017; Yeh et al. 2017] synthesize relatively low quality images from which the corresponding contents are cut and blended (e.g. with Poisson Blending [Pérez et al. 2003]) with the original contexts. In order to complete one image, other approaches [Yang et al. 2016; Yeh et al. 2017] need run thousands of optimization iterations or feed an incomplete image to CNNs repeatedly at multiple scales.

To overcome the above limitations, we propose a novel fully end-to-end progressive GAN to complete face images in high-resolution with multiple controllable attributes (see Figure 1). Our network is able to complete masked faces with high quality in a single forward pass without any post processing. It consists of two sub-networks: a completion network and a discriminator. Given face images with missing contents, the completion network tries to synthesize completed images that are indistinguishable from uncorrupted real faces, while keeping their contexts unchanged. The discriminator is trained simultaneously with the completion network to distinguish completed "fake" faces from real ones. Unlike most existing works [Denton et al. 2016; Iizuka et al. 2017; Li et al. 2017; Yang et al. 2016] that use the Encoder-Decoder structures, we introduce a new architecture based on the U-Net [Ronneberger et al. 2015] that better integrates information across all scales to generate higher quality images. Moreover, we design new loss functions inducing the network to blend the synthesized content with the contexts in a realistic way. We adopt the training methodology of growing GANs progressively [Karras et al. 2017] to generate high-resolution images. A conditional version of our network is also designed so that N attributes of the synthesized faces can be controlled by N-dimensional vectors (Figure 1). We compared our method with state-of-the-art approaches on a high-resolution face dataset CelebA-HQ [Karras et al. 2017]. The results of both qualitative evaluation and a pilot user study showed that our approach completed face images significantly more naturally than existing methods, with improved efficiency.

The main contributions of this paper are:

- We propose a novel approach that consolidates information across all scales to complete face images with arbitrary masks in much higher resolution than existing methods.
- We further design a conditional version of our architecture to control multiple attributes of the synthesized content.
- Our framework is able to complete images in a single forward pass, without any post-processing.

2 RELATED WORK

2.1 Image Generation

Generative models have been studied extensively to synthesize realistic and novel images by learning high-dimensional data distributions. Current work falls into three groups: variational autoencoders (VAE) [Kingma et al. 2016; Kingma and Welling 2013], autoregressive models [Oord et al. 2016; van den Oord et al. 2016] and GAN [Goodfellow et al. 2014]. VAE is easy to train, but the generated images are often blurry. Autoregressive models can synthesize sharp images, but they lack latent representations, which makes it more difficult to control the attributes of generated images than VAE or GAN. Additionally, the evaluation of autoregressive models is slow. GAN is able to generate sharp images from latent vectors (e.g. a 100-dimensional noise vector), whose architecture typically consists of two networks: a generator and a discriminator. The generator learns to synthesizes images that are indistinguishable from the training data while the discriminator is trained to differentiate between the generated images and real ones. The generator and discriminator are trained simultaneously and thus the training process is usually unstable.

Many methods try to stabilize GAN and generate high quality images by designing new objective functions and architectures. The Deep Convolutional GAN (DCGAN) [Radford et al. 2015] produces images using a set of fractional-strided convolutions. The Laplacian GAN [Denton et al. 2015] uses a Laplacian pyramid to generate

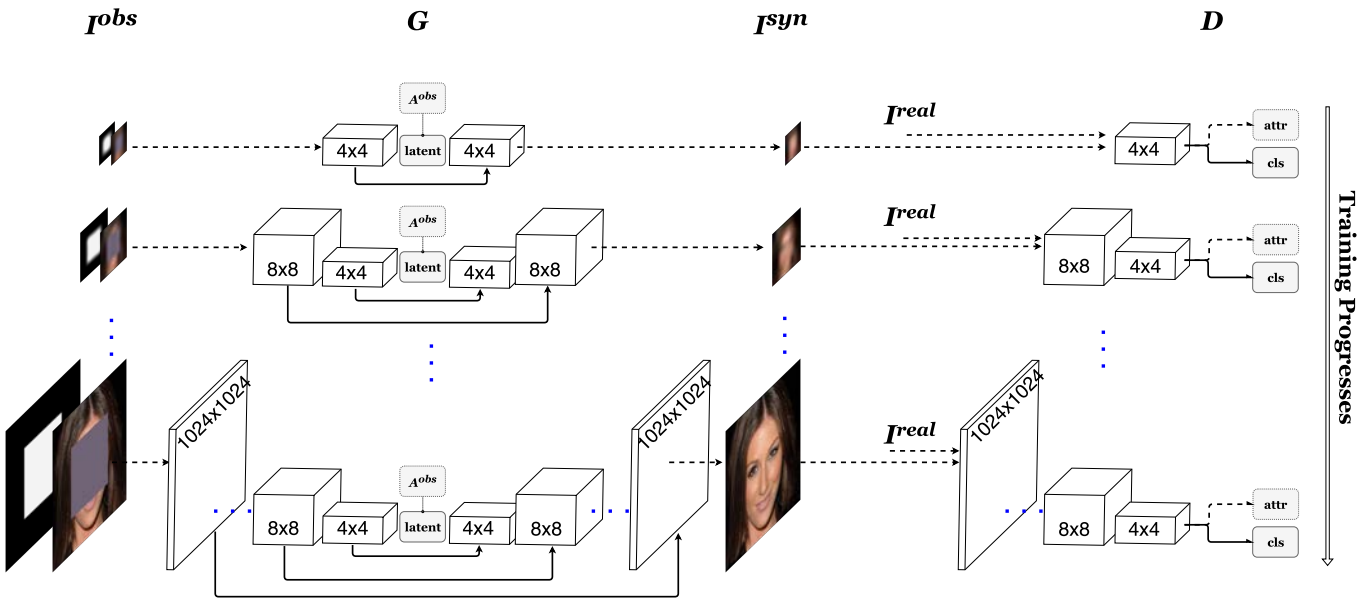

Fig. 2. The overall architecture and training process of our approach. The training of the completion network (or the "generator" G) and the discriminator D starts at low resolution (4×4). Higher layers are added to both G and D progressively to increase the resolution of the synthesized images. The r x r cubes in the figure represent convolutional layers that handle resolution r. For the conditional version, attribute labels A^{obs} are concatenated to the latent vectors. The discriminator D splits into two branches in the final layers: D_{cls} that classifies if an input image is real, and D_{attr} that predicts attribute vectors.

images from coarse to fine by adding high frequency information at different layers. Unfortunately, these techniques are unable to synthesize high-resolution images. The main challenge is that, in higher resolutions, the discriminator can tell the differences between real and fake images more accurately, which causes a vanishing gradient problem. Recently, Xiang et al. [Xiang and Li 2017] proposes a weight normalization approach and achieves better training performance. Karras et al. [Karras et al. 2017] put forward a progressive training mechanism to grow both the generator and discriminator from low to high resolution, and are able to generate realistic 1024×1024 images. The advantage of this methodology is that the networks do not have to handle information across all image scales at the same time, and instead can learn the holistic image structures first and then focus on producing finer details progressively. However, GANs cannot be applied to the image completion task directly because they aim at generating random natural images, but are not constrained by the image context.

2.2 Image Completion

There is a large body of image completion literature. Early non-learning based algorithms [Bertalmio et al. 2000, 2003; Efros and Leung 1999] complete missing content by propagating information from known neighborhoods, based on low level cues or global statistics [Levin et al. 2003]. Texture synthesis and patch matching based approaches [Barnes et al. 2009; Criminisi et al. 2003; Darabi et al. 2012; Efros and Leung 1999; Huang et al. 2014; Komodakis 2006; Kwatra et al. 2003; Wexler et al. 2007; Wilczkowiak et al. 2005] find similar structures from the context of the input image or from an external database [Hays and Efros 2007] and then paste them to fill in the holes. These methods assume that similar textures or patches of the missing contents can be found in the known regions. Recent work [Iizuka et al. 2017; Pathak et al. 2016; Yeh et al. 2017] has compared learning based methods with aforementioned approaches to produce large missing content. The results show that non-learning based approaches often synthesize content that is inconsistent with the global structures (e.g. using *mouths* to fill in holes at *eye* locations) while the learning based models can produce reasonable results.

Many researchers focus on the face completion problem. The Graph Laplace method [Deng et al. 2011] uses a spectral-graph-based algorithm to repair occluded face images. The Visio-lization [Mohammed et al. 2009] completes faces with realistic and variant characteristics using a combination of global and local models. However, these approaches can handle only low-resolution images with limited shapes of masks.

Recent learning based methods have shown the capability of CNNs to complete large missing content. The completion models are different from the generative models (e.g. GANs): the former need to complete corrupted images with plausible content while the latter focus on generating completely fake, yet realistic images from latent vectors. Based on existing GANs, the Context Encoder (CE) [Pathak et al. 2016] encodes the contexts of masked images to latent representations, and then decodes them to natural content images, which are pasted into the original contexts for completion. However, the images generated by CE are often blurry and have inconsistent boundaries along the seams between content and context. Given a trained generative model, Yeh et al. [Yeh et al. 2017]

proposed a framework to find the most plausible latent representations of contexts to complete masked images. But this work depends heavily on the quality of the pre-trained generative models. The Generative Face Completion model (GFC) [Li et al. 2017] and the Global and Local Consistent model (GL) [Iizuka et al. 2017] use both global and local discriminators, combined with post processing, to complete images more coherently. Though GFC and GL models are trained with random rectangular masks, they can handle masks with arbitrary shapes as well. Unfortunately, these two approaches can only complete face images in relatively low resolutions (e.g. 176 x 216 [Iizuka et al. 2017]). Yang et al. [Yang et al. 2016] combined a global content network and a texture network, and trained networks at multiple scales repeatedly to complete high-resolution images (512×512). Like the patch matching based approaches, Yang et al. assume that the missing content always shares some similar textures with the context, which is improbable for the face completion task.

2.3 Generative Models with Controllable Attributes

There are many researchers studying how to control the attributes of synthesized images from generative models. The Conditional GAN (CGAN) [Mirza and Osindero 2014] trains the networks conditioned on attribute vectors (e.g. one-hot class labels) that are used to control properties of produced images explicitly during evaluation. For instance, the trained model of CGAN on MNIST [LeCun 1998] can generate images of digits zero to nine depending on a label vector. Kaneko et al. [Kaneko et al. 2017] extend this work and design multi-dimensional controllers to manipulate the properties (e.g. young or old) of face images. Olszewski et al. [Olszewski et al. 2017] generates dynamic textures for a target face by referencing a source video sequence. Unlike the CGANs that make the generators or discriminators conditioned on the attribute code directly, the information model (InfoGAN) [Chen et al. 2016] is able to control continuous (e.g. width of digit) and discrete (e.g. category) attributes of images by preserving the attribute information during the generation process. In additional to distinguishing real images from fake ones, InfoGAN uses auxiliary networks to check whether the latent information predicted from the generated images is close to the input latent code. The Categorical GAN (CatGAN) [Springenberg 2015] controls the categories of generated images based on the assumption that real images have peaked class distributions while fake images should be uniformly distributed. Salimans et al. [Salimans et al. 2016] change the discriminator to a "*K+1*" classifier by adding a "generated" class to the original K image classes. Their model is able to generate images of multiple classes with one generative model. Recent image-to-image translation networks are able to transfer images to other domains by explicitly [Choi et al. 2017] or implicitly [Isola et al. 2016; Zhu et al. 2017] learning image characteristics. However, there are few completion models that are able to manipulate the properties of synthesized contents. Our conditional completion model is built on these generative approaches and can complete corrupted images with multiple controllable attributes.

3 APPROACH

In this section, we first formulate the problem of image completion. Then, we present details of the proposed fully end-to-end progressive generative adversarial network. The overall structure of our networks is shown in Figure 2.

3.1 Problem Formulation

Denote by Λ an image lattice (e.g., 1024×1024 pixels). Let I_Λ be an RGB image defined on the lattice Λ. Denote by Λ_t and Λ_c the target region to complete and the remaining context region respectively which form a partition of the lattice, i.e., $\Lambda_t \cap \Lambda_c = \emptyset$ and $\Lambda_t \cup \Lambda_c = \Lambda$. Without loss of generality, we assume Λ_t is a single connected component region (e.g., a rectangular mask in Figure 1). I_{Λ_t} is masked out with the same gray pixel value. Let M_Λ be a binary mask image with all pixels in M_{Λ_t} being 1 and all pixels in M_{Λ_c} being 0. For notational simplicity, we will omit the subscripts Λ, Λ_t and Λ_c when the text context is clear.

The objective of image completion is to generate a synthesized image I^{syn} that looks natural, realistic and appealing for an observed image I^{obs} with the target region $I^{obs}_{\Lambda_t}$ masked out. Furthermore, the generator can be controlled with respect to a set of attributes which are assumed to be independent from each other. Let $A = (a_1, \cdots, a_N)$ be a N-dim vector with $a_i \in \{0, 1\}$ encoding if a corresponding attribute appears ($a_i = 1$) or not ($a_i = 0$) such as the male and smiling attributes in Figure 1. We define the generator as,

$$I^{syn} = G(I^{obs}, M, A; \theta_G) \tag{1}$$

where θ_G collects all parameters of the generator (to be elaborated later), and the context regions, $I^{syn}_{\Lambda_c}$ and $I^{obs}_{\Lambda_c}$, are kept very similar.

Following the data distribution based generative methods, the generator needs to tightly approximate the underlying conditional probability model $p_G(I^{syn}|I^{obs}, M, A)$ in the high dimensional huge image space, and to implement a one-pass sampler which can generate a typical sample from the probability model (Eqn. 1), thus learning the generator $G(\cdot)$ is an extremely difficult task.

3.2 The Proposed Fully End-to-End Progressive GAN

Thanks to the recently proposed Generative Adversarial Networks (GAN) [Goodfellow et al. 2014], a generator that learns data distribution and computes typical sample can be trained under a minimax game setting. Denote by $p_G(I)$ and $p_{data}(I)$ the generator distribution and the data distribution respectively, and the former is trained to match the latter. GAN parametrizes $p_G(I)$ using a generator network G which transforms a noise random variable (e.g., white noise) z into a sample $G(z)$, overcoming the challenges of trying to compute probability to every I in the data distribution in an explicit way. Under the minimax game, an adversarial discriminator network D is trained simultaneously which aims to tell the generated sample $G(z)$ apart from the real data based on binary classification between real and fake. For a given generator G, the optimal discriminator is $D(I) = \frac{p_{data}(I)}{p_{data}(I)+p_G(I)}$ under the Nash equilibrium. In the work of Goodfellow et al. [Goodfellow et al. 2014], the minimax game is

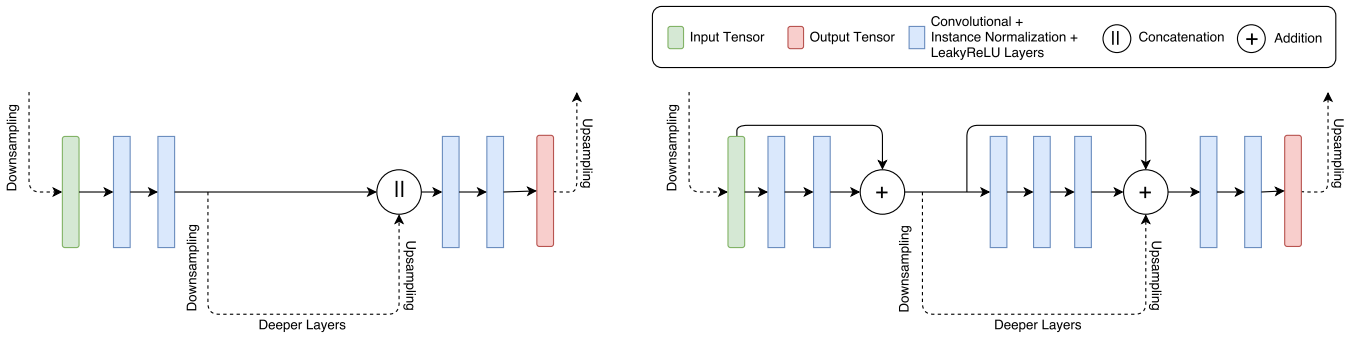

Fig. 3. Illustrations of a single layer of our architecture. There are skip connections between mirrored encoder and decoder layers. Left: the structure of the completion network; the skip connection is a copy-and-concatenate operation. This structure helps preserve the identity information between the synthesized images and real faces, resulting in little deformation. Right: the structure of the conditional completion network; residual connections are added to the encoder, and the skip connections are residual blocks instead of direct concatenation. The attributes of the synthesized contents can be manipulated more easily with this structure. Each blue rectangle represents a set of Convolutional, Instance Normalization and Leaky Rectified Linear Unit (LeakyReLU) [Maas et al. 2013] layers.

formulated by,

$$\min_G \max_D \mathcal{L}_{adv}(G,D) = E_{z\sim p_{noise}(z)}[1-\log(1-D(G(z)))] + E_{I\sim p_{data}(I)}[\log D(I)] \quad (2)$$

where $\mathcal{L}_{adv}(G,D)$ is the adversarial loss function and $E[\cdot]$ represents the expectation. The proposed method is built on the GAN.

In our model, the generator takes as input the observed corrupted image, the binary mask image and the attribute vector and outputs a completed image. It consists of two components,

$$G(I^{obs}, M, A; \theta_G) = G_{compl}(G_{enc}(I^{obs}, M; \theta_G^{enc}), A; \theta_G^{compl}) \quad (3)$$

where $G_{enc}(\cdot)$ encodes an input pair (I^{obs}, M) to a latent low dimensional vector. The latent vector is concatenated with the attribute vector. The concatenated vector plays the role of the noise random variable z in the original GAN. Then, $G_{compl}(\cdot)$ transforms the concatenated vector to a sample (i.e., the completed image). G_{enc} and G_{compl} are mirrored to each other. $\theta_G = (\theta_G^{enc}, \theta_G^{compl})$.

In our model, the discriminator takes as input either the ground-truth uncorrupted image or the completed image from the generator and has two output branches. It consists of three components: a shared feature backbone and two head classifiers. We have,

$$D(I;\theta_D) = \{D_{cls}(F(I;\theta_D^F);\theta_D^{cls}), D_{attr}(F(I;\theta_D^0);\theta_D^{attr})\} \quad (4)$$

Where the feature backbone, $F(I;\theta_D^0)$ computes the feature map for an input image. On top of the feature map, the first head classifier, $D_{cls}(I;\theta_D^{cls}) = D_{cls}(F(I;\theta_D^F);\theta_D^{cls})$ computes binary classification between real and fake, and the second one, $D_{attr}(I;\theta_D^{attr}) = D_{attr}(F(I;\theta_D^F);\theta_D^{attr})$ predicts an attribute vector. All the parameters of the discriminator are collected by $\theta_D = (\theta_D^F, \theta_D^{cls}, \theta_D^{attr})$. $\theta = (\theta_G, \theta_D)$ will be learned end-to-end. We note that the discriminator is only needed in training. We will omit the notations for parameters in equations when no confusion is caused.

Next, we elaborate on the details of training which includes the generation of I^{obs} and its attribute A^{obs}, the loss functions, the network architectures and the progressive training we propose for high resolution face completion.

3.2.1 Generating I^{obs} and A^{obs}. Let I_Λ^{real} and A^{real} be an uncorrupted face image and its annotated attribute vector. To generate I_Λ^{obs}, we first sample a mask M, then use it to mask out the target regions. We use approaches similar to the one proposed by the context encoder method [Pathak et al. 2016]. First, starting with an all-zero one-channel image, a rectangular region of random size and location is chosen. Second, a low resolution noise (e.g. 4×4, drawn from uniform distribution) image is generated and then up-sampled to the size of the chosen rectangle with bi-linear interpolation. In this way, we can construct a rectangular region with continuous random values. Then the image is converted to a binary mask with thresholding. Denote by $M \sim p_{mask}(M)$ a mask sample. Since we only know the occurrence of attributes in I_Λ^{real}, we can not infer how a mask M affects the occurrence. To create the attribute vector A^{obs} for (I_Λ^{obs}, M), we define it as a fake attribute vector,

$$A^{obs} = \begin{cases} A^{real}, & \text{if } p < 0.5. \\ (a_1, \cdots, 1-a_i, a_{i+1}, \cdots, a_N); \forall j, a_j \in A^{real}, & \text{otherwise.} \end{cases} \quad (5)$$

where $p \sim U(0,1)$ and $i \in [1, N]$ is a randomly chosen index. We will denote by $A^{obs} \sim p_{attr}(A^{real})$ an attribute vector sample.

3.2.2 Loss Functions. Beside extending the original adversarial loss function, we design three new loss functions to enforce sharp image completion.

Adversarial Loss. Given an uncorrupted image I^{real}, its attribute vector A^{real}, a mask M and the corresponding corrupted image I^{obs}, and a fake attribute vector A^{obs}, we define the loss by,

$$l(I^{real}, M, I^{obs}, A^{obs}|G,D) = (1-\log(1-D_{cls}(I^{syn})) + \log D_{cls}(I^{real}) \quad (6)$$

where $I^{syn} = G(I^{obs}, M, A^{obs})$). Similar to Eqn. 2, we have the expected loss,

$$\mathcal{L}_{adv}(G, D) = E_{\substack{I^{real}\sim p_{data}(I),\\ M\sim p_{mask}(M),\\ A^{obs}\sim p_{attr}(A^{real})}} [l(I^{real}, M, I^{obs}, A^{obs}|G, D)] \tag{7}$$

In the following, we will omit definitions of the expected losses of different loss terms.

Attribute Loss. For the attribute prediction head classifier in the discriminator, we define the attribute loss based on cross-entropy between the predicted attribute vector, $\hat{A}^{real} = D_{attr}(I^{real})$ and $\hat{A}^{obs} = D_{attr}(I^{obs})$ and the corresponding targets, A^{real} and A^{obs} for a real uncorrupted image and a synthesized image respectively. We have,

$$\begin{aligned} l_{attr}(I^{real}, A^{real}, M, I^{obs}, A^{obs}|G, D) = \\ \sum_{i=1}^{N}(a_i^{real} \log \hat{a_i}^{real} + (1 - a_i^{real}) \log(1 - \hat{a_i}^{real}))+ \\ \sum_{i=1}^{N}(a_i^{obs} \log \hat{a_i}^{obs} + (1 - a_i^{obs}) \log(1 - \hat{a_i}^{obs})). \end{aligned} \tag{8}$$

Reconstruction Loss. Since our method generates the entire completed face rather than only the target region, we define a wighted reconstruction loss l_{rec} to preserve both the target region and the context region, which is defined as,

$$\begin{aligned} l_{rec}(I^{real}, M, I^{obs}, A^{obs}|G) = \|\alpha \cdot M \cdot (I^{real} - I^{syn})\|_1 + \\ \|(1 - \alpha) \cdot (1 - M) \cdot (I^{real} - I^{syn})\|_1. \end{aligned} \tag{9}$$

where $\odot$ represents element-wise multiplication and α is the trade-off parameter.

Feature Loss. In additional to the reconstruction loss in terms of pixel values, we also encourage the synthesized image to have a similar feature representation [Johnson et al. 2016] based on a pre-trained deep neural network ϕ. Let ϕ_j be the activations of the jth layer of ϕ, the feature loss is defined by

$$l_{feat}(I^{real}, M, I^{obs}, A^{obs}|\phi, G) = \|\phi_j(I^{real}) - \phi_j(I^{syn}))\|_2^2. \tag{10}$$

In our experiments, ϕ_j is the $relu2\_2$ layer of a 16-layer VGG network [Simonyan and Zisserman 2014] pre-trained on the ImageNet dataset [Russakovsky et al. 2015].

Boundary Loss. To make the generator learn to blend the synthesized target region with the original context region seamlessly, we further define a close-up reconstruction loss along the boundary of the mask. Similar to [Yeh et al. 2017], we first create a weighted kernel w based on the mask image M. w is computed by blurring the mask boundary in M with a mean filter so that the pixels closer to the mask boundary are assigned larger weights. The kernel size of the mean filters is seven in our experiments.

$$l_{bdy}(I^{real}, M, I^{obs}, A^{obs}|G) = \|w \odot (I^{real} - I^{syn})\|_1. \tag{11}$$

Our model is trained end-to-end by integrating Eqn. 7 and the expected losses of Eqn. 8, Eqn. 9, Eqn. 10 and Eqn. 11 under the minimax game setting. We have,

$$\begin{aligned} \min_G \max_D \mathcal{L}(G, D) = \mathcal{L}_{adv}(G, D) + \lambda_{attr} \cdot \mathcal{L}_{attr}(G, D) + \\ \lambda_{rec} \cdot \mathcal{L}_{rec}(G) + \lambda_{feat} \cdot \mathcal{L}_{feat}(G, \phi) + \lambda_{bdy} \cdot \mathcal{L}_{bdy}(G). \end{aligned} \tag{12}$$

Where $\lambda.$'s are trade-off parameters between different loss terms.

Training without Multiple Controllable Attributes. To that end, since it is a special case of the proposed formulation stated above, we can simply remove the components involving attributes such as the attribute loss in a straightforward way. The resulting system still enjoys end-to-end learning.

3.2.3 Network Architectures. As illustrated in Figure 2, the generator G in our model is implemented by a U-shape network architecture consisting of the first component G_{enc} transforming the observed image and its mask to a latent vector and the second component G_{compl} transforming the concatenated vector (latent and attribute) to the completed image. There are residual connections between layers in G_{enc} and the counterpart in G_{compl} similar in the spirit to the U-Net [Ronneberger et al. 2015] and the Hourglass network [Newell et al. 2016] to consolidate information across multiple scales. Figure 3 illustrates the two structures of a layer in the generator for training without and with attributes respectively, which are adapted from the U-Net and Hourglass network. *Detailed specifications of the generator and the discriminator will be provided in the supplementary material.*

3.2.4 Progressive Training. To address the challenge of stabilizing the training of GANs with faster convergence rates, we follow the very recent work on training GANs progressively [Karras et al. 2017]. It starts with the lowest resolution (i.e. 4×4). After running a certain number of iterations, higher resolution layers are added to both the generator G and discriminator D at the same time. To avoid sudden changes to the trained parameters, the added layers are faded into the networks smoothly. All parameters are still trainable during the growing of networks.

At a resolution lower than 1024×1024, the input face images, masks and real images are all down-sampled with average pooling to fit the given scale. The advantage of progressive training is that the networks can learn the structures of missing contents from coarse to fine incrementally. The holistic and local information are aggregated at multiple spatial scales. We note that since we do not need the global-and-local discriminator structure like previous works [Iizuka et al. 2017; Li et al. 2017], the sizes and shapes of our masks could be arbitrary during the training, instead of being limited to rectangular regions with certain sizes.

One of the major challenges of generating high resolution images is the limitation of Graphics Processing Unit (GPU) memory. Most completion networks use Batch Normalization [Ioffe and Szegedy 2015] to avoid covariate shift. However, with the limited GPU memory, only a small number of batch sizes are supported at high resolution, resulting in low quality of generated images. We use the Instance Normalization [Ulyanov et al. 2016], similar to Zhu et al. [Zhu et al. 2017], and update D with a history of completed images instead of the latest generated one [Shrivastava et al. 2016] to stabilize training.

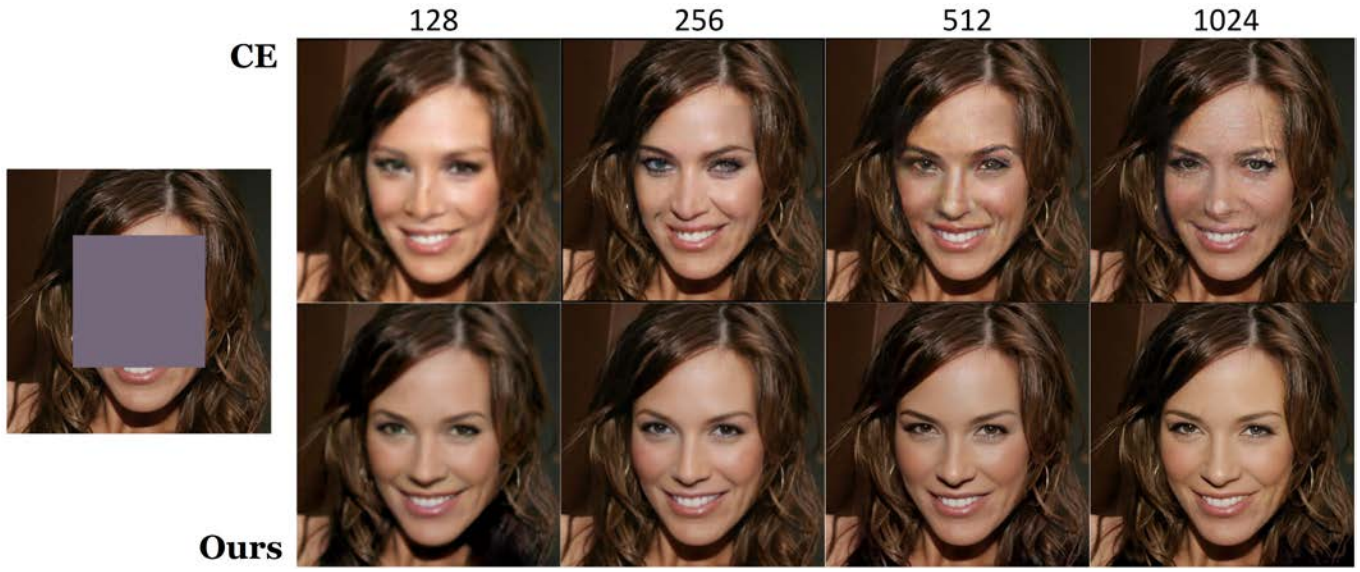

Fig. 4. Comparison with Context Encoder on high-resolution face completion. The top row are images generated by CE and the bottom row are our results. With increasing resolution (from 128×128 to 1024×1024), CE generated more distorted images while our method produced sharper faces with more details.

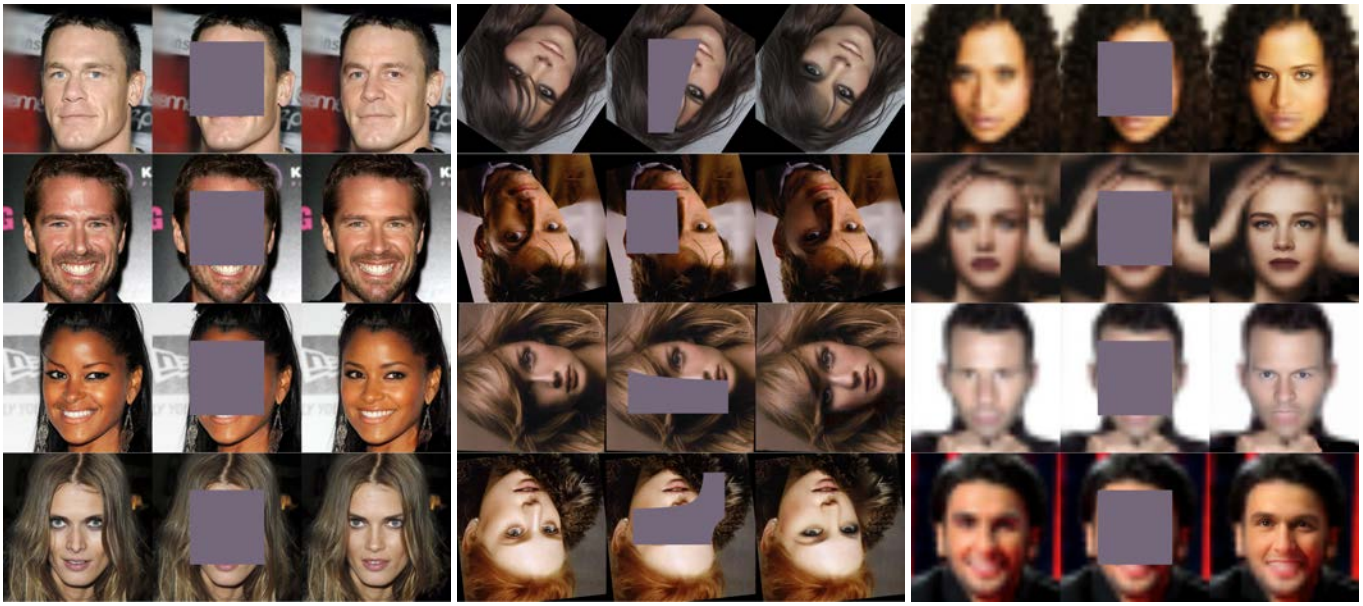

Fig. 5. Examples of high-resolution face completion results from our approach. All images are at 1024×1024 resolution. For each group, the left-most column are real images, the middle column are masked images and the right-most column are images synthesized by our model. Left group: images are masked with 512×512 holes in the center. Middle group: images are randomly flipped, rotated and covered by masks with arbitrary shapes and sizes. Right group: images with blurry contexts are completed by a model trained on clear contexts.

4 EXPERIMENTS

In this section, we first demonstrate our models' ability to complete high-resolution face images in several challenging scenarios through experiments. Additionally, we show examples of controlling the attributes of synthesized faces. In the end, we compare our method with state-of-the-art approaches in low resolution with a pilot user study.

4.1 Datasets and Experiment Settings

We used the CelebA-HQ [Karras et al. 2017] dataset for evaluation. It contains 30,000 aligned face images at 1024×1024 resolution. Similar

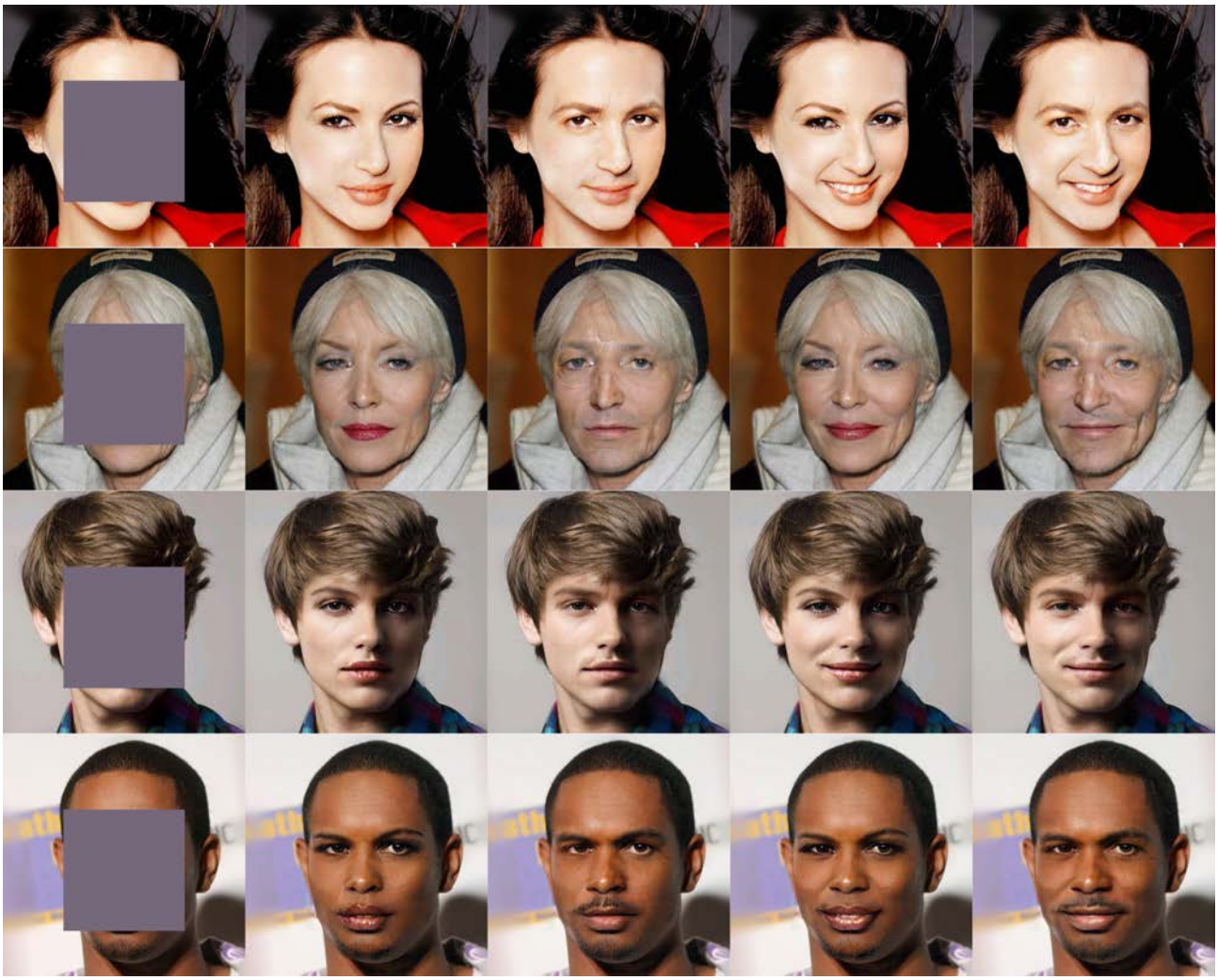

Fig. 6. Examples of images generated by our conditional model. All images are at 512×512 resolution. The leftmost column are masked images, and the rest are generated faces. The characteristics of synthesized images can be controlled by attribute vectors explicitly. Attributes ["Male", "Smiling"] are used in this example. The attribute vectors of column two to five are [0,0], [1,0], [0,1], and [1,1] ("1" denotes yes while "0" denotes no). Our model also learns to add detailed features to the context to make the face properties more consistent with the attribute labels, for instance adding chin beard to "Male" images in the second row, without affecting the holistic structures.

to previous methods [Yeh et al. 2017], we split the dataset randomly: 3,000 images for testing, and the remaining 27,000 for training. The CelebA-HQ was chosen over the original CelebA [Liu et al. 2015] dataset not only because CelebA-HQ has higher resolution images, but also because it is a cleaner dataset with significantly fewer artifacts and more consistent quality. The images were scaled to 128×128 for the user study and 512×512 for the attribute controlling task. The remaining experiments used 1024×1024 images. There were two types of masks: center and random. The center mask was a square region in the middle of the image with a side length of half the size of the image. The random masks, which were generated similar to the method of Pathak et al. [Pathak et al. 2016], were mostly continuous regions with arbitrary shapes, sizes and locations and covered about 10% to 30% of the original images.

In the experiments, the reconstruction trade-off parameter was set to $\alpha = 0.7$ to focus more on the target region. To balance the effects of different objective functions, we used $\lambda_{attr} = 2$, $\lambda_{rec} = 500$, $\lambda_{feat} = 10$, and $\lambda_{bdy} = 5000$. The Adam solver [Kingma and Ba 2014] was employed with a learning rate of 0.0001.

4.2 High-Resolution Face Completion

4.2.1 Comparison with the Context Encoder. Our method was compared with the Context Encoder (CE) [Pathak et al. 2016] on high-resolution face completion. Since the original networks of CE were designed for 128×128 images, we used a naive approach to

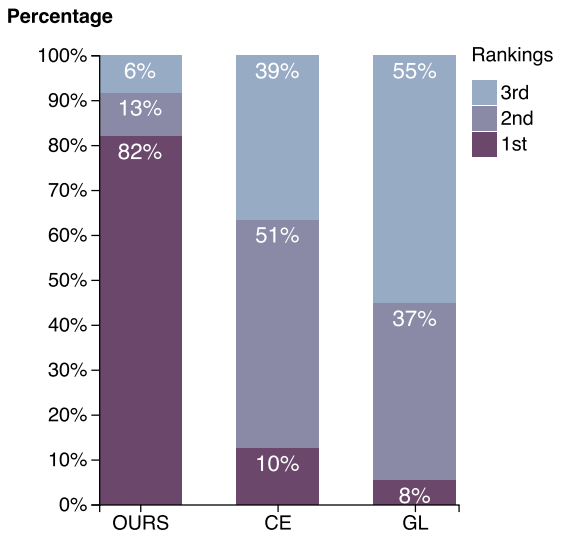

Fig. 7. The result of the user study to compare the naturalness of completion of three methods: ours, GL and CE. The three colors in each bar from bottom to top represent the percentage of each method being ranked first, second, and third. There are significantly more face images generated by our method being ranked the first than the other two approaches.

Table 1. The pairwise t-test results of the user study. Our method was ranked first significantly more often than either CE or GL. There was no statistically significant difference in the likelihood of CE being ranked first versus GL.

Method	Vs. Ours	Vs. CE	Vs. GL
Ours	-	t(31)=20.21	t(31)=18.65
	-	**p<0.001**	**p<0.001**
CE	t(31)=20.21	-	t(31)=0.59
	p<0.001	-	p=0.82
GL	t(31)=18.56	t(31)=0.59	-
	p<0.001	p=0.82	-

fit it to different resolutions. One, two, and three convolutional layers were added to the encoder, decoder and discriminator for 256×256, 512×512 and 1024×1024 networks respectively. The result (Figure 4) shows that, when the resolution increased, our method learned details incrementally and synthesized sharper faces, while CE generated poorer images with more distortions.

4.2.2 Semantic Completion. We first trained a high-resolution (1024×1024) model with center masks (examples shown in Figure 5). In order to test whether our model actually learned high-level semantics and structures of faces, or simply "remembered" face examples, we designed two more challenging experiments.

In the *first* experiment, training images were randomly flipped, rotated and covered with random masks. A new model was trained from scratch on this training set. The result (Figure 5) shows that our model was able to capture the anatomical structures of faces and generate content that is consistent with the holistic semantics.

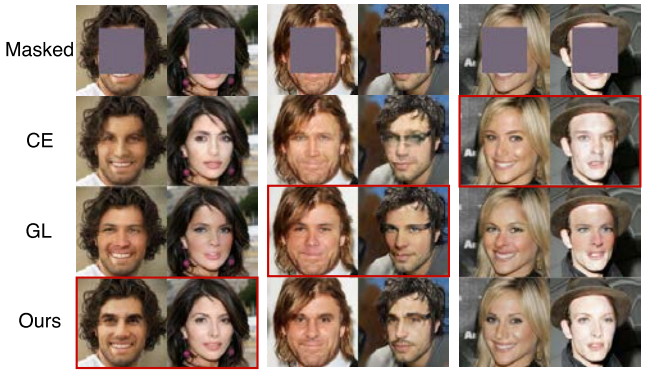

Fig. 8. Examples of images generated by different methods. The images being ranked first by users are annotated with red boxes.

In the *second* experiment, we studied the ability of our model to infer high-resolution content from blurry contexts. A model pre-trained on clear contexts was used for this task. The testing images were down-sampled to 32×32 from original size with average pooling, and then up-sampled to 1024×1024 using bilinear interpolation. The result (Figure 5) demonstrates that our model also learned up-sampling information and was insensitive to blurry contexts.

4.2.3 Attribute Controller. Our network can be converted to a conditional version to control the attributes of generated images. Unlike the traditional image completion techniques that aim at reconstructing the missing parts so that they look similar to the original content, our goal is to complete faces with structurally meaningful content whose properties are controllable depending on the input attribute vectors, while making minimum changes to the context. In our experiment, two attributes ("Male" and "Smiling") were chosen. This model was trained from scratch and the result was run at a 512×512 resolution (Figure 6). The result shows that the attributes of synthesized images were controlled by our model explicitly. Additionally, the model learned to add detailed features (e.g. chin beard for the "Male" label) to the context to make the properties of synthesized faces more consistent with their attribute code, without affecting the holistic structures.

4.2.4 Computation Time. Existing CNN-based high-resolution in-painting approaches often need significant time to process an image. For instance, it took about 1 min for the model of Yang et al. [Yang et al. 2016] to fill in a 256×256 hole of a 512×512 image with a Titan X GPU.

The advantage of our method is that our model, once trained, is able to complete a face image with a single forward pass, resulting in much higher efficiency. We tested the computation time of our model with a Titan Xp GPU by processing 3000 1024×1024 images with 512×512 holes. The mean completion time of one image is 0.007 seconds with a standard deviation of 0.0005 seconds.

4.3 User Study

We compared our method with two state-of-the-art CNN-based face completion approaches, CE and the Globally and Locally Consistent Method (GL) [Iizuka et al. 2017], with a pilot user study

at 128 × 128 resolution with center masks. For GL, we used Poisson Blending [Pérez et al. 2003] as post-processing. 32 subjects (21 male and 11 female participants, with ages from 22 to 36), including faculty and students, were volunteered to participate.

For each trial, we randomly selected one synthesized image from each method and presented them on screen with permuted order. A user was asked to rank the three faces based on how realistic they looked ("1" denoted the best while "3" denoted the worst). Each experiment started with a training session to help users become familiar with our user interface. The formal experiment consisted of 100 trials and there was no time limit. Most users finished the experiments within 20 minutes.

The result (Figure 7) shows that there were significantly more images generated by our method being favored by the viewers. Figure 8 shows some examples of face images produced by different methods. Overall, our approach generated sharper images with more details. However, sometimes users thought blurry images (e.g. female faces generated by CE) were more appealing. Without careful post processing, GL had a higher chance of producing content with inconsistent colors, for instance generating reddish faces while the contexts had pale skin. But, GL were better at object removal (e.g. removing glasses).

In order to confirm the intuition of our ranking results, we tested for statistical significance. To do this, we first collapsed each participant's rankings into a frequency list. For example, if a participant ranked our images first 77 times, CE images 12 times, and GL images 11 times, the frequency list would be 77, 12, 11. Given 32 participants, this resulted in 32 averages over 3200 samples.

Once frequency lists were built for all participants, the frequencies for each method (ours, CE, and GL) were again averaged over the 32 participants to produce a final list of averages from $n = 32$ samples.

Next, we used Welch's analysis of variance (ANOVA) [McDonald 2009] to test for statistically significant differences between the three ranking frequencies. We chose Welch's ANOVA rather than a standard ANOVA since we could not guarantee homogeneity of variance. Not surprisingly, results showed a significant difference in means for a standard $\alpha = 0.5$, with $F(2, 29) = 213.6, p < 0.001$.

Given a significant ANOVA, we concluded by computing Games-Howell post-hoc pairwise tests (Table 1) to see which methods' means differed significantly from one another (Games-Howell corrects for non-homogeneity of variance [Ruxton and Beauchamp 2008]). Results showed our method was significantly more likely to be ranked first versus both CE and GL. There was no statistically significant difference in the likelihood of CE being ranked first versus GL.

These results confirm that our method was ranked first significantly more often than either CE or GL.

4.4 Limitations

Though our method has low inference time, the training time is long due to the progressive growing of networks. In our experiment, it took about three weeks to train a 1024 × 1024 model on a Titan Xp GPU.

By carefully zooming in and inspecting our results, we find that our high-resolution model fails to learn low-level skin textures, such as furrows and sweat holes. Additionally, our model tends to generate blurry content while the context has abundant detailed textures (e.g. freckles). Moreover, occasionally, the model is unable to capture the symmetrical structure of faces (e.g. generating eyes with different colors). Some failure cases are shown in Figure 9. These issues are left for future work.

Fig. 9. Some failure cases of our approach. Our model tends to generate blurry images if the context has rich skin textures like freckles and wrinkles. It may also generate asymmetrical contents, for instance two eyes with different colors. The leftmost column are real images and the rightmost are synthesized faces by our approach.

5 CONCLUSION

We propose a deep learning approach for high-resolution face completion. Our model is trained progressively [Karras et al. 2017] and learns face structures from coarse to fine. By consolidating information across all scales, our model not only outperforms state-of-the-art methods by generating sharper images in low resolution, but is also able to synthesize faces in higher resolutions than existing techniques. A conditional version of our architecture allows users to control the properties of synthesized images explicitly with attribute vectors. Additionally, our architecture is designed in an end-to-end manner, in that it learns to generate completed faces directly with improved efficiency.

REFERENCES

Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. 2009. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. Graph.* 28, 3 (2009), 24–1.

Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. 2000. Image inpainting. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 417–424.

Marcelo Bertalmio, Luminita Vese, Guillermo Sapiro, and Stanley Osher. 2003. Simultaneous structure and texture image inpainting. *IEEE Transactions on image processing*

12, 8 (2003), 882–889.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*. 2172–2180.

Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. 2017. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. *arXiv preprint arXiv:1711.09020* (2017).

Antonio Criminisi, Patrick Perez, and Kentaro Toyama. 2003. Object removal by exemplar-based inpainting. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, Vol. 2. IEEE, II–II.

Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. 2012. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Trans. Graph.* 31, 4 (2012), 82–1.

Yue Deng, Qionghai Dai, and Zengke Zhang. 2011. Graph Laplace for occluded face completion and recognition. *IEEE Transactions on Image Processing* 20, 8 (2011), 2329–2338.

Emily Denton, Sam Gross, and Rob Fergus. 2016. Semi-supervised learning with context-conditional generative adversarial networks. *arXiv preprint arXiv:1611.06430* (2016).

Emily L Denton, Soumith Chintala, Rob Fergus, et al. 2015. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In *Advances in neural information processing systems*. 1486–1494.

Alexei A Efros and Thomas K Leung. 1999. Texture synthesis by non-parametric sampling. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, Vol. 2. IEEE, 1033–1038.

Stuart Geman, Daniel F Potter, and Zhiyi Chi. 2002. Composition systems. *Quart. Appl. Math.* 60, 4 (2002), 707–736.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.

James Hays and Alexei A Efros. 2007. Scene completion using millions of photographs. In *ACM Transactions on Graphics (TOG)*, Vol. 26. ACM, 4.

Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. 2014. Image completion using planar structure guidance. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 129.

Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2017. Globally and locally consistent image completion. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 107.

Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*. 448–456.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2016. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004* (2016).

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*. Springer, 694–711.

Takuhiro Kaneko, Kaoru Hiramatsu, and Kunio Kashino. 2017. Generative attribute controller with conditional filtered generative adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 2.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196* (2017).

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

Diederik P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. 2016. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*. 4743–4751.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

Nikos Komodakis. 2006. Image completion using global optimization. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, Vol. 1. IEEE, 442–452.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Neural Information Processing Systems (NIPS)*. 1106–1114.

Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick. 2003. Graphcut textures: image and video synthesis using graph cuts. In *ACM Transactions on Graphics (ToG)*, Vol. 22. ACM, 277–286.

Yann LeCun. 1998. The MNIST database of handwritten digits. *http://yann. lecun. com/exdb/mnist/* (1998).

Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, R. E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. 1989. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation* 1, 4 (1989), 541–551. https://doi.org/10.1162/neco.1989.1.4.541

Anat Levin, Assaf Zomet, and Yair Weiss. 2003. Learning how to inpaint from global image statistics. In *null*. IEEE, 305.

Yijun Li, Sifei Liu, Jimei Yang, and Ming-Hsuan Yang. 2017. Generative Face Completion. *arXiv preprint arXiv:1704.05838* (2017).

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*. 3730–3738.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. ICML*, Vol. 30.

John H McDonald. 2009. *Handbook of biological statistics*. Vol. 2. Sparky House Publishing Baltimore, MD.

Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).

Umar Mohammed, Simon JD Prince, and Jan Kautz. 2009. Visio-lization: generating novel facial images. In *ACM Transactions on Graphics (TOG)*, Vol. 28. ACM, 57.

Alejandro Newell, Kaiyu Yang, and Jia Deng. 2016. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*. Springer, 483–499.

Kyle Olszewski, Zimo Li, Chao Yang, Yi Zhou, Ronald Yu, Zeng Huang, Sitao Xiang, Shunsuke Saito, Pushmeet Kohli, and Hao Li. 2017. Realistic dynamic facial textures from a single image using gans. In *IEEE International Conference on Computer Vision (ICCV)*. 5429–5438.

Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. 2016. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759* (2016).

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. 2016. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2536–2544.

Patrick Pérez, Michel Gangnet, and Andrew Blake. 2003. Poisson image editing. In *ACM Transactions on graphics (TOG)*, Vol. 22. ACM, 313–318.

Alec Radford, Luke Metz, and Soumith Chintala. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434* (2015).

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 234–241.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 3 (2015), 211–252.

Graeme D Ruxton and Guy Beauchamp. 2008. Time for some a priori thinking about post hoc testing. *Behavioral Ecology* 19, 3 (2008), 690–693.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*. 2234–2242.

Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. 2016. Learning from simulated and unsupervised images through adversarial training. *arXiv preprint arXiv:1612.07828* (2016).

Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

Jost Tobias Springenberg. 2015. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390* (2015).

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. 2016. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022* (2016).

Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. 2016. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*. 4790–4798.

Yonatan Wexler, Eli Shechtman, and Michal Irani. 2007. Space-time completion of video. *IEEE Transactions on pattern analysis and machine intelligence* 29, 3 (2007).

Marta Wilczkowiak, Gabriel J Brostow, Ben Tordoff, and Roberto Cipolla. 2005. Hole Filling Through Photomontage.. In *BMVC*, Vol. 5. 492–501.

Sitao Xiang and Hao Li. 2017. On the Effects of Batch and Weight Normalization in Generative Adversarial Networks. *stat* 1050 (2017), 22.

Chao Yang, Xin Lu, Zhe Lin, Eli Shechtman, Oliver Wang, and Hao Li. 2016. High-Resolution Image Inpainting using Multi-Scale Neural Patch Synthesis. *arXiv preprint arXiv:1611.09969* (2016).

Raymond A Yeh, Chen Chen, Teck Yian Lim, Alexander G Schwing, Mark Hasegawa-Johnson, and Minh N Do. 2017. Semantic image inpainting with deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5485–5493.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint arXiv:1703.10593* (2017).