Supplemental Materials
DeepFaceEditing: Deep Face Generation and Editing with Disentangled Geometry and Appearance Control

SHU-YU CHEN*, Institute of Computing Technology, CAS and UCAS, China
FENGLIN LIU*, Institute of Computing Technology, CAS and UCAS, China
YU-KUN LAI, School of Computer Science and Informatics, Cardiff University, UK
PAUL L. ROSIN, School of Computer Science and Informatics, Cardiff University, UK
CHUNPENG LI, Institute of Computing Technology, CAS and UCAS , China
HONGBO FU, School of Creative Media, City University of Hong Kong, HK
LIN GAO†, Institute of Computing Technology, CAS and UCAS , China

1 OVERVIEW

In this supplemental document, we have listed more results of sketch-to-image generation, face synthesis results for hand-drawn sketches, results of user editing, inputs and results for non-frontal face generation, and more details of the implementation.

2 SKETCH-TO-IMAGE GENERATION

In Fig. 1, we show more results of sketch-to-image generation with various reference images and sketches.

3 RESULTS FOR HAND-DRAWN SKETCHES

In Fig. 2, we present more results for hand-drawn sketches and comparisons with DeepFaceDrawing [Chen et al. 2020], as well as applying style transfer technique [Kolkin et al. 2019] and swapping autoencoder [Park et al. 2020] after DeepFaceDrawing.

4 USER EDITING RESULTS

In Fig. 3, we show more editing results from users.

5 NON-FRONTAL FACE GENERATION

In Fig. 4, we show the source sketches and appearance images for non-frontal generation results.

6 NETWORK ARCHITECTURES

Our network consists of a Local Disentanglement Module and a Global Fusion Module.

*Authors contributed equally.
†Corresponding author.

Authors' addresses: Shu-Yu Chen, Fenglin Liu, Chunpeng Li, and Lin Gao are with the Beijing Key Laboratory of Mobile Computing and Pervasive Device, Institute of Computing Technology, Chinese Academy of Sciences. Yu-Kun Lai and Paul L. Rosin are with School of Computer Science and Informatics, Cardiff University. Hongbo Fu are with the School of Creative Media, City University of Hong Kong. Authors' e-mails: chenshuyu@ict.ac.cn, fenglinliu@std.uestc.edu.cn, LaiY4@cardiff.ac.uk, RosinPL@cardiff.ac.uk, cpli@ict.ac.cn, hongbofu@cityu.edu.hk, gaolin@ict.ac.cn.

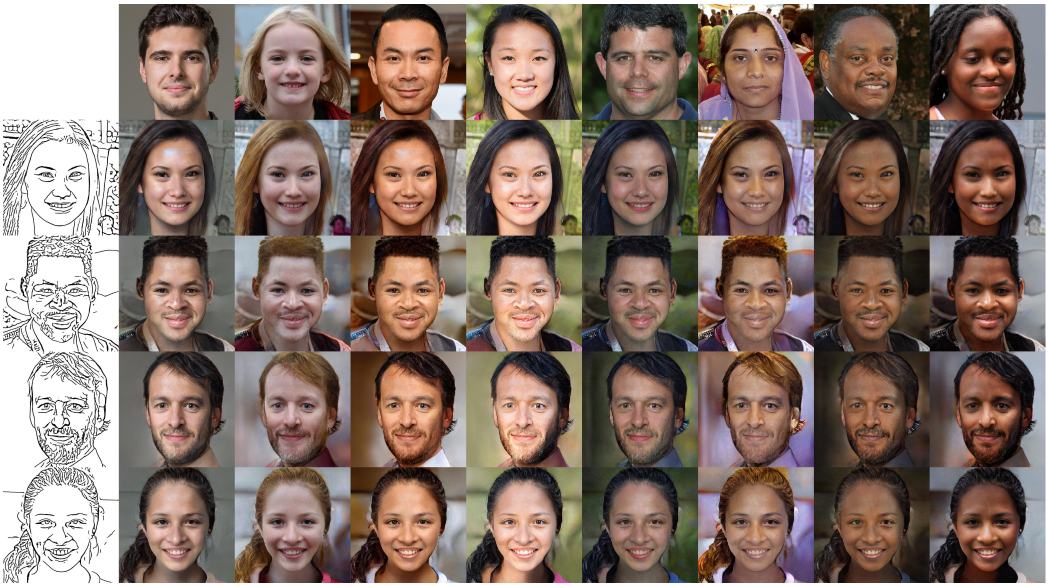

Fig. 1. Sketch-to-image generation with various reference images. With the geometry of sketches, our method can generate realistic faces with different skin color, lighting, and appearance according to diverse reference images.

6.1 Local Disentanglement

Local Disentanglement module extracts both the geometry and appearance features for each local component and synthesizes local image patches. This module consists of Geometry Encoder, Appearance Encoder and Image Synthesis Generator. We adopt the GAN architecture which contains a multi-scale discriminator.

6.1.1 Geometry Encoder. Geometry Encoder contains five encoding layers, as shown in Table 7. For sketches, the number of input channels is 1 and for images, it is 3. The Geometry Decoder (Table 8) takes the features generated by Geometry Encoder as input and generates corresponding sketches.

6.1.2 Appearance Encoder. In Table 9, we show the architecture of the Appearance Encoder. This network generates a vector which encodes appearance information.

6.1.3 Image Synthesis Generator. In Table 10, we show the architecture of the Image Synthesis Generator. Taking the feature maps generated by the Geometry Encoder as input, the Image Synthesis Generator uses adaptive instance normalization to inject the appearance information and generates local component images.

6.1.4 Discriminator. The discriminator employs a multi-scale discriminating approach: scale the input feature maps and the generated images at two different levels and go through two different sub-discriminators. More details are shown in Table 12.

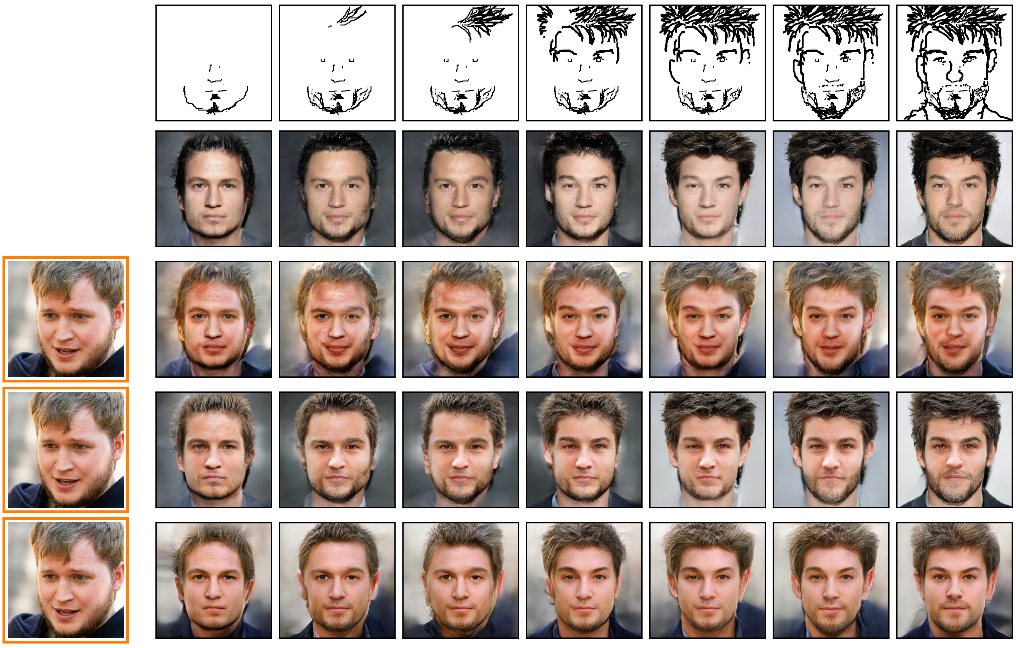

Fig. 2. Comparisons of image generation with hand-drawn sketches (1st row). The 2nd row shows the results by DeepFaceDrawing [Chen et al.2020] and the 3rd row is the results of using a style transfer technique [Kolkin et al.2019] after DeepFaceDrawing. The 4th row shows the results of using a swapping autoencoder [Park et al.2020] similar to the 3rd row. The bottom row shows our results using the image in the first column as appearance. Benefiting from our approach, geometry and appearance can be effectively disentangled, and thus our results are clear and realistic. The results combining DeepFaceDrawing and style transfer technique are not able to distinguish the geometry and appearance which leads to obvious artifacts, e.g. unremoved hair on the forehead. Similarly, the results of the swapping autoencoder also show unnatural lighting on eye regions.

6.2 Global Fusion

Our global fusion network also adopts a GAN utilizing a generator and a discriminator to generate real face images. The details of the generator are shown in Table 11. The discriminator is similar to the local disentanglement module and is illustrated in Table 12.

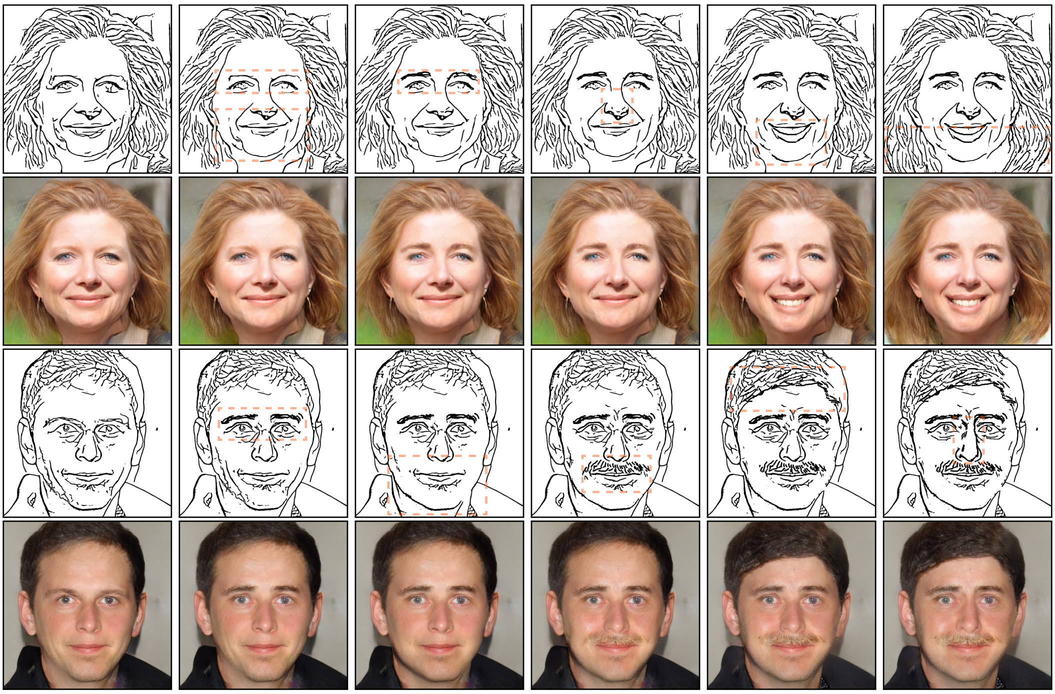

Fig. 3. Our model supports flexible face editing based on sketches, without requiring users to pre-define a region of interest.

F(x)
Conv2d
ReLU

Table 1. **Conv2D-Block**

F(x)
Conv2d
Instance Norm2d
ReLU

Table 2. **Conv2D-IN-Block**

F(x)
ConvTranspose2d
Adaptive Instance Norm2d
ReLU

Table 3. **ConvTrans2D-AdaIN-Block**

F(x)
ConvTranspose2d
Instance Norm2d
ReLU

Table 4. **ConvTrans2D-IN-Block**

F(x)
Conv2d
Adaptive Instance Norm2d
ReLU
Conv2d
Adaptive Instance Norm2d

Table 5. **Res-AdaIN-Block**

F(x)
Conv2d
Instance Norm2d
ReLU
Conv2d
Instance Norm2d

Table 6. **Res-IN-Block**

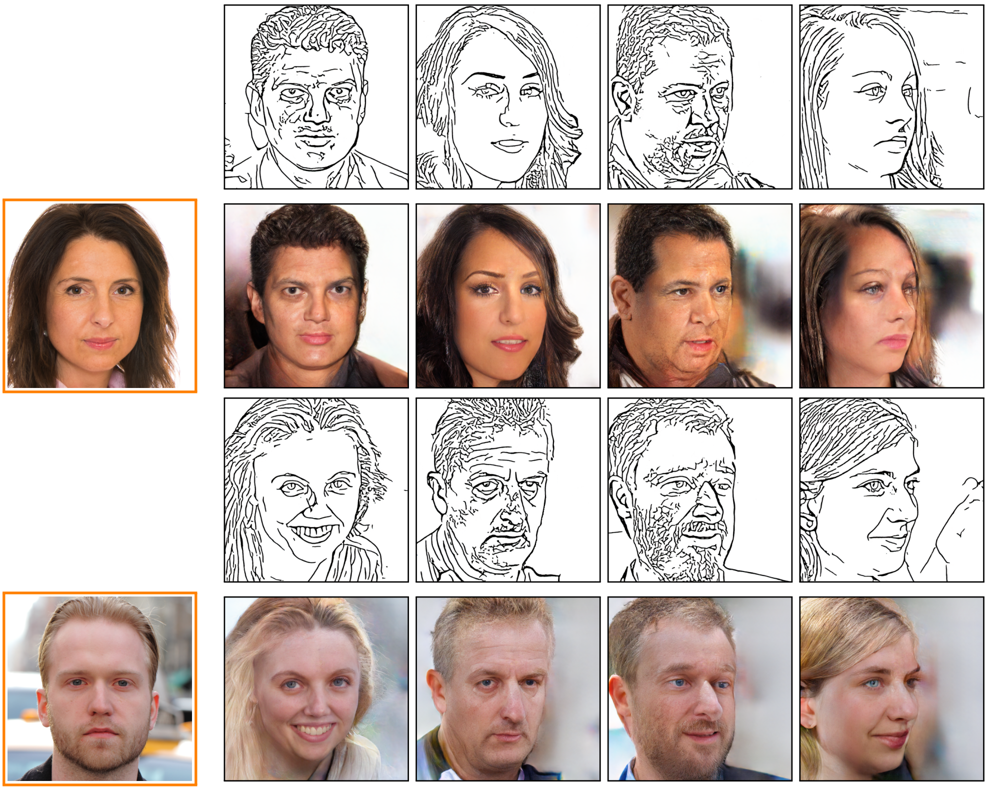

Fig. 4. The results for non-frontal face inputs. Source sketches in the 1st and 3rd rows provide geometry. Images in the 1st column provide appearance. When faces rotate within 30 degree from the frontal plane, relatively good results can be generated. With larger offset degree, some artifacts may be generated on the edges of hair and background.

Geometry Encoder		
Layer	**Output Size**	**Filter**
Input	$1/3 \times H \times W$	
Conv2D-IN-Block	$64 \times H \times W$	$1 \longrightarrow 64$
Conv2D-IN-Block	$128 \times \frac{H}{2} \times \frac{W}{2}$	$64 \longrightarrow 128$
Conv2D-IN-Block	$256 \times \frac{H}{4} \times \frac{W}{4}$	$128 \longrightarrow 256$
Conv2D-IN-Block	$512 \times \frac{H}{8} \times \frac{W}{8}$	$256 \longrightarrow 512$
Conv2D-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$512 \longrightarrow 1024$
Output	$1024 \times \frac{H}{16} \times \frac{W}{16}$	

Table 7. The architecture of the Geometry Encoder.

Geometry Decoder		
Layer	**Output Size**	**Filter**
Input	$1024 \times \frac{H}{16} \times \frac{W}{16}$	
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
ConvTrans2D-IN-Block	$512 \times \frac{H}{8} \times \frac{W}{8}$	$1024 \longrightarrow 512$
ConvTrans2D-IN-Block	$256 \times \frac{H}{4} \times \frac{W}{4}$	$512 \longrightarrow 256$
ConvTrans2D-IN-Block	$128 \times \frac{H}{2} \times \frac{W}{2}$	$256 \longrightarrow 128$
ConvTrans2D-IN-Block	$64 \times H \times W$	$128 \longrightarrow 64$
Conv2D	$1 \times H \times W$	$64 \longrightarrow 1$
Tanh	$1 \times H \times W$	
Output	$1 \times H \times W$	

Table 8. The architecture of the Geometry Decoder.

Appearance Encoder		
Layer	**Output Size**	**Filter**
Input	$3 \times H \times W$	
Conv2D-Block	$16 \times H \times W$	$3 \longrightarrow 16$
Conv2D-Block	$32 \times \frac{H}{2} \times \frac{W}{2}$	$16 \longrightarrow 32$
Conv2D-Block	$64 \times \frac{H}{4} \times \frac{W}{4}$	$32 \longrightarrow 64$
Conv2D-Block	$64 \times \frac{H}{8} \times \frac{W}{8}$	$64 \longrightarrow 64$
Conv2D-Block	$64 \times \frac{H}{16} \times \frac{W}{16}$	$64 \longrightarrow 64$
Conv2D-Block	$64 \times \frac{H}{32} \times \frac{W}{32}$	$64 \longrightarrow 64$
AdaptiveAvgPool2D	$64 \times 1 \times 1$	$64 \longrightarrow 64$
Conv-2D	$18304 \times 1 \times 1$	$64 \longrightarrow 18304$
Output	$18304 \times 1 \times 1$	

Table 9. The architecture of the Appearance Encoder.

Image Synthesis Generator		
Layer	**Output Size**	**Filter**
Input	$1024 \times \frac{H}{16} \times \frac{H}{16}$	
Res-AdaIN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-AdaIN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-AdaIN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-AdaIN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
ConvTrans2D-AdaIN-Block	$512 \times \frac{H}{8} \times \frac{W}{8}$	$1024 \longrightarrow 512$
ConvTrans2D-AdaIN-Block	$256 \times \frac{H}{4} \times \frac{W}{4}$	$512 \longrightarrow 256$
ConvTrans2D-AdaIN-Block	$128 \times \frac{H}{2} \times \frac{W}{2}$	$256 \longrightarrow 128$
ConvTrans2D-AdaIN-Block	$64 \times H \times W$	$128 \longrightarrow 64$
Conv2D	$3 \times H \times W$	$64 \longrightarrow 3$
Tanh	$3 \times H \times W$	
Output	$3 \times H \times W$	

Table 10. The architecture of the Image Synthesis Generator.

Global Fusion Network

Layer	Output Size	Filter
Input	$64 \times H \times W$	
Conv2D-IN-Block	$64 \times H \times W$	$64 \longrightarrow 64$
Conv2D-IN-Block	$128 \times \frac{H}{2} \times \frac{W}{2}$	$64 \longrightarrow 128$
Conv2D-IN-Block	$256 \times \frac{H}{4} \times \frac{W}{4}$	$128 \longrightarrow 256$
Conv2D-IN-Block	$512 \times \frac{H}{8} \times \frac{W}{8}$	$256 \longrightarrow 512$
Conv2D-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$512 \longrightarrow 1024$
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
Res-IN-Block	$1024 \times \frac{H}{16} \times \frac{W}{16}$	$1024 \longrightarrow 1024$
ConvTrans2D-IN-Block	$512 \times \frac{H}{8} \times \frac{W}{8}$	$1024 \longrightarrow 512$
ConvTrans2D-IN-Block	$256 \times \frac{H}{4} \times \frac{W}{4}$	$512 \longrightarrow 256$
ConvTrans2D-IN-Block	$128 \times \frac{H}{2} \times \frac{W}{2}$	$256 \longrightarrow 128$
ConvTrans2D-IN-Block	$64 \times H \times W$	$128 \longrightarrow 64$
Conv2D	$3 \times H \times W$	$64 \longrightarrow 3$
Tanh	$3 \times H \times W$	
Output	$3 \times H \times W$	

Table 11. The architecture of the Global Fusion Network.

Discriminating Unit (DisUnit)

Layer	Output Size	Filter
Input	$(3+1) \times H \times W$	
Conv2D-Block	$64 \times (\frac{H}{2}+1) \times (\frac{W}{2}+1)$	$(3+1) \longrightarrow 64$
Conv2D-IN-Block	$128 \times (\frac{H}{4}+1) \times (\frac{W}{4}+1)$	$64 \longrightarrow 128$
Conv2D-IN-Block	$256 \times (\frac{H}{8}+1) \times (\frac{W}{8}+1)$	$128 \longrightarrow 256$
Conv2D-IN-Block	$512 \times (\frac{H}{8}+2) \times (\frac{W}{8}+2)$	$256 \longrightarrow 512$
Conv2D	$1 \times (\frac{H}{8}+3) \times (\frac{W}{8}+3)$	$512 \longrightarrow 1$

Discriminator

Layer	D1 Output Size	D2 Output Size
Input	$4 \times H \times W$	$4 \times H \times W$
AvgPool	-	$4 \times \frac{H}{2} \times \frac{W}{2}$
DisUnit	$1 \times (\frac{H}{8}+3) \times (\frac{W}{8}+3)$	$1 \times (\frac{H}{16}+3) \times (\frac{W}{16}+3)$
Output	$1 \times (\frac{H}{8}+3) \times (\frac{W}{8}+3)$	$1 \times (\frac{H}{16}+3) \times (\frac{W}{16}+3)$

Table 12. The architecture of the Discriminator.