Supplementary Material

Table 1. The structure of the employed classification network.

Input	Layers	Output	
x	InceptionResnetV1	(384, 3, 3)	
$\uparrow$	Conv2d	(1792, 3, 3)	A
$\uparrow$	AvgPool	(1792)	
$\uparrow$	FC1	(512)	f
$\uparrow$	FC2	(8631)	c
A, c	Distraction	(1792, 3, 3)	$\hat{A}$
$\uparrow$	AvgPooling	(1792)	
$\uparrow$	FC1	(512)	$\hat{f}$

Table 2. The structure of the employed generator G.

Input	Layers	Output	
Z_a	E	(512)	f_a
Z_{id}, f_a	Concat	(1024)	
$\uparrow$	FC	(13056)	O_0
Z_g	ResBlock	(64, 128, 128)	O_1
$\uparrow$	ResBlock	(128, 64, 64)	O_2
$\uparrow$	ResBlock	(256, 32, 32)	O_3
$\uparrow$	ResBlock	(512, 16, 16)	O_4
$\uparrow$, O_0	ResBlock×4	(512, 16, 16)	
$\uparrow$, O_4, O_0	ResBlock	(512, 32, 32)	
$\uparrow$, O_3, O_0	ResBlock	(256, 64, 64)	
$\uparrow$, O_2, O_0	ResBlock	(128, 128, 128)	
$\uparrow$, O_1, O_0	ResBlock	(64, 256, 256)	
$\uparrow$	Conv2d	(3, 256, 256)	$\hat{x}$

Table 3. The structure of the employed appearance encoder E.

Input	Layers	Output	
Z_a	ResBlock	(64, 128, 128)	
$\uparrow$	ResBlock	(128, 64, 64)	
$\uparrow$	ResBlock	(256, 32, 32)	
$\uparrow$	ResBlock	(512, 16, 16)	
$\uparrow$	ResBlock×2	(512, 4, 4)	
$\uparrow$	SumPooling	(512)	f_a

1. Network Architecture

Identity Feature Anonymization. The pre-trained FaceNet classification network [4] is used for identity feature anonymization. As shown in the top part of Table 1, the InceptionRenetV1 [6] is employed to implement FaceNet [1].

[1] https://github.com/timesler/facenet-pytorch

Table 4. The structure of the employed discriminator D.

Input	Layers	Output
(x,S_x) or ($\hat{x}$,S_x)	Concat	(6, 256, 256)
$\uparrow$	ResBlock	(64, 128, 128)
$\uparrow$	ResBlock	(128, 64, 64)
$\uparrow$	ResBlock	(256, 32, 32)
$\uparrow$	ResBlock	(512, 16, 16)
$\uparrow$	ResBlock×2	(512, 4, 4)
$\uparrow$	ResBlock	(512, 4, 4)
$\uparrow$	SumPooling	(512)
$\uparrow$	FC	(1)

For simplicity, we use InceptionResnetV1 to denote all the layers before the last convolutional layer (i.e.Conv2d), the first fully connected (FC) layer (i.e. FC1) is used for feature extraction and the last FC layer (i.e. FC2) is used for classification. As shown in the bottom part of Table 1, the distraction layer is used to distract the attention of its CAM heatmap to recast the identity feature.

Generator. As shown Table 2, generator G is built by stacking downsampling and upsampling ResBlocks. Z_{id} and $f_a = E(Z_a)$ are concatenated followed by a FC layer to produce O_0. Z_g first goes through multiple downsampling ResBlocks and then their outputs are fed to the corresponding upsampling ResBlocks. O_0 is fused into the upsampling process by using the AdaIN operation [3]. As shown in Table 3, the network structure of appearance encoder E is built by stacking ResBlocks and SumPooling layer [2].

Discriminator. As shown in Table 4, discriminator D takes real data pair (x, S_x) or fake data pair $(\hat{x}, S_x)$ as input, goes through a Concat layer, multiple ResBlocks and a SumPooling layer to tell the realism of the input data. S_x is the geometry structure of x.

2. More Results

When the semantic segmentation model [7] is not available, our approach can still work by using the detected landmarks [1] to perform segmentation. The results are demonstrated in Figure 2 and Figure 1, where Figure 1 demonstrates the results of only changing the geometry structure. Significant geometry changes would make the resulted faces look different from their original version. Thus, it is reasonable to pick up the delegate geometry structrues that are relatively far from the original one to enhance anonymization.

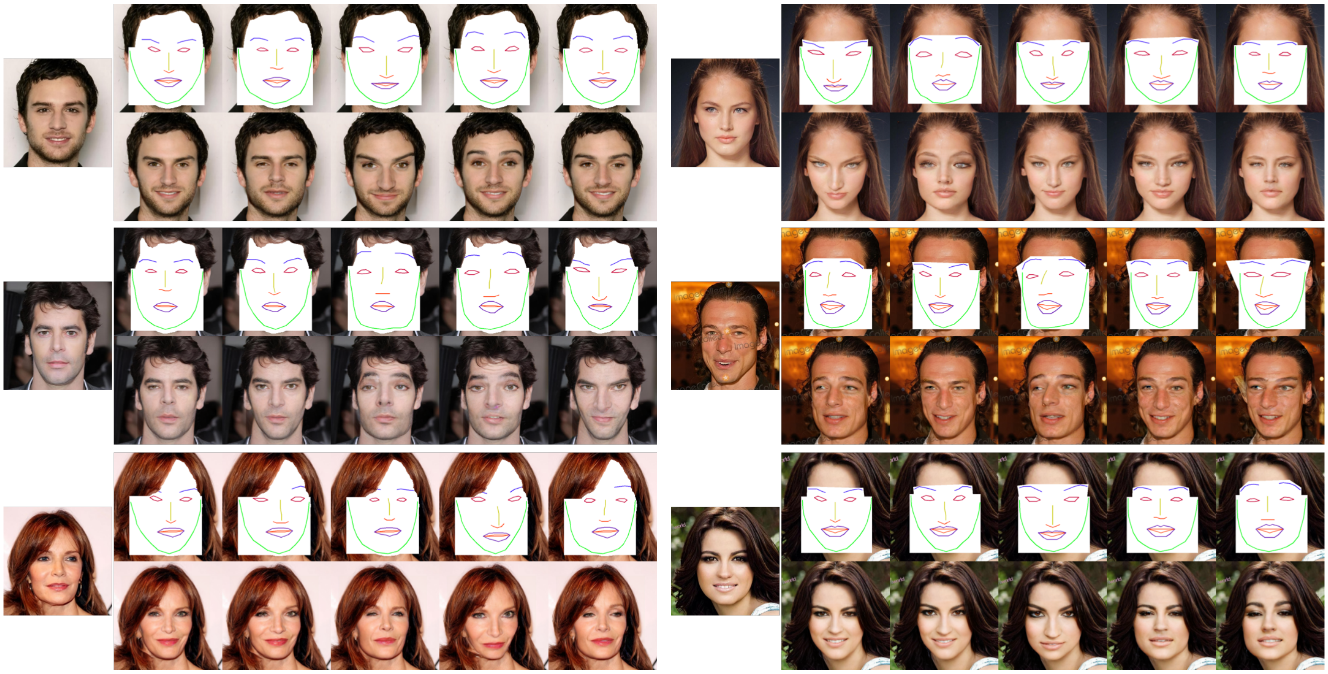

Figure 1. Demonstration of faces generated by only changing the geometry structure with (left) and without (right) semantic segmentation.

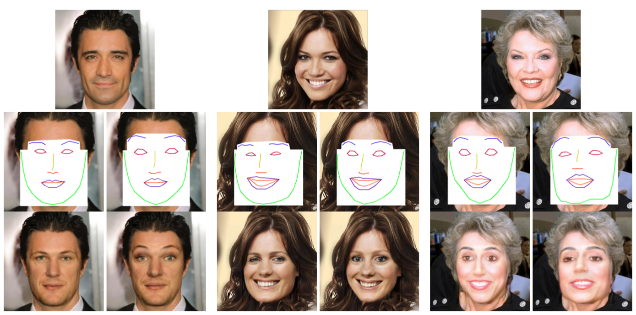

Figure 2. Illustration of our results without semantic segmentation.

References

[1] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem?(and a dataset of 230,000 3d facial landmarks). In *ICCV*, pages 1021–1030, 2017. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1

[3] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pages 1501–1510, 2017. 1

[4] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: a unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 1

[5] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017.

[6] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017. 1

[7] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *ECCV*, pages 325–341, 2018. 1