Supplementary material – CIAGAN: Conditional Identity Anonymization Generative Adversarial Networks

Maxim Maximov*
Technical University Munich

Ismail Elezi*
University of Venice

Laura Leal-Taixé
Technical University Munich

Abstract

In this supplemental document, we compare the generation quality of our method to different baselines (Section 1), qualitatively show the diversity of our generation method (Section 2), demonstrate results on facial occlusions (Section 3), explain the limitations of our model (Section 4), and detail the architecture of our models (Section 5).

1. FID of baseline methods

In Table 1 we show quantitative results on the quality of the generated images. We use the *FID* score [2], a metric that compares the statistics of generated samples to those of real samples. The lower the *FID*, the better. To quantify the quality of generated samples, they are first embedded into a feature space given by (a specific layer) of Inception Net. Then, viewing the embedding layer as a continuous multivariate Gaussian, the mean and covariance are estimated for both the generated data and the real data. The Fréchet distance between these two Gaussians is then used to quantify the quality of the samples:

$$FID(x, g) = ||\mu_x - \mu_g||_2^2 + Tr(\Sigma_x + \Sigma_g - 2(\Sigma_x\Sigma_g)^{1/2}) \quad (1)$$

where (μ_x, Σ_x), and (μ_g, Σ_g) are the mean and covariance of the sample embeddings from the data distribution and model distribution, respectfully. The authors show that the score is consistent with human judgment and is robust to noise. Finally, *FID* can detect intra-class mode dropping, *e.g.* a model that generates only one image per class will have a high FID.

Our method reaches a very low *FID* score quantitatively showing that the quality of the generated images is very high. Pixelization methods unsurprisingly reach a very high *FID* score. The higher the number of pixels merged together is, the higher is the *FID* score. Similar behavior is seen for the blurring methods where the higher the level of blur is, the higher is the *FID* score.

Model	FID (↓)
Pixelization 8 by 8	510.83
Pixelization 16 by 16	318.43
Blur 9 by 9	54.50
Blur 17 by 17	152.03
Pix2Pix [3]	121.41
CycleGAN [7]	185.26
Ours	**2.08**

Table 1: Quality and diversity measurements of different methods.

We train GAN-based image translation methods [3, 7] where the source domain are the landmarks and the target domain are the images. We were surprised to see that the *FID* score of image translation methods is very high, comparable to blurring methods. We investigate this by checking the visual quality of the generated images. We see that the models learn to generate only slight variations of the same face (that can be considered an average face). We conjecture that the main reason for this problem is the sparse signal of the source domain. We show a qualitative evaluation of the baseline methods in Fig. 2 where our method is the only one that generates realistically looking images.

2. Diversity of the generated images

In Fig. 1 of the main paper, we showed triplets of images, with the first image in the triplet being the source image, and the other two images being different anonymized versions of it.

In Fig. 3 we perform a similar experiment, but this time instead of showing only 2 anonymized versions of the source image, we present 9 different versions of it. We see that the generated images have still different identities that are sufficiently different from each other.

* Authors contributed equally.

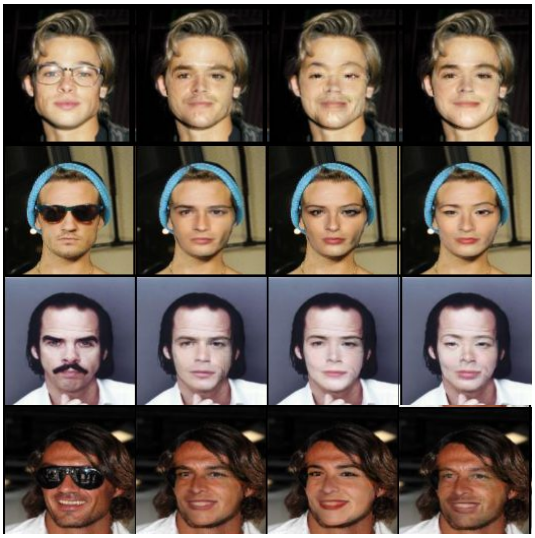

Figure 1: First column contains source images, all other columns are anonymized versions of our method with different control vectors.

3. Removing glasses and mustache

In Fig. 1 we perform an experiment where the source images contain faces of people with glasses, or with a mustache. Considering that our generator uses as input the landmarks of the face (instead of the entire face), it has no knowledge whatsoever that the source image might contain glasses. Unsurprisingly, the generated images do not contain glasses or mustache. Note that the quality of the generated images does not suffer and during the blending process, the generator inpaints the region where the source image contained the glasses.

4. Limitations

A weakness of all current de-identification methods [4, 5, 1] is that they need the original faces to be initially detected before they can be anonymized. Consequently, any face that has not been detected can not be anonymized. Thus, all the aforementioned methods are not deployable in systems where the anonymization must be guaranteed. Our method suffers from a similar issue, if a face does not get detected, then its landmark will not get generated, and so the model will not generate the anonymized version of the face. As future work, we plan on investigating this problem in two directions: deploying an ensemble of detection networks to minimize the probability of faces not getting detected; and anonymizing entire images without the need of detecting the original faces on them.

Another limitation of landmarks usage is the occurrence of occlusions in front of a face. Since the generated face is based on the landmarks, these occlusions are going to be removed (*e.g.* glasses as in Fig. 1). The problem can be resolved by detecting such occlusions and treating them as part of the background mask.

Finally, like in other deep learning anonymization frameworks, the more different the images are to the images of the training dataset, the worse is the quality of the generation. CelebA dataset offers multiple images per identity with good quality but with a significant bias towards frontal faces (since they are using photos of celebrities). Consequently, our method performs best when used in similar datasets. For our model to perform as well in extreme poses, it needs to be additionally trained in a dataset containing such poses.

5. Network architecture

In Table 2 we show the architecture of the generator. The generator uses an encoder-decoder architecture, and receives as input a 6-dimensional image that is created by concatenating the *landmark image* with the *masked background image*. It encodes the input image into a 3-dimensional tensor that has 256 channels. In the bottleneck, the encoded image is concatenated with the identity embedding (that is an output of the transposed convolution network). Finally, the network decodes the combined embedding, to produce the anonymized version of the source image.

In Table 3 we show the architecture of the discriminator. The network receives as input a 3-dimensional image that is created by combining the *generated face* with the *masked background image*. Then it uses a series of residual blocks. Finally it uses 2 fully-connected layers. The network has the same architecture as the siamese network that we use for identity guidance.

In Table 4 we show the architecture of the embedding network. The network receives as input the label of the desired identity (given in one-hot format) and produces a 3-dimensional tensor. This tensor is fed into the bottleneck of the generator.

Finally, we give the architecture of the residual blocks used in all the networks. The architectures of the ”residual block down”, ”residual block up” and ”residual block” are given on Tables 5, 6 and 7 respectively.

Generator	
Layers	**Output size**
Input	6 x 128 x128
Residual Block Down	32 x 64 x 64
Residual Block Down	64 x 32 x 32
Residual Block Down	128 x 16 x 16
Residual Block Down	256 x 8 x 8
Residual Block Down	256 x 4 x 4
Concatenate with the ID embedding	512 x 4 x 4
3x3 stride1 conv + ReLU	256 x 4 x 4
Residual Block	256 x 4 x 4
Residual Block	256 x 4 x 4
Residual Block	256 x 4 x 4
Residual Block	256 x 4 x 4
Residual Block Up	256 x 8 x 8
Residual Block Up	128 x 16 x 16
Residual Block Up	64 x 32 x 32
Residual Block Up	32 x 64 x 64
Residual Block Up	16 x 128 x 128
3x3 stride1 conv	3 x 128 x 128

Table 2: The network architecture of our generator.

Discriminator	
Layers	**Output size**
Input	3 x 128 x128
Residual Block Down	32 x 64 x 64
Residual Block Down	64 x 32 x 32
Residual Block Down	128 x 16 x 16
Residual Block Down	256 x 8 x 8
Residual Block Down	512 x 4 x 4
FC + LeakyReLU	1024
FC + LeakyReLU	1

Table 3: The network architecture of our discriminator.

Transposed Convolutional Neural Network	
Layers	**Output size**
Input	N
(FC + LeakyReLU) x 7	512
Reshape	32 x 4 x 4
3x3 stride 1 conv + LeakyReLU + IN	64 x 4 x 4
3x3 stride 1 conv + LeakyReLU + IN	128 x 4 x 4
3x3 stride 1 conv + LeakyReLU + IN	256 x 4 x 4
Concatenate with the landmark embedding	512 x 4 x 4

Table 4: The architecture of our transposed convolutional neural network.

Residual Block Down	
Input	
3x3 stride1 conv + ReLU	1x1 stride1 conv
3x3 stride1 conv	2x2 average pooling
2x2 average pooling	
Summation	
Instance Normalization	

Table 5: The network architecture for the residual block down module.

Residual Block Up	
Input	
IN + ReLU	2x2 Upsample
2x2 Upsample	1x1 stride1 conv
3x3 stride1 conv + IN + ReLU	
3x3 stride1 conv	
2x2 average pooling	
Summation	
Instance Normalization	

Table 6: The network architecture for the residual block up module.

Residual Block
Input
3x3 stride1 conv + IN + ReLU
3x3 stride1 conv + IN
Summation with Input

Table 7: The network architecture for the residual block module. IN stands for Instance Normalization [6]

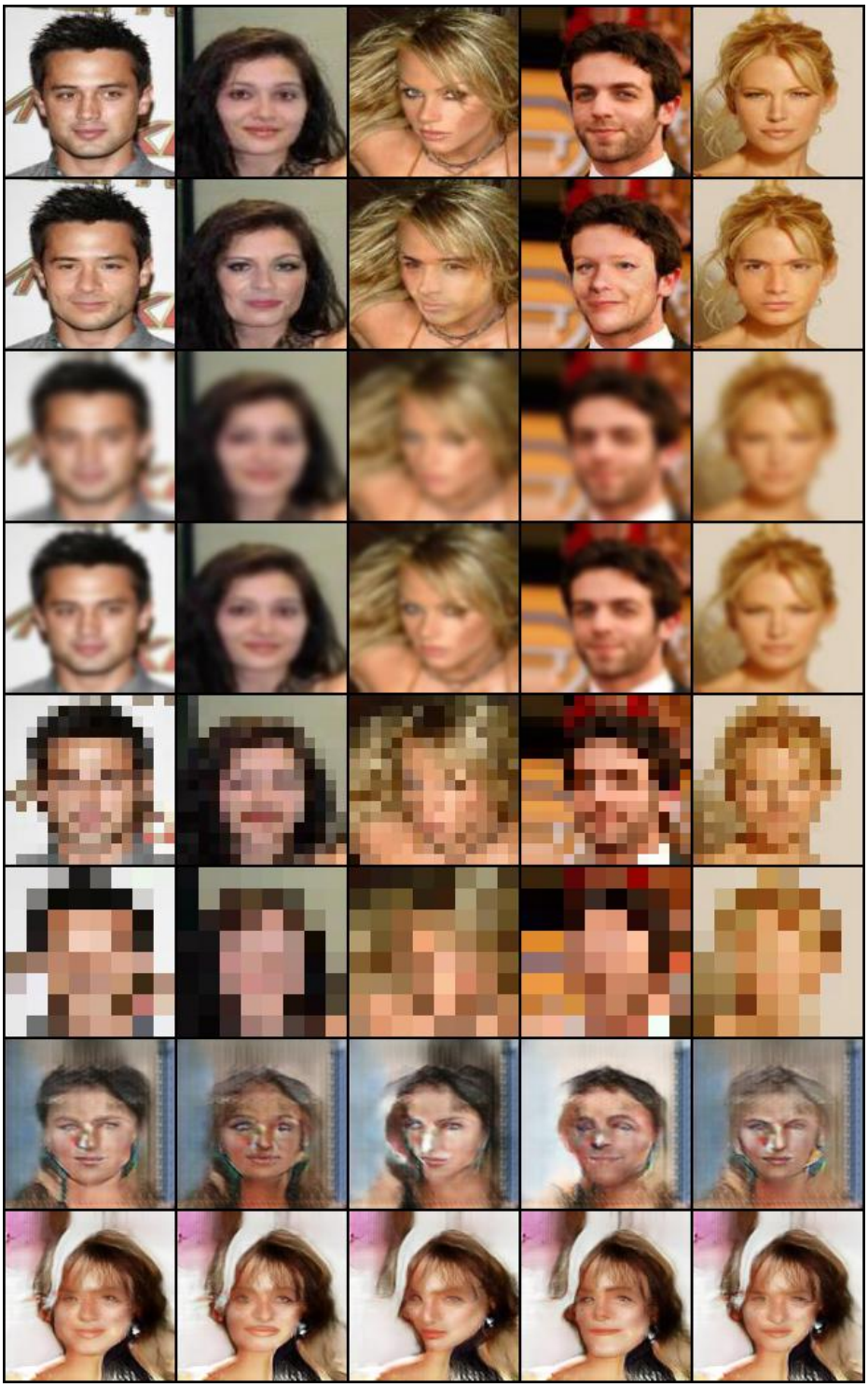

Figure 2: Our method compared with other anonymization baselines. From up to down: Original, CIAGAN, Blur 17 by 17, Blur 9 by 9, Pixelization 16 by 16, Pixelization 8 by 8, Pix2Pix, CycleGAN.

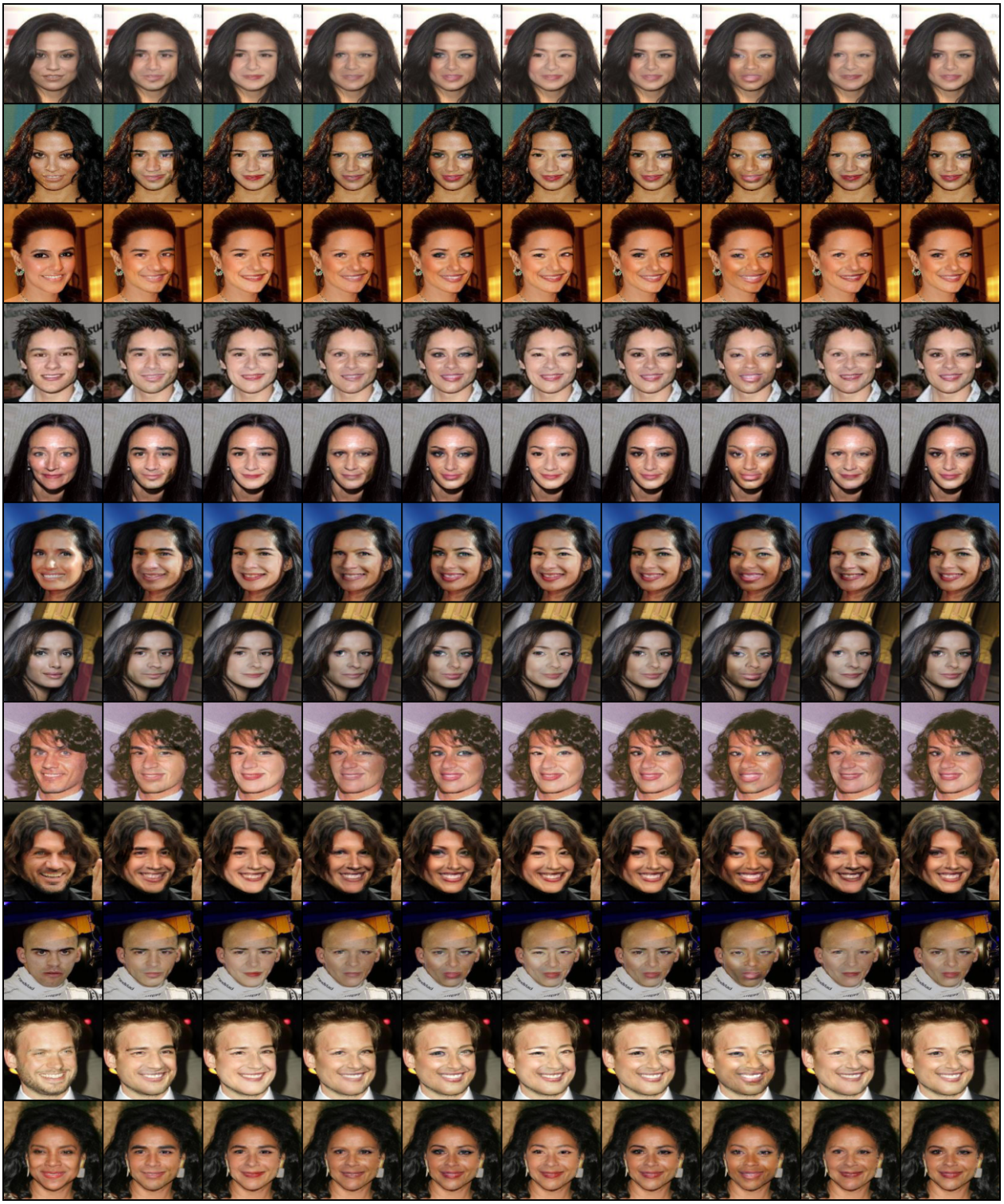

Figure 3: The images in the first column are the source images, all the other images are anonymized versions of the source images, where the anonymization process uses different control vectors.

References

[1] Oran Gafni, Lior Wolf, and Yaniv Taigman. Live face de-identification in video. In *IEEE International Conference on Computer Vision, ICCV 2019, Seoul, South Korea, October 27 - November 2, 2019*, 2019. 2

[2] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6626–6637, 2017. 1

[3] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5967–5976, 2017. 1

[4] Zhongzheng Ren, Yong Jae Lee, and Micheal S. Ryoo. Learning to anonymize faces for privacy preserving action detection. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[5] Qianru Sun, Liqian Ma, Seong Joon Oh, Luc Van Gool, Bernt Schiele, and Mario Fritz. Natural and effective obfuscation by head inpainting. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5050–5059, 2018. 2

[6] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016. 3

[7] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2242–2251, 2017. 1