Face De-Identification

Seyyed Mohammad Sadegh Moosavi Khorzooghi
seyyedmohammads.moosavikhorzoog@mavs.uta.edu
University of Texas at Arlington

Huadi Zhu
huadi.zhu@mavs.uta.edu
University of Texas at Arlington

David Jaime
david.jaime@mavs.uta.edu
University of Texas at Arlington

ABSTRACT

There has been a growing popularity in photo and video sharing on social media, which let the privacy content of the photos and videos be accessible by many people especially those with malicious intents who want to take advantage of them. Moreover, those adversaries can easily access sensitive information about the victims using advanced machine learning or deep learning methods. Therefore, there is a need to remove sensitive and identifiable information from photos and videos. The key point in face obfuscation is maintaining utilities while obfuscating face well enough. There have been several methods such as traditional methods (pixelation and blurring), K-Same methods, and GAN methods such as UPGAN used for this purpose. So far, UPGAN has been the most successful method in this regard. However, it still needs improving in terms of obfuscation and utility preserving. StyleGAN is currently the most advance method in generating high quality and real looking pictures. Moreover, it can mix the latent vectors of two images and make the resulting pictures have mixed styles inheriting from the inputs. Therefore, if only inherits coarse styles from the target face and the rest from a random face, the target can be easily obfuscated. Therefore, in this paper, FFHQ dataset is used to produce mixed styles high quality faces. Then, the resulting dataset is under 3 different attacks to evaluate StyleGAN in face obfuscation and compare it with other methods.

1 INTRODUCTION

There has been a growing popularity in photo and video sharing on social media, which let the privacy content of the photos and videos be accessible by many people especially those with malicious intents who want to take advantage of them. Moreover, those adversaries can easily access sensitive information about the victims using advanced machine learning or deep learning methods. Therefore, there is a need to remove sensitive and identifiable information from photos and videos. There have been some traditional methods such as pixelation or blurring which pixelate or blur the face with different window sizes. The problem with these methods is that they only hide objects well if the window size is large enough, which makes the visual effects of the face or the video unpleasant. Therefore, newer methods such as K-Same and K-Same-Net have been proposed which increase obfuscation while preserving utilities, which are non-identifiable features of the face such as pose, expression, gender, age, etc. However, these methods don't do a good job in preserving pose and give slightly blurry results too. Therefore, GAN methods such as UPGAN have been proposed which can reduce blurriness and support pose. However, the mentioned methods still have difficulty in obfuscation along with giving clear and natural results. StyleGAN is a type of GAN which can produce clear and natural looking faces by mixing styles from two given faces [1]. It does that by mixing the styles extracted from the latent vectors of these two faces. We can categorize the styles into 3 categories: 1- coarse styles, median styles, and fine styles. Coarse styles are usually related to non-identifiable features such as pose, expression, hair style, eyeglasses etc. while the other styles can have identifiable information too. StyleGAN hasn't been proposed for face obfuscation, but it is assumed that if we mix coarse styles from the face we want to obfuscate with the rest from another face, we could get good enough face obfuscation while having a clear result maintaining utilities. Therefore, we are going to get the results of StyleGAN on their face datasets with different levels of style mixing and use the results for different attacks to evaluate the obfuscation level of the method, and finally, reaching the best obfuscation using the best style mixing. (Section 1).

2 BACKGROUND AND RELATED WORK

As previously stated we will be using StyleGAN as our main focus to see how it compares to other obfuscation methods. Although StyleGAN is not an obfuscation method currently being used based on our results it might be applicable. StyleGAN itself if a type of Generative Adversarial Network (GAN).

2.1 Generative Adversarial Networks

Generative adversarial networks (GANs) continue to increase in popularity. It was first introduced by Goodfellow et al [1] in 2014. The structure of GANs can be seen in Figure 1. According to Zhengwei Wang et al. [10] it has applications in computer vision, natural language processing, time series synthesis, semantic segmentation, etc. Its characterized as two neural networks in a constant battle between each other. The generator generates items while the discriminator evaluates such items for legitimacy. As they continue to fight, both continue to show improvements. A good example is a counterfeiter versus law enforcement [1]. Where over time the counterfeiters get so good their fakes cannot be detected. This optimization process is a minimax game process where both networks in this case are making optimal decisions to reach a point of nash

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

equilibrium. This equilibrium is when neither network can benefit from optimization if the other does not [9].

Further research has coupled GANs together calling them CoGANs with the purpose of learning a joint distribution of multi-domain images [5]. Prior to this, tuples of corresponding images from different domains were used. Many challenges came with the corresponding images that limited application. CoGAN solves this issue only needing images from the marginal distribution of the individual domains. CoGAN has found applications in movie production, game production, image transformation and domain adaptation.

Text-to-image synthesis methods have been implemented through the use of GAN [10]. However, previous methods encodes entire text descriptions into a sentence vector as a condition for image generation. This method presents good results, but because of the focus on the sentence level important information on the word level is lost. Attentional Generative Adversarial Network or AttnGAN was created to solve this issue by Tao Xu et al [11]. It allows for multi-stage refinement and fine-grained text-to-image generation.

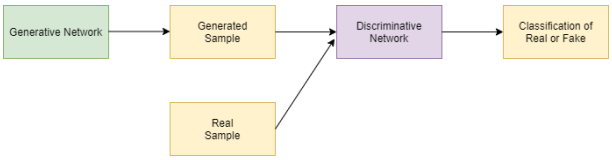

Figure 1: Image of Generative Adversarial Structure

2.2 Analysis of Face Obfuscation

Early research on applying GANs for face obfuscation began by applying parametric face models [8]. The first step was replacing the original face with their new rendered face models. The new rendered face models maintained similar facial expressions as the original. Then using a GAN clean up the image by adding fine details and that outputs a photo realistic image.

A more extensive study on the commonly used obfuscation methods has been conducted [3]. They focused on three attack scenarios being obfuscated face identification, obfuscated and clear face verification, and obfuscated face reconstruction. As well as three threat models with increasing knowledge of the obfuscation methods. Then applying these to eight obfuscation techniques, including GAN based. Results showed that UP-GAN a GAN based obfuscation method performed among the best against such attacks. K-same best methods performed substantially better but did not retain the pose of the subjects and results may be blurry. However, UP-GAN retained pose and clear pictures. We perform similar work following the attack scenarios and threat models but instead focus on a new GAN, StyleGAN developed by Hao et al. [2].

A similar study conducted by McPherson et al [6]. They test to see how modern image recognition systems are able to gather information from images through Pixelation(mosaic), P3, and face blurring (Gaussian blur). They trained a separate neural network model for each method. The ATandT and Facescrub data sets are most similar to our generated faces data set. For each method being tested obfuscation of the entire data set was performed, then split it between training set and test set. It was proven that the methods used still retain enough information in the image.

2.3 Style Based Generators

New developments on the generator architecture of GANs have been developed [2]. In the past most research and development has focused on the discriminator architecture of GANs. Hao et al., developed StyleGAN with these points in mind. A GAN that takes two styles each from a given face and mixes them together based on the latent vectors. This new GAN allows for more control over high-level attributes like pose and identity as well as stochastic variation like freckles and hair. In our study we use the developed StyleGAN technology to compare how it does against current obfuscation methods. Since StyleGANs creation Hao et al., it has further been developed and named StyleGAN2. StyeGan2 improves the quality of images and removes artifacts previously seen in StyleGAN. As well as improvements to the generator and projection of images to the latent space [4].

3 METHODOLOGY/DESIGN

3.1 System Design

The goal of this paper to investigate if StyleGAN is suitable for face obfuscation. StyleGAN uses latent vectors which are random vectors that are fed to a multi-layer perception. By doing so, we are able to change the mean and variance of the output of different convolutional layers of a GAN whose input is a trained constant. Different convolutional layers are believed to have different styles, which support different feature levels of the face images. For example, coarser styles are related to coarser features such as shape, pose, and expression, while the finer ones are related to the fine texture and color of the face.

Style mixing is used to mix styles of two or more images. This characteristic can be used for face obfuscation. For this purpose, the resulting face has the coarse styles of the intended face and the rest from a second face. As illustrated in Figure 2, by finding the 'w' vector for our latent vector and for that of each image, we can manipulate the result and feed it to the network. Therefore, since coarser styles are related to utilities, it is believed that it will have good obfuscation.

Currently, we are able to produce different faces resulting from mixing styles with different levels. Then we are going to create a dataset created from faces of the FFHQ dataset, which will be used to be tested under 4 different attacks: 1- authentication attack in which it is tried to recognise the obfuscated face. 2- verification attack which is used to say if 2 one obfuscated face and one non-obfuscated are from the same face, 3- reconstruction attack which is used for reconstructing obfuscated faces, 4- given the second face and the obfuscated face, we test if it is possible to recognise or reconstruct the original face.

3.2 Threat Model and Attack Simulation

Our threat model is trained using clear images, with or without obfuscated images. To evaluate its performance on recognizing the real faces from the obfuscated images, we test the model with only obfuscated images.

To simulate attacks, we train artificial neural networks to perform the image recognition tasks as our attack model, following McPherson et al.'s work [6]. Two activation units: ReLU (rectified

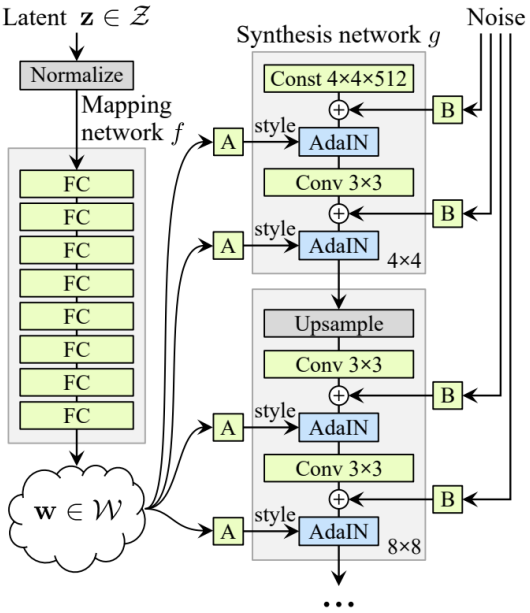

Figure 2: System architecture

linear units) and LeakyReLU are used in this network. Figure 3 an example of the convolutional neural network, which is consisted of max-pooling, sub-sampling, and convolutional layers.

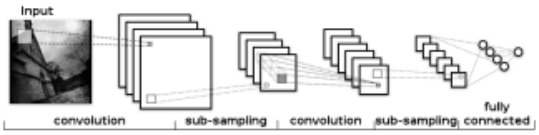

Figure 3: An example of the convolutional neural network

4 EVALUATION

We use FFHQ as our original dataset, which is open-source on GitHub and can be downloaded from a Google Drive link. The dataset consists of 70,000 high-quality PNG images at 1024×1024 resolution and contains considerable variation in terms of age, ethnicity and image background. It also has good coverage of accessories such as eyeglasses, sunglasses, and hats [7].

Using the aforementioned method, we are able to generate our own dataset, as shown in Figure 4, with different style mixing levels. These are examples with the same human faces. By controlling the style mixing levels we can obtain different outputs. A higher level derives the output more likely to the random selected image. Our dataset is consisted of 5 classes, each having 1,000 face images generated using StyleGAN with multiple Style Mixing Levels ranging from 0-2 to 0-6. Among them, 600 are used for training, 200 are for validation, and the other 200 are for testing.

Figure 5 shows each layer, its output shape, and the number of parameters of our implemented CNN model. Figure 6 plots our attack result. With 5 or more sample sets, we observe the attack success rate, presented by the accuracy, higher than 90%. This result indicate that the StyleGAN model is vulnerable against our threat model. A possible explanation is that having only one source

Figure 4: Different style mixing levels: 0-2, 0-3, 0-5, and 0-6, respectively

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 126, 126, 32)      896
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 63, 63, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 61, 61, 64)        18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 30, 30, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 28, 28, 128)       73856
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 14, 14, 128)       0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 12, 12, 128)       147584
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 6, 6, 128)         0
_________________________________________________________________
flatten (Flatten)            (None, 4608)              0
_________________________________________________________________
dense (Dense)                (None, 512)               2359808
_________________________________________________________________
dense_1 (Dense)              (None, 5)                 2565
=================================================================
Total params: 2,603,205
Trainable params: 2,603,205
Non-trainable params: 0
```

Figure 5: Parameters of our CNN attacking model

image for each class is insufficient, as CNN memorizes the same utilities. We consider the potential solutions as adding noise to the related styles, or creating real source images with different poses, expressions, etc.

5 LIMITATIONS

5.1 Number of Source Images

For each of our classes only one source image was used. We believe the reason the model got 100% validation accuracy was due to that our for each class the sample images used had the same utility. In

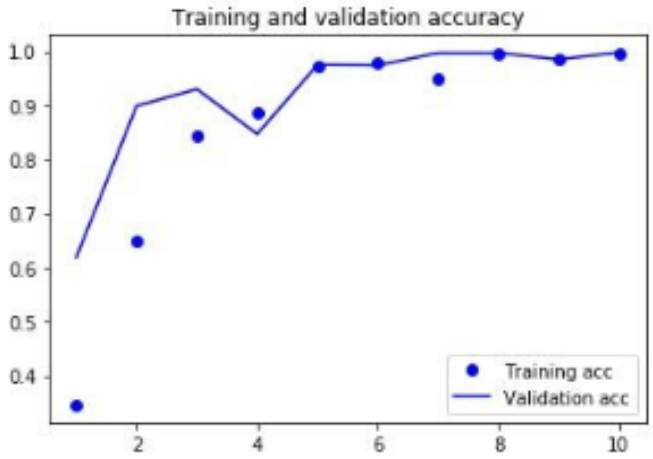

Figure 6: Training and validation accuracy to the size

this case utility is the same head shape, pose and or eye glasses. This is a problem because the deep learning model then believes that for the given class in question the generated images that has the same utility are recognized as the class image. There are two possible solutions this issue. First we use a data set that contains more than one image of an individual. Where each individual has multiple images with different utilities. Second we are to add noise to the data set.

5.2 Background of Generated Images

After a specific image has been obfuscated, its background was changing as well. This can be seen in Figure 4 This was a part of the StyleGAN process. However, if we are using this method to obfuscate images of people we do not want the background to change as well. When we look towards future work we plan on working on a process that does not affect the background of the subject in question.

6 CONCLUSION

There has been growing demands for advanced image obfuscation techniques in photos and videos for privacy and security concerns. In this paper, we exploit an existing algorithm, namely StyleGAN, to generate artificial images of human faces from the FFHQ dataset, and evaluate its robustness against our deep learning threat model. We train our system model with 600 generated images, either obfuscated or not, and test it using another 200 obfuscated images. Our result indicates that the obfuscation was vulnerable to our threat model, with the attack success rate over 90%. We further propose potential solutions as either adding noise to the related styles, or creating real source images with different poses, or expressions, which can be a direction of our future work.

REFERENCES

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu andDavid Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. *CoRR* abs/1406.2661 (2014). arXiv:1406.2661 https://arxiv.org/abs/1406.2661

[2] Hanxiang Hao, David Güera, Amy R. Reibman, and Edward J. Delp. 2019. Robustness Analysis of Face Obscuration. *CoRR* abs/1905.05243 (2019). arXiv:1905.05243 http://arxiv.org/abs/1905.05243

[3] Tero Karras, Samuli Laine, and Timo Aila. 2018. A Style-Based Generator Architecture for Generative Adversarial Networks. *CoRR* abs/1812.04948 (2018). arXiv:1812.04948 http://arxiv.org/abs/1812.04948

[4] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. 2019. Analyzing and Improving the Image Quality of StyleGAN. (2019). arXiv:cs.CV/1912.04958

[5] Ming-Yu Liu and Oncel Tuzel. 2016. Coupled Generative Adversarial Networks. *Advances in Neural Information Processing Systems 29* NIPS 2016 (2016). http://papers.nips.cc/paper/6544-coupled-generative-adversarial-networks.pdf

[6] Richard McPherson, Reza Shokri, and Vitaly Shmatikov. 2016. Defeating Image Obfuscation with Deep Learning. *CoRR* abs/1609.00408 (2016). arXiv:1609.00408 https://arxiv.org/abs/1609.00408

[7] NVLabs. Flickr-Faces-HQ Dataset (FFHQ). (????). https://github.com/NVlabs/ffhq-dataset#flickr-faces-hq-dataset-ffhq.

[8] Qianru Sun, Ayush Tewari, Weipeng Xu, Mario Fritz, Christian Theobalt, and Bernt Schiele. 2018. A Hybrid Model for Identity Obfuscation by Face Replacement. *ECCV* LNCS 2016 (2018). http://openaccess.thecvf.com/content_ECCV_2018/papers/Qianru_Sun_A_Hybrid_Model_ECCV_2018_paper.pdf

[9] Kunfeng Wang, Chao Gou, Yanjie Duan, Yilun Lin, Xinhu Zheng, and aFei Yue Wang. 2017. Generative Adversarial Networks: Introduction and Outlook. *JOURNAL OF AUTOMATICA SINICA* 4 (2017). https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8039016

[10] Zhengwei Wang, Qi She, and Tomas E. Ward. 2019. Generative Adversarial Networks in Computer Vision: A Survey and Taxonomy. *CoRR* abs/1906.01529 (2019). arXiv:1906.01529 https://arxiv.org/abs/1906.01529

[11] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. 2018. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks. *CVPR 2018* (2018). http://openaccess.thecvf.com/content_cvpr_2018/papers/Xu_AttnGAN_Fine-Grained_Text_CVPR_2018_paper.pdf