Exploring Face De-Identification using Latent Spaces

Una M. Kelly
University of Twente,
Enschede, The Netherlands
u.m.kelly@utwente.nl

Luuk Spreeuwers
University of Twente,
Enschede, The Netherlands
l.j.spreeuwers@utwente.nl

Raymond Veldhuis
University of Twente,
Enschede, The Netherlands
r.n.j.veldhuis@utwente.nl

Abstract

We explore a new method to hide identity information in a facial image from face recognition (FR) systems, while only minimally changing the appearance of the image as perceived by humans. We train a decoder network that reverses the mapping of an FR system and use the dissimilarity score function of this FR system to teach the decoder to return images with as little identity information as possible, while using a visual loss to change the image as little as possible visually. We show that these obfuscation attacks are also successful when the FR system is unknown. We analyse the obfuscated images in latent space and show that our approach as well as an existing method can be easily circumvented by applying the same obfuscation method to the enrolled faces as to the probe images. We suggest an adaptation that can help prevent this circumvention.

1. Introduction

Facial images uploaded on for example social media may be "scraped" from the internet by the social media platform or by other entities such as ClearView AI [7]. If someone were to upload an image and would like their friends and family to recognise them, without revealing the identity of the image to the social media platform owner or other malicious users, they can apply de-identification to their images before uploading. Such *de-identified* images are also called *adversarial attacks* or *identity-obfuscated* images. If an obfuscation method is public (e.g. [1]) we show that the obfuscation can be easily circumvented by by using a face recognition (FR) system to compare a probe image to a gallery image *and* the obfuscated gallery image. Our goal is to hide identity information from FR systems so that automated, mass collection of identity information is prevented. At the same time, we change the visual appearance of the image as little as possible so that to humans the identity information seems unchanged.

Our contributions include: a new approach to appearance-preserving de-identification, analysing the latent space of an FR system to understand the effect of obfuscation, exposing a vulnerability of our and an existing method, and an approach to make appearance-preserving de-identification harder to circumvent. We also show that our obfuscated images are succesful in black-box settings, i.e. knowing the FR system is not necessary.

2. Related Work

Identity obfuscation of facial images has been achieved using methods that obfuscate identity of faces in images or videos by replacing the faces [18], processing the faces [5, 6], blurring or pixelating the face region, etc. [14]. These methods result in faces that can no longer be recognised by humans nor by automated FR systems. Some methods to circumvent pixelation, occlusion etc. have been proposed as well, showing that the identity obfuscation using these methods can be reversed [12].

A Generative Adversarial Network (GAN) is used in Privacy-protecing GAN [20] to hide identity information, while trying to maintain visual appearance as much as possible. It focuses on keeping soft biometric information such as ethnicity, age and gender. AnonymousNet [11] also changes identity information to achieve de-identification and introduces metrics that can be used to measure how succesful de-identification is.

These methods have in common that they can successfully hide identity information, but also visually change the image so that the identity is no longer visible to humans either. However, our aim is to only hide identity from automated FR, not from humans. With such a tool people could edit their images and still share or upload images to e.g. social media platforms.

2.1. Optimisation vs. learning

When using deep-learning-based FR systems, we can use the network gradients to generate images whose embeddings in the FR latent space are significantly different from the original image's embedding. This can be achieved using optimisation, in which case an image-specific adversarial perturbation can be computed. Another approach is

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.

to use learning to train a network that takes an image as input and outputs a de-identified image. The disadvantage of learning (i.e. training a decoder or other network) over optimisation (gradient descent) is that the decoder network has to learn one model that can de-identify many different images, while optimisation applies an adversarial perturbation tailored specifically to one image. On the other hand, the advantage of learning over optimisation is that there is no need for retraining or calculating image-specific gradients every time a new image is de-identified. Furthermore, the trained network can be shared, so that individuals can apply it on their own images, without needing to share their original images. Both approaches could also be combined [21].

Learning-based de-identification

AdvFaces [3] uses a GAN-based approach to generate subtle perturbations in a face image, both for obfuscation and impersonation attacks. The resulting images can succesfully fool several (unseen) state-of-the-art FR systems, where some systems are fooled more succesfully than others. While our approach shares some similarities with AdvFaces used for obfuscation, we focus on preventing reversibility of obfuscation and simplify the training by excluding the GAN element. We compare our approach to AdvFaces and show that unless obfuscated vs. obfuscated images are taken into account during training, the obfuscation can essentially be bypassed by applying the same obfuscation technique on both images that are being compared.

Optimisation-based de-identification

An approach based on gradient descent is used in [1] to hide identities from two (pretrained and frozen) identification networks. The Penalized Fast Gradient Value Method (P-FGVM) is used on a loss function that consists of an adversarial loss term and a second term that measures the “realism” of the image. This method was applied on a subset of CelebA that contains 900 images of 30 identities, resulting in images that preserve facial image quality and at the same time succesfully confuse the identity classifier that was trained to classify the 30 identities. This as such is not very useful, since it only enables identity obfuscation for these 30 identities, and only for the identity classifier trained specifically for these identities. This approach could potentially be extended so that it can be applied to new identities as well as other FR systems.

In [17] “cloaks”, which are imperceptible pixel-level changes, are added to images. If training data is “scraped” from the internet to train FR models, then such images “poison” the training data. Resulting FR system no longer identify (uncloaked) images correctly.

Both a universal image perturbation and an image-specific perturbation are combined in [22] to generate adversarial face images that fool FR systems. Optimisation is used to maximise the mean distance between the embedding of a probe image and the embedding of an enrolled image. It is briefly mentioned that adversarial vs. adversarial comparisons are nowhere near as succesful as adversarial vs. original attacks and a solution to this problem is proposed, which consists of adding a noise vector to the adversarial image embedding before optimising the distance to the original image embedding.

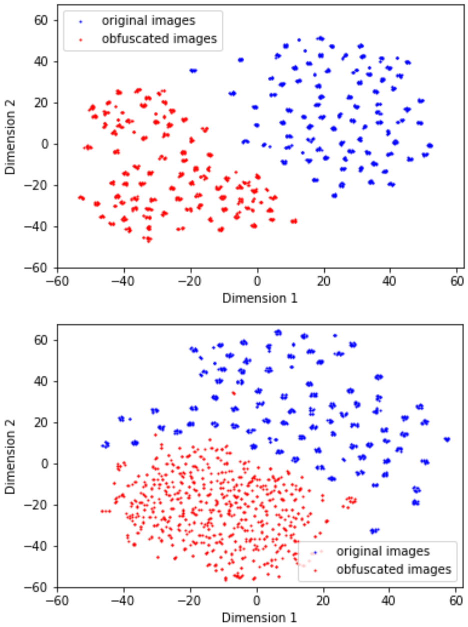

Figure 1. Visualisation of embeddings in the FR latent space using T-SNE [8] for dimensionality reduction. Top: using only a simple latent loss (Eq. 4). Bottom: using an improved loss (Eq. 8). Using the improved loss leads to less clustering by identity, as well as less of a separation between normal and identity-obfuscated images.

3. Proposed System

FR systems based on Convolutional Neural Networks (CNN) take an image as input and return a latent vector that contains identity information. Embeddings corresponding to images of the same person are clustered in the FR latent space (see Fig. 1). Our aim is to adapt the images in such a way that embeddings are no longer clustered by identity.

Since embeddings in the FR latent space contain information essential to a person’s identity, we would like to use them as input to a Decoder that we train to visually reconstruct the original facial images. We aim to prevent recognition of these images by applying appropriate loss functions. However, since some information such as background or expression is not relevant to identity, the FR latent embed-

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.

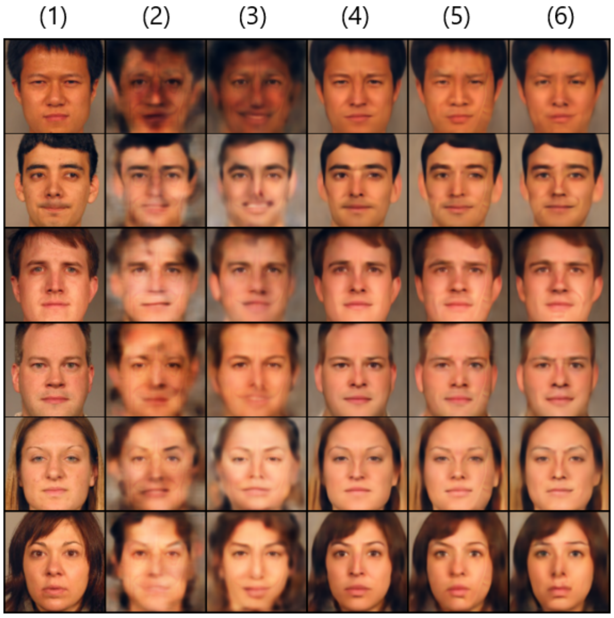

Figure 2. Original FRGC [13] and obfuscated images under different settings. The autoencoder used for obfuscation consists of a MobileFaceNet network as the encoder and a decoder that maps from latent space back to image space. 1: Original images. 2: Inverting the FR latent space using $\mathcal{L}_{\text{pixel}}$, $\mathcal{L}_{\text{percept}}$ and $\mathcal{L}_{\text{latent}}$. 3: Same as 2, but adding 4 fully connected layers before applying deconvolutions. 4: Same as 2, but with a supporting Encoder. 5: Same as 4, but including $\mathcal{L}_{\text{aux}}$ in the loss. 6: Same as 5, but including $\mathcal{L}_{\text{vis}}$ in the loss. All examples are generated using random images from the validation set.

ding alone is not sufficient to reconstruct a facial image with sufficient quality. We show in Fig. 2 to what extent faces can be reconstructed using only the embedding of the FR system.

To improve the image reconstructions we train a supporting encoder network that compresses the image, but whose latent space also contains information about attributes such as background, expression, etc. Using both the embedding of the FR system as well as the embedding of the encoder as input, we train an improved decoder, see Fig. 3.

Instead of adding a supporting encoder we also tried increasing the complexity of the Decoder network by adding several fully connected layers before applying deconvolutions to the latent embeddings. If the information in the latent space is simply too entangled to faithfully reconstruct images, this would allow the network to first disentangle this information. While this more complex network did results in visually improved reconstructions, simply adding a supporting Encoder network as described above resulted in reconstructions of far better quality. It is likely possible to achieve better results using only the FR latent embedding as input, perhaps by including skip-connections as in U-Net [15] or reusing information from the latent space at each deconvolution as is done in StyleGAN [10]. We choose to keep our approach more simple, focusing on studying the effects of obfuscation in latent space to help us understand how obfuscation can be improved. The losses we suggest and conclusions drawn in this paper can be applied to existing methods, or to more complex networks in future research to further improve results.

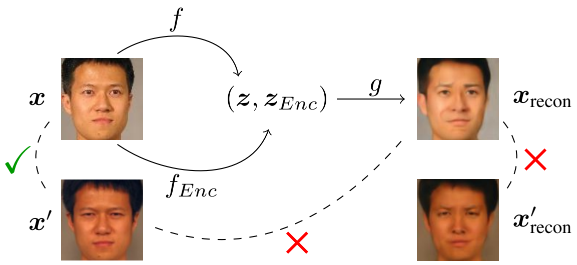
f
$\boldsymbol{x}$
$(\boldsymbol{z}, \boldsymbol{z}_{Enc})$
g
$\boldsymbol{x}_{\text{recon}}$
✓
f_{Enc}
✗
$\boldsymbol{x}'$
✗
$\boldsymbol{x}'_{\text{recon}}$

Figure 3. Overview of our obfuscation approach. $\boldsymbol{x}$ and $\boldsymbol{x}'$ are original images of the same person. On the right the two corresponding reconstructed (i.e. obfuscated) images are shown. $\boldsymbol{x}'_{\text{recon}}$ was obfuscated in the same way as $\boldsymbol{x}_{\text{recon}}$. The FR system correctly matches the two original images, but not the obfuscated images. Dashed lines represent FR comparisons.

3.1. Training

Let f denote the FR mapping from image to latent space, i.e. $f(\boldsymbol{x}) = \boldsymbol{z}$. Let f_{Enc} be the mapping of the supporting encoder, i.e. $f_{\text{Enc}}(\boldsymbol{x}) = \boldsymbol{z}_{\text{Enc}}$. Furthermore, let g denote the decoder mapping back to image space. Let d be the dissimilarity score function used by the FR system to compare latent embeddings. We train the decoder using the following losses

$$\mathcal{L}_{\text{pixel}} = \mathbb{E}_{\boldsymbol{x}}\left[||\boldsymbol{x} - \boldsymbol{x}_{\text{recon}}||_2^2\right], \tag{1}$$
$$\mathcal{L}_{\text{percept}} = \mathbb{E}_{\boldsymbol{x}}\left[||f^1(\boldsymbol{x}) - f^1(\boldsymbol{x}_{\text{recon}})||_2^2\right], \tag{2}$$
$$\mathcal{L}_{\text{latent}} = \mathbb{E}_{\boldsymbol{x}}\left[-\max(T, d(\boldsymbol{z}, \boldsymbol{z}_{\text{recon}}))\right], \tag{3}$$

where $\boldsymbol{x}_{\text{recon}} = g(\boldsymbol{z}, \boldsymbol{z}_{\text{Enc}})$, $\boldsymbol{z}_{\text{recon}} = f(\boldsymbol{x}_{\text{recon}})$ and T is an upper bound on d. The perceptual loss [9] uses the first hidden layer of the FR system, denoted f^1. It has the same purpose as the pixel loss, which is to faithfully reconstruct the faces on a visual level.

Since $d(\boldsymbol{z}, \boldsymbol{z}_{\text{recon}})$ is potentially unbounded we set an upper bound T when calculating $\mathcal{L}_{\text{latent}}$. The higher T is set, the more artifacts appear in the reconstructed images, so we empirically choose a value for T that is at the lower end of dissimilarity scores for impostor pairs, since this is sufficient for an obfuscated image to be rejected. The total loss $\mathcal{L}$ is

$$\mathcal{L} = \gamma_1 \mathcal{L}_{\text{pixel}} + \gamma_2 \mathcal{L}_{\text{percept}} + \gamma_3 \mathcal{L}_{\text{latent}}. \tag{4}$$

In Fig. 4 and Table 1 we show that while models trained with this loss function produce obfuscated images that are very succesful when they are compared to normal images,

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.

they are much less succesful when obfuscated images are compared with each other. In the next section we suggest additions to the loss function to alleviate this.

3.2. Improving Identity Obfuscation

We extend the loss function with the following term

$$\mathcal{L}_{\text{aux}} = \mathbb{E}_{\boldsymbol{x}} \left[-\max\left(T, d((f(\boldsymbol{x}_{\text{recon}}), f(\boldsymbol{x}'_{\text{recon}})))\right] . \quad (5)$$

Here, $\boldsymbol{x}_{\text{recon}}$ and $\boldsymbol{x}'_{\text{recon}}$ are two obfuscated images of the same identity. This loss ensures that two images of the same identity cannot be successfully matched if they are both reconstructed. The total loss is now the sum of four loss terms

$$\mathcal{L} = \gamma_1 \mathcal{L}_{\text{pixel}} + \gamma_2 \mathcal{L}_{\text{percept}} + \gamma_3 \mathcal{L}_{\text{latent}} + \gamma_4 \mathcal{L}_{\text{aux}} . \quad (6)$$

In the case that the FR system uses a similarity score function, the same approach can be applied by changing the sign of the latent losses, and changing $\max$ to $\min$ in Eq. 1 and 5.

While this improved loss function indeed ensures that obfuscated images are not only successful when they are compared to normal images, but also when they are compared to other identity-obfuscated images. However, the extra loss also leads to visible artifacts in the reconstructed images (see 5th column in Fig. 2). Therefore, we add another term to the loss function that penalises these artifacts.

3.3. Reducing Artifacts

Including $\mathcal{L}_{\text{aux}}$ leads to visible artifacts in the reconstructed images (3$^{\text{rd}}$ column in Fig. 2). While larger values for γ_3 and γ_4 in Eq. 6 lead to better obfuscation, this also makes the artifacts more visible. Therefore, we add another term to the loss function by taking the difference between the mean of the reconstructed images and the mean of the original images. If there are systematic artifacts in the reconstructions what is left are artifacts, which is what we want to minimise.

$$\mathcal{L}_{\text{vis}} = \left(\mathbb{E}_{\boldsymbol{x}}\left[\boldsymbol{x}\right] - \mathbb{E}_{\boldsymbol{x}}\left[\boldsymbol{x}_{\text{recon}}\right]\right)^2, \quad (7)$$

The total loss is now the sum of five loss terms

$$\begin{aligned} \mathcal{L} = \gamma_1 \mathcal{L}_{\text{pixel}} + \gamma_2 \mathcal{L}_{\text{percept}} + \gamma_3 \mathcal{L}_{\text{latent}} \\ + \gamma_4 \mathcal{L}_{\text{ref}} + \gamma_5 \mathcal{L}_{\text{vis}} . \end{aligned} \quad (8)$$

We use MobileFaceNet [2] with a dissimilarity score function that calculates the angle between latent embedding vectors to implement the loss functions. We evaluate our obfuscated images with three FR systems that were not used during training: FaceNet [16], ArcFace [4] and a Commercial-Off-The-Shelf (COTS) FR system. We use FRGC [13] for training and testing. Our training set comprises 18143 images of 482 identities, the validation set contains 3629 images of 86 identities. There is no overlap in identities between training and validation set. Images are cropped and aligned as shown in Fig. 2. Distributions and visualisations were estimated using the validation set.

```
$\theta_D, \theta_{\text{Enc}} \leftarrow$ initialize network parameters
repeat
    $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}$    ▷ Draw $N$ samples from the dataset
    $\boldsymbol{x}'^{(1)}, \ldots, \boldsymbol{x}'^{(N)}$    ▷ Draw another $N$ samples (same IDs)
    $\boldsymbol{z}^{(i)} = f(\boldsymbol{x}^{(i)}), \quad i = 1, .., N$
    $\boldsymbol{z}'^{(i)} = f(\boldsymbol{x}'^{(i)}), \quad i = 1, .., N$    ▷ Get FR embeddings
    $\boldsymbol{z}^{(i)}_{\text{Enc}} = f_{\text{Enc}}(\boldsymbol{x}^{(i)}), \quad i = 1, .., N$
    $\boldsymbol{z}'^{(i)}_{\text{Enc}} = f_{\text{Enc}}(\boldsymbol{x}'^{(i)}), \quad i = 1, .., N$    ▷ Get Encoder emb.
    $\boldsymbol{x}^{(i)}_{\text{recon}} = D(\boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i)}_{\text{Enc}}), \; i = 1, .., N$    ▷ Reconstr. images
    $\boldsymbol{z}^{(i)}_{\text{recon}} = f(\boldsymbol{x}^{(i)}_{\text{recon}}), \quad i = 1, .., N$
    $\boldsymbol{z}'^{(i)}_{\text{recon}} = f(\boldsymbol{x}'^{(i)}_{\text{recon}}), \quad i = 1, .., N$    ▷ Get new FR emb.

    $\mathcal{L}_{\text{pixel}} = \frac{1}{N} \sum_{i=0}^{N} \text{MSE}(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(i)}_{\text{recon}})$    ▷ Compute losses
    $\mathcal{L}_{\text{percept}} = \frac{1}{N} \sum_{i=0}^{N} ||f^1(\boldsymbol{x}^{(i)}) - f^1(\boldsymbol{x}^{(i)}_{\text{recon}})||,$
    $\mathcal{L}_{\text{latent}} = -\frac{1}{N} \sum_{i=0}^{N} \max(T, d(\boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i)}_{\text{recon}}))$
    $\mathcal{L}_{\text{aux}} = -\frac{1}{N} \sum_{i=0}^{N} \max(T, d(\boldsymbol{z}^{(i)}_{\text{recon}}, \boldsymbol{z}'^{(i)}_{\text{recon}}))$
    $\mathcal{L}_{\text{vis}} = \text{MSE}(\frac{1}{N} \sum_{i=0}^{N} \boldsymbol{x}^{(i)}, \frac{1}{N} \sum_{i=0}^{N} \boldsymbol{x}^{(i)}_{\text{recon}})$
    $\mathcal{L} = \gamma_1 \mathcal{L}_{\text{pixel}} + \gamma_2 \mathcal{L}_{\text{percept}} - \gamma_3 \max(T, \mathcal{L}_{\text{latent}})$
        $-\gamma_4 \max(T, \mathcal{L}_{\text{aux}}) + \gamma_5 \mathcal{L}_{\text{vis}}$
    $\theta_D \leftarrow \theta_D - \nabla_{\theta_D} \mathcal{L}$    ▷ Gradient update on decoder
    $\theta_{\text{Enc}} \leftarrow \theta_{\text{Enc}} - \nabla_{\theta_{\text{Enc}}} \mathcal{L}$    ▷ Gradient update on encoder
until convergence
```

4. Evaluation Metrics

To measure and compare the performance of our models, we compute the Attack Success Rate (Eq. 9) using dissimilarity scores of the face recognition system used for training, as well as three other FR systems.

$$\text{Attack Success Rate} = \frac{\#(\text{Comparison scores} < \tau)}{\text{Total \#Comparisons}} \quad (9)$$

Here, τ is the decision threshold at which the False Non-Match Rate of the corresponding FR system is minimal under the constraint that the False Match Rate $<0.1\%$.

To measure how well images are reconstructed, we use the Structural Similarity Index Measure (SSIM) [19]. We only report SSIM values for our own models, since other models such as AdvFaces were trained using different data.

5. Experiments

We train all models for 200 epochs using a batch size of 64. Results after training a decoder using only latent embeddings of the FRS as input are shown in the 2nd and 3rd column in Fig. 2. Results after training a decoder using a supporting encoder network are shown in the same figure, in columns 4-6, where the last column depicts results after improving visual quality using the loss in Eq. 7.

The dissimilarity scores of reconstructed (i.e. obfuscated) images compared to the original images are shown in Fig. 4.

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.

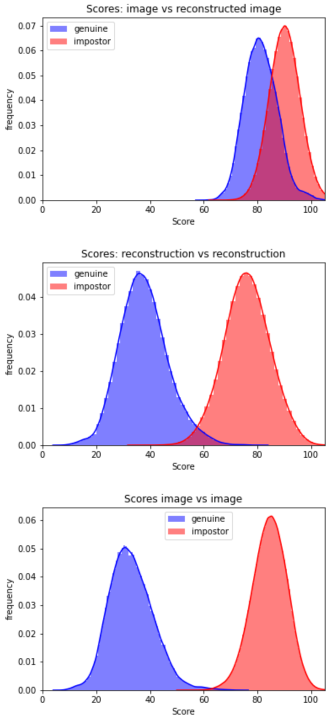

Figure 4. Top row: MobileFaceNet (used during training) dissimilarity scores for the reconstructed images (reconstruction vs. normal image). Second row: MobileFaceNet genuine and impostor comparison scores where all images are reconstructed images (reconstruction vs. reconstruction). The fact that the two distributions have been separated shows that the identity obfuscation is essentially reversed. Bottom row: scores on original images.

These results indicate that this method is very useful to obfuscate identity while at the same time maintaining visual appearance. However, when both probe and enrolled images are obfuscated, the dissimilarity scores show that the identities can be extracted, circumventing the obfuscation. This is shown in Fig. 4 for our suggested approach. The same effect is shown for the existing approach AdvFaces [3] in the second row in Fig. 5.

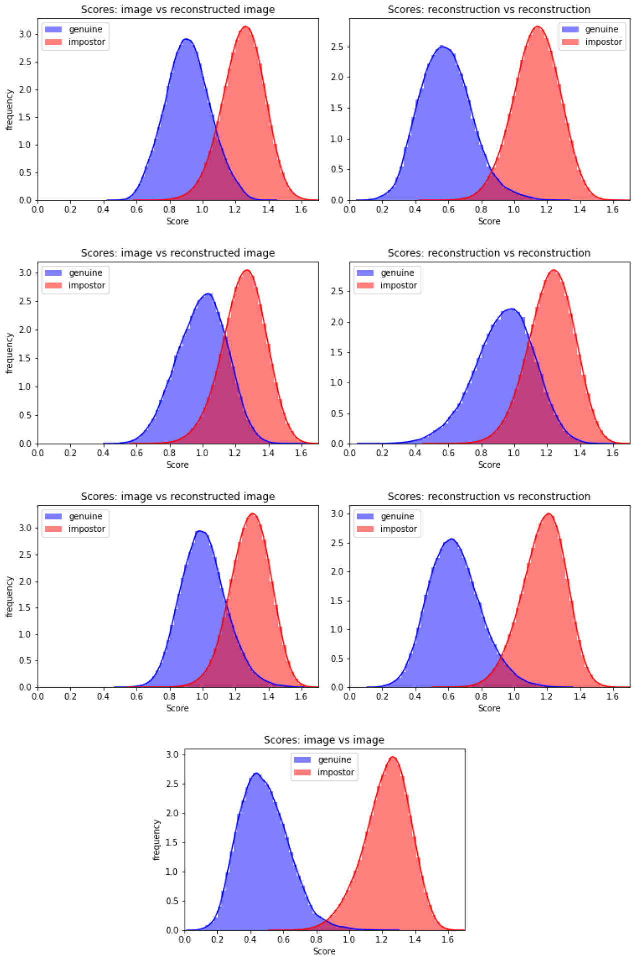

Figure 5. These distributions were estimated using an unseen FR system (Inception). Top row: results of our basic obfuscation approach. Second row: our improved obfuscation approach (including artifact improvement). Third row: AdvFaces [3]. The last histogram shows the genuine and impostor scores for the original images. The genuine score distributions for our basic approach and AdvFaces are much more similar to the original genuine distribution, leading to less successful identity obfuscation.

5.1. Results of Improved Identity Obfuscation

We add the loss proposed in Eq. 5 and retrain the decoder network. The reconstructions produced by the decoder trained with the loss in Eq. 6 can be seend in the 5th column in Fig. 2. While these reconstructions are quite good, we see certain patterns/artifacts that we would like te remove, since our aim is to visually change the images a little as possible. This could be achieved by for example adding a GAN type loss, that learns to distinguish between real and reconstructed images. This would involve training an additional Discriminator-type network. In order to keep our approach more simple, we instead add an extra visual loss that compares the mean of all original images with the mean of all reconstructed images. This seems to successfully reduce the artifacts caused by adding the extra loss from Eq. 7. The improved reconstructions are shown in the last column in Fig. 2.

We empirically determine a good balance of the different losses: $\gamma_1 = 1, \gamma_2 = 0.05, \gamma_3 = 0.05, \gamma_4 = 5$, $\gamma_5 = 0.2$. Generally speaking, the performance (obfuscation) improves for larger γ_2 and γ_3, but leads to decreased visual quality. We set the upper limit $T = 80$, since this is a sufficient score for any comparisons to be rejected.

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.

Table 1. Attack Success Rates (%).

MobileFaceNet	Obf. vs. normal	Obf. vs. obf.	SSIM
Simple loss (Eq. 4)	100.0	0.7	0.93
Auxiliary loss (Eq. 6)	100.0	85.3	0.91
Improved loss (Eq. 8)	99.8	83.5	0.91
Inception (black box)			
Simple loss (Eq. 4)	88.5	17.8	
Auxiliary loss (Eq. 6)	89.5	78.1	
Improved loss (Eq. 8)	94.7	85.5	
AdvFaces	97.9	23.6	
ArcFace (black box)			
Simple loss (Eq. 4)	99.2	3.3	
Auxiliary loss (Eq. 6)	99.6	76.4	
Improved loss (Eq. 8)	99.4	79.6	
AdvFaces	94.6	3.6	
COTS (black box)			
Simple loss (Eq. 4)	43.7	0.1	
Auxiliary loss (Eq. 6)	71.1	7.2	
Improved loss (Eq. 8)	84.0	14.0	
AdvFaces	1.1	0.0	

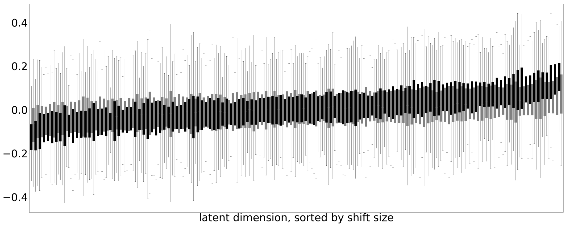

Figure 6. A boxplot comparing the latent shifts in each dimension of the FR latent space. The shifts when using only the simple latent loss (Eq. 4) are shown in black, the shifts when using the improved loss (Eq. 8) are shown in grey. The grey shifts are more centered around zero, while the black shifts show a more consistent pattern. This supports our hypothesis that when only original and obfuscated images are compared with each other during training, then obfuscation essentially leads to a translation of embeddings in latent space, but is less successful at undoing clustering of embeddings.

5.2. Analysing obfuscation in latent space

We hypothesise that the reason identity obfuscation does not work well when obfuscated images are compared to each other instead of to unaltered images, is that the latent FR embeddings of obfuscated images are shifted in latent space, but that the embeddings of identities are still clustered together. What we actually want to achieve, is to prevent clustering of embeddings according to their identity. Including the auxiliary loss from Eq. 5 during training should prevent this. We analyse the shifts in latent space by subtracting latent embeddings z from latent embeddings of the corresponding reconstructed images z_{recon}. We examine these shifts for each latent dimension (128 in our case) in Fig. 6. We also apply T-SNE to visualise the latent space, see Fig. 1.

The boxplot in Fig. 6 shows that the shifts without using the auxiliary loss indeed show a specific pattern. When the auxiliary loss is included, this pattern becomes less obvious, and the shifts are more centered around 0 in all dimensions. When obfuscated images are also compared to each other during training, this leads to a better scattering of the embeddings of obfuscated images and therefore improves robustness of identity obfuscation. Comparing these results with the values in Table 1, we see that at the same time, including the auxiliary loss actually increases the attack success rate.

6. Conclusion & Future Work

We successfully generated de-identified images and showed that also in black-box scenarios face recognition systems can no longer reliably extract identity information from them, while visually they remain very similar. The proportion of resulting de-identified images that is no longer correctly identified by deep learning-based FR is significant. Malicious FR systems that were trained with "scraped" data tend to fall into this class, e.g. [7]. This de-identification comes at a very low cost, since our method can be shared and automated easily, especially compared to optimisation-based approaches.

We showed that a very simple way to bypass identity obfuscation is to apply the same obfuscation method to the enrolled face image. Training with a loss that penalises the similarity according to the FR between two obfuscated images (of the same identity) reduces the visual quality a little, but makes de-identitification much harder to circumvent.

Our exploration of the effects of de-identification on the embeddings in the latent space of an FR system has led to useful insights and can be used to improve other methods for de-identication. For example, the auxiliary loss we propose in Eq. 5 could also be used to improve an optimisation-based method for de-identification. The visual loss we propose in Eq. 7 is a simpler and computationally less expensive way to reduce artifacts caused by de-identification than using a GAN.

References

[1] E. Chatzikyriakidis, C. Papaioannidis, and I. Pitas. Adversarial face de-identification. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 684–688, 2019.

[2] S. Chen, Y. Liu, X. Gao, and Z. Han. MobileFaceNets: Efficient CNNs for accurate real-time face verification on mobile devices. *CoRR*, abs/1804.07573, 2018.

[3] D. Deb, J. Zhang, and A. K. Jain. Advfaces: Adversarial face synthesis. In *2020 IEEE International Joint Conference on Biometrics (IJCB)*, pages 1–10, 2020.

[4] J. Deng, J. Guo, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *CoRR*, abs/1801.07698, 2018.

[5] O. Gafni, L. Wolf, and Y. Taigman. Live face de-identification in video. In *2019 IEEE/CVF International*

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.

Conference on Computer Vision (ICCV), pages 9377–9386, 2019.

[6] R. Gross, L. Sweeney, J. Cohn, F. De la Torre, and S. Baker. *Face De-identification*, pages 129–146. 07 2009.

[7] K. Hill. The secretive company that might end privacy as we know it. `https://www.nytimes.com/2020/01/18/technology/clearview-privacy-facial-recognition.html`, Jan. 2020. Accessed: 2022-4-29.

[8] G. Hinton and S. Roweis. Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002.

[9] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision – ECCV 2016*, pages 694–711, Cham, 2016. Springer International Publishing.

[10] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. *CoRR*, abs/1812.04948, 2018.

[11] T. Li and L. Lin. Anonymousnet: Natural face de-identification with measurable privacy. *CoRR*, abs/1904.12620, 2019.

[12] R. McPherson, R. Shokri, and V. Shmatikov. Defeating image obfuscation with deep learning. *CoRR*, abs/1609.00408, 2016.

[13] P. J. Phillips, P. J. Flynn, T. Scruggs, K. W. Bowyer, Jin Chang, K. Hoffman, J. Marques, Jaesik Min, and W. Worek. Overview of the face recognition grand challenge. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 947–954 vol. 1, 2005.

[14] S. Ribaric and N. Pavesic. An overview of face de-identification in still images and videos. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 04, pages 1–6, 2015.

[15] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015.

[16] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015.

[17] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao. Fawkes: Protecting personal privacy against unauthorized deep learning models. *CoRR*, abs/2002.08327, 2020.

[18] Q. Sun, A. Tewari, W. Xu, M. Fritz, C. Theobalt, and B. Schiele. A hybrid model for identity obfuscation by face replacement. *ArXiv*, abs/1804.04779, 2018.

[19] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[20] Y. Wu, F. Yang, and H. Ling. Privacy-protective-GAN for face de-identification. *CoRR*, abs/1806.08906, 2018.

[21] W. Xia, Y. Zhang, Y. Yang, J. Xue, B. Zhou, and M. Yang. Gan inversion: A survey. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (01):1–17, jun 5555.

[22] J. Zhang, J. Sang, X. Zhao, X. Huang, Y. Sun, and Y. Hu. *Adversarial Privacy-Preserving Filter*, page 1423–1431. Association for Computing Machinery, New York, NY, USA, 2020.

Architecture

Operation	Kernel	Stride	Size
$f_{\text{Enc}}(\boldsymbol{x})$			$112 \times 112 \times 3$ (Input)
Convolution	2×2	2×2	$56 \times 56 \times 32$
Convolution	2×2	2×2	$28 \times 28 \times 32$
Convolution	2×2	2×2	$14 \times 14 \times 64$
Convolution	2×2	2×2	$7 \times 7 \times 64$
$g(\boldsymbol{z})$			$1 \times 1 \times 128$ (Input)
Fully connected	-	-	$7 \times 7 \times 128$
Concat. with $f_{\text{Enc}}(\boldsymbol{x})$			$7 \times 7 \times 192$
Upsample & Conv.	3×3	1×1	$14 \times 14 \times 128$
Upsample & Conv.	3×3	1×1	$28 \times 28 \times 64$
Upsample & Conv.	3×3	1×1	$56 \times 56 \times 64$
Upsample & Conv.	3×3	1×1	$112 \times 112 \times 32$
Upsample & Conv.	3×3	1×1	$112 \times 112 \times 3$
Optimizer	RMSProp(lr $= 10^{-4}$, $\alpha = 0.9$)		
Batch size	64		
Bias	False, except in the last layer of the decoder, where it is untied.		
BatchNorm	After each convolution.		
Weight init.	Isotropic gaussian ($\mu = 0, \sigma = 0.01$)		
Bias init.	Sigmoid$^{-1}(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_i)$, returns the average of images $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ when passed through the sigmoid activation.		
Nonlinearity	Leaky ReLU of slope 0.02, except for the last convolution in g, which is followed by a sigmoid activation function.		

Authorized licensed use limited to: UNIVERSITY OF TWENTE.. Downloaded on January 27,2023 at 07:48:37 UTC from IEEE Xplore. Restrictions apply.