Published in *Proc. of 11th IAPR International Conference on Biometrics (ICB 2018). Gold Coast, Australia, Feb. 2018*

Semi-Adversarial Networks: Convolutional Autoencoders for Imparting Privacy to Face Images

Vahid Mirjalili [1] mirjalil@msu.edu
Sebastian Raschka [1] raschkas@msu.edu
Anoop Namboodiri [2] anoop@iiit.ac.in
Arun Ross [1] rossarun@cse.msu.edu

[1] Michigan State University, East Lansing, USA
[2] International Institute of Information Technology, Hyderabad, India

Abstract

In this paper, we design and evaluate a convolutional autoencoder that perturbs an input face image to impart privacy to a subject. Specifically, the proposed autoencoder transforms an input face image such that the transformed image can be successfully used for face recognition but not for gender classification. In order to train this autoencoder, we propose a novel training scheme, referred to as semi-adversarial training in this work. The training is facilitated by attaching a semi-adversarial module consisting of an auxiliary gender classifier and an auxiliary face matcher to the autoencoder. The objective function utilized for training this network has three terms: one to ensure that the perturbed image is a realistic face image; another to ensure that the gender attributes of the face are confounded; and a third to ensure that biometric recognition performance due to the perturbed image is not impacted. Extensive experiments confirm the efficacy of the proposed architecture in extending gender privacy to face images.

1. Introduction

Biometric face recognition refers to the use of face images for recognizing an individual in an automated manner [10]. A typical face recognition system employs a *face matcher* that compares two face images and determines the degree of similarity or dissimilarity between them. This comparison operation can be used to (a) *verify* the claimed identity of an input face image or (b) determine the *identity* of an unknown face image by comparing it against a set of known face images.

While face images collected by a biometric system are expected to be used only for *recognition* of individuals [12], recent research has established the possibility of automatically deducing additional information about an individual from their face image [2]. For example, information about a person's age, gender, race, or health can be obtained by using a *soft biometric classifier* (*e.g.*, a gender classifier) that can extract this information from a single face image [28]. While the extraction of soft biometric data (sometimes referred to as *attributes*) can be used to improve the performance of a biometric system [26, 15], it also raises several *privacy* concerns associated with gleaning information without an individual's consent. Further, such an automated analysis can be potentially misused for age-based or gender-based profiling that can undermine the use of biometrics in many applications [4].

Given these concerns, researchers have discussed the possibility of *de-identifying* a face image prior to storing it in a database [20]. While de-identification has tremendous applications in surveillance systems, it can irrevocably compromise the biometric utility of a face image [6]. However, in many applications, it is necessary to *retain* the biometric utility of the face image while *suppressing* the possibility of gleaning additional information, such as gender [18]. This type of *differential privacy* [21] is expected to enhance the privacy of face images stored in a database while at the same time ensuring that biometric recognition is not unduly affected.

In this work, we develop a convolutional autoencoder (CAE) that generates a perturbed face image that can be successfully used by a *face matcher* but not by a *gender classifier*. The proposed CAE is referred to as a **semi-adversarial network** since its output is adversarial to the gender classifier but not to the face matcher. The proposed network can be easily appropriated for use with other attributes (such as age or race). In principle, the design of the semi-adversarial network can be utilized in other problem domains where there is a need to confound some classifiers while retaining the utility of other classifiers.

1.1. Related work

A number of aspects of privacy protection has been studied in the biometric literature [19, 24, 21, 18]. On one hand, there are face de-identification techniques [11, 20, 7] where a face image is modified in order to confound a face matcher. On the other hand, as inspired by the work of Othman and Ross [21] and later promoted by Sim and Zhang [27], the goal is to selectively confound or preserve a set of attributes that can be deduced from face images. Specifically, a few methods for suppressing the gender attribute have been presented [25, 29, 21]. Recently, a new method for protecting privacy with practical applications for biometric databases was proposed in [18], where input face images were modified with respect to a specific gender classifier. In this case, perturbations were derived based on a specific gender classifier, the perturbations did not significantly impact the match scores of a face matcher.

In this paper, we provide an alternative solution by designing a convolutional autoencoder that transforms input images such that the performance of an *arbitrary* gender classifier is impacted, while that of an *arbitrary* face matcher is retained. The contributions of this paper, in this regard, are the following: (a) formulating the privacy-preserving problem in terms of a convolutional autoencoder that does *not* require prior knowledge about the gender classifier nor the face matcher being used; (b) incorporating an explicit term related to the matching accuracy in the objective function which ensures that the *utility* of the perturbed images is not negatively impacted; (c) developing a *generalizable* solution that can be trained on one dataset and applied to other previously unseen datasets.

To the best of our knowledge, this is the first work where adversarial training is used to design a generator component that is able to maximize the performance with respect to one classifier while minimizing the performance with respect to another. Experimental results show that the proposed method of semi-adversarial learning for multi-objective functions is efficient for deriving perturbations that are generalizable to other classifiers that were not used (or not available) during training.

2. Proposed method

2.1. Problem formulation

Let $X \in \mathbb{R}^{m \times n \times c}$ denote a face image having c channels each of height m and width n. Let $f_G(X)$ denote a binary gender classifier that returns a value in the range $[0, 1]$, where 1 indicates a "Male" and 0 indicates a "Female". Let $f_M(X_1, X_2)$ denote a face matcher that computes the match score between a pair of face images, X_1 and X_2. The goal of this work is to construct a model $\phi(X)$, that perturbs an input image X such that the perturbed image $X' = \phi(X)$ has the following characteristics: (a) from a human perspective, the perturbed image X' must look similar to the original input X; (b) the perturbed image X' is most likely to be misclassified by an arbitrary gender classifier $f_G(X)$; (c) the match scores, as assessed by an arbitrary biometric matcher f_M, between perturbed image X' and other unperturbed face images from the same subject, are not impacted thereby retaining verification accuracy.

This goal can be expressed as the following objective function, which minimizes a loss function J consisting of three disjoint terms corresponding to the three characteristics listed above:

$$J(X, y, X'; f_G, f_M) = \lambda_D J_D(X, X') + \lambda_G J_G(y, X'; f_G) + \lambda_M J_M(X, X'; f_M), \quad (1)$$

where, X is the input image, y is the gender label of X, and X' is the perturbed image. The term $J_D(X, X')$ measures the dissimilarity between the input image and the perturbed image produced by a decoder $\phi(X)$ to ensure that the perturbed images still appear as realistic face images. The second term, $J_G(y, X'; f_G)$, measures the loss associated with correctly predicting gender of perturbed image X' using f_G, to ensure that the accuracy of the gender classifier on the perturbed image X' is reduced. The third term, $J_M(X, X'; f_M)$, measures the loss associated with the match score between X and X' computed by f_M. This term ensures that the matching accuracy as assessed by f_M is not substantially diminished due to the perturbations introduced to confound the gender classifier.

In order to optimize this objective function, *i.e.*, minimizing gender classifier accuracy while maximizing the biometric matching accuracy and generating realistic looking images, we design a novel convolutional neural network architecture that we refer to as a semi-adversarial convolutional autoencoder.

2.2. Semi-adversarial network architecture

The semi-adversarial network introduced in this paper is significantly different from Generative Adversarial Networks (GANs). A typical GAN has two components: a discriminator and a generator. The *generator* learns to generate realistic looking images from the training data, while the *discriminator* learns to distinguish between the generated images and the corresponding training data [5, 26]. In contrast to regular GANs consisting of a generator and a single discriminator, the proposed semi-adversarial network attaches two independent classifiers to a generative subnetwork. Unlike the generator subnetwork of GANs that is trained based on the feedback of one classifier, the semi-adversarial configuration proposed in this paper learns to generate image perturbations based on the feedback of two classifiers, where one classifier acts as an adversary of the other. Hence, the semi-adversarial network architecture we propose consists of the following three different subnet-

works (Fig. 1): (a) a trainable generative component in form of a convolutional autoencoder (subnetwork I) for adversarial learning; (b) an auxiliary CNN-based gender classifier (subnetwork II); (c) an auxiliary CNN-based face matcher (subnetwork III).

The auxiliary gender classifier as well as the auxiliary matcher[1] are detachable parts in this network architecture used only during the *training* phase. In contrast to GANs, the generative component of this proposed network architecture is a convolutional autoencoder (section 2.2.1), which is initially pre-trained to produce an image that closely resembles an image from the training set after incorporating gender prototype information (section 2.2.2). Then, during further training, feedback from both an auxiliary CNN-based gender classifier and an auxiliary CNN-based face matcher are incorporated into the loss function (see Eqn. (1)) to perturb the regenerated images such that the error rate of the auxiliary gender classifier increases while that of the auxiliary face matcher is not unduly affected.

An overview of this semi-adversarial architecture is shown in Fig. 1, and the details are further described in the following subsections.

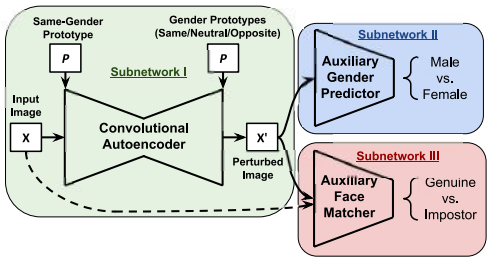

Figure 1. Schematic representation of the semi-adversarial neural network architecture designed to derive perturbations that are able to confound gender classifiers while still allowing biometric matchers to perform well. The overall network consists of three sub-components: a convolutional autoencoder (subnetwork I), an auxiliary gender classifier (subnetwork II), and an auxiliary matcher (subnetwork III).

2.2.1 Convolutional autoencoder

The architecture of the convolutional autoencoder subnetwork that modifies and reconstructs the input image in three different ways is shown in Fig. 2. The input to this sub-network is a gray-scale face image of size 224×224 concatenated with a same-gender prototype, P_{SM} (Fig. 3). The input is then processed through the encoder part consisting of two convolutional layers; each layer is followed by a leaky ReLU activation function and an average pooling layer, resulting in feature maps of size $56 \times 56 \times 12$. Next, the outputs of the encoder are passed through a decoder with two convolutional layers each, followed by a leaky ReLU activation and an upsampling layer using two-dimensional nearest neighbor interpolation. The output of the decoder is a $224 \times 224 \times 128$ dimensional feature map.

The feature maps from the decoder output are then concatenated with either same-gender (P_{SM}), neutral-gender (P_{NT}), or opposite-gender (P_{OP}) prototypes in the *proto-combiner* module (see Fig. 2 and Fig. 3). The proto-combiner module is followed by a final convolutional layer and a sigmoid activation function yielding a reconstructed image X'_{SM}, X'_{NT}, or X'_{OP}, depending on the gender-prototype used. The autoencoder described in this section contains five trainable layers. Those layers are pre-trained using an information bottleneck approach [8] to retain the relevant information from both the original image and the same-gender prototype. This is sufficient to reconstruct realistic looking images by minimizing $J_D(X, X')$, which measures the dissimilarity between the gray-scale input images and the perturbed images by computing the sum of the element-wise cross entropy between input and output (perturbed) images. After pre-training, this subnetwork is further trained by passing its reconstructed images to two other sub-networks: the auxiliary gender predictor and the auxiliary face matcher (Fig. 1). The gender prototypes, as well as the two subnetworks, are described in the following subsections.

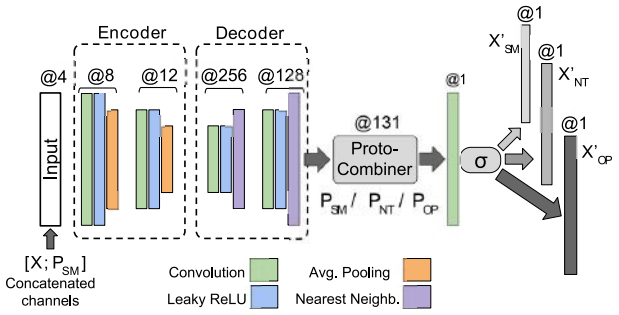

Figure 2. Architecture of the autoencoder augmented with gender-prototype images. The encoder receives a one-channel gray-scale image as input, which is concatenated with the RGB channels of the same-gender prototype image. After the compressed representation is passed through the decoder part of the autoencoder for reconstruction (128 channels), the proto-combiner concatenates it with the RGB channels of a same-, neutral-, or opposite-gender prototype resulting in 131 channels that are then passed to a final convolutional layer.

[1]The term "auxiliary" is used to indicate that these subnetworks do not correspond to pre-trained gender classifiers or face matchers, but rather classifiers that are generated from the training data. Note that such a formulation makes the semi-adversarial network generalizable.

2.2.2 Gender prototypes

The 224×224 male and female RGB gender prototypes (P_{male}, P_{female}) were computed as the average of all $65{,}160$ male images and $92{,}190$ female images, respectively, in the CelebA training set [14]. Then, the same-gender (P_{SM}) and opposite-gender (P_{OP}) prototypes, which are being concatenated with the input image and combined with the autoencoder output (Fig. 2), are constructed based on the ground-truth label y, while the neutral-gender prototype is computed as the weighted mean of male and female prototypes (Fig. 3):

- Same-gender prototype, P_{SM}: $yP_{\text{male}} + (1-y)P_{\text{female}}$
- Opposite-gender prototype, P_{OP}: $(1-y)P_{\text{male}} + yP_{\text{female}}$
- Neutral prototype, P_{NT}: $\alpha_F P_{\text{female}} + \alpha_M P_{\text{male}}$

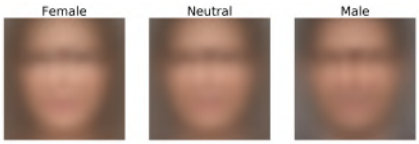

Figure 3. Gender prototypes used to confound gender classifiers while maintaining biometric matching during the semi-adversarial training of the convolutional autoencoder.

Here, α_F is the proportion of females in the CelebA training set and α_M is the proportion of males. The convolutional autoencoder network (summarized in Fig. 1 and further illustrated in Fig. 2) is provided with same-gender prototype images (female or male corresponding to the ground truth label of the input image), which are concatenated with the input image before being transmitted to the encoder module in order to derive a compressed representation of the original image along with the same-gender prototype information. After the decoder reconstructs the original images, the three different gender-prototypes are added as additional channels via the proto-combiner (Fig. 2).

The final convolutional layer of the autoencoder produces three different perturbed images: X'_{SM} (obtained when the same-gender prototype is used), X'_{NT} (when the neutral prototype is used), and X'_{OP} (when the opposite-gender prototype is used).

Pre-training: During pre-training, to ensure that the convolutional autoencoder is capable of reconstructing the original images, only the same gender perturbations (X'_{SM}) were considered in the cross-entropy cost function.

Training: For the further training of the autoencoder, to confound the auxiliary gender classifier and ensure high matching accuracy of the auxiliary matcher, both the perturbed outputs using same- and opposite-gender prototypes were passed through the auxiliary gender classifier, to ensure that the perturbation made using the same-gender prototype produces accurate gender prediction while perturbations made using the opposite-gender prototype confounds the gender prediction. The perturbed outputs due to the neutral prototypes are not incorporated in the loss function, and are only used for evaluation purposes.

2.2.3 Auxiliary CNN-based gender classifier

The architecture of the auxiliary CNN-based gender classifier, which consists of six convolutional layers and two fully connected (FC) layers, is summarized in Fig. 4. Each convolutional layer is followed by a leaky ReLU activation function and a max-pooling layer that reduces the height and width dimensions by a factor of 2, resulting in feature maps of size $4 \times 4 \times 256$. Passing the output of the second FC layer through a sigmoid function results in class-membership probabilities for the two labels: 0:"Female" and 1:"Male". This network was independently trained on the CelebA-train dataset by minimizing the cross-entropy cost function, until its convergence after five epochs; the gender prediction accuracy of the auxiliary network when tested on the CelebA-test set was 96.14%. During training, two dropout layers with drop probability of 0.5 were added to the FC layers for regularization. However, these dropout layers were removed when this subnetwork was used for deriving perturbations as part of the three-subnetwork neural network architecture shown in Fig 1.

As this CNN-based gender classifier was only used for training the convolutional autoencoder for generating perturbed face images, and not for further evaluation of this model, it is referred to as *auxiliary gender classifier* to distinguish it from the gender classifiers used for evaluation.

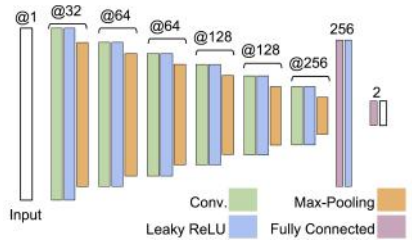

Figure 4. Architecture of the CNN-based auxiliary gender classifier that was used during the training of the convolutional autoencoder. This classifier was used as an auxiliary (fixed) component in the final model to derive the image perturbations according to the objective function described in Section 2.1.

2.2.4 Auxiliary CNN-based face matcher

As discussed in Section 2.1, the loss function contains a term $J_M(X, X'; f_M)$ to ensure good face matching accuracy despite the perturbations introduced to confound the gender classifier. To provide match scores during the training of the autoencoder subnetwork, we used a publicly available VGG model as described by Parkhi *et al*. [22] consisting of 16 weight layers. This VGG subnetwork produces face descriptors which are vector representations of size 2622 extracted from RGB face images. The publicly available weight parameters of this network were used without further performance tuning.

In addition, as the open-source VGG-face network expects RGB images as inputs, we modified the convolutional filters of the first layer by adding the three filter matrices related to the input channels, for compatibility with the single-channel gray-scale input images. As this CNN-based face matcher was only used for training the convolutional autoencoder for generating perturbed face images, and not for further evaluation of this model, it is referred to as *auxiliary matcher* to distinguish it from the commercial matching software used for evaluation.

2.3. Loss function

After pre-training the convolutional autoencoder described in Section 2.2.1, it is connected to the other two subnetworks (the auxiliary CNN-based gender classifier described in Section 2.2.3 and the auxiliary CNN-based face matcher described in Section 2.2.4) for further training. During the pre-training stage, the loss term $J_D(X, X')$ was used to ensure that the convolutional autoencoder is capable of producing images that are similar to the input images. The loss term is computed as the element- or pixel-wise cross entropy, S, between input and output (perturbed) images:

$$J_D(X, X'_{SM}) = \sum_{k=1}^{224^2} S\left(X^{(k)}, X'^{(k)}_{SM}\right). \tag{2}$$

Next, to generate the perturbed images X'_{SM}, X'_{NT}, or X'_{OP} (based on the type of gender-prototype used) such that gender classification is confounded but biometric matching remains accurate, two loss terms, J_G and J_M, were used. The first loss term is associated with suppressing gender information in X'_{OP} and preserving it in X'_{SM}:

$$\begin{gathered} J_G\left(y, X'_{SM}, X'_{OP}; f_G\right) = \\ S\left(y, f_G(X'_{SM})\right) + S\left(1-y, f_G(X'_{OP})\right), \end{gathered} \tag{3}$$

where, $S(t, \hat{p})$ denotes the cross-entropy cost function using target label t and the predicted class-membership probability $\hat{p}$. Note that in this loss function, we use the ground truth labels for X'_{SM} so that the gender of X'_{SM} is correctly predicted, while we use flipped labels for X'_{OP} so that the gender of perturbed image X'_{OP} is incorrectly predicted. We found that without the use of this configuration for X'_{SM} and X'_{OP}, the network will perturb the input image, X, such that perturbations are overfit to the auxiliary gender classifier that is used during training.

The second loss term, J_M, measures the matching similarity between input image X and the perturbed image X'_{SM} generated from the same-gender prototype:

$$J_M(X, X'_{SM}; R_{vgg}) = \left\| R_{vgg}(X'_{SM}) - R_{vgg}(X) \right\|_2^2, \tag{4}$$

where, $R_{vgg}(X)$ indicates the vector representation of image X obtained from the VGG-face network [22]. The total loss is then the weighted sum of the two loss terms J_G and J_M:

$$\begin{gathered} J_{total}\left(X, y, X'_{SM}, X'_{OP}; f_G, R_{vgg}\right) = \\ \lambda_G J_G(y, X'_{SM}, X'_{OP}; f_G) + \lambda_M J_M(X, X'_{SM}; R_{vgg}). \end{gathered} \tag{5}$$

J_{total} was then used to derive the loss gradients with respect to the parameter weights of the convolutional autoencoder during the training stage, to generate perturbations according to the objective function (Section 2.1). Note that the coefficients λ_M and λ_G in Eqn 5 constitute additional tuning parameters to re-weight the contributions of J_G and J_M toward the total loss. In this work, we did not optimize λ_M and λ_G, however, and used a constant of 1 to weight both J_G and J_M equally.

2.4. Datasets

The original dataset source used in this work is the large-scale CelebFaces Attributes (CelebA) dataset [14], which consists of 202,599 face images in JPEG format for which gender attribute labels were already available with the dataset. The dataset was randomly divided into 162,079 training images (CelebA-train) and 40,520 images for testing (CelebA-test). The CelebA-train dataset was used to train the gender classifier (Section 2.2.3), as well as the convolutional autoencoder (Section 2.2.1).

In addition to the CelebA-test dataset, three publicly available datasets were used for evaluation only: MUCT [17], LFW [9] and AR-face [16] databases. The final compositions of these datasets, after applying a preprocessing step using a deformable part model (DPM) as described by Felzenszwalb *et al*. [3] to ensure that all images have the same dimensions (224×224), are summarized in Table 1. The resulting perturbed images obtained from the CelebA-test, MUCT, LFW, and AR-face datasets, were used to measure the effectiveness of modifying the gender attribute as assessed by a commercial gender classifier (G-COTS) and a commercial biometric matcher (M-COTS, excluding AR-images labeled as occluded due to sunglasses or scarfs).

Table 1. Sizes of the datasets used in this study for training and evaluation. CelebA-train was used for training only, while the other four datasets were used to evaluate the final performance of the trained model.

Dataset	Train	# Images	# Male	# Female
CelebA-train	yes	157,350	65,160	92,190
CelebA-test	no	39,411	16,318	23,093
MUCT	no	3754	131	145
LFW	no	12,969	4205	1448
AR-face	no	3286	76	60

2.5. Implementation details and software

The convolutional autoencoder (Section 2.2.1), auxiliary CNN-based gender classifier (Section 2.2.3) and the auxiliary CNN-based face matcher (Section 2.2.4) were implemented in TensorFlow [1] based on custom code for the convolutional layers and freezing the parameters of the gender classifier and face matcher during training of the autoencoder subnetwork [23].

3. Experimental Results

After training the autoencoder network using the CelebA-train dataset as described in Section 2.2.1, the model was used to perturb images in other, independent datasets: CelebA-test, MUCT, LFW, and the AR-face database. For each face image in these datasets, a set of three output images was reconstructed using same-gender, neutral-gender, and opposite-gender prototypes. Furthermore, our results are compared with the face-mixing approach proposed in [21]. Examples of these reconstructed outputs for two female face images, and two male face images are shown in Fig. 5.

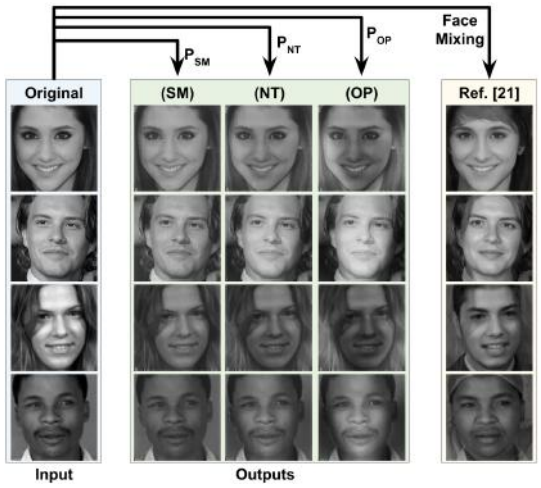

Figure 5. Example input images with their reconstructions using same, neutral, and opposite gender prototypes from the CelebA-test (first two rows) and MUCT (last two rows) datasets.

3.1. Evaluation and verification

The previously described auxiliary CNN-based gender classifier (Section 2.2.3) and auxiliary CNN-based face matcher (Section 2.2.4) were not used for the evaluation of the proposed semi-adversarial autoencoder as these two subnetworks were used to provide semi-adversarial feedback during training. The performance of the semi-adversarial autoencoder is expected to be optimally biased when tested using the auxiliary gender classifier and auxiliary face matcher. Thus, we used independent gender classification and face matching software for evaluation and verification instead, to represent a real-world use case scenario.

Two sets of experiments were conducted to assess the effectiveness of the proposed method. First, two independent software for gender classification were considered: the popular research software IntraFace [13] as well as a state-of-the-art commercial software, which we refer to as *G-COTS*. Second, a state-of-the-art commercial matcher that has shown excellent recognition performance on challenging face datasets was used to evaluate the face matching performance; we refer to this commercial face matching software as *M-COTS*.

3.1.1 Perturbing gender

In order to assess the effectiveness of the proposed scheme in perturbing gender, the reconstructed images using the proposed semi-adversarial autoencoder from the four datasets were analyzed. The Receiver Operating Characteristic (ROC) curves for predicting gender using IntraFace and G-COTS from the original images and the perturbed images are shown in Fig. 6.

We note that gender prediction via IntraFace is heavily impacted when using different gender prototypes for image reconstruction. We observe that the performance of IntraFace on AR-face images after opposite-gender perturbation is very close to random (as indicated by the near-diagonal ROC curve in Fig. 6(a)-(d)). The performance of G-COTS proves to be more robust towards perturbations, compared to IntraFace; however, the ROC curve corresponding to the opposite-gender prototype, shows a substantial deviation from the ROC curve of the original images (Fig. 6(e)-(h)). This observation indicates that the opposite-gender prototype perturbations have a substantial, negative impact on the performance of state-of-the-art G-COTS software, thereby extending gender privacy.

The exact error rates in predicting the gender attribute of face images using both IntraFace and G-COTS software are provided in Table 2 for the original images and the perturbed images using opposite-gender prototypes. The quantitative comparison of the error rates indicates a substantial

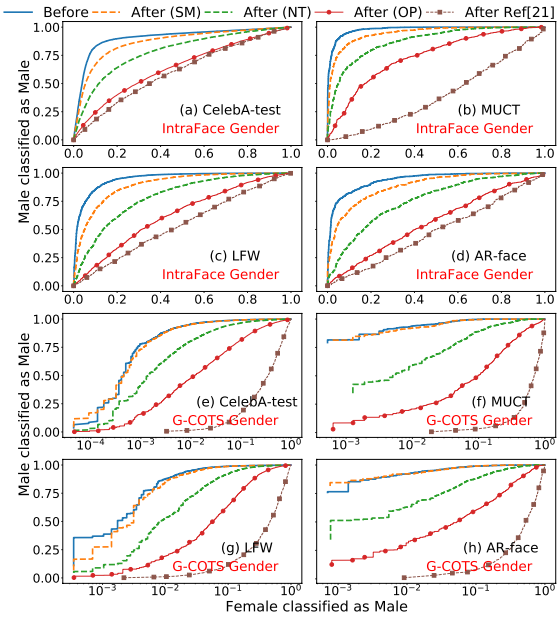

Figure 6. ROC curves comparing the performance of IntraFace (a-d) and G-COTS (e-h) gender classification software on original images ("Before") as well as images perturbed via the convolutional autoencoder model ("After") on four different datasets: CelebA-test, MUCT, LFW, and AR-face.

Table 2. Error rates in gender prediction using IntraFace and G-COTS gender classification softwares on the original datasets before and after perturbation. Note the substantial increase in the prediction error upon perturbation via the convolutional autoencoder model using opposite-gender prototypes.

Software	Dataset	Original (before)	Perturbed (after OP)	Ref. [21]
IntraFace	CelebA-test	19.7%	39.3%	44.6%
	MUCT	8.0%	39.2%	57.7%
	LFW	33.4%	72.5%	70.9%
	AR-face	16.9%	53.8%	54.2%
G-COTS	CelebA-test	2.2%	13.6%	42.4%
	MUCT	5.1%	25.4%	53.9%
	LFW	2.8%	18.8%	46.1%
	AR-face	9.3%	26.9%	40.6%

increase in the prediction error rates when image datasets were perturbed using opposite-gender prototypes. Note that in the case of G-COTS software, perturbations made by the face mixing scheme proposed in [21] result in higher error rates. On the other hand, the additional advantage of our approach is in preserving the identity, as we will see in the next section.

3.2. Retaining matching accuracy

The match scores were computed using a state-of-the-art M-COTS software and the resulting ROC curves are

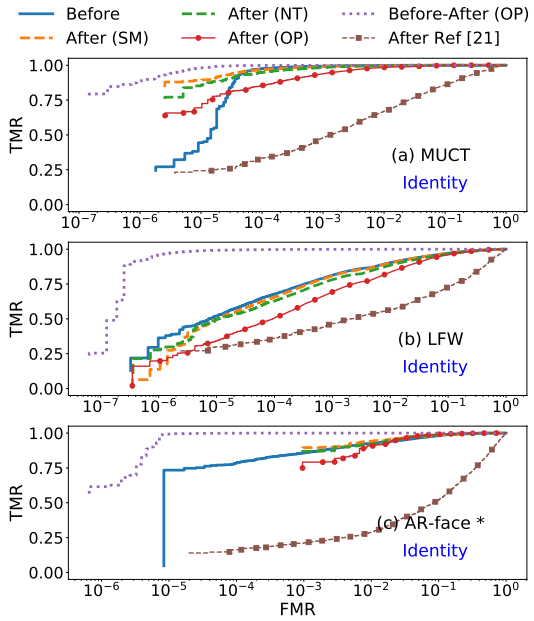

Figure 7. ROC curves showing the performance (true and false matching rates) of M-COTS biometric matching software on the original images ("Before") compared to the perturbed images ("After") generated by the convolutional autoencoder model using same-, neutral-, or opposite-gender prototypes for three different datasets: (a) MUCT, (b) LFW, and (c) AR-face.

shown in Fig. 7. While the matching term, J_M, in the loss function is directly applied to reconstructed outputs from same-gender prototype, X'_{SM}, the reconstructions that use neutral- or opposite-gender prototypes are not directly subject to this loss term (see Section 2.3). As a result, the ROC curve of the reconstructed images coming from same-gender prototype appear much closer to the original input compared to the reconstructed images from neutral- and opposite-gender prototypes. Overall, we were able to retain a good matching performance even when using opposite-gender prototype. On the other hand, the ROC curves obtained from outputs of the mixing approach proposed in [21] are heavily impacted, resulting in de-identified outputs (which is not desirable in this work).

Finally, the True Match Rate (TMR) values at a False Match Rate of 1% are reported in Table 3. The perturbed images from all three datasets show TMR values that are very close to the value obtained from the unperturbed original dataset.

4. Conclusions

In this work, we focused on developing a semi-adversarial network for imparting soft-biometric privacy to face images. In particular, our semi-adversarial network perturbs an input face image such that gender prediction is

Table 3. True (TMR) and false (FMR) matching rates (measured at values of 1%) of the independent, commercial M-COTS matcher after perturbing face images via the convolution autoencoder using same (SM), neutral (NT), and opposite (OP) gender prototypes, indicating that the biometric matching accuracy is not substantially affected by confounding gender predictions.

Dataset	Original (before)	Perturbed (SM)	(NT)	(OP)
MUCT	99.88 %	99.79%	99.57%	98.44%
LFW	90.29%	90.02%	88.47%	83.45%
AR-face	94.97%	94.11%	91.95%	90.81%

confounded while the biometric matching utility is retained. The proposed method uses an auxiliary CNN-based gender classifier and an auxiliary CNN-based face matcher for training the convolutional autoencoder. The trained model is evaluated using two independent gender classifiers and a state-of-the-art commercial face matcher which were unseen during training. Experiments confirm the efficacy of the proposed architecture in imparting gender privacy to face images, while not unduly impacting the face matching accuracy.

5. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Number 1618518.

References

[1] M. Abadi et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] A. Dantcheva, P. Elia, and A. Ross. What else does your biometric data reveal? A survey on soft biometrics. *IEEE Transactions on Information Forensics and Security*, 11(3):441–467, 2016.

[3] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Cascade object detection with deformable part models. In *Computer vision and pattern recognition (CVPR)*, pages 2241–2248.

[4] C. Garvie. *The Perpetual Line-Up: Unregulated Police Face Recognition in America*. Georgetown Law, Center on Privacy & Technology, 2016.

[5] I. Goodfellow et al. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[6] R. Gross et al. Integrating utility into face de-identification. In *International Workshop on Privacy Enhancing Technologies*, pages 227–242. Springer.

[7] R. Gross et al. Model-based face de-identification. In *Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2006.

[8] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[9] G. Huang et al. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[10] A. Jain, A. A. Ross, and K. Nandakumar. *Introduction to biometrics*. Springer Science & Business Media, 2011.

[11] A. Jourabloo, X. Yin, and X. Liu. Attribute preserved face de-identification. In *International Conference on Biometrics (ICB)*, pages 278–285, 2015.

[12] E. J. Kindt. *Privacy and data protection issues of biometric applications*. Springer, 2013.

[13] D. la Torre et al. Intraface. In *11th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–8, 2015.

[14] Z. Liu et al. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015.

[15] A. Madry et al. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[16] A. Martinez and R. Benavente. AR face database, 2000. *http://www2.ece.ohio-state.edu/ aleix/ARdatabase.html*.

[17] S. Milborrow, J. Morkel, and F. Nicolls. The MUCT landmarked face database. *PRASA*, 2010.

[18] V. Mirjalili and A. Ross. Soft biometric privacy: Retaining biometric utility of face images while perturbing gender. In *Proc. of International Joint Conference on Biometrics (IJCB)*, 2017.

[19] I. Natgunanathan et al. Protection of privacy in biometric data. *IEEE Access*, 4:880–892, 2016.

[20] E. M. Newton, L. Sweeney, and B. Malin. Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):232–243, 2005.

[21] A. Othman and A. Ross. Privacy of facial soft biometrics: Suppressing gender but retaining identity. In *European Conference on Computer Vision Workshop*, pages 682–696. Springer, 2014.

[22] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conference*, volume 1, page 6, 2015.

[23] S. Raschka and V. Mirjalili. *Python Machine Learning, 2nd Ed.* Packt Publishing, Birmingham, UK, 2017.

[24] N. K. Ratha, J. H. Connell, and R. M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40(3):614–634, 2001.

[25] D. A. Rowland and D. I. Perrett. Manipulating facial appearance through shape and color. *IEEE Computer Graphics and Applications*, 15(5):70–76, 1995.

[26] A. Rozsa et al. Are facial attributes adversarially robust? *arXiv preprint arXiv:1605.05411*, 2016.

[27] T. Sim and L. Zhang. Controllable face privacy. In *11th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, volume 4, pages 1–8, 2015.

[28] Y. Sun et al. Demographic analysis from biometric data: Achievements, challenges, and new frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[29] J. Suo et al. High-resolution face fusion for gender conversion. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(2):226–237, 2011.