

# Perceptual Indistinguishability-Net (PI-Net): Facial Image Obfuscation with Manipulable Semantics

Jia-Wei Chen<sup>1,3</sup> Li-Ju Chen<sup>3</sup> Chia-Mu Yu<sup>2</sup> Chun-Shien Lu<sup>1,3</sup>

<sup>1</sup>Institute of Information Science, Academia Sinica <sup>2</sup>National Yang Ming Chiao Tung University

<sup>3</sup>Research Center for Information Technology Innovation, Academia Sinica

## Abstract

*With the growing use of camera devices, the industry has many image datasets that provide more opportunities for collaboration between the machine learning community and industry. However, the sensitive information in the datasets discourages data owners from releasing these datasets. Despite recent research devoted to removing sensitive information from images, they provide neither meaningful privacy-utility trade-off nor provable privacy guarantees. In this study, with the consideration of the perceptual similarity, we propose perceptual indistinguishability (PI) as a formal privacy notion particularly for images. We also propose PI-Net, a privacy-preserving mechanism that achieves image obfuscation with PI guarantee. Our study shows that PI-Net achieves significantly better privacy utility trade-off through public image data.*

## 1. Introduction

More and more facial image datasets are becoming available in a wide variety of communities. The availability of these datasets presents enormous opportunities for collaboration between data owners and the machine learning community (e.g., social relation recognition [43]). However, the inherent privacy risk of such datasets prevents data owners from sharing the data. For example, Microsoft’s facial image dataset MS-Celeb-1M, Duke’s MTMC, and Stanford’s Brainwash were taken down due to the potential privacy concern [30, 36].

**Facial Image Obfuscation.** Many facial image obfuscation (aka. face anonymization and face de-identification) solutions have been proposed. An approach to mitigating privacy risk of releasing facial images is to use Generative Adversarial Networks (GANs) to synthesize visually similar images [7, 25]. Recently, GANs that can manipulate semantics have been proposed to even have a fine-grained control of attributes such as age and gender [17, 37, 50]. Based on inpainting [28, 42, 52], one can also anonymize the image, while retaining the facial semantics, by removing the region of interest (ROI) with sensitive semantics in the image and restoring it. Image forgery methods such as

Deepfakes [45], if used to replace sensitive semantics, can also mitigate privacy risks for identity disclosure [3, 15].

However, all of the above methods share a common weakness of syntactic anonymity, or say, lack of formal privacy guarantee. Recent studies report that obfuscated faces can be re-identified through machine learning techniques [33, 19, 35]. Even worse, the above methods are not guaranteed to reach the analytical conclusions consistent with the one derived from original images, after manipulating semantics. To overcome the above two weaknesses, one might resort to differential privacy (DP) [9], a rigorous privacy notion with utility preservation. In particular, DP-GANs [1, 6, 23, 46] shows a promising solution for both the provable privacy and perceptual similarity of synthetic images. Unfortunately, DP-GANs can only be shallow, because of its rapid noise accumulation, hindering model accuracy. One can also apply DP to the image, leading to pixelation [10]. Consequently, the images generated in such a way are of low quality.

**Key Insights.** Basically, anonymizing facial images, while retaining necessary information for tasks such as detection, recognition, and tracking is very challenging. Notably, our result is in possession of the following novelties.

First, based on metric privacy [5], we introduce perceptual indistinguishability (PI), a variant of DP with a particular consideration of perceptual similarity of the facial images. More specifically, PI, while retaining perceptual similarity, achieves the indistinguishability result that an adversary, when seeing an anonymized image, can hardly infer the original image, thereby protecting the privacy of the image content. On the other hand, inherited from DP, PI can also ensure high data utility (detection, classification, tracking, etc.). As far as we know, this is the first time perceptual similarity or more concretely, facial attributes, is used to define an indistinguishability notion from the image adjacency point of view in the context of DP, which enables the reconciliation among privacy and utility. Although facial attributes have been also exploited for face de-identification [27, 51], in addition to relying on different privacy notions, our study is different from theirs in that (1) [27] needs pre-processing for face alignment and cropping and (2) [51] learns privacy representation instead of releasing private image dataset.

Second, we introduce PI-Net, a novel encoder-decoder architecture to achieve image obfuscation with PI. PI-Net is featured by its operations in latent space and can also be seen as a latent-coded autoencoder. In particular, we inject noise to the latent code derived from GAN inversion. PI-Net is also featured by the use of triplet loss that clusters the faces with similar facial attributes. This novel architecture enables the network to create anonymized images that look realistic and satisfy user-defined facial attributes.

In summary, we make the following contributions:

- We present a notion for image privacy, perceptual indistinguishability (PI), defining adjacent images by the perceptual similarity between images in latent space.
- We propose PI-Net to anonymize faces with the selected semantic attributes manipulation. The architecture of PI-Net generates realistic looking faces with the selected attributes preservation.

## 2. Related work

**Face Anonymization.** Face anonymization has been studied extensively. Straightforward methods such as pixelization, blurring, and masking obviously harm the visual quality. Through the adversarial learning that uses the generator as an anonymizer to modify sensitive information and the discriminator as a facial identifier, the trained generator can synthesize high-quality facial images with different identities [38]. On the other hand, many GAN-based face anonymization methods were proposed. For example, the facial attributes of an image are modified via GANs such that the distribution of facial attributes matches the desired  $t$ -closeness in Anonymousnet [26, 27]. One can generate faces via GAN such that the probability of a particular face being re-identified is at most  $1/k$  in  $k$ -same family of algorithms [14, 24, 34]. DeepPrivacy [20] and CIAGAN [29] are conditional GANs (CGANs), generating anonymized images. The former is based on the surrounding of the face and sparse pose information, while the latter relies on an identity control discriminator. Gafni et al.’s proposal [13] is an adversarial autoencoder, coupled with a trained face-classifier. However, their anonymized images, while successfully fooling the recognition system, are unlikely to hide the identity of the presented faces.

**Differentially Private Machine Learning.** Model inversion attack [12] aims to reconstruct the training dataset by accessing or querying a model. Defenses against model inversion include DP and adversarial learning [51]. Here, we pay our attention only to DP-GAN [1, 6, 23, 46], as it can generate synthetic images with reasonable data utility. Since only the discriminator has access to the training data, one cannot distinguish whether an individual training data participated in training after injecting DP noise into the gradients. Moreover, variants of DP (e.g., Renyi-DP [31]) and different network structures (e.g., [46]) can be employed to generate images with higher quality. The 1-Lipschitz property of the Wasserstein distance is also helpful in determin-

ing clipping bounds in gradient sanitization, further improving image quality [6]. Unfortunately, DP-GANs use standard DP [9] to define image adjacency and only cares the accuracy of the model fed by synthetic images. Therefore, one can only expect low perceptual similarity between the original and synthetic images, not to speak of the attribute preservation.

**Differentially Private Multimedia Analytics.** There is little attention to multimedia analytics with DP. Until recently, the first study of defining image adjacency was proposed [10]. More specifically, by treating an image as a database, and by treating the features within the image (e.g., pixels) as database records, PixDP [10] defines image adjacency with standard DP. DP noise injection in the pixel domain obviously brings catastrophic consequence on image quality, and so the follow-up method [11] defines image adjacency based on the singular vectors of the SVD transformed images. SVD-Priv [11] slightly improves the quality of anonymized images. In addition to images, DP can also be applied to video in that video adjacency is defined according to whether a sensitive visual element appears in the video [48]. Unfortunately, as video adjacency is defined in pixel domain, the video quality will be largely destroyed.

## 3. Preliminary

DP [9] provides a provable guarantee of privacy that can be quantified and analyzed, and the adjacency is a critical concept that defines the information to be hidden. Unfortunately, applying DP, together with Laplace noise [9], to images leads to unacceptable quality. In our work, inspired by metric privacy [5], we define image adjacency as the difference in the high-level features learned by GANs; *i.e.*, the latent codes, used to capture changes in image semantics. In other words, we attempt to find the best-suited latent code in the latent space for a given image. Then, we give a clear semantic definition of the latent space so that the semantics of the image can be captured for privacy analysis. Below we present formal definitions of the above terminologies that will be used throughout this paper. The notations frequently used in this paper are described in in Notation Table in the Appendix.

**Definition 3.1. (Semantic Space [40])** Suppose a semantic scoring function defined as  $f_S : \mathcal{I} \rightarrow \mathcal{S}$  can evaluate each image’s facial attribute components, such as old age, smiling, where  $\mathcal{S} \subset \mathbb{R}^m$  is called the semantic space formed by  $m$  facial attributes (In our case, for a semantic label  $s \in \mathcal{S}$ , it is a binary vector with each entry indicating whether an attribute exists or not), and  $\mathcal{I}$  is the image space. Moreover, given a well-trained GAN model, we denote a generator as  $G : \mathcal{X} \rightarrow \mathcal{I}$ , such that the latent space  $\mathcal{X} \subset \mathbb{R}^d$  can be bridged with the semantic space  $\mathcal{S}$  through  $s = f_S(G(x))$ , and so the semantic of the latent code  $x$  can be evaluated.

**Definition 3.2. (GAN Inversion [2])** Given a trained GAN model, the goal of GAN inversion is to find the most accurate latent code  $x' \in \mathcal{X}$  to recover the input image  $I \in \mathcal{I}$ .

In our case, an optimization-based method is used to minimize the reconstruction error of the features between the generated image and the given image by directly optimizing the latent code  $x'$ . We use  $F$  to denote the optimization of this inversion:  $F : \mathcal{I} \rightarrow \min_{x'} \mathcal{L} = \|\mathcal{I} - G(x')\|_2 + \alpha \cdot \|V(\mathcal{I}) - V(G(x'))\|_2$  where  $V$  is the VGG16 [41].

**Definition 3.3. ( $d_{\mathcal{X}}$ -Privacy [5])** A randomized mechanism  $K : \mathcal{X} \rightarrow \mathcal{P}(\mathcal{Z})$  satisfies  $d_{\mathcal{X}}$ -privacy, iff  $\forall x, x' \in \mathcal{X}$ :

$$K(x)(Z) \leq e^{d_{\mathcal{X}}(x,x')} K(x')(Z), \quad \forall Z \in \mathcal{F}_{\mathcal{Z}}, \quad (1)$$

where  $d_{\mathcal{X}}$  is a distance metric for  $\mathcal{X}$ ,  $\mathcal{Z} \subseteq \mathcal{X}$  is a set of query outcomes over  $\mathcal{X}$ ,  $\mathcal{F}_{\mathcal{Z}}$  is a  $\sigma$ -algebra over  $\mathcal{Z}$ , and  $\mathcal{P}(\mathcal{Z})$  is the set of probability measures over  $\mathcal{Z}$ .

Here, metric privacy [5] generalizes DP to protect secrets in an arbitrary domain  $\mathcal{X}$ . Essentially, based on a distance metric  $d_{\mathcal{X}}$ ,  $K$  ensures a level of indistinguishability proportional to the distance. An adversary observing the outputs (e.g.,  $Z$ ) can hardly infer the exact input.

**Definition 3.4. ( $\Delta$ -Sensitivity [5])** A deterministic function  $f : \mathcal{X} \rightarrow \mathcal{Y}$  is  $\Delta$ -sensitive with respect to  $d_x$  and  $d_y$  iff  $d_y(f(x), f(x')) \leq \Delta d_x(x, x')$  for all  $x, x' \in \mathcal{X}$ . The smallest  $\Delta$  is called the sensitivity of  $f$  with respect to  $d_x$  and  $d_y$ .

**Definition 3.5. ( $\Delta d_{\mathcal{X}}$ -Privacy [5])** Assume that  $f$  is  $\Delta$ -sensitive with respect to  $d_x$  and  $d_y$ , and the mechanism  $H$  satisfies  $d_{\mathcal{Y}}$ -privacy. Then, the mechanism  $M : \mathcal{X} \rightarrow \mathcal{P}(\mathcal{Z})$  defined as  $M(x) = (H \circ f)(x)$  satisfies  $\Delta d_{\mathcal{X}}$ -privacy, where  $\circ$  is the composition operator.

## 4. Our Solution: PI-Net

In this section, we detail our image obfuscation mechanism based on perceptual distance. First, we formalize our proposed privacy notion, perceptual indistinguishability (PI). Then, we describe our proposed semantic transformation for manipulating the chosen facial attributes. Finally, we illustrate the mechanism used to obfuscate the identifiable information on the face.

### 4.1. Privacy Formulation

Standard DP [9] has been a de facto standard for data privacy, and widely used to protect the privacy of tabular (non-multimedia) data. Nevertheless, tabular data (composed of records) intrinsically are different from multimedia data (composed of low-level features like pixels or composed of high-level features like semantic features). Thus, standard DP is not applicable to privacy protection of multimedia, and a privacy notion for multimedia is desirable.

In an image, there are many facial attributes that can be captured by latent codes learned from GANs. For these latent codes that can correctly represent attributes, an interpretation is that they correctly learn the semantic information embedded in the image and have the corresponding semantic scores. We are particularly interested in semantic information that can also be used to represent an individual's

identity. Studies in semantic editing [2, 40] have concluded that the semantics captured in the latent code are related to individual identity and can be used to change the identity of the individual visually. Therefore, one can expect a properly-designed distance function that can measure the dissimilarity between latent codes, and use it to protect an individual identity.

**Definition 4.1. (Perceptual Distance)** Given a well-trained GAN model, the Euclidean distance  $\rho(x, x')$  between two latent codes  $x$  and  $x'$  is called the perceptual distance, where  $x, x' \in \mathcal{X}$ , and  $\mathcal{X}$  is latent space learned in GANs.

Perceptual distance is a metric for measuring perceptual similarity between images. Studies [13, 22, 51] have proposed a variety of definitions for perceptual distance, with an attempt to improve image quality or privacy protection. Despite no standardized definition for perceptual distance, they all can be seen as a distance between high-level features learned from neural networks. In our work, we adopt Euclidean distance between latent codes learned from GANs model as the perceptual distance. There are two reasons behind such a design choice. The first comes with the constraint that the distance in metric privacy is required to satisfy the triangular inequality. Second, as the latent space of GAN models has arithmetic properties [40], our use of Euclidean distances, together with a proper noise injection, suffices to alter individual identities.

In the following, we use the  $d_{\mathcal{X}}$ -privacy in Definition 3.3 to formulate a notion of privacy based on perceptual distance. In our consideration, privacy is proportional to semantic information; *i.e.*, images that are semantically similar are more indistinguishable from each other, and thus have a higher probability of generating the same obfuscated output. Following the above principle, we define the image adjacency and privacy notion below.

**Definition 4.2. (Adjacent images)** Two facial images are adjacent, if the perceptual distance between the corresponding latent codes  $\rho(x, x') \leq \beta$ . As  $\beta$  goes smaller, two faces will have more similar facial attributes, while the identities will also be similar.

**Definition 4.3. ( $\epsilon$ -Perceptual Indistinguishability)** A randomized mechanism  $K$  satisfies  $\epsilon$ -Perceptual Indistinguishability ( $\epsilon$ -PI) if  $\forall x, x' \in \mathcal{X}$  that are adjacent :

$$\ln \left| \frac{K(x)(Z)}{K(x')(Z)} \right| \leq \epsilon \rho(x, x'), \quad (2)$$

where  $K(x)(Z)$  is the probability that the obfuscated latent code belongs to the set  $Z \subseteq \mathcal{X}$  when the original latent code is  $x$ , and  $K(x)$  is the probability distribution over  $\mathcal{X}$ .

An interpretation of Definition 4.3 is that the closer the perceptual distance between the latent codes (the more similar the semantic information measured), the closer the probability of producing the same obfuscated output, thus making it more difficult for an adversary to distinguish between true codes.

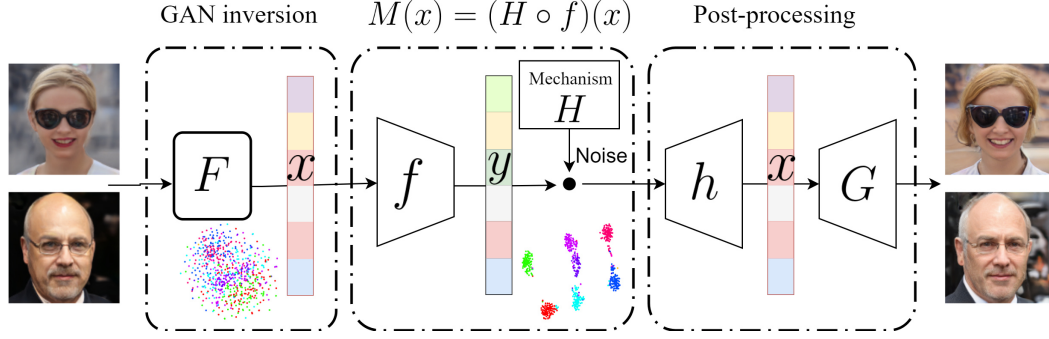


Figure 1. PI-Net: (Left dash-dot block) GAN inversion, (Middle dash-dot block) Encoding network, and (Right dash-dot block) Decoding network and GAN. The mechanism  $H$  is not required during training. The  $\mathcal{X}$  space and  $\mathcal{Y}$  space are visualized with  $t$ -SNE [47].

## 4.2. Training PI-Net

An ideal facial image obfuscation would obfuscate the latent code’s identifiable information without interfering with other facial attributes. Nevertheless, studies [2, 40] have observed that facial attributes are entangled in the latent codes learned from GANs. Consequently, adding noise directly to the latent codes will also interfere with the facial attributes we desire to preserve, resulting in the poor utility of the obfuscated images. For example, the two attributes of “old age” and “wearing glasses” are entangled in the latent code. When we desire to preserve the “wearing glasses” attribute, but once the noise is added to the “old age”, the “wearing glasses” attribute is changed by the unwanted interference.

**Transform.** We propose a transformation framework that can cluster the attribute information we desire to preserve from the entangled latent space, allowing the mechanism to determine the strength of the obfuscation without interference with other attributes for better data utility.

The overall architecture is shown in Figure 1, and can be separated into three components: (1) GAN inversion, (2) Encoding network, and (3) Decoding network and GAN. The operation path is as follows: First, the image  $I \in \mathcal{I}$  is inverted to an appropriate code  $x$  in the latent space  $\mathcal{X}$  by Definition 3.2. The next step is to cluster the codes with the same specified facial attributes through the encoding network  $f$ , which serves as a transformation. In the last step, the transformed code in the  $\mathcal{Y}$  space is mapped back to a latent space  $\mathcal{X}$  using the decoding network  $h$ , and can then be handed over to the generator  $G$  for image synthesis. Note that mechanism  $H$  presented in Figure 1 is not required in the training procedure and will be described in Section 4.3.

The key idea is that, given a semantic defined in Definition 3.1 and the specified attributes to be preserved, the transformed codes  $y = f(x)$  with the same specified attributes should be close to each other, and the other transformed codes should be far from each other, as illustrated in Figure 1. As the transformed codes have been clustered by the specified attributes, each transformed code is surrounded by other codes with the same specified attributes. As a consequence, adding noise from the distribu-

tion  $(H \circ f)(x)$  results in a code that preserves the same attributes but changes its identity.

**Triplet Training.** When training the encoding network  $f$ , we use the triplet loss shown in Equation 3 as:

$$L_{triplet}(x, \theta) = \sum_{i=1}^N \|f_{\theta}(x_i^a) - f_{\theta}(x_i^p)\|_2^2 - \|f_{\theta}(x_i^a) - f_{\theta}(x_i^n)\|_2^2 + \mu. \quad (3)$$

In the triplet loss training, given a semantic formed by  $m$  attributes, a triplet  $(x_i^a, x_i^p, x_i^n)$  contains two latent codes of the same specified attributes, called anchor  $x_i^a$  and positive sample  $x_i^p$ , and a third latent code must be from different attributes, called negative sample  $x_i^n$ . The triplet loss function requires the anchor point  $x_i^a$  to be closer to the positive sample  $x_i^p$  than the negative sample  $x_i^n$ , and  $\mu$  is a threshold that encourages the negative sample to be further away from the anchor point than the positive sample.

Then, we need to map the code in the space  $\mathcal{Y}$  back to the latent space  $\mathcal{X}$  as close as possible, allowing the generator to synthesize similar images. Thus, we train a decoding network  $h_{\omega}$  by using Equation 4 as:

$$L_{recon}(x, \omega) = \sum_{i=1}^N \|x_i - h_{\omega}(f_{\theta}(x_i))\|_2, \quad (4)$$

which requires that the transformed code  $h_{\omega}(f_{\theta}(x_i))$  must be similar to the original code  $x$ .

Finally, since triplet loss only considers the clustering of codes with the same specified attributes, irrelevant to classification results. Therefore, we additionally penalize misclassification by the cross entropy loss that is defined as:

$$L_{ce}(x, \theta, \omega) = - \sum_{i=1}^N s_i \log(f_A(h_{\omega}(f_{\theta}(x_i))))), \quad (5)$$

where  $f_A$  is the trained attribute classification model, and  $s_i \in \mathbb{R}^m$  is the corresponding correct semantic, which is a vector formed by  $m$  binary attributes.

The above neural network training procedures are summarized in Algorithm 1.

---

**Algorithm 1** Training PI-Net

---

**Input:**

Images:  $\mathcal{I} = \{I_1, \dots, I_N\}$   
Semantic space:  $\mathcal{S} = \{s_1, \dots, s_N\}$ , where  $s_i \in \mathbb{R}^m$   
Learning rate:  $\alpha, \beta$ .

**Output:**

Model weights:  $\theta, \omega$

```
1:  $\mathcal{D} = \{(I_1, s_1), \dots, (I_N, s_N)\}$ 
2: for  $I_i$  in  $\mathcal{I}$  do
3:   Invert the image  $I_i$  to get the latent code  $x_i$ 
     using  $x_i = F(I_i)$  and  $F$  in Definition 3.2
4: end for
5: while not done do
6:   Shuffle  $\{(x_1, s_1), \dots, (x_N, s_N)\} \subseteq \mathcal{D}$ 
7:   Split the shuffled dataset  $D$  to get batches  $B_{1, \dots, m}$ 
8:   for  $B_i$  in  $B_{1, \dots, m}$  do
9:      $b_i = |B_i|$ 
10:    Get  $b_i$  datapoints  $(x_1, s_1), \dots, (x_{b_i}, s_{b_i}) \in B_i$ 
11:    Evaluate  $\nabla_{\theta} L_{\text{triplet}}$  and  $\nabla_{\theta} L_{\text{ce}}$  using  $B_i$ 
     and  $L_{\text{triplet}}, L_{\text{ce}}$  in Equations 3 and 5
12:    Update  $\theta \leftarrow \theta - \alpha(\nabla_{\theta} L_{\text{triplet}} + \nabla_{\theta} L_{\text{ce}})$ 
13:    Evaluate  $\nabla_{\omega} L_{\text{recon}}$  using  $B_i$ 
     and  $L_{\text{recon}}$  in Equation 4
14:    Update  $\omega \leftarrow \omega - \beta \nabla_{\omega} L_{\text{recon}}$ 
15:   end for
16: end while
```

---

### 4.3. Image Obfuscation via PI-Net

As training PI-Net does not involve noise injection mechanism  $H$ , this section clarifies three issues during image obfuscation: (1) How to privatize  $f$ ? (2) How to guarantee PI? and (3) The design of noise injection mechanism  $H$ .

**How to Privatize  $f$ .** As shown in Figure 1, when the data owner wants to share the images by a trained PI-Net, the latent codes  $x$  are transformed through the encoding network  $f : \mathcal{X} \rightarrow \mathcal{Y}$  for high data utility. To obtain a transformed code from  $f$ , one can define  $M(x) = (H \circ f)(x)$ , where  $M : \mathcal{X} \rightarrow \mathcal{P}(\mathcal{Y})$  and  $H : \mathcal{Y} \rightarrow \mathcal{P}(\mathcal{Y})$ . Here, the noise injection mechanism  $H$  will be described later in Equation 6, which assigns a noise sampling from a particular distribution to the transformed code  $y$ , so as to satisfy  $d_{\mathcal{Y}}$ -privacy (see more details later). The  $d_{\mathcal{Y}}$ -privacy, whose interpretation of  $d_{\mathcal{Y}}$ -privacy is the same as Definition 3.3 but on a different metric space, ensures that the level of privacy provided by  $H$  is proportional to the metric  $d_{\mathcal{Y}}$  on the  $\mathcal{Y}$  space. Note that in the framework of metric privacy, we can express the privacy of the “mechanism” itself in its own space, without the need to consider the deterministic function or  $\Delta$ -sensitivity.

**How to Guarantee PI.** When  $H$  satisfies  $d_{\mathcal{Y}}$ -privacy,  $M$  can be proven to achieve  $\Delta d_{\mathcal{X}}$ -privacy through  $\Delta$ -sensitivity, where  $\mathcal{X}$  denotes the space of defining the perceptual distance, according to Definition 3.5. The calcula-

tion of  $\Delta$ -sensitivity is shown below. Based on Definition 3.5,  $\Delta$ -sensitivity, and  $d_{\mathcal{Y}}$ -privacy, we can prove that the mechanism  $M$  satisfies  $\epsilon$ -PI in Definition 4.3. The detailed proof is omitted here (Please see Appendix.).

In particular, the  $\Delta$ -sensitivity in Definition 3.4 is also known as the Lipschitz continuity. Although the neural network can add constraints such as spectral normalization or gradient penalty in the training phase to achieve Lipschitz continuity [16, 32], these methods are not applicable to our case, where each cluster requires its own  $\Delta$ -sensitivity. More specifically, as the encoding network  $f$  will generate clusters according to the number of specified attributes, we need to calculate each cluster’s  $\Delta$ -sensitivity separately. Assume there are  $J$  clusters in  $\mathcal{Y}$  space. In order to bound the sensitivity of encoding network  $f$  on code  $x_j$  that belongs to  $j$ -th cluster,  $j = 1, \dots, J$ , we clip the vector  $\mathbf{g}_j$  by  $\mathbf{g}_j / \max(1, \|\mathbf{g}_j\|/C_j)$  to ensure  $\|\mathbf{g}_j\|_2 \leq C_j$ , where  $\mathbf{g}_j = f(x_j) - f(x'_j) \in \mathbb{R}^k$ , and  $C_j = \Delta \|x_j - x'_j\|_2 \in \mathbb{R}$  is the perceptual distance. This clipping step is also used in DP-GANs [1, 6, 46].

**Generating Noise in  $H$ .** Our mechanism  $H$  in Figure 1 follows the method [11] proposed for sampling on the hypersphere. In a  $k$ -dimensional space  $\mathcal{Y}$ , the mechanism  $H = D_{\epsilon, k}$  samples  $y$  given  $y_0$  according to the probability density function defined as:

$$D_{\epsilon, k}(\mathbf{y}_0)(\mathbf{y}) = C_{\epsilon, k} e^{-\epsilon \cdot d_{\mathcal{Y}}(\mathbf{y}_0, \mathbf{y})}, \quad (6)$$

where  $d_{\mathcal{Y}}$  is the  $k$ -dimensional Euclidean distance, and

$$C_{\epsilon, k} = \frac{1}{2} \left( \frac{\epsilon}{\sqrt{\pi}} \right)^k \frac{\left( \frac{k}{2} - 1 \right)!}{(k-1)!}. \quad (7)$$

Equation 6 is a variant of the multivariate Laplace mechanism that satisfies  $d_{\mathcal{Y}}$ -privacy, which means that the level of indistinguishability provided by the mechanism  $H$  is proportional to the  $k$ -dimensional Euclidean distance. The same way to measure privacy using metrics can also be seen in the Definition 3.3, and the mechanism  $H \circ f$  satisfies  $\epsilon$ -PI in Definition 4.3, as we prove in the Appendix.

Sampling according to Equation 6 can be achieved as follows. We first convert the Cartesian coordinates of  $y$  to the hyper-spherical coordinate system with  $y_0$  at the origin, resulting in 1 radial coordinate and  $k - 1$  angular coordinates. Then, combined with the formula for  $n$ -sphere surface area, we can deduce that the marginal probability of radial is the gamma-beta distribution  $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$ , where the shape  $\alpha$  corresponds to dimension  $k$ , the ratio  $\beta$  corresponds to  $\epsilon$ , and the variable  $x$  corresponds to the radial  $r$ . Two steps are taken next: (1) Sampling radial coordinates according to marginal distributions, *i.e.*, gamma-beta distribution. (2) Uniformly sample a point on the unit  $(k - 1)$ -sphere. Since the radial and angle are independent of each other, multiplying the results of (1) and (2) gives the sampled noise.

## 5. Experiments

In this section, we first describe the experimental settings. After that, we visually compare the obfuscated images between PI-Net and other DP-based methods. At the end, we perform the evaluation of PI-Net in terms of image quality, attribute re-classification accuracy, and trade-off between privacy and utility.

### 5.1. Experimental Setting

Here, we describe the proposed network model of our method and the datasets used in the experiments.

**Network Architecture.** Our proposed PI-Net is composed of three components, including (1) GAN inversion, (2) autoencoder, and (3) generator in trained GANs, as illustrated in Figure 1. The GAN inversion has shown in Definition 3.2, and the autoencoder is composed of the fully connected layers and the ReLU activation function. As PI-Net is independent of specific GAN models, we have freedom to employ arbitrary GANs as the generator. In our experiment, we used the generator of StyleGAN [25] in PI-Net, and set StyleGAN’s  $W$  space as PI-Net’s latent space  $\mathcal{X}$ . The reason behind such a design choice is that StyleGAN has been studied to be effective in capturing high-level features such as age, smiling and other facial attributes [2, 40]. Moreover, the dataset used for training our PI-Net is also the synthetic images that generated by StyleGAN. The following describes the dataset used in our evaluations.

**Training Dataset.** Based on a pre-trained StyleGAN model, we randomly sampled and synthesized 50K images from the latent space. There are two reasons of adopting this way to generate synthetic dataset: (1) To collect sufficient samples of specific facial attributes to avoid bias from the unbalanced dataset, and (2) To avoid the use of GAN inversion for latent code calculation of real images, saving a considerable amount of time in building the dataset. In our training dataset generation, since the randomly selected latent codes do not have face attribute labels, we used the CelebA [53] to train the attribute prediction model based on ResNet-50 [21] to assign attribute labels to each sampled latent code. Meanwhile, to avoid assigning false face labels, we added only the selected samples with sufficiently high confidence levels to the training dataset.

**Testing Dataset.** We used CelebA to evaluate the performance of our trained PI-Net. CelebA has facial attribute data from more than 200K celebrity images, with different celebrity identities labeled in the dataset, and each celebrity image has 40 binary attributes. It has been widely used for different computer vision tasks, including face detection, face recognition, and attribute prediction. In our experiments, we used the celebrity identities provided by the dataset and the corresponding binary attributes to evaluate our privacy protection and data utility.

### 5.2. Visualization

We show the visualization results obtained from PI-Net and compare our results with the ones derived from PixDP

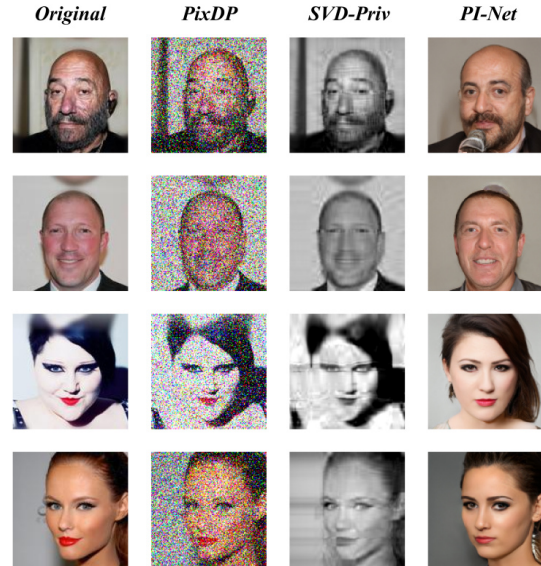


Figure 2. Example images and the corresponding obfuscated results. From left to right, the first column represents original images. PixDP [10] (2nd column), SVD-Priv [11] (3rd column), and PI-Net (4th column) obfuscate low-level feature (pixels), mid-level feature (geometric structures), and high-level feature (image semantics), respectively. To better observe the shape of faces, we actually used less noises on PixDP and SVD-Priv than PI-Net.

[10] and SVD-Priv [11]. We particularly note that only PixDP and SVD-Priv were chosen for comparison because they are the solutions that provide formal privacy. As can be seen from Figure 2, PI-Net generates obfuscated images that have much better realistic-looking.

The visual difference among PI-Net, PixDP, and SVD-Priv can be attributed to three factors. (1) PixDP defines the indistinguishability of identifiable features on the pixel space in a straightforward manner. Consequently, after adding DP noise, the image will be mostly pixelated, resulting in pessimistic image quality. (2) SVD-Priv defines the indistinguishability on the singular value space. Since the singular value of an image determines its geometric structure, obfuscating the singular value will interfere with the overall geometry structure of the image. (3) PI-Net defines the indistinguishability on the high-level feature space, learned from neural networks. As high-level features can capture abstract features such as image semantics, compared to the mid-level and lower-level features such as geometric structures and pixels, image quality will not be dramatically degraded after obfuscation.

### 5.3. Evaluation

In addition to its formal privacy guarantee from PI, PI-Net was also evaluated empirically in terms of (1) quality of the obfuscated image, (2) accuracy of attribute re-classification, and (3) trade-off between privacy and utility. First, quality of the obfuscated image is used to measure the similarity degradation between the resultant privacy-

protected image and its original one in terms of Frechet Inception Distance (FID) [18] and Structural Similarity Image Measurement (SSIM) [49]. Second, accuracy of attribute re-classification is used to verify if the attributes, intended to be preserved, can be retained well. We used a pre-trained attribute prediction model based on ResNet-50 [21] to extract the attributes and calculate the attribute preservation ratio, which is defined as the number of privacy-protected images whose attributes are re-classified correctly divided by the number of total outputs. Third, we examined the trade-off between privacy and utility of PI-Net, where privacy is measured in terms of face recognition via pre-trained FaceNet [39] based on Inception-Resnet backbone [44] in that low recognition rate implies high privacy, and utility is measured in terms of face detection.

**Quality of The Obfuscated Image.** We can see from Table 1 that PI-Net yields the lowest FID and highest SSIM when compared with PixDP [10] and SVD-Priv [11]. The above FID and SSIM results are also consistent with Figure 2, where PI-Net generates visually pleasing images.

$\epsilon$	SSIM			FID		
	0.1	0.5	1.0	0.1	0.5	1.0
PixDP	0.01	0.02	0.03	445	445	442
SVD-Priv	0.07	0.21	0.31	333	271	250
PI-Net	0.29	0.35	0.37	61	59	59

Table 1. Image quality (higher SSIM and lower FID implies better image quality) vs. privacy budget (lower  $\epsilon$  implies better privacy).

**Accuracy of Attribute Re-Classification.** Figure 3 shows the attribute preservation ratios under different privacy budgets  $\epsilon$  and different numbers of attributes to be preserved. The face attributes that are specified to be preserved are depicted in Table 2. The red, blue, and green curves in Figure 3 denote the results for retaining two-attribute, three-attribute, and four-attributes, respectively. The solid curves were generated using the triplet and cross-entropy loss functions but the dash curves were not. Such an ablation study shows that the use of triplet and cross-entropy loss indeed helps in retaining the desired facial attributes.

Preserved face semantic	
2 attributes	Young, Smiling
3 attributes	Young, Smiling, HeavyMakeup
4 attributes	Young, Smiling, HeavyMakeup HighCheekbones

Table 2. Three cases of attributes selected to be preserved.

Moreover, it is worth mentioning that the attribute preservation ratio is not 100% due to the following two reasons. We used the attribute “Smiling” as an example

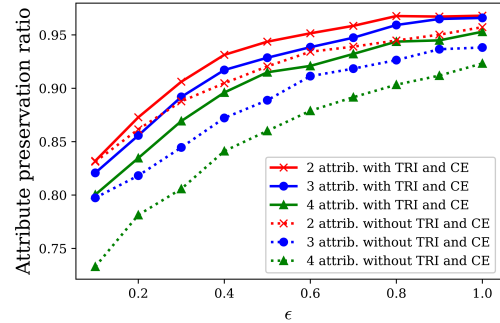


Figure 3. Attribute preservation ratio vs. privacy budget  $\epsilon$  with/without triplet (TRI) and cross-entropy (CE) losses.

in four-attributes setting for the following discussions. As shown in Figure 4(a), the original image possesses the attribute “non-Smiling”, and this attribute is desired to be preserved after applying PI-Net. We can visually see from Figure 4(b) that the privacy-protected image still has the attribute “non-Smiling”. We find that both the original and privacy-protected images belong to the same cluster in the  $\mathcal{Y}$  space. However, the specified attribute in Figure 4(b) is inaccurately predicted to be “Smiling” due to attribute prediction for attribute label assignment.

On the other hand, the original attribute “non-Smiling” has been changed in the resultant privacy-protected image in Figure 4(c), which possesses “Smiling”. The reason of failing to retain the desired attribute comes from the mapping error from the space  $\mathcal{Y}$  to the latent space  $\mathcal{X}$ .

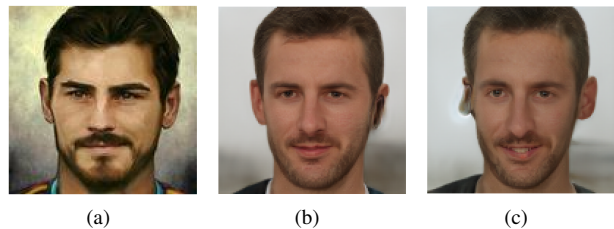


Figure 4. Reasons of failing to preserve desired attributes perfectly. (a) Original image with attribute “non-Smiling”; (b) Privacy protected image with the attribute “non-Smiling” but is predicted to be “Smiling”; (c) Privacy protected image with the attribute “Smiling” due to erroneous mapping to the latent space.

**Ablation Study on Loss Functions.** Figure 5 illustrates how the triplet loss in the PI-Net enables the obfuscated image to be re-classified correctly for face attributes. The results in the top row were obtained in PI-Net by removing the triplet loss and keeping only the MSE loss, whereas those in the bottom row were generated using both the triplet and cross-entropy loss functions.

It can be observed from Figures 5(d)-5(f) that because the triplet loss brings transformed codes with the same semantics close enough, the facial attributes do not deviate significantly after image obfuscation. Therefore, the specified facial attributes of the obfuscated images can be identi-

cal to those of the corresponding original ones. On the contrary, for the model that only uses MSE as the loss function, the distribution of transformed codes, as shown in Figures 5(a)-5(c), becomes chaotic without forming clusters.

Furthermore, it is worth noting that some clusters cannot be visualized because they contain too few points. This is due to some combinations of attributes being almost non-existent. For example, Figure 5(f) specifies that four binary attributes should be preserved, but only seven clusters appear. The reason behind this is that some clusters are empty.

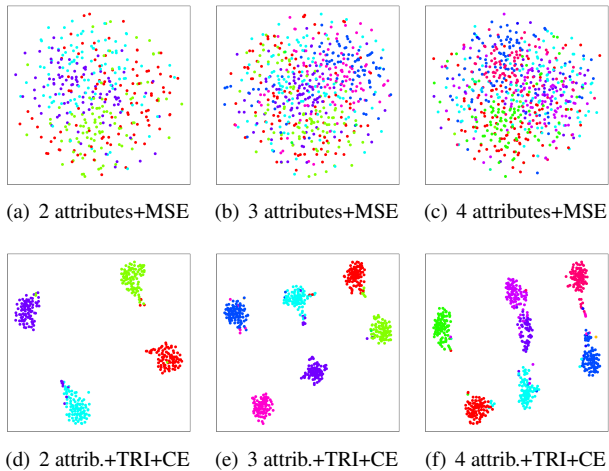


Figure 5. Visualizing the distribution of the transformed codes with *t*-SNE [47]. (a)/(b)/(c) correspond to the latent code distribution with 2/3/4 attributes under the use of MSE as loss function. (d)/(e)/(f) correspond to the latent code distribution with 2/3/4 attributes under the use of triplet and cross-entropy loss functions.

**Trade-off between Privacy and Utility.** Ideally, PI-Net only obfuscates the identifiable features, while preserving the integrity of multiple face attributes. However, due to the factors such as the noise scale introduced by PI and the imperfection of attribute clustering, PI-Net might generate the images with distortion on the attributes not selected by the user. Following [29], we also examined two important capabilities that an anonymization method should have, *i.e.*, high detection rate and low identification rate.

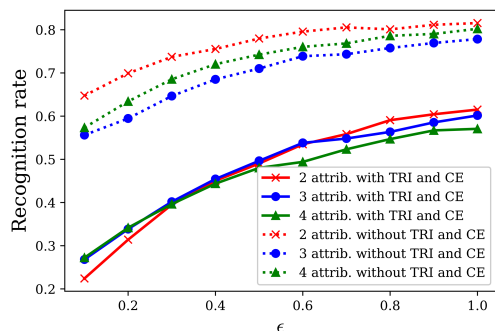


Figure 6. Re-identification ratio vs. privacy budget  $\epsilon$  with/without triplet (TRI) and cross-entropy (CE) losses.

First, we show the trade-off between privacy and face recognition in Figure 6. We used FaceNet based on Inception-Resnet backbone as our recognition model, which is released from OpenFace [4]. One can make the following two observations: (1) The more attributes that are preserved, the higher the percentage of obfuscated images that are re-identified, which means that fewer identifiable features are available for obfuscation. (2) The use of the triplet loss function and cross-entropy will increase the percentage of obfuscated images that are re-identified. This is caused by the preservation of more attributes, and therefore fewer identifiable features are available for obfuscation. To show the ability of privacy protection, the cosine similarity of Facenet embeddings could present the familiarity between origin and after protection image. Second, we examined the face detection rate of PI-Net. The results show that, unlike the conventional pixelization and blurring techniques (see Figure 2 in [29]), the detection rates using HOG [8] of PI-Net are almost 100%, meaning that the face structure is maintained very well. We also find that the use of triplet and cross-entropy loss functions only sacrifices negligible detections. When compared with PixDP [10] and SVD-Priv [11], Table 3 shows that these two methods fail to preserve face image quality effectively, their face detection rates are remarkably lower than ours under different settings of privacy budgets.

	$\epsilon = 0.1$	$\epsilon = 0.3$	$\epsilon = 0.5$	$\epsilon = 1.0$
PixDP	0.00	0.00	0.00	0.00
SVD-Priv	0.11	0.18	0.25	0.39
PI-Net	0.99	1.00	1.00	1.00

Table 3. Face detection rate (higher rate implies better preservation of a face in obfuscated image) vs. privacy budget (lower  $\epsilon$  implies better privacy)

## 6. Conclusion

In this paper, we have proposed Perceptual Indistinguishability-Net (PI-Net) to study how differential privacy (DP) can be employed for facial image obfuscation while maintaining certain utility. In PI-Net, perceptual indistinguishability is presented to define the adjacency between images in the latent space for injecting noises to achieve obfuscation, and facial semantic attributes are manipulated to generate realistic looking faces. Experimental results demonstrate that our method can satisfy the trade-off between privacy and utility. To our knowledge, we are first to introduce perceptual similarity to define image indistinguishability in the context of DP.

**Acknowledgements** This work was supported by Ministry of Science and Technology, Taiwan, ROC, under Grants MOST 109-2221-E-001-023, 107-2221-E-001-015-MY2, and 110-2636-E-009-018.



## References

- [1] Martín Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. *ACM Conference on Computer and Communications Security (CCS)*, 2016.
- [2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4431–4440, 2019.
- [3] Darius Afchar, Vincent Nozick, J. Yamagishi, and I. Echizen. Mesonet: a compact facial video forgery detection network. *IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7, 2018.
- [4] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.
- [5] Konstantinos Chatzikoakakis, Miguel Andrés, Nicolás Emilio Bordenabe, and Catuscia Palamidessi. Broadening the scope of differential privacy using metrics. In *Privacy Enhancing Technologies Symposium (PETS)*, pages 82–102. Springer, 2013.
- [6] Dingfan Chen, Tribhuvanesh Orekondy, and Mario Fritz. Gswgan: A gradient-sanitized approach for learning differentially private generators. In *NIPS*, 2020.
- [7] Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, S. Kim, and J. Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8789–8797, 2018.
- [8] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 886–893, 2005.
- [9] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9:211–407, 2014.
- [10] Liyue Fan. Differential privacy for image publication. In *Theory and Practice of Differential Privacy (TPDP) Workshop*, 2019.
- [11] Liyue Fan. Practical image obfuscation with provable privacy. *IEEE International Conference on Multimedia and Expo (ICME)*, pages 784–789, 2019.
- [12] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. *USENIX Symposium on Security*, 2014.
- [13] Oran Gafni, Lior Wolf, and Yaniv Taigman. Live face de-identification in video. *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9377–9386, 2019.
- [14] Ralph Gross, Latanya Sweeney, Fernando de la Torre, and Simon Baker. Model-based face de-identification. *Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, pages 161–161, 2006.
- [15] David Guera and E. Delp. Deepfake video detection using recurrent neural networks. *IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6, 2018.
- [16] Ishaan Gulrajani, F. Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of wasserstein gans. In *NIPS*, 2017.
- [17] Zhenliang He, W. Zuo, M. Kan, S. Shan, and X. Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE Transactions on Image Processing*, 28:5464–5478, 2019.
- [18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.
- [19] Steven Hill, Zhimin Zhou, Lawrence Saul, and Hovav Shacham. On the (in)effectiveness of mosaicing and blurring as tools for document redaction. *Privacy Enhancing Technologies Symposium (PETS)*, 2016.
- [20] Håkon Hukkelås, Rudolf Mester, and Frank Lindseth. Deepprivacy: A generative adversarial network for face anonymization. *International Symposium on Visual Computing (ISVC)*, 2019.
- [21] Shaoqing Ren, JKaiming He, Xiangyu Zhang and Jian Sun. Deep residual learning for image recognition. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [22] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.
- [23] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations (ICLR)*, 2019.
- [24] Amin Jourabloo, Xi Yin, and X. Liu. Attribute preserved face de-identification. *International Conference on Biometrics (ICB)*, pages 278–285, 2015.
- [25] Tero Karras, S. Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405, 2019.
- [26] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. *IEEE International Conference on Data Engineering (ICDE)*, pages 106–115, 2007.
- [27] Tao Li and Lei Lin. Anonymousnet: Natural face de-identification with measurable privacy. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019.
- [28] Yijun Li, Sifei Liu, Jimei Yang, and Ming-Hsuan Yang. Generative face completion. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5892–5900, 2017.
- [29] Maxim Maximov, Ismail Elezi, and Laura Leal-Taixé. Ciagan: Conditional identity anonymization generative adversarial networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5447–5456, 2020.
- [30] Cade Metz. Facial recognition tech is growing stronger, thanks to your face. *The New York Times*, July 13, 2019.
- [31] Ilya Mironov. Rényi differential privacy. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275, 2017.
- [32] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

- [33] Lily Hay Newman. Ai can recognize your face even if you're pixelated. *WIRED*, 2016.
- [34] Elaine Newton, L. Sweeney, and Bradley Malin. Preserving privacy by de-identifying face images. *IEEE Transactions on Knowledge and Data Engineering*, 17:232–243, 2005.
- [35] Seong Joon Oh, Rodrigo Benenson, Mario Fritz, and Bernt Schiele. Faceless person recognition: Privacy implications in social media. *European Conference on Computer Vision (ECCV)*, 2016.
- [36] Kenny Peng. Facial recognition datasets are being widely used despite being taken down due to ethical concerns. here's how. *Freedom to Tinker*, October 21, 2020.
- [37] Shengju Qian, Kwan-Yee Lin, W. Wu, Yangxiaokang Liu, Q. Wang, F. Shen, Chen Qian, and R. He. Make a face: Towards arbitrary high fidelity face manipulation. *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10032–10041, 2019.
- [38] Zhongzheng Ren, Y. Lee, and M. Ryoo. Learning to anonymize faces for privacy preserving action detection. In *European Conference on Computer Vision (ECCV)*, 2018.
- [39] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015.
- [40] Yujun Shen, Jinjin Gu, X. Tang, and B. Zhou. Interpreting the latent space of gans for semantic face editing. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9240–9249, 2020.
- [41] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*, 2015.
- [42] Qianru Sun, Liqian Ma, Seong Joon Oh, Luc Van Gool, Bernt Schiele, and Mario Fritz. Natural and effective obfuscation by head inpainting. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [43] Qianru Sun, Bernt Schiele, and Mario Fritz. A domain based approach to social relation recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [44] Christian Szegedy, S. Ioffe, V. Vanhoucke, and Alexander Amir Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, 2017.
- [45] Ruben Tolosana, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *Information Fusion*, 64:131–148, 2020.
- [46] Reihaneh Torkzadehmahani, Peter Kairouz, and B. Paten. Dp-cgan: Differentially private synthetic data and label generation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 98–104, 2019.
- [47] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.
- [48] Han Wang, Shangyu Xie, and Yuan Hong. Videodp: A flexible platform for video analytics with differential privacy. *Privacy Enhancing Technologies Symposium (PETS)*, 2020(4):277–296, 2020.
- [49] Zhou Wang, A. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13:600–612, 2004.
- [50] Po-Wei Wu, Yu-Jing Lin, Che-Han Chang, Edward Chang, and Shih-Wei Liao. Relgan: Multi-domain image-to-image translation via relative attributes. *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5913–5921, 2019.
- [51] Taihong Xiao, Yi-Hsuan Tsai, Kihyuk Sohn, M. Chandraker, and Ming-Hsuan Yang. Adversarial learning of privacy-preserving and task-oriented representations. In *The AAAI Conference on Artificial Intelligence (AAAI)*, 2020.
- [52] Raymond A. Yeh, Chen Chen, Teck-Yian Lim, Alexander G. Schwing, M. Hasegawa-Johnson, and M. Do. Semantic image inpainting with deep generative models. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6882–6890, 2017.
- [53] Xiaogang Wang Ziwei Liu, Ping Luo and Xiaoou Tang. Deep learning face attributes in the wild. *International Conference on Computer Vision (ICCV)*, pages 3730–3738, 2015.