Differentially Private Imaging via Latent Space Manipulation

Tao Li and Chris Clifton
Department of Computer Science, Purdue University, West Lafayette, Indiana, USA
{taoli,clifton}@purdue.edu

Abstract

There is growing concern about image privacy due to the popularity of social media and photo devices, along with increasing use of face recognition systems. However, established image de-identification techniques are either too subject to re-identification, produce photos that are insufficiently realistic, or both. To tackle this, we present a novel approach for image obfuscation by manipulating latent spaces of an unconditionally trained generative model that is able to synthesize photo-realistic facial images of high resolution. This manipulation is done in a way that satisfies the formal privacy standard of local differential privacy. To our knowledge, this is the first approach to image privacy that satisfies ε-differential privacy for the person.

Figure 1: Can you identify the authors? These are images of the authors, with noise added that satisfies differential privacy sufficient to prevent identification of the authors if you do not already know who they are.

1. Introduction

Image obfuscation techniques have been used to protect sensitive information, such as human faces and confidential texts. However, recent advances in machine learning, especially deep learning, make standard obfuscation methods such as pixelization and blurring less effective at protecting privacy [1]; it has been showed that over 90% of blurred faces can be re-identified by deep convolutional neural networks or commerical face recognition systems [2].

We envision scenarios where the image should convey the general tone and activity (e.g., facial expressions), but not identify individuals. For example, one could post photos on social media retaining images of friends, but protecting identity of bystanders while maintaining the general feel of the image; an example is given in fig. 2. In a very different scenario, surveillance footage could be viewed by police to identify suspicious acts, but identity of those in the image would only be available through appropriate court order, protecting against (possibly unintended) profiling and "guilt by association". In both scenarios, blurring/pixelization fails to preserve desired semantics (e.g., facial expression), and fails to provide the desired privacy protection.

Many attempts have been made to obfuscate images and some privacy guarantees are provided. A pixelization method proposed in [3] satisfies pixel-wise ϵ-differential privacy [4]. However, the utility of the pixelized images is far from satisfactory, due to the high perturbation noise needed to reasonably hide the original; the images appear like traditional pixelization or blurring techniques. A more serious problem is that this provides differential privacy for *pixels*, not for the individuals pictured in the image. Not only are the images highly distorted, as with ad-hoc approaches to pixelization and blurring they are subject to re-identification of the individuals in the image [5, 6, 7].

In this paper, we show how differential privacy can be provided at the level of the individual in the image. The key idea is that we transform the image into a semantic latent space. We then add random noise to the latent space representation in a way that satisfies ε-differential privacy. We then generate a new image from the privatized latent space representation. This ensures a formal privacy guarantee, while providing an image that preserves important characteristics of the original.

Other obfuscation methods have been proposed recently to balance privacy and utility. For example, adding noise to an SVD-transformation is proposed in [8]; however, the

(a) Original (b) Differentially private bystander

Figure 2: Protecting bystanders on social media. The person in the background has been replaced with a differentially private version, while the subject of the image is maintained. Note that in real use, the background and lighting would be blended (as a postprocessing step); for clarity we are showing only the facial image manipulation.

approach does not formalize privacy in the sense of identifying *individuals*. [9] makes use of generative adversarial networks to obfuscate a face in the context of detecting and depicting (anonymized) actions. [10] adopted generative adversarial networks (GANs) for facial image obfuscation by identifying a face and accordingly inpainting it with a synthesized face alternative. This unfortunately has the potential to lose important characteristics of the original image. Another approach leveraged a conditional GAN to manipulate facial attributions in accordance with different privacy requirements [11]. These approaches suffer the common failing that they do not provide a *formal* privacy guarantee. As such, they may be subject to re-identification or re-construction attacks.

Building on top of previous works, this paper presents a practical image obfuscation method with provable guarantees and some level of photo-realism. Unlike [10] which replaces the entire face with an arbitrary substitute, and [11] which obfuscates facial images on a discrete attribute space, this work further extends facial image manipulation to a continuous latent space. Applying differential privacy in this latent semantic space provides greater photo-realism while satisfying rigorous privacy guarantees.

A key to formal privacy methods is that there is randomness in the approach: the same image privatized twice will not look the same (as demonstrated in fig. 1, which includes multiple images of some authors.) This randomness is a key component to preventing reconstruction attacks. We show that randomized manipulations in the latent semantic space can be expected to provide realistic images. The method guarantees similarity-based indistinguishability among images, providing privacy guarantees in worst-case scenarios and boosting the utility of the obfuscated image.

Our main contributions are:

- The first definition of ε-differential privacy for images that protects *individuals* in the image;
- A practical framework for real-world differentially private imaging that maintains a level of image semantics;
- We introduce a clipping step in image latent space that enables a formal guarantee of ε-differential privacy with significantly improved fidelity.

In the rest of the paper: Section 2 discusses related works; We formalize our approach in section 3 and show how it satisfies differential privacy with a practical framework; Section 4 details our implementation and demonstrates the results; Section 5 concludes the paper.

2. Related Work

Our work bears some similarity to differentially private synthetic data generation [14, 15, 16, 17], perhaps most notably the use of generative networks for synthetic data [18, 19]. However, the problem addressed in those works is generating synthetic data representing a *set* of individuals, rather than the local differential privacy we achieve. If applied directly to an image, such approaches would provide privacy for *pixels*, not for persons - similar to the work of [3, 11] discussed in the introduction. We also noted other transformation-based approaches [8, 9, 10] that do not provide formal privacy guarantees.

Other work has shown that ML models can memorize (and subsequently leak) parts of their training data [20]. This can be exploited to expose private details about members of the training dataset [21]. These attacks have spurred a push towards differentially private model training [22], which uses techniques from the field of differential privacy to protect sensitive characteristics of training data. This is

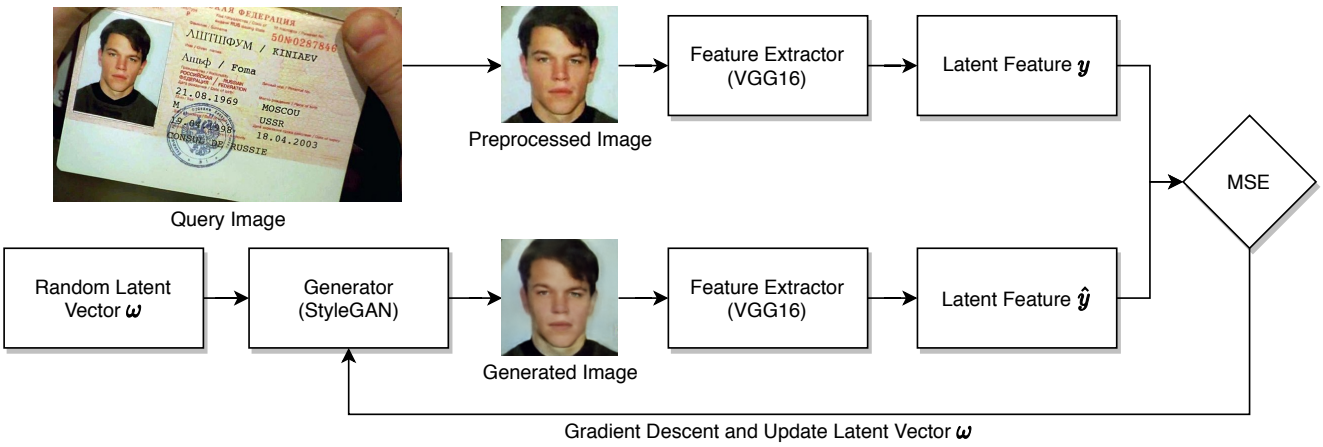


Figure 3: Feature optimization pipeline to encode an arbitrary image. We first crop and align the query image and compare it with a random image generated by StyleGAN [12] using a loss function (e.g., mean squared error). Instead of comparing the two images in a pixel-wise fashion, we leverage a deep feature extractor (i.e., VGG16 [13]) to obtain latent representations of the images, and then apply gradient descent to optimize the latent code $\hat{\mathbf{y}}$ of the random image until the synthesized image is close enough to the query one.

a very different problem, our goal is to protect images that are *not* contained in the training data.

There is also work targeted to defeating existing face recognition systems. Much of the work in image privacy results in substantial distortion. As with pixelization, these often produce images that are not visually pleasing. Methods include distorting images to make them unrecognizable [11, 23, 24], and producing adversarial patches in the form of bright patterns printed on sweatshirts or signs, which prevent facial recognition algorithms from even registering their wearer as a person [25, 26]. Finally, given access to an image classification model, "clean-label poison attacks" can cause the model to misidentify a single, pre-selected image [27, 28]. However, these are targeted against facial recognition systems designed without regard to the privacy protection, and could be subject to targeted re-identification attacks such as [5, 6, 7].

3. Differentially Private Imaging

From the above, it should now seem obvious how we can get differential privacy: Add noise to the latent vector $\mathbf{z}$ in a way that satisfies differential privacy. This leaves three questions, addressed in this section. The first is what mechanism do we use to add noise? There are multiple mechanisms providing differential privacy; the right choice depends on how noise impacts the final result. The second question is how much noise do we need to add? This requires understanding the *sensitivity* of the latent vector $\mathbf{z}$: How much it can vary across different input images. This will be covered in section 3.2. The final issue is how to obtain the latent vector $\mathbf{z}$ in the first place? Answering these questions requires a better understanding of the latent space.

3.1. Latent Space and Image Encoding

It has been widely observed that when linearly interpolating two latent codes $\mathbf{z}_1$ and $\mathbf{z}_2$, the appearance of the corresponding synthesis changes continuously [29, 30, 12]. [29] and [31] identifies some vector arithmetic phenomenon in a GAN's latent space, such as addition and subtraction invariance, implying the linearity property of latent spaces, as well as monotonicity and Euclidean distance. [32] provides a proof. The linear interpolation between $\mathbf{z}_1$ and $\mathbf{z}_2$ forms a direction in latent space $\mathcal{Z}$, which further defines a hyperplane, and the hyperplane splits a binary semantics.

Given a real world facial image, the problem of finding its corresponding latent representation can be considered as an optimization problem where we search the latent space to find a latent vector, from which the reconstructed image is close enough (and hopefully identical) to the query image. Figure 3 illustrates the optimization pipeline. Being trained in a reasonably large and representative image dataset (e.g., FlickrFaces-HQ (FFHQ) [12]), a GAN model is presumed to memorize and represent the universe of facial images. We first start with a random latent vector ω and place it in a generator (e.g., StyleGAN [12]) to obtain a synthesized image. Instead of comparing the query image with its synthesized counterpart in a pixel-wise manner, we leverage a feature extractor (VGG16 [13]) to obtain latent representations of each image and compare the loss function (i.e., MSE) in the feature space, as deep feature loss has been shown superior to pixel loss in practice [33]. Afterward, we use gradient descent to update the latent vector ω until the generated image converges to the query one.

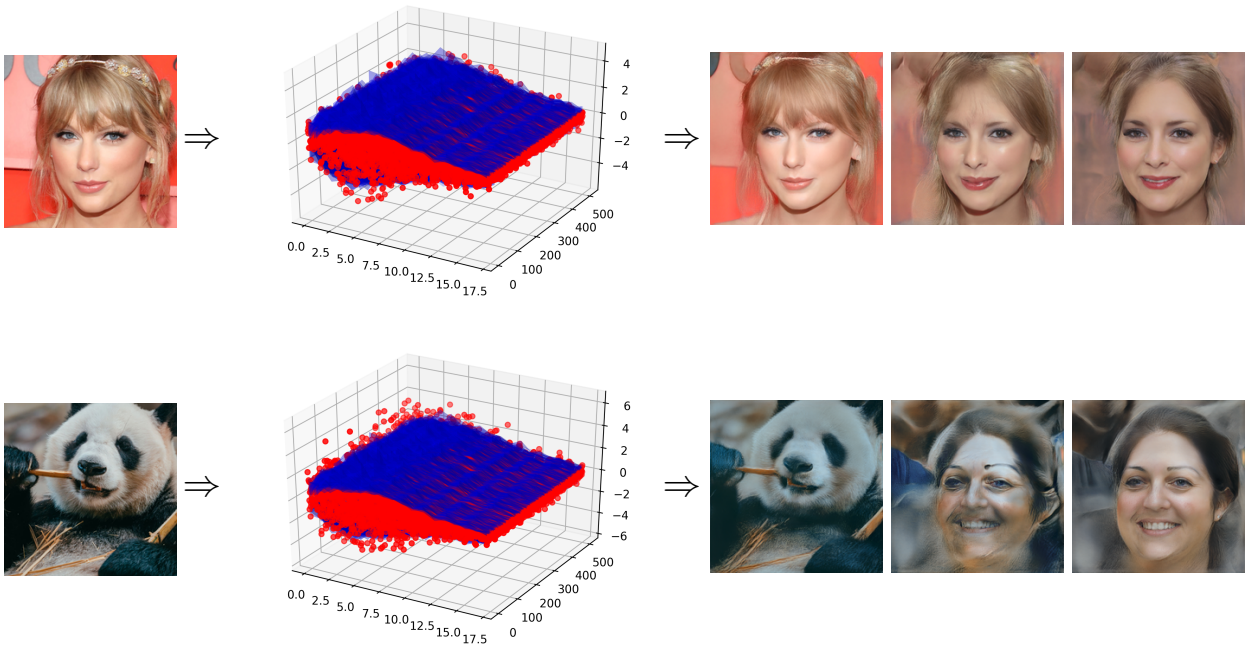


Figure 4: Clipping in the latent space. Given an input image, we first obtain its latent code (red points) using the feature optimization pipeline in fig. 3. We then clip the latent code to make all the components fall within the lower and upper bounds (blue surfaces) given by their distributions in the dataset.The clipped latent code are fed into the GAN model to generate the outputs. Note that none of the examples are in the training dataset. The three outputs (from left to right) are clipping at $0\%/100\%$, $15\%/85\%$, and $30\%/70\%$, respectively.

3.2. Privacy Mechanism

A key issue in using the Laplace mechanism for ε-differential privacy is determining the sensitivity: How much changing one individual can impact the result. Sensitivity is the maximum amount that the latent space could change by replacing one image to e privatized with **any** other image.

More formally, we want to determine the maximum the sensitive values could change if we replaced any possible input image with any other input image. This would be the *LDP sensitivity*, and adding noise commensurate with that difference would give us ε-differential privacy.

Unfortunately, the maximum possible difference in the latent space between any two input images is not only difficult to bound, but would result in untenable levels of noise. Imagine, for example, a completely black and completely white image - vastly different, and not really interesting from a privacy point of view. Furthermore, while those may be the greatest difference in the original space, we need to determine the greatest difference in the latent space, which is not directly related to pixel-level differences in the input.

We use an idea from *maximum observed sensitivity*[34]. They use the largest sensitivity observed across a dataset of significant size as a surrogate for the range of possible values. In our case, the training data (for which we aren't concerned about privacy) is the large dataset; the maximum difference between any two images in the latent space could stand in for the range of possible values. Unfortunately, this does not provide ε-differential privacy: If we were given an unusual input image (say, someone standing on their head, or with particularly unusual features) it could fall outside

Algorithm 1: Sensitivity Calculation.

Require: Dataset with identities $D = \{(X^{(i)}, id_i)\}_{i=1}^{n}$;
Require: Encoder $f : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{X}$ represents the image space and $\mathcal{Z} \subseteq \mathbb{R}^m$ is the latent space with m latent semantics;
1: **for** each image $X^{(i)}$ **do**
2: latent vector $Z^{(i)} \leftarrow f(X^{(i)})$;
3: **end for**
4: **for** each latent semantics Z_j **do**
5: local sensitivity $LS_j \leftarrow \max_{d(x,y)\leq 1} ||f(x) - f(y)||_1$;
6: **end for**
7: LDP sensitivity $S_L \leftarrow \max_x \ LS_f(x)$;

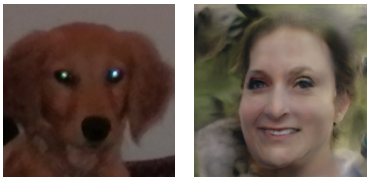


Figure 5: **Left** is the dog in fig. 2, and **right** is its differentially private face.

these bounds, and result in a recognizable image.

Instead, we use the maximum observed sensitivity to *clip* images in the latent space. Any values that fall outside the observed bounds are clipped to the observed bounds, guaranteeing that the range of the input to the differential privacy mechanism is known, allowing us to determine sensitivity. This allows us to fully satisfy ε-differential privacy.

Clipping the images does not come without cost. While it ensures we satisfy differential privacy, an image that falls outside the "normal" values observed in the training data may be significantly distorted. We show examples in fig. 4. The original image is on the left, followed by the 3D point cloud visualization of the latent code in the second column; The third column are images clipped to the maximum and minimum values (i.e., 0% and 100%) observed in a sample of 3500 of the training data images; the fourth and fifth columns are clipped at 15% and 85%, and 30% and 70%, respectively.

Note that we do not claim the clipping itself provides differential privacy. It enables us to bound the range of the input to the mechanism, so that the Laplace mechanism can be used to satisfy differential privacy. In particular, it enables us to provide privacy for outliers. Clipping forces outliers into the range of the training data, causing an image to be generated that resembles the training data when the actual input image is far from the training data. An example is privatizing the dog from fig. 2, giving fig. 5. This results in very high distortion for outlying images, perhaps suggesting they are outliers, but still satisfying the formal definition of differential privacy. It allows us to satisfy differential privacy with higher fidelity for images that bear closer resemblance to the training data.

3.3. Algorithm

We can now discuss how we provide ε-local differential privacy. The idea is that the privacy budget ε is divided among the various components in the latent space. Each is used, along with the sensitivity derived from the clipping values for that component (based solely on the public training data), to determine a random draw of Laplace noise for that component, which is again clipped (a postprocessing step). This gives an ε-differentially private version of the image *in the latent space*.

Algorithm 2: Differentially Private Imaging with Laplace Mechanism

Require: Input image $X^{(i)}$;
Require: Encoder $f : \mathcal{X} \to \mathcal{Z}$;
Require: Generator $g : \mathcal{Z} \to \mathcal{X}$;
Require: Latent space sensitivities S_{Lj};
Require: Latent space weights w_j s.t. $\sum w_j = 1$;
Require: Privacy parameter ε;
Require: Laplace distribution $Lap(0, \lambda)$;
Require: Clipping function $f_c(i, j, \alpha)$;
1: latent vector $Z^{(i)} \leftarrow f(X^{(i)})$;
2: **for** each latent semantics $Z_j^{(i)}$ **do**
3: obtain a random δ from $Lap(S_{Lj} \cdot w_j/\varepsilon)$;
4: $Z_j'^{(i)} \leftarrow Z_j^{(i)} + \delta$;
5: $Z_j''^{(i)} \leftarrow f_c(Z_j'^{(i)})$;
6: **end for**
7: desired noisy image $X'^{(i)} \leftarrow g(Z''^{(i)})$;

We use this differentially private latent space version, with no reference to the original image or the latent space transformation of the original image, to generate an image using the previously described generative network. The overall algorithm is given in algorithm 2. Note that the feature optimization pipeline in fig. 3 serves as the encoder f.

3.4. Privacy Guarantee

Remember that our goal is not to protect the individuals in the training data (these are assumed to be public, e.g., for the experiments in this paper the encoder and generator were trained using the FlickrFaces-HQ (FFHQ) dataset.) The goal is to protect the individual in a *new* image. Therefore we assume that nothing in algorithm 2 depends on the individual in the input image $X^{(i)}$ except what is explicitly shown in the algorithm.

Theorem 1. Algorithm 2 provides ε-local differential privacy.

Proof. $\mathcal{M}$ is the randomized mechanism in algorithm 2. Using the notations in [35] and above, we have

$$\begin{aligned}\frac{\Pr[\mathcal{M}(v, f, \varepsilon) = s]}{\Pr[\mathcal{M}(v', f, \varepsilon) = s]} &= \frac{\Pr[Lap(S_L \cdot w_f/\varepsilon)] = s - f(v)}{\Pr[Lap(S_L \cdot w_f/\varepsilon)] = s - f(v')} \\ &= \frac{\frac{S_L \cdot w_f}{\varepsilon} \cdot \exp(-\frac{|s-f(v)|\varepsilon}{S_L \cdot w_f})}{\frac{S_L}{\varepsilon} \cdot \exp(-\frac{|s-f(v')|\varepsilon}{S_L \cdot w_f})} \\ &= \exp(\frac{\varepsilon|f(v') - f(v)|}{S_L \cdot w_f}) \leq \exp(\varepsilon) \\ &= \exp(\frac{\varepsilon|f(v') - f(v)|}{S_L}) \leq \exp(\varepsilon \cdot w_f)\end{aligned}$$

□

Each component in the semantic space transformation of the image has noise added from a Laplace distribution. From [4], we have that each component $Z_j'^{(i)}$ is $(\varepsilon \cdot w_f)$-differentially private. Sequential composition gives $Z'^{(i)}$ is $\sum \varepsilon \cdot w_j = \varepsilon \sum w_j$ differentially private. Since $\sum w_j = 1$, this shows that $Z'^{(i)}$ is ε-differentially private. The remaining image generation step uses only the *noise-added* version of the image in the semantic space. Since no information from the individual in question is used in this or the generator g, the postprocessing theorem of differential privacy tells us that the output image is ε-differentially private.

Some may ask why we do not use a parallel composition argument, since the noise is added independently to each component. The problem is that parallel composition requires that the components be from disjoint individuals; this would be like saying we want to avoid identifying an individual's hairstyle and smile, rather than protecting against identifying the individual.

Note that this makes the assumption that not only is the image to be protected not in the training data, but that the *individual pictured* is not in the training data (or more specifically, not in the data used to train the image generator g.)

4. Empirical Results

The previous section shows how we can achieve a differentially private image, and why we might expect it to produce reasonable images.. We evaluate the proposed method and its results, both qualitatively and quantitatively, using real world images.

4.1. Dataset

For experiments, we adopt the FlickrFaces-HQ (FFHQ) dataset [12] collected by NVIDIA, consisting of 70,000 high-resolution (1024×1024) human facial images. This dataset covers a wide spectrum of human faces, including variations in age, ethnicity, and image backgrounds; crawled from Flickr. To resolve computational issues, we use a randomly-selected subset of 3500 images for these experiments. We align and crop the images using Dlib[1]. All results reported in this paper are based on the aligned and cropped dataset. All the images shown in the paper are publicly available or taken by the authors, and none of them are in the training dataset.

4.2. Evaluation

We first show what happens with small values of ε. Figure 6 demonstrates $\varepsilon = 1, \ldots, 5000$; we can see that the images are not very useful. (Although with sufficient clipping, they are recognizably human.) Note that $\varepsilon = 1$ is very strict; a way to think of this is that if the adversary knows the image is of either Barack Obama or Hillary Clinton, we are

[1] http://dlib.net/

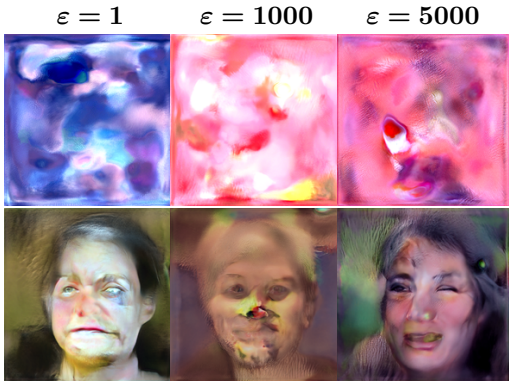


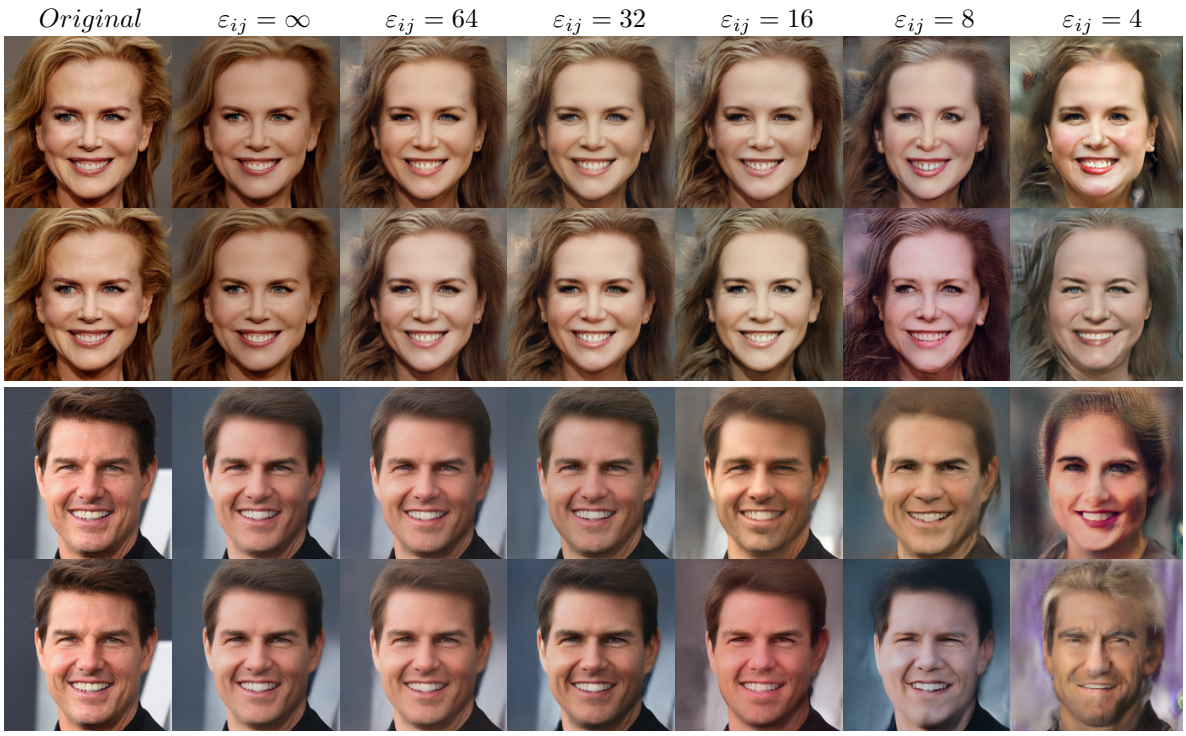
Figure 6: Examples of image privatized using small values of ε, providing privacy against an adversary with strong expectations of who the image might be. The first row has no clipping; the second row is clipped at $10\%/90\%$. In comparison, fig. 7 uses larger values of ε, which leads to better perceptual quality.

adding enough noise that the adversary's best guess would be right 75% of the time (as opposed to 50% without seeing the image.) This does hold for these images, even knowing that fig. 6 is either Obama or Clinton, anyone (correctly) guessing that it was generated from an image of President Obama would have little confidence in that guess.

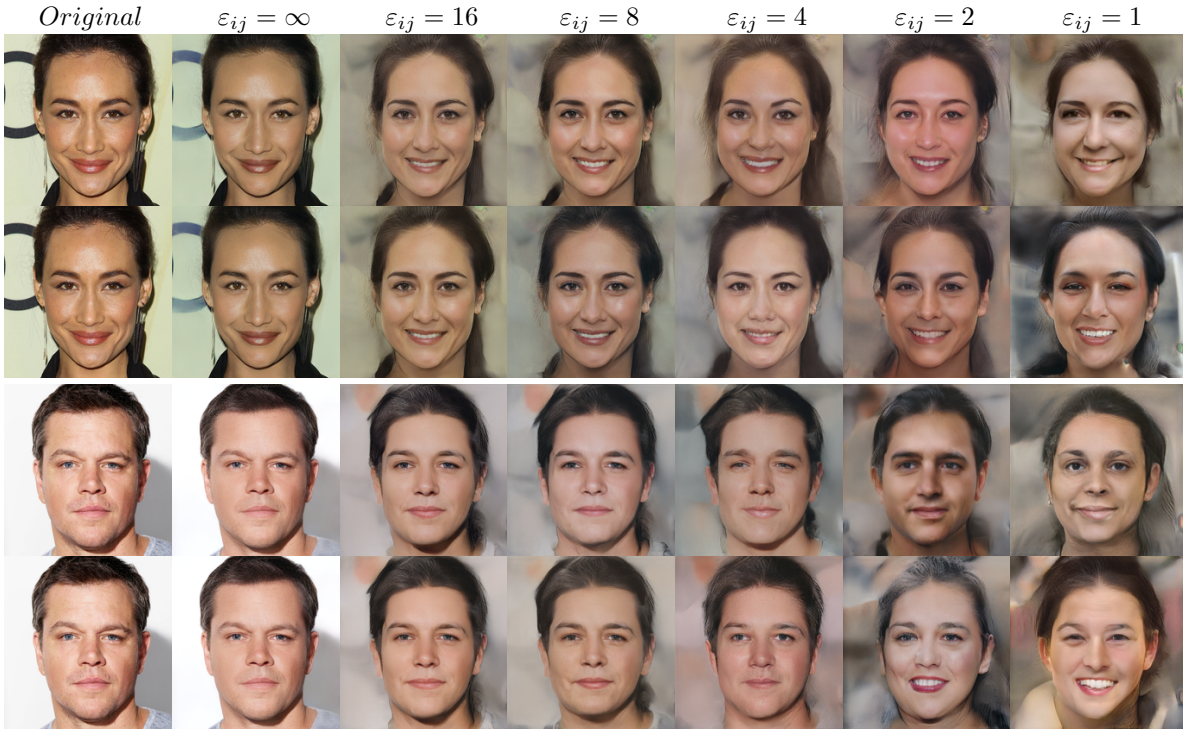
Our use case is much different; the image may be known to be part of a large community (e.g., it is taken on a college campus, so likely belongs to someone on that campus), but could belong to any of thousands of people in that group. This enables a much larger ε without a significant risk of re-identification *in the absence of other information providing a strong expectation on who the image belongs to.* For more discussion of setting ε, see [36, 37].

The remainder of the images we show are with much larger ε. For example, in fig. 1, if you knew the names of the authors, you would have a good shot at guessing which picture went with which author. But in a double-blind review process, where the authors could be any of the thousands of people who might submit to Privacy Enhancing Technologies, identifying who the authors are is infeasible even at $\varepsilon = 9216$.

With a larger of ε and proper clipping, fig. 7 has much more visually pleasing results. It shows results under various settings. For each identity, we show two groups of experimental results under the same setting. They have different outputs because of the randomness of the mechanism. The first column is the inputs and the second column is the generated results from the image encoding pipeline, where the synthesized images are optimized to be as close to the original as possible (i.e., $\varepsilon = \infty$). The remaining columns showcase outputs under different noise levels. With a large



Clipping at 12.5% and 87.5%.



Clipping at 25% and 75%.

Figure 7: Experimental results with different privacy and clipping settings. For each identity, two groups of experimental results under the same settings are given. They produce different outputs because of the randomness of the mechanism. Note that the number of latent components is $18 \times 512 = 9216$ for our experiments and privacy loss $\varepsilon = \sum \varepsilon_{ij} = 9216 \cdot \varepsilon_{ij}$.

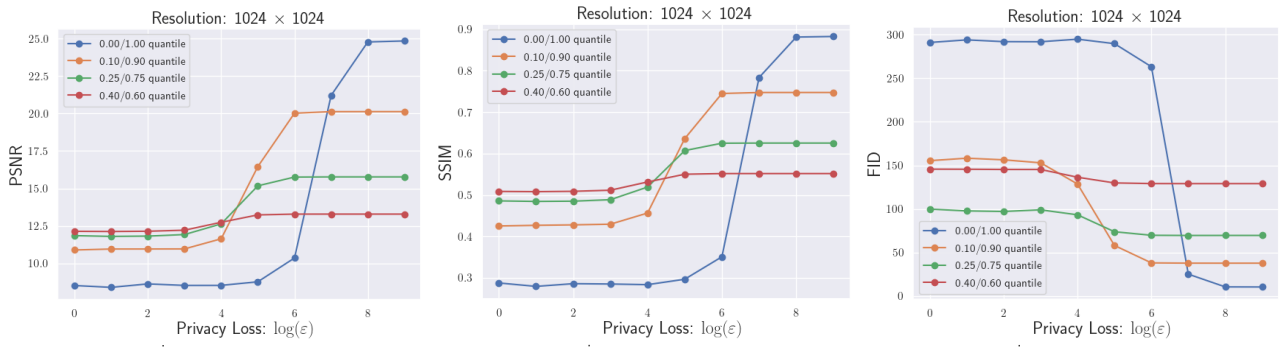


Figure 8: Trade-offs between privacy and utility. By varying privacy parameter ε in the latent space, the pixels vary accordingly. We show SSIM, PSNR, and FID with respect to privacy loss, and the larger the privacy loss is, the closer the image to its original. Above figures demonstrate that both pixel-wise distortion and perceptual distance become smaller as the privacy loss ε increases, indicating that the less noise added, the closer the generated image to its original identity, and vice versa. This aligns with our intuition, as presumably each latent value controls a group of pixels.

noise (i.e., a smaller ε), the output image is less similar to the original (i.e., more private) while still maintaining some fidelity (e.g., it is still a human face sharing some features with the original).

Clipping also plays an important role in this process to maintain the perceptual quality of the image. Under the same noise level, a heavily clipped output has better visual quality than one without clipping, although it loses more specifics of the original. Notice that in fig. 4, an input image of an animal after clipping results in a human face. Even a white noise input, with substantial clipping, appears to show a human face (since this is what the training data consists of.) This basically shows what surfaces in the latent space look like.

Figure 8 quantitatively evaluate the outputs from the proposed method with different privacy and clipping settings. PSNR and SSIM measure the level of distortion at the pixel level; while FID captures the differences at the semantic level. Four clipping settings are tested, $0\%/100\%$, $10\%/90\%$, $25\%/75\%$, and $40\%/60\%$, each of which corresponds to a line in the figures. The trends are clear that clipping makes the outputs more robust to noise; and those without clipping would have greater distortion as noise increase. These results align with fig. 7 as well as our intuition.

5. Conclusion

In this work, we provide the first meaningful formal definition of ε-differential privacy for images by leveraging the latent space of images and Laplace mechanism. A practical framework is presented to tackle real world images. Experimental results show that the proposed mechanism is able to preserve privacy in accordance with privacy budget ε while maintain high perceptual quality for sufficiently large values of ε.

For a practical example of such a mechanism, assume a differentially private high-ε image is posted on a social media site. A face recognition system to automatically tag people in the image may be able to correctly tag the poster, and friends of the poster – subjects that the poster would probably not choose to anonymize anyway. But even with high ε, attempts to identify others in the image who are anonymized, while significantly better than a random guess, would still have extremely low confidence.

There is still considerable room for improvement. We have split the privacy budget evenly between components; varying this split may result in significantly better quality. Varying privacy budget between semantic components could be used to adjust what is preserved (e.g., preserving pose or emotional state at the expense of lower fidelity to age or gender.) Methods based on the exponential mechanism of differential privacy rather than noise addition are likely to provide more realistic images, but with less relationship to the original. The key is that all of these build on the same basic concept: Defining modifications in the latent space such that privacy is provided for the *person*.

References

[1] Richard McPherson, Reza Shokri, and Vitaly Shmatikov. Defeating image obfuscation with deep learning. *arXiv preprint arXiv:1609.00408*, 2016. 1

[2] Tao Li and Min Soo Choi. DeepBlur: A simple and effective method for natural image obfuscation. *arXiv preprint arXiv:2104.02655*, 2021. 1

[3] Liyue Fan. Image pixelization with differential privacy. In *IFIP Annual Conference on Data and Applications Security and Privacy*, pages 148–162. Springer, 2018. 1, 2

[4] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analy-

sis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006. 1, 6

[5] Ralph Gross, Edoardo Airoldi, Bradley Malin, and Latanya Sweeney. Integrating utility into face de-identification. In *International Workshop on Privacy Enhancing Technologies*, pages 227–242. Springer, 2005. 1, 3

[6] Ralph Gross, Latanya Sweeney, Fernando De la Torre, and Simon Baker. Model-based face de-identification. In *2006 Conference on computer vision and pattern recognition workshop (CVPRW'06)*, pages 161–161. IEEE, 2006. 1, 3

[7] Ralph Gross, Latanya Sweeney, Jeffrey Cohn, Fernando De la Torre, and Simon Baker. Face de-identification. In *Protecting privacy in video surveillance*, pages 129–146. Springer, 2009. 1, 3

[8] Liyue Fan. Practical image obfuscation with provable privacy. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 784–789. IEEE, 2019. 1, 2

[9] Zhongzheng Ren, Yong Jae Lee, and Michael S Ryoo. Learning to anonymize faces for privacy preserving action detection. In *Proceedings of the european conference on computer vision (ECCV)*, pages 620–636, 2018. 2

[10] Qianru Sun, Liqian Ma, Seong Joon Oh, Luc Van Gool, Bernt Schiele, and Mario Fritz. Natural and effective obfuscation by head inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5050–5059, 2018. 2

[11] Tao Li and Lei Lin. AnonymousNet: Natural face de-identification with measurable privacy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2, 3

[12] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 3, 6

[13] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 3

[14] Jun Zhang, Graham Cormode, Cecilia M Procopiuc, Divesh Srivastava, and Xiaokui Xiao. Privbayes: Private data release via bayesian networks. *ACM Transactions on Database Systems (TODS)*, 42(4):1–41, 2017. 2

[15] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2018. 2

[16] Gergely Acs, Luca Melis, Claude Castelluccia, and Emiliano De Cristofaro. Differentially private mixture of generative neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 31(6):1109–1121, 2018. 2

[17] Differential privacy synthetic data challenge, 2018. 2

[18] Jun-Yan Zhu and Jim Foley. Learning to synthesize and manipulate natural images. *IEEE computer graphics and applications*, 39(2):14–23, 2019. 2

[19] Reihaneh Torkzadehmahani, Peter Kairouz, and Benedict Paten. Dp-cgan: Differentially private synthetic data and label generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[20] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 587–601, 2017. 2

[21] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015. 2

[22] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318, 2016. 2

[23] Qianru Sun, Ayush Tewari, Weipeng Xu, Mario Fritz, Christian Theobalt, and Bernt Schiele. A hybrid model for identity obfuscation by face replacement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 553–569, 2018. 3

[24] Yifan Wu, Fan Yang, and Haibin Ling. Privacy-protective-gan for face de-identification. *arXiv preprint arXiv:1806.08906*, 2018. 3

[25] Simen Thys, Wiebe Van Ranst, and Toon Goedemé. Fooling automated surveillance cameras: adversarial patches to attack person detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 3

[26] Zuxuan Wu, Ser-Nam Lim, Larry Davis, and Tom Goldstein. Making an invisibility cloak: Real world adversarial attacks on object detectors. *arXiv preprint arXiv:1910.14667*, 2019. 3

[27] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. In *Advances in Neural Information Processing Systems*, pages 6103–6113, 2018. 3

[28] Chen Zhu, W. Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. Transferable clean-label poisoning attacks on deep neural nets. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7614–7623. PMLR, 2019. 3

[29] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. 3

[30] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthe-

sis. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. 3

[31] Piotr Bojanowski, Armand Joulin, David Lopez-Paz, and Arthur Szlam. Optimizing the latent space of generative networks. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 599–608. PMLR, 2018. 3

[32] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. *arXiv preprint arXiv:1907.10786*, 2019. 3

[33] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 3

[34] Raj Chetty and John Friedman. A practical method to reduce privacy loss when disclosing statistics based on small samples. *Journal of Privacy and Confidentiality*, 9(2), 2019. 4

[35] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067, 2014. 5

[36] Jaewoo Lee and Chris Clifton. Differential identifiability. In *The 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1041–1049, Beijing, China, Aug. 12-16 2012. 6

[37] Ninghui Li, Wahbeh Qardaji, Dong Su, Yi Wu, and Weining Yang. Membership privacy: A unifying framework for privacy definitions. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, CCS '13, page 889–900, New York, NY, USA, 2013. Association for Computing Machinery. 6