Privacy Safe Representation Learning via Frequency Filtering Encoder

Jonghu Jeong, Minyong Cho, Philipp Benz, Jinwoo Hwang, Jeewook Kim, Seungkwan Lee and Tae-hoon Kim

Deeping Source Inc., 508, Eonju-ro, Gangnam-gu, Seoul, Republic of Korea

Abstract

Deep learning models are increasingly deployed in real-world applications. These models are often deployed on the server-side and receive user data in an information-rich representation to solve a specific task, such as image classification. Since images can contain sensitive information, which users might not be willing to share, privacy protection becomes increasingly important. Adversarial Representation Learning (ARL) is a common approach to train an encoder that runs on the client-side and obfuscates an image. It is assumed, that the obfuscated image can safely be transmitted and used for the task on the server without privacy concerns. However, in this work, we find that training a reconstruction attacker can successfully recover the original image of existing ARL methods. To this end, we introduce a novel ARL method enhanced through low-pass filtering, limiting the available information amount to be encoded in the frequency domain. Our experimental results reveal that our approach withstands reconstruction attacks while outperforming previous state-of-the-art methods regarding the privacy-utility trade-off. We further conduct a user study to qualitatively assess our defense of the reconstruction attack.

Keywords

privacy-preserving machine learning, adversarial representation learning, image frequency filtering

1. Introduction

Service providers, such as Amazon Rekognition and Microsoft Cognitive Services, frequently deploy deep learning models in real-world applications in recent years. The models run on the providers' server can receive and process user information in an information-rich representation to solve a specific task. For example, the users send their face images from their smartphone (client) to the server and receive the processed results, such as face identification. However, the raw images can also contain additional information which users do not consent to reveal or share, violating the users' privacy. An adversary could take over and abuse the images of the users. In one possible attack scenario, adversaries can train a new attacker model (*e.g.* neural network) that retrieves private attributes, such as gender, emotional state, and race. Even the service provider could have malicious intent without the users' knowledge. Hence, an obfuscation method should be used to protect the users' privacy.

For privacy protection with deep learning models, several prior works exist ranging from federated learning [1, 2], split learning [3, 4], differential privacy [5, 6, 7], and homomorphic encryption [8, 9, 10] to instance hiding mechanisms [11, 12, 13, 14], GAN-based obfuscation techniques [15, 16] and adversarial representation learning [17]. Among these works, however, adversarial representation learning (ARL) is the one suitable for the service

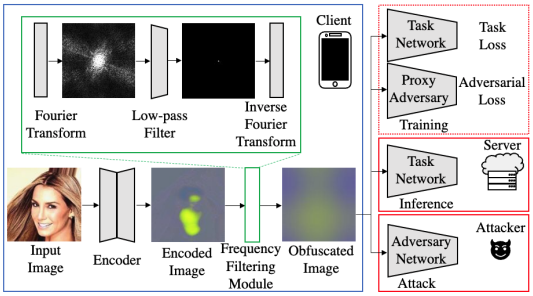

Figure 1: An overview of our proposed method. The proposed method trains an encoder that obfuscates an input image through a neural net and leverages a frequency filtering module to safely transmit a privacy-sensitive image from a client-side to a server-side. The frequency filtering module helps the encoder to remove private information effectively from the image while retaining its utility to be used for a particular task of interest (*utility task*) on the server-side. The encoder is trained with the conventional ARL scheme and then deployed to the client-side. Even with the possibility of data leakage during data transmission, malicious attackers can not abuse the obfuscated image for a privacy breach attack (*privacy task*) since the transmitted data contains information that is only useful for the utility task.

The IJCAI-ECAI-22 Workshop on Artificial Intelligence Safety (AISafety 2022), July 24-25, 2022, Vienna, Austria

jonghu.jeong@deepingsource.io (J. Jeong);
minyong.cho@deepingsource.io (M. Cho);
philipp.benz@deepingsource.io (P. Benz);
jinwoo.hwang@deepingsource.io (J. Hwang);
jeewook.kim@deepingsource.io (J. Kim);
seungkwan.lee@deepingsource.io (S. Lee);
pete.kim@deepingsource.io (T. Kim)

© 2022 © Copyright 2022 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3215/10.pdf

provider to serve users with an obfuscation method. For example, federated learning and instance hiding focus on model training with privacy-safe data, not on inference with obfuscated data [1, 11]. Furthermore, several existing methods suffer under privacy leakage [18, 19, 20], and the degree of computational complexity is too large to be deployed in practice [8, 9, 10]. With ARL, the service provider can train an obfuscator model and deploy it to make data obfuscation possible on the user side [21, 22].

Most previous ARL methods solve the problem of privacy-safe transmission by optimizing 1) utility task loss and 2) proxy adversary task loss [23, 21, 24, 22]. They also introduce specific loss-design formulations, model architecture design, and training schemes. The methods are evaluated quantitatively with performance on both utility and adversary tasks. Note that there usually exists a trade-off between privacy and utility. We use a reconstruction attack, to test the quality of the obfuscation. In a reconstruction attack, a new model is trained that takes the obfuscated representation as an input and outputs the original image. As demonstrated in Figure 2, the original data of existing ARL methods can successfully be recovered from the obfuscated representation. This result suggests that the private information is still encoded in the obfuscated representations.

We present a novel ARL method that leverages frequency filtering, leveraging an extreme low-pass frequency filter (Figure 1). The representation filtering on the frequency domain effectively limits the amount of information to be encoded. Our experimental results show that our approach outperforms previous state-of-the-art methods regarding the privacy-utility trade-off. We also present that our proposed method withstands the reconstruction attack better than existing ARL methods, which are evaluated through visual metrics and a user study.

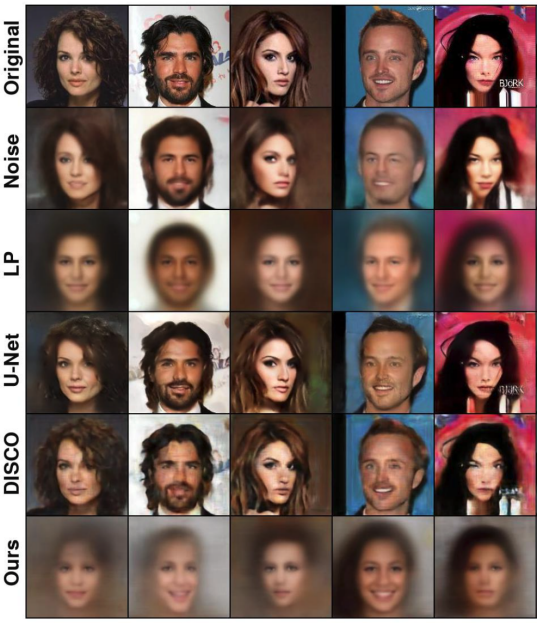

Figure 2: Results of the reconstruction attack with various methods on CelebA. For a successful defense, the reconstructed image should not reveal 1) the identity of the original image and 2) the privacy attribute (in this case, *gender*). Our method successfully defends the reconstruction attack while all other approaches fail. Detailed results are further discussed in Section 5.

2. Related Work

Data-privacy in Computer Vision For privacy-safe data transmission, several approaches have been proposed to tackle the problem of raw image sharing. Federated learning [1, 2] and split learning [3, 4] aim to train a machine learning model without directly sharing raw images through sharing gradients or a processed representation. These methods usually focus on the model training, and not on inference with obfuscated data. Homomorphic encryption [8, 9, 10] attempts to train models on encrypted data, such that the data can be shared in encrypted form and be processed without decryption. Currently, this method suffers from a considerably high computational cost. Instance hiding mechanisms [11, 12, 13, 14] introduce random pixel mixing and clipping algorithm to perturb images. The perturbed images are used only for the training, and the original images are used for the inference which means that there are still potential threats for data breaches when inferring the target.

Adversarial Representation Learning (ARL) Another line of work focuses on the training framework of ARL to address the utility-privacy trade-off of (a) mitigation of privacy disclosure while (b) maintaining task utility. ARL methods have found their application in practical scenarios, such as information censoring [25], learning fair representations [26, 27], the mitigation of information leakage [23, 21, 24], collaborative inference [28, 29, 22], and GAN-based obfuscation techniques [15, 16]. Commonly, the ARL framework consists of three entities: 1) an obfuscator, which transforms input data to a private representation that retains utility, 2) a task model, performing the utility task on the data representation, 3) a proxy adversary, attempting to extract sensitive attributes. Recent approaches [30, 31, 32, 24] represent each component as deep neural networks (DNNs). MaxEnt [23] formulate the ARL problem as an adversarial non-zero-sum game and minimizes the amount of non-utility information, which they quantify through entropy. Adversarial representation learning with non-

linear functions through kernel representation with theoretical guarantees are introduced in [33]. While most of the previous methods represent the obfuscated output as the intermediate feature of a DNN, Bertran et al. [21] leverages domain-preserving transformations, *i.e.* images to images. Above mentioned ARL methods mainly focused on designing special loss functions or model architectures. To the best of our knowledge, our method is the first ARL method that focuses on the effective encoding of privacy-safe representation in the frequency domain.

There are three common attacks on privacy in machine learning. The first is the membership inference attack [34], which attempts to infer whether a data sample is used for the machine learning model training. This attack is more related to the attack on the server-side model, not the transmitted data. The second is the inversion attack [35] which attempts to infer raw data from processed representation. This is the same attack scenario as the aforementioned reconstruction attack. The last is the information leakage attack [23], for which adversaries attempt to infer privacy-related information from obfuscated representation. In this work the inversion attack and the information leakage attack are considered as they are potential threats to transmitted privacy-sensitive images.

Frequency Perspective in Computer Vision Prior works have explored the behavior of DNNs from a frequency perspective. Overall, there is solid evidence that both high-frequency features and low-frequency features can be helpful for classification [36, 37]. It has been demonstrated that DNNs have an increased bias toward texture compared to the object's shape [38]. On the other hand, DNNs trained only on low-pass filtered images also generalize well and are capable of achieving high accuracies [36]. Yin et al. [36] shows that adversarial training and Gaussian data augmentation shift DNNs towards utilizing low-frequency information in the input. Wang et al. [37] points out that convolutional neural networks (CNNs) mainly exploit high-frequency components. Similarly, Abello et al. [39] find that mid or high-level frequencies are disproportionately critical for CNNs. Ilyas et al. [40] also show similar findings that human-imperceptible features with high-frequency properties are sufficient for the model to exhibit high generalization capability.

In this work, we leverage previous insights that information can be encoded in different frequency ranges of images. We propose encoding information in the low-frequency band of images to securely transfer them between different parties.

3. Problem Formulation

We consider an image dataset $x \sim \mathcal{X} \in \mathbf{R}^{H\times W\times 3}$, where H and W represent width and height, respectively, along with a number of various attributes $y \sim \mathcal{Y}$. Some of the attributes are private attributes $y_p \sim \mathcal{Y}_p$ and some are utility attributes $y_t \sim \mathcal{Y}_t$, such that $\mathcal{Y} = \mathcal{Y}_t \cup \mathcal{Y}_p$. Given a utility task model f_t, we search for an intermediate representation $\hat{x}$, from which f_t can infer the utility attributes, but not the privacy attributes. This transformation can also be represented through a DNN o, termed obfuscator, resulting in $o(x) = \hat{x}$. Note that in prior works, the intermediate representation $\hat{x}$ was often represented as a feature map differing in shape from the original input images. However, similar to [21], we represent the obfuscated representation in the same shape as the original input image. This setting allows us to leverage existing image transformation techniques, such as transforming them into a 2D Fourier representation. Additionally, this form of intermediate representation allows us to analyze the representations visually.

Threat Model Given the above problem formulation, an attacker can attempt to retrieve information about the private attributes from the intermediate representation. This can be realized either by directly inferring private information from the intermediate representation (*information leakage attack*) or through the reconstruction of the original input images from the intermediate representations (*reconstruction attack*). In the *information leakage attack* scenario an attacker is able to obtain data pairs consisting of the corresponding intermediate representation and their respective private attributes $\{\hat{x}, y_p\}$. In this scenario an attacker can attempt to train a model f_a, which leaks the private information from the representations $f_a(\hat{x}) = y_p$. In the *reconstruction attack*, given image pairs of the original image and the intermediate representation $\{x, \hat{x}\}$ the attacker attempts to obtain a model f_r which retrieves the original image x from the intermediate representation $f_r(\hat{x}) = x$. In this work, we represent both attacker models f_a and f_r through DNNs, since they are proven to be powerful for image processing tasks.

4. Methodology

Fourier Transformation Fourier transform is a common tool to perform frequency analysis [41]. We consider the 2D discrete Fourier transformation $\mathcal{F} : \mathbf{R}^{W\times H} \rightarrow \mathbf{C}^{W\times H}$ and the inverse Fourier transformation as $\mathcal{F}^{-1}$. After applying $\mathcal{F}$ on an image, low frequencies are located in the center of a Fourier image, while high frequencies are located toward the boundaries. For low-pass filtering, we set all frequency components outside of a central

circle with radius r in the frequency domain to zero and apply $\mathscr{F}^{-1}$ afterward. We normalize the radius to be in the range of [0, 1] by considering the center of the image as 0 and the corner as 1. We indicate low-pass filtering as *LP*.

Frequency Obfuscation We depict our proposed methodology in Figure 1. Given an input image, the objective is to obfuscate the image to achieve the best privacy-utility trade-off. Our obfuscator module consists of an encoder architecture followed by frequency-filtering. We choose the commonly used U-Net [42] architecture as our encoder and pass the original image through it. Formally, we express this as $e(x)$, where we indicated the encoder with e. The subsequent frequency filtering is realized via a low-pass filter $LP(e(x))$. This procedure completes the generation of the intermediate representation through the obfuscator $\hat{x} = o(x) = LP(e(x))$. During obfuscator training, we leverage a task model and a proxy adversary. The objective of the task model is to predict the utility attribute from the intermediate representation. The respective task loss can be calculated with $l_t = \mathbb{E}[\mathscr{L}_t(f_t(o(x)), y_t)]$, where $\mathscr{L}_t$ indicates the task loss function, which is the cross-entropy function in our setup. The objective of proxy adversary model is to leak the privacy attribute from the intermediate representation. The proxy adversary loss can be calculated as $l_p = \mathbb{E}[\mathscr{L}_p(f_a(o(x)), y_p)]$, where $\mathscr{L}_p$ indicates the privacy loss function, which is also represented as the cross-entropy function. The obfuscator loss is represented as $l_o = l_t - l_p$.

Similar to the scenario introduced in DISCO [22] a practical application scenario of our proposed approach is when the obfuscator module is present on a trusted client device, which sends the intermediate feature representations to a server. Since an adversary can intercept the communication between client and server, or the server can also be malicious, we consider the server-side an untrusted entity.

Evaluation Protocol In the following, we outline our evaluation protocol. We follow the general ARL evaluation protocol [22, 23]. Given an image classification dataset, we specify certain classes as the utility and privacy tasks, respectively. Based on the chosen tasks, following our proposed method we obtain an obfuscator and a utility task model. Note that this includes training proxy adversaries. After training, we evaluate the models on the utility task and report the accuracy as *utility*. Then we freeze the weights of the obfuscator and train an adversary model to predict the privacy attributes and report the accuracy as *privacy*. To assess the privacy-utility trade-off, we measure their difference (Δ).

Additionally, we report the *performance bounds*. Theoretically, the utility (higher the better) is upper bounded by 100%. In practice, however, we consider the upper bound as the utility performance of a ResNet18 [43] model trained on the original images. For privacy (lower the better), we consider the lower bound as the random guess for the privacy attribute.

We also perform a reconstruction attack on the obfuscated images to recover corresponding original images. We evaluate the reconstruction attacks quantitative and qualitatively by calculating similarity scores between the original and reconstructed images and conducting a user study on the reconstructed images.

5. Experiments

5.1. Setup

Datasets We conduct experiments on CelebA [44], FairFace [45], and CIFAR10 [46]. Following the utility and privacy task setting from DISCO [22], we set "Smiling" as the utility attribute and "Male" as the privacy attribute for CelebA, "Gender" as the utility attribute, and "Race" as the privacy attribute for FairFace. For CIFAR10, the utility task is defined as classifying living objects (*e.g.* "bird", "cat", *etc.*) or non-living objects (*e.g.* "airplane", "automobile", *etc.*) and the privacy task as classifying the separate 10 classes.

Implementation details The encoder is a lightweight variant of U-Net [42], with 4× fewer intermediate feature channels than the original version. We use an extreme low pass filter with radius, $r = 0.01$ for CelebA and FairFace, and $r = 0.05$ for CIFAR10. We apply a center-circled filter, which can adjust the level of obfuscation by changing its radius (bandwidth). Section 6.2 discusses the effect of the radius. We normalize the radius by the length from the filter's center to the corner to make the value in the range $[0, 1]$. For both the utility and privacy task models, we use ResNet-18 [43], and use the same dataset for training both models. We use Adam [47] optimizer for all 3 models with learning rate 10^{-4} for U-Net and 10^{-3} for the ResNet-18 models. We evaluate the top-1 accuracy for both utility and privacy tasks. We used the lightweight U-Net as the reconstructor for the reconstruction attack. The reconstructor adversary is trained with the MSE loss between the original and the reconstructed images. The reconstructed images are evaluated using MSE, L_1, SSIM [48], MS-SSIM [49], PSNR [50], and LPIPS [51]. MSE, L_1, and PSNR compare the images pixel-wise while SSIM and MS-SSIM compare structural similarity (*e.g.*, brightness, contrast) between the images. LPIPS uses a pre-trained neural network's feature map for comparison. These metrics are commonly used for comparing the similarity between images [22, 24, 52] and we consider them

Method	Fairface			CelebA			CIFAR10		
	Privacy ↓	Utility ↑	Δ ↑	Privacy ↓	Utility ↑	Δ ↑	Privacy ↓	Utility ↑	Δ ↑
Perf. Bounds	19.03	90.16	71.13	57.43	93.32	35.89	10.00	98.79	78.79
Noise	42.61	74.33	31.72	91.71	85.38	-6.33	54.37	87.77	33.40
LP	31.93	64.77	32.84	76.52	63.69	-12.83	47.05	85.76	38.71
U-Net	51.52	86.40	34.88	87.21	93.12	5.91	85.05	95.45	10.40
DISCO	19.00	81.50	62.50	61.20	91.00	29.80	22.30	91.98	69.68
Ours	23.63	89.67	**66.04**	61.60	93.27	**31.67**	22.58	92.95	**70.37**

Table 1
Evaluation of the privacy-utility trade-off. The upper/lower arrow suggests that each value is higher/lower the better. Our method shows the biggest gap between privacy and utility accuracy among all the datasets. Note that the privacy accuracy is based on the newly trained adversary model which is trained with the fully trained and frozen obfuscation model.

as a proxy of human vision.

Compared Methods We compare our method with various baselines. As a simple baseline obfuscator, we add Gaussian noise sampled from $\mathcal{N}(0, \sigma^2)$ to the input image while obeying the image range of pixels in the range $[0, 1]$. We indicate this method with *Noise*. We use $\sigma^2 = 4$ for CelebA and FairFace and $\sigma^2 = 0.64$ for CIFAR10, which obfuscate the images sufficiently. To investigate the sole effect of the low-pass filtering, we apply only the low-pass filter to the raw images. We name this baseline as *LP*. Complementary, we also compare the U-Net without the low-pass filtering module as an obfuscator. We call it *U-Net*. This setup is similar to DeepObfuscator [24] which uses an encoder, task model, and a proxy adversary. However, since DeepObfuscator has not open-sourced their code, we used our U-Net encoder as a method to compare. Finally, we compare our method to the state-of-the-art ARL method *DISCO* [22], which selectively removes features via channel pruning in the latent space.

5.2. Results

Table 1 shows a comparison between the privacy and utility accuracy of each obfuscation method. Our method resulted in the highest gap between utility and privacy accuracy on all datasets. For the methods without encoder (*i.e. Noise* and *LP*), the accuracy for both utility and privacy decreases compared to training with the original image since these methods obfuscate images without any prior knowledge of the tasks. These methods cannot selectively restrict information for high utility and low privacy leakage. *U-Net* showed high utility accuracy but failed to defend against the privacy attack, although it is trained with a proxy adversary. We conjecture that simply taking the guidance of the proxy model loss is not enough for the encoder to learn to restrict information.

Method	MSE ↑	L_1 ↑	SSIM ↓	MS-SSIM ↓	PSNR ↓	LPIPS ↑
Noise	584.88	16.97	0.6017	0.7776	20.46	0.3714
LP	1889.15	32.10	0.4632	0.5390	15.37	0.5537
U-Net	390.34	13.81	0.7505	0.8839	22.22	0.1809
DISCO	567.17	15.94	0.5765	0.7611	20.60	0.4351
Ours	**3689.50**	**48.08**	**0.4240**	**0.4728**	**12.47**	**0.6145**

Table 2
Similarity scores between the original image and the reconstructed ones on CelebA. The upper/lower arrow suggests that each value is higher/lower the better, respectively. Our approach shows the best dissimilarity among all the metrics.

Our method is a combination of *LP* and *U-Net*, and learns to encode a representation into the restricted bandwidth, which is limited by the frequency filtering module. This limited bandwidth helps the encoder to learn how to extract utility information effectively and remove privacy attributes to fully leverage the limited bandwidth. While the same data is used to train both utility and adversary models, which is a generous and unrealistic condition for the attackers to have, we found the adversary model performed poorly. *DISCO* shows the lowest privacy accuracy among all the datasets. However, the utility accuracy is lower than our method, so the utility-privacy gap is smaller than ours.

In terms of the visual quality, our obfuscated representations appear as simple globs of color, making them unrecognizable to human observers (Figure 1). The obfuscated representations from other methods also appear obfuscated to the human eye. However, applying our best effort reconstruction attack, it is possible to reconstruct the original image or infer the privacy attribute (*i.e.* gender) from reconstructed images. (Figure 2). The reconstructed images from our method successfully defend identity reconstruction and privacy attribute leakage, with the reconstructed images all being relatively similar to each other. The quantitative results of the reconstruc-

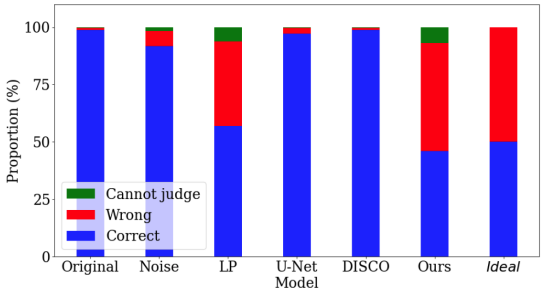

Figure 3: Result of the user study on reconstructed images of CelebA. We asked the participants to classify *gender* (male/female) on 180 images such as Figure 2. The participants correctly distinguished the gender of original images and reconstructed images from the three methods (*Noise*, *U-Net*, and *DISCO*) with more than 90% accuracy. Our method and *LP* effectively confused the participants with gender-neutral faces (45.83% and 56.9% of correct answers ratio each), while ours is slightly better than *LP* in terms of obfuscation. We also plot the ideal case of the user study to show our method's near-perfect superiority against the reconstruction attack.

tion attack in Table 2 further confirm this since all scores achieve the best results in terms of dissimilarity for our approach. We note that an adversary model trained with the reconstructed images to infer the privacy attributes performs worse than directly training the model with the obfuscated images since the reconstructed images are processed from the obfuscated images.

5.3. User Study

We present a user study to show our method's robustness against the reconstruction attack on CelebA. Since the privacy task for the dataset is gender classification, the reconstructed image's gender should not be correctly classified by a human observer if the obfuscation is successful. To conduct the experiment, we randomly sampled 30 images (15 for male and 15 for female), for which ResNet18 classifies the gender correctly. By doing so, we balanced each class and addressed the ambiguity of the labels to prevent unfair results. Then, we obfuscated the images using each of the techniques and reconstructed them with their respective attacker models from Section 5.1. Examples of reconstructed images are shown in Figure 2. We presented 180 reconstructed images to a group of people and asked them to identify whether the person in the reconstructed image is male, female, or cannot be judged. We provided the last option to let the users skip the examples that are hard to judge. The test subjects were randomly selected and consist of 30 people who live in Seoul, South Korea, and are in their 20s and 30s.

As shown in Figure 3, people correctly identify the gender for the original images and the reconstructed ones from the methods *Noise*, *U-Net*, and *DISCO*. More than 90% of answers were correct for the three methods. *LP* showed a relatively low correct ratio (56.9%) and a high "cannot judge" ratio (6.19%). Our method showed the best for both, the lowest correct ratio of 45.83% and the highest "cannot judge" ratio of 7.02%. We consider the 50% ratio for each "correct" and "wrong" answer as a random guess since the labels for the test datasets are balanced. Additionally, we note that "cannot judge" can be considered as a random guess since without this option, the users would have done a random choice. The results indicate that our approach successfully protects against reconstruction attacks in terms of human vision. The results also align with the quantitative results (Table 2). In terms of obfuscation, our method shows the best results, followed by *LP*. It reconfirms the usefulness of our architecture design, the combination of the encoder and the frequency filtering module.

Method	Privacy ↓	Utility ↑	Δ ↑
HP (r=0.80)	26.19	89.03	62.84
HP (r=0.85)	26.28	89.13	62.85
HP (r=0.90)	28.94	88.00	59.06
HP (r=0.95)	24.96	88.12	63.16
HP (r=0.99)	19.03	52.88	33.85
LP (r=0.01)	23.63	89.67	**66.04**

Table 3
The privacy-utility gap of the high-pass filtering module on FairFace. Our low-pass filtering module shows the best privacy-utility gap compared to the high-pass filter with the various filter radii.

6. Ablation Study

6.1. High-pass filter

Previously, we presented the effect of the low-pass frequency filtering module on ARL. The module appropriately limits the amount of encoded information in the obfuscated image. It retains the information at a low-frequency range. Using a high-pass filter, we can leverage the same intuition, by limiting the information to be encoded in the high-frequency bandwidth. However, in the following, we will present results indicating that the low-pass filter is the superior method to use.

We conduct the same experiment from Section 5.2 on FairFace with a high-pass filtering module for 5 radii (0.80, 0.85, 0.90, 0.95, 0.99). Contrary to the low-pass filtering, the filter removes frequencies inside the filter radius, which leads to a radius of 0.99 as the most extreme high-pass filter. We call this method *HP*.

The respective results are presented in Table 3. As the filtering gets more extreme, the utility accuracy decreases

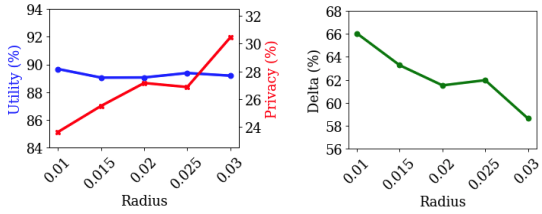

Figure 4: (Left) Privacy and utility accuracy under each radius of the low-pass filter. The experiments are conducted on FairFace. (Right) Privacy-utility trade-off. Delta represents the performance gap between utility and privacy.

together with the privacy accuracy. The table also shows that our approach with a low-pass filter from Table 1 outperforms all results from the high-pass filter regarding the privacy-utility gap. The best privacy-utility gap with the high-pass filter is 63.16% with a radius of 0.95, which is 2.88%p lower than for the approach with low-pass filtering. It has been demonstrated that DNNs can learn from low-pass filtered images more efficiently than high-pass filtered ones [36]. Especially with the extreme high-pass (r=0.99), the model did not learn for both, the utility and privacy tasks.

Furthermore, from a practical point of view, we need to reduce the size of the obfuscated image to reduce the cost of transmission or storage. The most commonly used JPEG compression algorithm leverages the filtering of high frequency. If we use a high-pass filter ARL method, encoded information in the high-frequency range would be lost. To this end, encoding information into the low-frequency range is more suitable than the opposite to utilize the conventional compression algorithms further.

6.2. The effect of filter radius

One of the key points of our proposed method is the frequency filtering module. The module has only one parameter to consider, the filter's radius. To gain insight into choosing the parameter, we conducted experiments with various radii. The same experiment from Section 5 on FairFace is done with 5 radii (0.01, 0.015, 0.02, 0.025, 0.03). The radius of 0.01 is the most extreme low-pass filter.

Figure 4 (left) shows a trend of consistent utility accuracy and increasing privacy accuracy. The utility accuracies are around 89% with a small variance. The privacy accuracies show an increasing tendency from 23.64% to 30.45% as the radius increases. It leads the privacy-utility gap to decrease (Figure 4, right).

The increased privacy accuracy aligns with our intuition of limiting information in the obfuscated representation. The wider radius allows the representation to have more information, leading the adversary to exploit it for a privacy attack easily. Note that the utility accuracy did not decrease even with the harshest filter. We speculate that the extremely low-pass filtered representation is enough for these specific utility tasks. Figure 4 and Table 3 confirm that the radius is a crucial factor of privacy and utility accuracy. Thus the radius is a hyperparameter that should be tuned based on the privacy-utility gap.

7. Conclusion

This work proposes a novel ARL method based on frequency filtering, which is robust to privacy leakage attacks while maintaining task utility. Our experiments suggest that a combination of neural-net encoder and low-pass filter improves ARL training for the quantitative and qualitative metrics. The method outperforms other compared methods for the quantitative measure of privacy-utility trade-off and reconstruction attack (Section 5). Our user study suggests that the proposed method effectively defends against reconstruction attacks (Section 5.3). The ablation experiments justified the use of a low-pass filter and also showed that the filter radius adjusts the privacy-utility trade-off (Section 6).

For future work we consider the optimization of the client-side model to reduce the computation burden by using a lightweight architecture such as MobileNetV3 [53]. Furthermore, an adaptive selection of the frequency-filtering hyperparameter might increase the utility accuracy and decrease the privacy accuracy.

References

[1] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, D. Bacon, Federated learning: Strategies for improving communication efficiency, arXiv preprint arXiv:1610.05492 (2016).

[2] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., Advances and open problems in federated learning, Foundations and Trends® in Machine Learning (2021).

[3] O. Gupta, R. Raskar, Distributed learning of deep neural network over multiple agents, Journal of Network and Computer Applications (2018).

[4] P. Vepakomma, O. Gupta, T. Swedish, R. Raskar, Split learning for health: Distributed deep learning without sharing raw patient data, arXiv preprint arXiv:1812.00564 (2018).

[5] C. Dwork, Differential privacy: A survey of results, in: International conference on theory and applications of models of computation, 2008.

[6] Z. Ji, Z. C. Lipton, C. Elkan, Differential privacy and machine learning: a survey and review, arXiv preprint arXiv:1412.7584 (2014).

[7] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, L. Zhang, Deep learning with differential privacy, in: ACM SIGSAC conference on computer and communications security, 2016.

[8] E. Hesamifard, H. Takabi, M. Ghasemi, Cryptodl: Deep neural networks over encrypted data, arXiv preprint arXiv:1711.05189 (2017).

[9] C. Juvekar, V. Vaikuntanathan, A. Chandrakasan, {GAZELLE}: A low latency framework for secure neural network inference, in: USENIX Security Symposium, 2018.

[10] K. Nandakumar, N. Ratha, S. Pankanti, S. Halevi, Towards deep neural network training on encrypted data, in: Conference on Computer Vision and Pattern Recognition Workshops (CVPR-W), 2019.

[11] Y. Fu, H. Wang, K. Xu, H. Mi, Y. Wang, Mixup based privacy preserving mixed collaboration learning, in: International Conference on Service-Oriented System Engineering (SOSE), 2019.

[12] Y. Huang, Z. Song, K. Li, S. Arora, Instahide: Instance-hiding schemes for private distributed learning, in: International Conference on Machine Learning (ICML), 2020.

[13] M. Shin, C. Hwang, J. Kim, J. Park, M. Bennis, S.-L. Kim, Xor mixup: Privacy-preserving data augmentation for one-shot federated learning, arXiv preprint arXiv:2006.05148 (2020).

[14] E. Borgnia, J. Geiping, V. Cherepanova, L. Fowl, A. Gupta, A. Ghiasi, F. Huang, M. Goldblum, T. Goldstein, Dp-instahide: Provably defusing poisoning and backdoor attacks with differentially private data augmentations, arXiv preprint arXiv:2103.02079 (2021).

[15] T.-h. Kim, D. Kang, K. Pulli, J. Choi, Training with the invisibles: Obfuscating images to share safely for learning visual recognition models, arXiv preprint arXiv:1901.00098 (2019).

[16] C. Xu, J. Ren, D. Zhang, Y. Zhang, Z. Qin, K. Ren, Ganobfuscator: Mitigating information leakage under gan via differential privacy, Transactions on Information Forensics and Security (2019).

[17] J. Donahue, K. Simonyan, Large scale adversarial representation learning, Advances in Neural Information Processing Systems 32 (2019).

[18] L. Lyu, H. Yu, Q. Yang, Threats to federated learning: A survey, arXiv preprint arXiv:2003.02133 (2020).

[19] D. Pasquini, G. Ateniese, M. Bernaschi, Unleashing the tiger: Inference attacks on split learning, in: ACM SIGSAC Conference on Computer and Communications Security, 2021.

[20] O. Li, J. Sun, X. Yang, W. Gao, H. Zhang, J. Xie, V. Smith, C. Wang, Label leakage and protection in two-party split learning, arXiv preprint arXiv:2102.08504 (2021).

[21] M. Bertran, N. Martinez, A. Papadaki, Q. Qiu, M. Rodrigues, G. Reeves, G. Sapiro, Adversarially learned representations for information obfuscation and inference, in: International Conference on Machine Learning (ICML), 2019.

[22] A. Singh, A. Chopra, E. Garza, E. Zhang, P. Vepakomma, V. Sharma, R. Raskar, Disco: Dynamic and invariant sensitive channel obfuscation for deep neural networks, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[23] P. C. Roy, V. N. Boddeti, Mitigating information leakage in image representations: A maximum entropy approach, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2586–2594.

[24] A. Li, J. Guo, H. Yang, F. D. Salim, Y. Chen, Deepobfuscator: Obfuscating intermediate representations with privacy-preserving adversarial learning on smartphones, in: International Conference on Internet-of-Things Design and Implementation, 2021.

[25] H. Edwards, A. Storkey, Censoring representations with an adversary, in: International Conference on Learning Representations (ICLR), 2016.

[26] C. Louizos, K. Swersky, Y. Li, M. Welling, R. Zemel, The variational fair autoencoder (2016).

[27] D. Madras, E. Creager, T. Pitassi, R. Zemel, Learning adversarially fair and transferable representations, in: International Conference on Machine Learning (ICML), 2018.

[28] P. Vepakomma, A. Singh, O. Gupta, R. Raskar, Nopeek: Information leakage reduction to share activations in distributed deep learning, in: 2020 International Conference on Data Mining Workshops (ICDMW), 2020.

[29] S. A. Osia, A. S. Shamsabadi, S. Sajadmanesh, A. Taheri, K. Katevas, H. R. Rabiee, N. D. Lane, H. Haddadi, A hybrid deep learning architecture for privacy-preserving mobile analytics, IEEE Internet of Things Journal (2020).

[30] F. Pittaluga, S. Koppal, A. Chakrabarti, Learning privacy preserving encodings through adversarial training, in: Winter Conference on Applications of Computer Vision (WACV), 2019.

[31] S. Liu, J. Du, A. Shrivastava, L. Zhong, Privacy adversarial network: representation learning for mobile data privacy, ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (2019).

[32] Z. Wu, Z. Wang, Z. Wang, H. Jin, Towards privacy-preserving visual recognition via adversarial training: A pilot study, in: European Conference on Computer Vision (ECCV), 2018.

[33] B. Sadeghi, R. Yu, V. Boddeti, On the global optima of kernelized adversarial representation learning, in: International Conference on Computer Vision (ICCV), 2019.

[34] R. Shokri, M. Stronati, C. Song, V. Shmatikov, Membership inference attacks against machine learning models, in: Symposium on security and privacy (SP), 2017.
[35] M. Fredrikson, S. Jha, T. Ristenpart, Model inversion attacks that exploit confidence information and basic countermeasures, in: ACM SIGSAC conference on computer and communications security, 2015.
[36] D. Yin, R. G. Lopes, J. Shlens, E. D. Cubuk, J. Gilmer, A fourier perspective on model robustness in computer vision, in: Advances in neural information processing systems (NeurIPS), 2019.
[37] H. Wang, X. Wu, Z. Huang, E. P. Xing, High-frequency component helps explain the generalization of convolutional neural networks, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
[38] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, W. Brendel, Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness., in: International Conference on Learning Representations (ICLR), 2019.
[39] A. A. Abello, R. Hirata, Z. Wang, Dissecting the high-frequency bias in convolutional neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 863–871.
[40] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, A. Madry, Adversarial examples are not bugs, they are features, Advances in neural information processing systems (NeurIPS) (2019).
[41] J. S. Lim, Two-dimensional signal and image processing, Englewood Cliffs (1990).
[42] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, 2015.
[43] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Conference on computer vision and pattern recognition (CVPR), 2016.
[44] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: International Conference on Computer Vision (ICCV), 2015.
[45] K. Karkkainen, J. Joo, Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation, in: Winter Conference on Applications of Computer Vision (WACV), 2021.
[46] A. Krizhevsky, Learning multiple layers of features from tiny images, Technical Report, 2009.
[47] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).
[48] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, Transactions on Image Processing (2004).
[49] Z. Wang, E. P. Simoncelli, A. C. Bovik, Multiscale structural similarity for image quality assessment, in: The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, volume 2, Ieee, 2003, pp. 1398–1402.
[50] A. Horé, D. Ziou, Image quality metrics: Psnr vs. ssim, in: International Conference on Pattern Recognition, 2010.
[51] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 586–595.
[52] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila, Analyzing and improving the image quality of stylegan, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.
[53] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, et al., Searching for mobilenetv3, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019.