Deep Poisoning: Towards Robust Image Data Sharing against Visual Disclosure

Hao Guo[1], Brian Dolhansky[2], Eric Hsin[2], Phong Dinh[2], Cristian Canton Ferrer[2], Song Wang[1]
[1] University of South Carolina, [2] Facebook AI

Abstract

Due to respectively limited training data, different entities addressing the same vision task based on certain sensitive images may not train a robust deep network. This paper introduces a new vision task where various entities share task-specific image data to enlarge each other's training data volume without visually disclosing sensitive contents (e.g. illegal images). Then, we present a new structure-based training regime to enable different entities learn task-specific and reconstruction-proof image representations for image data sharing. Specifically, each entity learns a private Deep Poisoning Module (DPM) and insert it to a pre-trained deep network, which is designed to perform the specific vision task. The DPM deliberately poisons convolutional image features to prevent image reconstructions, while ensuring that the altered image data is functionally equivalent to the non-poisoned data for the specific vision task. Given this equivalence, the poisoned features shared from one entity could be used by another entity for further model refinement. Experimental results on image classification prove the efficacy of the proposed method.

1. Introduction

Deep networks, e.g. Convolutional Neural Networks (CNNs), have achieved state-of-the-art results on many computer vision tasks [12, 13, 18, 24, 27, 39, 40], which can be used in many critical production systems [2, 7, 25]. Traditionally, training of these networks requires large-scale task-specific datasets with many images [44]. However, for certain vision tasks with restricted images, one entity (institution/company) may not properly collect sufficient images for robust deep model learning. To deal with this, various entities could enlarge training data volume by sharing image data with each other. Nevertheless, the task-specific images may contain overly sensitive visual contents that should not be spread, making the sharing of raw images inappropriate. Thus, this paper introduces a new task on vision integrity of preventing image data sharing from visually disclosing sensitive image contents.

The introduced task of image data sharing against visual disclosure includes two objectives: 1) the image data (e.g. convolutional features) shared by different entities should be same task-specific, so that one entity could utilize data shared from various entities for the model learning; 2) one entity should not be able to visually observe the image contents by recovering images from the shared data, e.g. image reconstruction – defined as visual disclosure of sensitive image contents. There are broad practical applications of this task. For instance, deep model based child exploitation image (CEI) detection and terrorist propaganda image detection help comfort the online social community. The model learning requires a large volume of training images, which may not be easily collected by an individual company, due to the highly sensitivity of the images. The introduced task allows various companies to share image data for sensitive vision tasks without spreading uncomfortable images, which may also violate laws.

Different from the task of privacy preserving, which disentangles utility information from privacy information (e.g. facial expression v.s. face identity) on a constant dataset, the introduced task focuses on a collaborative dataset, allowing collaborators (entities) securely sharing and using the image data to refine deep model learning without undesired image spreading. Privacy-preserving methods usually convert the raw images to image representations by a certain operation, such as anonymization [21, 48], encryption [8, 16, 57], privacy-preserving representation learning [43, 5, 54, 37, 3]. For the introduced task, shared image data should be in the same feature space (the same task specific), which requires the above operation being shared among collaborators. Thus, one entity may reverse the operation, e.g. like a black box, to reconstruct the original images from data shared by other entities, leading to visual disclosure of image contents. Besides, comparing with federated learning [29] using extra hardware for simultaneous model learning from multiple entities without sharing image data, the introduced task enables image data sharing without visual disclosure and allows various entities to train models individually, leading to enlarged applications.

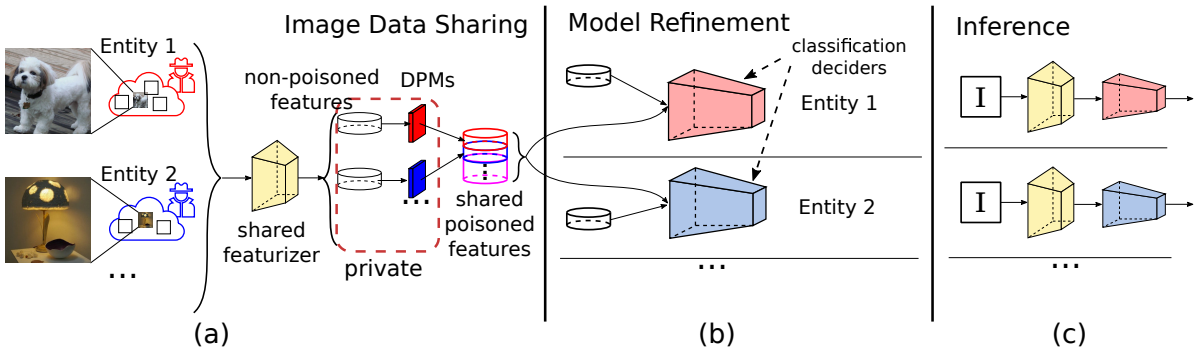

Figure 1. An illustration of image data sharing for collaboratively addressing the same task. (a) Each entity uses an individual private DPM for feature poisonining – training details discussed later; (b) One entity uses its own data and image data shared by others to refine respective models, e.g. classification decider; (c) Using the refined models for image inference.

This paper focuses on the regime of preventing visual disclosure of sensitive image contents (e.g. private faces, illegal images) during image data sharing for a particular vision task. To illustrate, we take the image classification as an example of the vision task to be addressed – various entities collaborate on training models on image classification by sharing their respectively collected image data to each other. First, one entity pre-trains a specific image classification network, and split the network at a specific point into the image featurizer (consisting of certain starting layers) and the classification decider (the remaining layers), similar to [33, 34]. Entities can use the shared featurizer to produce feature maps for their images. Instead of sharing these deep features, which can be easily reconstructed to the original images by reversing the featurizer, we design a regime that requires each entity to learn a respective Deep Poisoning Module (DPM) with various architectures to poison the deep features. By means of adversarial training, each DPM is optimized to ensure that the poisoned features are functionally equivalent to the original features for image classification, while they can not be recovered to the original images by reversing the featurizer. As shown in Fig. 1, with keeping the DPM in private – a partial-release strategy, the poisoned features shared from one entity can be used by another entity for refining the classification decider. Meanwhile, without accessing to the DPM, others can not reconstruct images from the poisoned features, which ensures the image data sharing against visual disclosure.

Finally, we conduct experiments to verify that the proposed DPM can prevent visual disclosure of sensitive contents during the image data sharing with a minimal loss in image classification performance. By simulating the process of collaborators exploiting the shared image data from other entities, our experiments demonstrate that the proposed framework is an effective way for image data sharing for collaboration without visually disclosing sensitive contents.

2. Related Work

To protect information privacy, privacy-preserving data publishing (PPDP) [6, 56] has been studied for a long time. It collects a set of individual records and publishes the records for further data mining, without disclosing individual attributes such as gender, disease, or salary [1, 23, 31, 36, 49]. Existing work on PPDP mainly focuses on anonymization [4, 10, 52] and data slicing [20]. While it usually handles individual records related to identification, it is not explicitly designed for general high-dimensional data, such as images.

Recently, the privacy issue has been attracting an increasing attention by the computer vision and deep learning community. It is an important task for vision ethics. For example, MS-Celeb-1M [11, 35] and Duke MTMC [42] were withdrawn from public release due to privacy issues. Existing methods on preserving privacy in images and videos usually alter the images or learn image representations so that the private information is degraded in the data. Intuitive perturbations, such as blurring and blocking [22, 28, 30], modify the images to reduce privacy information. De-identification methods [21, 48] partially alter images, for example by obfuscating faces. Encryption-based approaches [8, 16, 57] train models directly on encrypted data. Optimal transformations for producing super low-resolution images or videos in order to avoid leaking sensitive information are learned in [45]. Inspired by Generative Adversarial Nets (GAN) [9], adversarial approaches [17, 19, 32, 38, 41, 47, 50, 53, 43, 5, 54, 37, 3] learn deep obfuscators for images or corresponding convolutional features.

Even though the above methods achieve promising results on preserving privacy, they can not address the introduced task of image data sharing against visual disclosure. To enable that one entity utilize image data shared from various entities, they should apply the same privacy-preserving

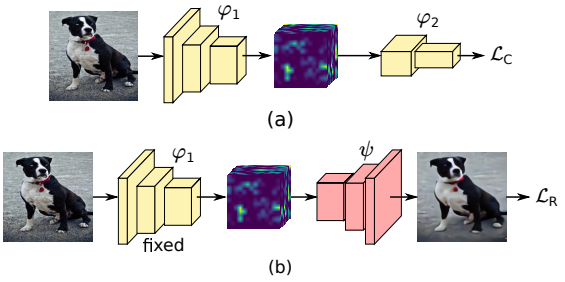

Figure 2. Illustration of the initial pre-training.

operation to convert images to the same feature space. However, the information to be shared in the image data and the visual depictions to be protected may not be disentangled efficiently as privacy-preserving methods distinguish utility and privacy. Thus, the image data shared by one entity could be reconstructed to visual depictions (the original images) by reversing the shared privacy-preserving operation (e.g. by means of reversing a black box), which still leads to undesired image spreading.

The more similar task to the introduced one is the federated learning [29] – multiple entities simultaneously update a model without sharing data to each other. Different from the introduced task, it usually learns a model on a third-party hardware, e.g. an extra server, or from cloud side. Image data are not required to be shared between entities (clients). This well addresses the data island issue, but limits the independence of each entity. Compared with federated learning, this paper proposes a new solution for various entities collaborating with each other to address the same vision task based on image data sharing with not visually disclosing sensitive image contents.

3. Proposed Regime

3.1. Overview

The overall image data sharing regime, proposed in this paper, for collaborations between various entities consists of three steps:

1 An initial deep network is pre-trained by one entity for a specific vision task, from which a featurizer is extracted to convert images to convolutional features;

2 A private Deep Poisoning Module is learned by each entity to poison the image features for image data sharing with not visually disclosing sensitive contents;

3 Each entity exploits image data shared from others for deep model refinement to better address the specific vision task.

3.2. Step 1: Initial Pre-training

The collaboration between entities aims to expand the volume of training images for each entity addressing the specific vision task, i.e. image classification. To achieve this goal, we expect that all entities share the same operation to converting raw images to certain image representations. At the beginning of the collaboration, one of the entities with the most of training samples pre-trains a classification network Φ based on a specific architecture, such as VGGNet [46], ResNet [13], ResNeXt [55] or DenseNet [15], and the conventional classification loss is adopted for model optimization:

$$\mathcal{L}_C(x, y_i) = -\log\left(e^{p(\Phi(x)=y_i)} / \sum_j e^{p(\Phi(x)=y_j)}\right), \quad (1)$$

where y_i represents the annotation of the input image x. As illustrated in Fig 2(a), the pre-trained model Φ is divided into two sequential modules: the image featurizer φ_1 and the classification decider φ_2:

$$\Phi(x) = \varphi_2(\varphi_1(x)). \quad (2)$$

Based on the same featurizer, various entities can convert their raw images to the image data in the same feature space. However, the convolutional feature maps can be easily reconstructed to the original images, due to the rich visual information remembered in the features. As shown in Fig. 2(b), the image reconstructor ψ is learned to reverse the featurizer by minimizing the L1 loss between the original image and the reconstructed image:

$$\mathcal{L}_R = \|x - \psi(\varphi_1(x))\|_1 . \quad (3)$$

After the pre-training, each entity share the image featurizer, classification decider and the image reconstructor with fixed parameters.

3.3. Step 2: DPM Training

The reconstruction in Fig. 2(b) shows that if one entity shares the conventional features directly to collaborators, the collaborators can easily recover the original images, leading to visually spreading images. To deal with this issue, this paper proposes the regime that each entity designs a private Deep Poisoning Module (DPM) to poison the image features before sharing. The DPM consists of a sequential of convolutional layers and activation layers. It is an extra opertaion that disturbs the original features. Each entity could keep its DPM in private, so that other entities are denied to learn an image reconstructor to reverse the poisoned features. The private DPM makes the ground truth (the pairs of poisoned features and original images) unavailable for reconstructor learning. Meanwhile, each DPM is also learned to ensure that the poisoned features are functionally equivalent to the original features (produced directly from the image featurizer) for the classification decider.

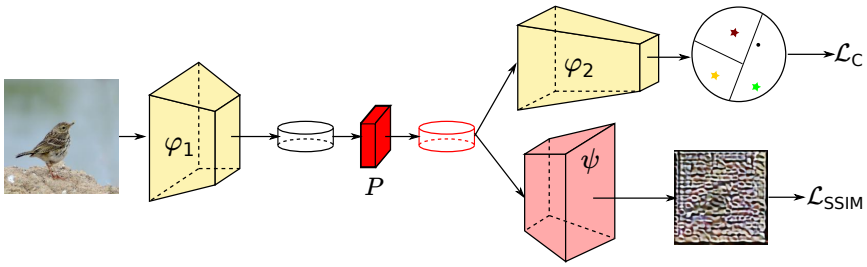

Figure 3. The learning framework of a DPM. The parameters of φ_1, φ_2 and ψ are fixed during DPM training.

Specifically, as shown in Fig. 3, the learning of a DPM is achieved by adversarially optimizing two loss functions. By fixing the parameters of φ_1 and φ_2 and minimizing the classification loss in Eq. (1), the poisoned features achieve *classification equivalence* to the original features:

$$\varphi_2(P(\varphi_1(x))) = \varphi_2(\varphi_1(x)), \tag{4}$$

where P is the private DPM.

To make the poisoned features not be reversed to the original images by the pre-trained image reconstructor ψ (an inverse of the shared featurizer φ_1), the DPM is also learned to achieve reconstruction disparity – it makes the reconstructed image from poisoned features $\psi(P(\varphi_1(x)))$ visually dissimilar to the original one x, by minimizing the Structural Similarity Index Measure (SSIM) [14, 51]. We use $SSIM(\cdot,\cdot) \in [0,1]$ between two images as the loss function:

$$\mathcal{L}_{SSIM} = SSIM(\psi(P(\varphi_1(x))), x). \tag{5}$$

Besides, the parameters of the pre-trained image reconstructor is also fixed during DPM learning.

Then, the DPM is finally optimized by the linear combination of the classification loss and the reconstruction loss:

$$\theta_P = \underset{\theta_P}{\operatorname{argmin}}\, \mathcal{L}_C + \lambda \underset{\theta_P}{\operatorname{argmin}}\, \mathcal{L}_{SSIM}, \tag{6}$$

where λ is a hyper-parameter to balance two losses, and θ_P represents the parameters of the DPM.

Generally, with fixed parameters in the pre-trained featurizer, classification decider and image reconstructor, DPMs of various entities allows various collaborators to convert their images to deep representations in the same feature space (i.e. classification equivalence) by feeding them to partially different operations – the same featurizer but different DPMs, respectively. This classification equivalence ensures that the poisoned features from different entities can be combined for classification decider refinement, while the partial sharing of the operations (keeping DPMs in private) makes the inverse of the operation infeasible, which defend against the image reconstruction from the poisoned features. Compared with deep obfuscated representations [19, 32, 38, 41, 47, 50, 53], which retain image classification-related information and suppress the reconstruction-related information adversarially, the proposed private DPM framework relies on a specific structure to deny the image reconstructor learning and thus is more reliable for preventing visual disclosure. To retain the functional ability of convolutional features for image classification, Since the reconstruction-related information can not be eliminated thoroughly to retain the functional ability of convolutional features for image classification, once the obfuscator is shared to collaborators for making classification equivalence, they can still reverse the image features to original images.

3.4. Step 3: Classification Decider Refinement

With the shared image featurizer φ_1 and private DPMs $\{P_1, P_2, ...\}$, each entity featurizes and poisons its images, and share the poisoned features to other collaborators, as shown in Fig. 1(a). Then, each collaborator could combine the shared image features (poisoned) and its own image features (original) to refine the classification decider, as illustrated in Fig. 1(b). The sequential combination of the shared featurizer and its refined classification decider allows an entity to form a more robust deep model for image classification task in Fig. 1(c).

4. Experiments

In this section, we first conduct experiments to prove that the proposed DPM can effectively poison the image features for image data sharing as expected: 1) the poisoned features are functionally equivalent to the original ones for a specific vision tasks, i.e. image classification; 2) the poisoned image features can not be reconstructed to the original images for visual disclosure of sensitive image contents. Then, we compare the the proposed DPM with existing methods, including conventional perturbations and adversarial obfuscation, to clarify its reliability. Furthermore, by simulating entities exploiting poisoned and shared image features from others to refine model learning, we also verify the effectiveness of using DPM for image data sharing among collaborators addressing the same vision task.

4.1. Configurations

Instead of collecting some specific sensitive images, e.g. CEIs, which may be inappropriate for exhibition in conference papers, we use the widely-used ImageNet dataset [44] (1000-category classification) for simulation. Specifically, to simulate the task of various entities collaborating to address the image classification task, the dataset is randomly split into two sets, each with images of exclusive 500 categories (around 640K training images). One of the 500-category image sets $\mathbb{S}$ simulates the image datasets for addressing the image classification task, while the other one $\mathbb{Q}$ simulates a public image dataset. We suppose that image contents in $\mathbb{S}$ should not be visually disclosed during the image data sharing. Both $\mathbb{S}$ and $\mathbb{Q}$ contain training and validation subsets.

Due to its general applicability for computer vision tasks, we adopt a ResNet [13] architecture as the backbone network. Following the expressions in Table 1 of [13], we use $conv[\cdot]\_[\cdot]$ to represent the hook point that splits the architecture into the image featurizer and the classification decider. For example, $conv4\_1$ indicates that the featurizer consists of the layers from the start of the architecture until the first building block of $layer4$ in the ResNet architecture.

4.2. DPM for Image Data Sharing against Visual Disclosure

4.2.1 Effectiveness of DPM

Suppose we are an individual entity with the collected image set $\mathbb{S}$ and would like to share these image data to others for image classification model training, i.e. the classification decider in Fig. 1. Initially, following the Step 1 in Sec. 3.2, we conventionally train the 500-category image classification models based on ResNet50 and ResNet101, respectively. Given the input images with spatial dimension 224×224, the top-1 and top-5 precision on the validation set of $\mathbb{S}$ achieved by ResNet50 are 79.39% and 94.18%, respectively, while that achieved by ResNet101 are 81.13% and 95.03%, respectively.

Then, the layer point $conv4\_1$ [13] for both models are selected for network split. The image reconstructor ψ to reverse the featurizer φ_1 is composed from the inverse of bottleneck blocks in ResNet [13]. Specifically, two inverse bottleneck blocks (CONV1 $\times$ 1 – BN – CONV3 $\times$ 3 – BN – CONV1 $\times$ 1 – ReLU) are used before upscaling the spatial dimension of feature maps by the factor of 2. As the feature maps with spatial dimension 14×14 are upscaled to 224×224, a CONV1 $\times$ 1 – BN – ReLU – CONV1 $\times$ 1 module is appended to produce the reconstructed image. This image reconstructor is learned as shown in Fig. 2(b) and optimized by the loss in Eq. (3). Since we would like to simulate the process of an entity reversing the featurizer instead of the specific set of image features, the image reconstructor is trained on the image set $\mathbb{Q}$.

Backbone	Acc. Metric (%)	Convolutional Features	
		Original	Poisoned
ResNet50	top-1	79.39	78.88
	top-5	94.18	94.11
ResNet101	top-1	81.13	80.78
	top-5	95.03	94.86

Table 1. Image classification results based on the original convolutional features and the poisoned convolutional features, by feeding them to the pre-trained classification decider, respectively.

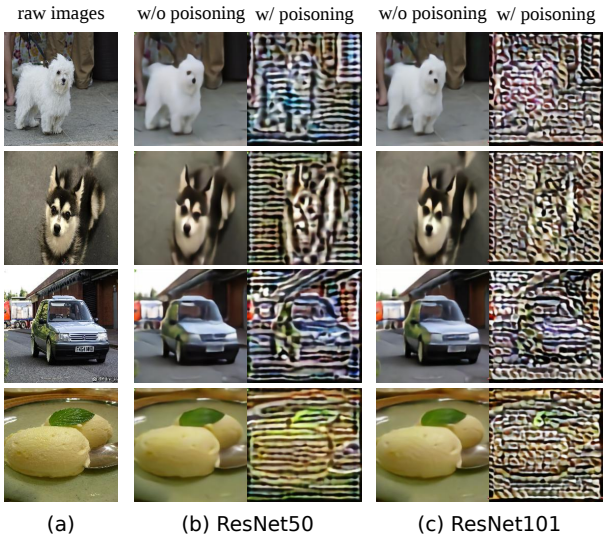

Figure 4. Qualitative comparison of image reconstruction from the original convolutional features (second and fourth columns) and the poisoned convolutional features (third and fifth columns).

According to the Step 2 in Sec. 3.3, we design a specific DPM (P) with 4 residual blocks from ResNet and insert it to the pre-trained model. It is optimized based on Eq. (6) with the hyper-parameter $\lambda = 1.0$.

As shown in Table 1, when we feed the original features $\varphi_1(x)$ and the poisoned features $P(\varphi_1(x))$ to the pre-trained classification decider φ_2, the achieved performance from the same architecture for image classification (top-1 and top-5 accuracy) is quite close, which indicates that the poisoned features are functionally equivalent to the original features for the specific image classification.

$conv4\_1$	L1 Distance (↑)		SSIM (↓)	
	Original	Poisoned	Original	Poisoned
ResNet50	0.0443	0.2928	0.6730	0.0070
ResNet101	0.0406	0.2886	0.7009	0.0069

Table 2. Quantitative comparison of image reconstruction results from the original and poisoned features.

Meanwhile, we use the L1 distance (Eq. (3)) and SSIM (Eq. (5)) to quantify the similarity between the reconstructed images and the original images. As shown in the second and fourth columns of Table 2, the image reconstructor is learned as an excellent inverse to the featurizer. It well recovers images from the original features. As the DPM is adopted to poison the features, the inverse of the featurizer can not recover the images (third and fifth columns in Table 2). The increasing L1 distance and decreasing SSIM indicate that the image similarity between the reconstructed images and the original images is reduced. While the DPM is not shared to collaborators and the original images are prohibited to sharing, collaborators are prevented from learning image reconstructors to reverse the DPM. Thus, we use the pre-trained image reconstructor (i.e. the inverse of the image featurizer) to recover images from the poisoned features. Besides, Fig. 4 visually compares the image reconstruction results from the original features and the poisoned features. The image contents in the reconstructed images from the poisoned features can hardly be recognized by human eyes. The visual disclosure of sensitive contents is well defended if these images contain sensitive contents.

To summarize, the above experimental results demonstrate that the learned DPM poisons the features so that the poisoned features: 1) are classification equivalent to the original features, and 2) defend against visual disclosure of sensitive image contents when being shared.

4.2.2 Sufficiency of DPM

During the above DPM training and evaluation, the same fixed reconstructor is defended. This pre-trained reconstructor is an easy objective to optimize against, and in practice, there may be different networks to reverse the featurizer. Therefore, we further design and train five more reconstructors based on different architectures to reverse the pre-trained featurizer. These reconstructors are only used for testing the defensive ability of the DPM, but not used for DPM training. We denote the only reconstructor used for DPM training as px_2s_2, where p indicates the type of blocks used for building the reconstructor (in this case, a plain inverse bottleneck block without residual operation), x_2 represents two blocks before upscaling, and s_2 means that the upscaling factor is 2. Similarly, other reconstructors are denoted as px_4s_2, rx_2s_2, rx_4s_2, rx_4s_4 and $rx_2s_2\_c$, where r indicates inverse residual bottleneck blocks, and c means the normalization strategy during reconstructor training is clamp instead of min-max normalization. We feed the features produced by φ_1 and their corresponding poisoned features created with P to each of the above reconstructors. The reconstruction results in Fig. 5 indicate that the learned DPM can defend various reconstructors, which have not been learned to defend during its training.

4.2.3 Comparison with Existing Methods

In our design, each entity learns an individual DPM to poison image features for image data sharing. To clarify the advantage of the proposed DPM, we compare it to existing approaches for addressing visual disclosure of image contents, including stationary perturbations, such as Gaussian filter (GF), Gaussian noise (GN) and mean filter (MF), and adversarial representation learning, e.g. DeepObfuscator [19].

By replacing the DPM with directly applying the above stationary perturbations to convolutional features and feeding the perturbated features to the classification decider and the image reconstructor, the results shown in the top part of Table 3 indicate that the perturbations can defend against image reconstruction to some extent, but also suppress the ability of the features for image classification. While the proposed DPM can achieve classification equivalence between poisoned features and original features, it achieves more promising results for both remaining image classification performance and defending against image reconstruction. Furthermore, combining stationary perturbations with DMP still achieves promising results – defending against the image reconstruction with minimal loss in classification performance, as shown in the bottom rows of Table 3. Meanwhile, the visual comparison of image reconstruction from image features altered by stationary perturbations and the proposed DPM in Fig. 6 further validates the advantage of the proposed DPM over conventional perturbations.

Additionally, we compare the proposed method with a privacy-preserving method, i.e. DeepObfuscator [19] for image reconstruction under both conventional privacy preserving and the introduced image data sharing for collaboration. The comparison experiment is conducted on CelebA dataset [26] for face attribute recognition. Similar to [19], the first 20 attributes are regarded as the utility in privacy preserving or the vision task to be addressed in image data sharing, while the remaining 20 attributes are the privacy, and the image reconstruction is another privacy and visual

Poisoning	classification (%)		reconstruction	
	top-1	top-5	L1	SSIM
w/o	81.13	95.03	0.0406	0.7009
GN	25.47	45.99	0.1905	0.2635
GF	15.60	30.24	0.1055	0.4699
MF	4.44	10.78	0.1169	0.4334
DPM	80.78	94.86	0.2886	0.0069
GN+DPM	78.10	93.64	0.3339	0.0047
GF+DPM	78.77	93.78	0.3450	0.0041
MF+DPM	71.23	89.96	0.3564	0.0186

Table 3. Result comparison between perturbations and the proposed DPM for remaining image classification performance and defending image reconstruction simultaneously.

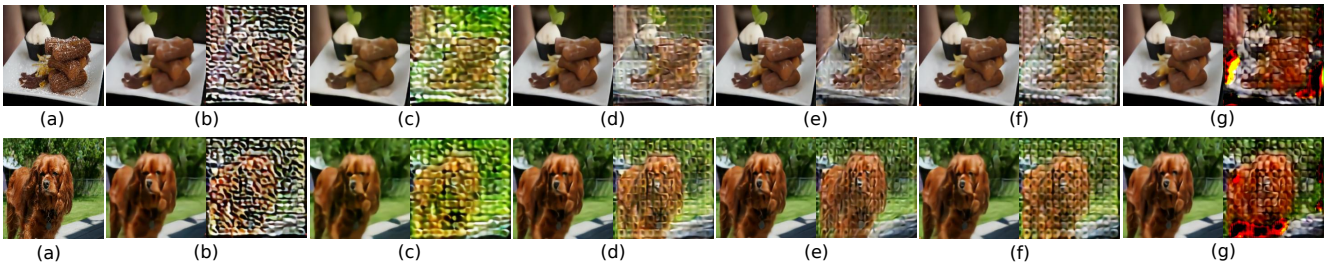

Figure 5. Comparison of image reconstruction results from the original (left columns) and poisoned (right columns) convolutional features by reconstructors: (b) px_2s_2, (c) px_4s_2, (d) rx_2s_2, (e) rx_4s_2, (f) rx_4s_4 and (g) $rx_2s_2\_c$. (a): raw images.

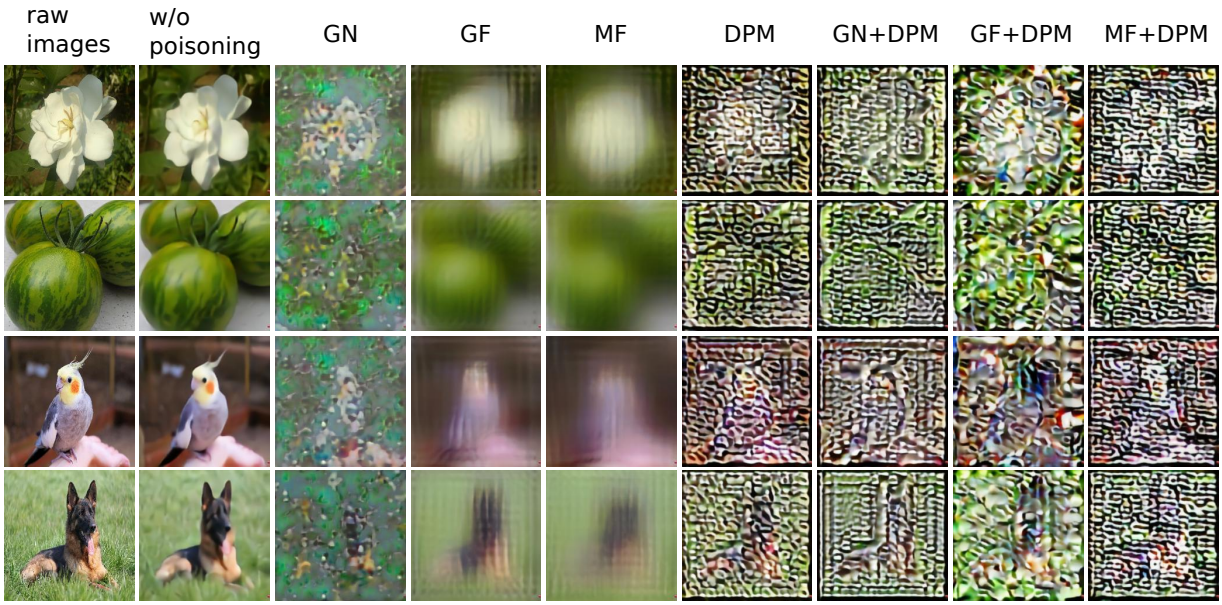

Figure 6. Comparison of image reconstruction from different poisoning operations.

disclosure for image data sharing. The VGGNet [46] is adopted as the backbone for fair comparison and we split the model at the same point as DeepObfuscator. As shown in columns 2 to 5 of Table 4, the proposed DPM achieves better overall results on privacy preserving. Specifically, the DPM better remains the utility recognition and defends the reconstruction that leaks privacy. When being adopted to image data sharing for task-specific collaboration against visual disclosure, sharing the obfuscator of [19] among collaborators results in the reversing of the obfuscator, which makes the obfuscated features reconstructed to the original images, as shown in Fig. 7. Also, in columns six and seven of Table 5, when an entity learns an image reconstructor to recover images from the original image features, the reconstructed images are highly similar to the original images, with L1 distance of 0.0200 and SSIM of 0.9177. Similarly, reversing the obfuscator of DeepObfuscator also achieves great reconstruction quality, which is not desired. This is because there is a large overlap between the recognition-related information and reconstruction-relation information in the convolutional features. To guarantee the recognition performance, the reconstruction-related informaiton can not be removed thoroughly from the image features. Compared with DeepObfuscator, the proposed DPM allows each entity to keep its own DPM in private and maintain the poisoned features from different DMPs in the same feature space. Keeping DPM in private denies reversing it for the image reconstruction from poisoned features. Thus, the proposed DPM can also be used for privacy preserving, while the privacy-preserving methods can not address the introduced task of image data sharing against visual disclosure.

4.3. Entity Collaboration – Model Refinement

In this section, we conduct experiments to verify that the poisoned features shared from each entity helps others improve the image classification performance. To simulate the collaborators with limited training images, we define three subsets of images from the image set $\mathbb{S}$, each of which is

	utility recognition mAPs (↑, %)	privacy recognition mAPs (↓, %)	privacy reconstruction		image sharing reconstruction	
			L1 (↑)	SSIM (↓)	L1 (↑)	SSIM (↓)
w/o protection	81.08	80.54	0.0200	0.9177	0.0200	0.9177
DeepObfuscator [19]	73.71	65.07	0.2614	0.3997	0.0359	0.8232
DPM	79.53	65.95	0.2573	0.0064	Denied	

Table 4. Result comparison between the privacy-preserving obfuscation and the proposed DPM.

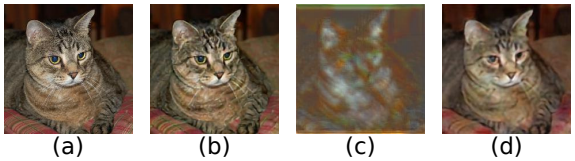

Figure 7. (a) the original image; (b) image reconstruction from the non-poisoned features; (c) DeepObfuscator defending its adversarial reconstructor; (d) image reconstruction from obfuscated features (DeepObfuscator) by reversing the obfuscator.

assumed to be collected by an entity (in $\{E_1, E_2, E_3\}$) with 10% images (64056 images) from $\mathbb{S}$, denoted as $\mathbb{S}_1$, $\mathbb{S}_2$ and $\mathbb{S}_3$. Besides, a fourth entity (E_4) collects 1% images of $\mathbb{S}$, denoted as $\mathbb{S}_4$. As shown in rows (1 – 4) of Table 5, when various entities learn classification models individually from their own images, respectively, the classification performance is severely limited by the number of training images.

Suppose the first entity uses $\mathbb{S}_1$ pre-trains the classification network and splits it according to Sec.3.2, then each of the entity follows the similar procedure to train individual DPMs (P_1, P_2, P_3 and P_4) for feature poisoning as Sec. 3.3. A shared pool of image features among various collaborators is established as: $\{P_1(\varphi_1(\mathbb{S}_1)), P_2(\varphi_1(\mathbb{S}_2)), P_3(\varphi_1(\mathbb{S}_3)), P_4(\varphi_1(\mathbb{S}_4))\}$. Then, each entity can combine its own image data and the image data shared from other entities to refine the classification decider φ_2, as shown in Fig. 1(b).

As shown in the rows 7 and 8 in Table 5, two entities E_1 and E_2 use the poisoned image features from each other and achieve much better classification performance than training models based on its own data (rows 1 and 2), respectively. Compared with combining image features directly (row 6), there is certain loss in performance improvement due to the poisoning operation leading to information loss. Besides, when the number of collaborators increases, e.g. E_3 (in rows 9 and 10) exploits data from E_1 and E_2, the performance can be further improved and is much better than just using its own data $\mathbb{S}_3$ in row 3. Specially, when entity E_4 train its classification model based on its own images, the top-1 accuracy is 13.5%. With other entities sharing image data to E_4, it can easily achieve 61.6% top-1 accuracy (in row 12). These comparison results indicate that the image data sharing does benefit entities since they can use others' data for its model refinement.

rows	image data	top-1	top-5
1 – E_1	raw images $\mathbb{S}_1$ (10%)	52.9	77.7
2 – E_2	raw images $\mathbb{S}_2$ (10%)	52.7	77.8
3 – E_3	raw images $\mathbb{S}_3$ (10%)	53.2	77.9
4 – E_4	raw images $\mathbb{S}_4$ (1%)	13.5	30.3
5	raw images $\mathbb{S}$ (100%)	81.1	95.0
6	$\varphi_1(\mathbb{S}_1) \cup \varphi_1(\mathbb{S}_2)$	61.2	83.8
7 – E_1	$\varphi_1(\mathbb{S}_1) \cup P_2(\varphi_1(\mathbb{S}_2))$	59.4	82.6
8 – E_2	$P_1(\varphi_1(\mathbb{S}_1)) \cup \varphi_1(\mathbb{S}_2)$	59.8	83.0
9 – E_3	$P_1(\varphi_1(\mathbb{S}_1)) \cup P_2(\varphi_1(\mathbb{S}_2)) \cup \varphi_1(\mathbb{S}_3)$	62.3	84.5
10 – E_3	$P_1(\varphi_1(\mathbb{S}_1)) \cup P_2(\varphi_1(\mathbb{S}_2)) \cup P_3(\varphi_1(\mathbb{S}_3)) \cup \varphi_1(\mathbb{S}_3)$	62.7	84.7
11 – E_4	$P_1(\varphi_1(\mathbb{S}_1)) \cup P_2(\varphi_1(\mathbb{S}_2)) \cup \varphi_1(\mathbb{S}_4)$	58.4	82.1
12 – E_4	$P_1(\varphi_1(\mathbb{S}_1)) \cup P_2(\varphi_1(\mathbb{S}_2)) \cup P_3(\varphi_1(\mathbb{S}_3)) \cup \varphi_1(\mathbb{S}_4)$	61.6	84.4

Table 5. Classification performance of image classification models trained/fine-tuned on image data in different representations.

5. Conclusion and Future Work

This paper introduced a new vision task of image data sharing against visual disclosure of sensitive contents. The task aims to make various entities refine respective models by using image data shared by others, but not visually observe the sensitive content in the shared data. To achieve this goal, we proposed that each entity inserts an extra Deep Poisoning Module to the pre-trained network. The DPMs were learned to make the poisoned features be functionally equivalent to the non-poisoned features and defend against the image reconstruction. Being kept in private, DPMs provided a structure-based approach to prevent the shared image data from visually disclosing sensitive image contents. Our experiments verified the effectiveness of the proposed method by simulating the process of sharing data to benefit collaborators' model refinement without visually disclosing sensitive image contents.

Besides, this paper is an initial exploration of the introduced task, there are still many problems to be addressed, such as the potential non-i.i.d. issue among images from various entities, expanding to other vision applications, etc. These issues may be discussed in the future work.

References

[1] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *ACM Sigmod Record*, volume 29, pages 439–450. ACM, 2000.

[2] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2722–2730, 2015.

[3] Jiawei Chen, Janusz Konrad, and Prakash Ishwar. Vgan-based image representation learning for privacy-preserving facial expression recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1570–1579, 2018.

[4] Rui Chen, Benjamin C.M. Fung, Noman Mohammed, Bipin C. Desai, and Ke Wang. Privacy-preserving trajectory data publishing by local suppression. *Information Sciences*, 231:83–97, 2013.

[5] Elliot Creager, David Madras, Jörn-Henrik Jacobsen, Marissa A Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. Flexibly fair representation learning by disentanglement. *arXiv preprint arXiv:1906.02589*, 2019.

[6] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Computing Surveys*, 42(4):14, 2010.

[7] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012.

[8] Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy. In *International Conference on Machine Learning*, pages 201–210, 2016.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[10] Bernardo Cuenca Grau and Egor V. Kostylev. Logical foundations of privacy-preserving publishing of linked data. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[11] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. MS-Celeb-1M: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pages 87–102. Springer, 2016.

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[14] Alain Hore and Djemel Ziou. Image quality metrics: PSNR vs. SSIM. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369. IEEE, 2010.

[15] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4700–4708, 2017.

[16] Miran Kim, Yongsoo Song, Shuang Wang, Yuhou Xia, and Xiaoqian Jiang. Secure logistic regression based on homomorphic encryption: Design and evaluation. *JMIR medical informatics*, 6(2):e19, 2018.

[17] Tae-hoon Kim, Dongmin Kang, Kari Pulli, and Jonghyun Choi. Training with the invisibles: Obfuscating images to share safely for learning visual recognition models. *arXiv preprint arXiv:1901.00098*, 2019.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[19] Ang Li, Jiayi Guo, Huanrui Yang, and Yiran Chen. DeepObfuscator: Adversarial training framework for privacy-preserving image classification. *arXiv preprint arXiv:1909.04126*, 2019.

[20] Tiancheng Li, Ninghui Li, Jian Zhang, and Ian Molloy. Slicing: A new approach for privacy preserving data publishing. *IEEE Transactions on Knowledge and Data Engineering*, 24(3):561–574, 2010.

[21] 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.

[22] Yifang Li, Nishant Vishwamitra, Bart P Knijnenburg, Hongxin Hu, and Kelly Caine. Blur vs. block: Investigating the effectiveness of privacy-enhancing obfuscation for images. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1343–1351. IEEE, 2017.

[23] Yehuda Lindell and Benny Pinkas. Privacy preserving data mining. In *Annual International Cryptology Conference*, pages 36–54. Springer, 2000.

[24] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *European Conference on Computer Vision*, pages 21–37. Springer, 2016.

[25] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. DeepFashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, June 2016.

[26] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision*, December 2015.

[27] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[28] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5188–5196, 2015.

[29] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

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

[31] Ricardo Mendes and João P Vilela. Privacy-preserving data mining: methods, metrics, and applications. *IEEE Access*, 5:10562–10582, 2017.

[32] Seong Joon Oh, Mario Fritz, and Bernt Schiele. Adversarial image perturbation for privacy protection a game theory perspective. In *2017 IEEE International Conference on Computer Vision*, pages 1491–1500. IEEE, 2017.

[33] Seyed Ali Osia, Ali Shahin Shamsabadi, Ali Taheri, Kleomenis Katevas, Hamid R Rabiee, Nicholas D Lane, and Hamed Haddadi. Privacy-preserving deep inference for rich user data on the cloud. *arXiv preprint arXiv:1710.01727*, 2017.

[34] Seyed Ali Osia, Ali Taheri, Ali Shahin Shamsabadi, Minos Katevas, Hamed Haddadi, and Hamid RR Rabiee. Deep private-feature extraction. *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[35] Jordan Pearson. Microsoft deleted a massive facial recognition database, but it's not dead. *Vice*, Jun 2019.

[36] Benny Pinkas. Cryptographic techniques for privacy-preserving data mining. *ACM Sigkdd Explorations Newsletter*, 4(2):12–19, 2002.

[37] Francesco Pittaluga, Sanjeev Koppal, and Ayan Chakrabarti. Learning privacy preserving encodings through adversarial training. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 791–799. IEEE, 2019.

[38] Nisarg Raval, Ashwin Machanavajjhala, and Landon P. Cox. Protecting visual secrets using adversarial nets. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1329–1332. IEEE, 2017.

[39] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 779–788, 2016.

[40] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[41] 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*, pages 620–636, 2018.

[42] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision workshop on Benchmarking Multi-Target Tracking*, 2016.

[43] Proteek Chandan Roy and Vishnu Naresh Boddeti. Mitigating information leakage in image representations: A maximum entropy approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2586–2594, 2019.

[44] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[45] Michael S Ryoo, Brandon Rothrock, Charles Fleming, and Hyun Jong Yang. Privacy-preserving human activity recognition from extreme low resolution. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[46] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[47] Pablo Speciale, Johannes L Schonberger, Sing Bing Kang, Sudipta N Sinha, and Marc Pollefeys. Privacy preserving image-based localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5493–5503, 2019.

[48] 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*, pages 553–569, 2018.

[49] Jaideep Vaidya and Chris Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 639–644. ACM, 2002.

[50] Haotao Wang, Zhenyu Wu, Zhangyang Wang, Zhaowen Wang, and Hailin Jin. Privacy-preserving deep visual recognition: An adversarial learning framework and a new dataset. *arXiv preprint arXiv:1906.05675*, 2019.

[51] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[52] Raymond Chi-Wing Wong, Jiuyong Li, Ada Wai-Chee Fu, and Ke Wang. (α, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 754–759. ACM, 2006.

[53] Zhenyu Wu, Zhangyang Wang, Zhaowen Wang, and Hailin Jin. Towards privacy-preserving visual recognition via adversarial training: A pilot study. In *Proceedings of the European Conference on Computer Vision*, pages 606–624, 2018.

[54] Taihong Xiao, Yi-Hsuan Tsai, Kihyuk Sohn, Manmohan Chandraker, and Ming-Hsuan Yang. Adversarial learning of privacy-preserving and task-oriented representations. *arXiv preprint arXiv:1911.10143*, 2019.

[55] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1492–1500, 2017.

[56] Yang Xu, Tinghuai Ma, Meili Tang, and Wei Tian. A survey of privacy preserving data publishing using generaliza-

tion and suppression. *Applied Mathematics & Information Sciences*, 8(3):1103–1116, 2014.

[57] Ryo Yonetani, Vishnu Naresh Boddeti, Kris M Kitani, and Yoichi Sato. Privacy-preserving visual learning using doubly permuted homomorphic encryption. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2040–2050, 2017.