Poster: Image Disguising for Privacy-preserving Deep Learning

Sagar Sharma, Keke Chen
Data Intensive Analysis and Computing (DIAC) Lab, Kno.e.sis Center, Wright State University
{sharma.74,keke.chen}@wright.edu

ABSTRACT

Due to the high training costs of deep learning, model developers often rent cloud GPU servers to achieve better efficiency. However, this practice raises privacy concerns. An adversarial party may be interested in 1) personal identifiable information encoded in the training data and the learned models, 2) misusing the sensitive models for its own benefits, or 3) launching model inversion (MIA) and generative adversarial network (GAN) attacks to reconstruct replicas of training data (e.g., sensitive images). Learning from encrypted data seems impractical due to the large training data and expensive learning algorithms, while differential-privacy based approaches have to make significant trade-offs between privacy and model quality. We investigate the use of image disguising techniques to protect both data and model privacy. Our preliminary results show that with block-wise permutation and transformations, surprisingly, disguised images still give reasonably well performing deep neural networks (DNN). The disguised images are also resilient to the deep-learning enhanced visual discrimination attack and provide an extra layer of protection from MIA and GAN attacks.

ACM Reference Format:
Sagar Sharma, Keke Chen. 2018. Poster: Image Disguising for Privacy-preserving Deep Learning. In *2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18), October 15–19, 2018, Toronto, ON, Canada.* ACM, New York, NY, USA, Article 39, 3 pages. https://doi.org/10.1145/3243734.3278511

1 INTRODUCTION

Deep Neural Networks (DNN) generate robust modeling results across diverse domains such as image classification and natural language processing. However, DNN training is resource and time consuming. Model developers often utilize AWS's elastic GPUs and Google Cloud platform to train large-scale models. In such a setting, a big concern is the privacy of sensitive training data and the model that can be possibly used to explore private data [2].

One possible approach to addressing the privacy issue is to learn models from encrypted data, however, it is too expensive to be practical for deep learning yet. Recent advances in cryptography have provided a few constructs for learning from encrypted data, such as homomorphic encryptions, garbled circuits, and secret sharing [3, 10]. A few attempts have been made to adopt these constructs in deep learning, for example, secure gradient descent [8]. However, due to the large training data and number of iterations in learning DNNs, the protocols normally have impractical costs.

CCS '18, October 15–19, 2018, Toronto, ON, Canada

ACM ISBN 978-1-4503-5693-0/18/10.
https://doi.org/10.1145/3243734.3278511

Differential privacy has been applied in deep learning [1, 11], however the protocols are vulnerable to model inversion (MIA) [2] and Generative Adversarial Network(GAN) attacks [5]. Furthermore, there is a significant tradeoff between utility and privacy - large noises are needed to achieve meaningful privacy, which leads to low-quality models [9, 11]. In the centralized setting, PrivyNet [7] tries to hide private data by users constructing local shallow NNs and sharing the intermediate representations to cloud for learning the final model. However, the results show that the intermediate representations are still visually identifiable.

Scope and contributions. We take a unique approach to balancing privacy and utility with image disguising. The intuition is that deep learning is so powerful that it can pick up the unique features for distinguishing even disguised image training data. The question is how to design the proper disguising mechanisms that can make the original content not (visually and algorithmically) recognizable anymore, while still preserving the features that allow DNNs to distinguish disguised images. We have studied a suite of image disguising mechanisms that enable learning high-quality DNN models on the disguised images, which can be applied in the outsourced setting to protect both data and model privacy. Each outsourced dataset gets a secret image transformation key. As long as data owners keep their keys secret, the disguised images are resilient to the well-known attacks. Fascinatingly, the models learned on the disguised images are high quality and work well in classifying the disguised images, comparable to the models built on undisguised images. Our contributions are as follows:

(1) We have designed a suite of image disguising mechanisms for preserving both privacy and utility of image-based DNN learning in the outsourced setting.
(2) We have developed a toolkit for calibrating the the privacy and utility of certain parameter settings for the disguising mechanisms.
(3) With our approach, the current MIA and GAN attacks generate images in the disguised image forms, thus, providing no additional information than the disguised training images.
(4) Our preliminary evaluation shows that the disguising mechanisms can effectively preserve data privacy and result in surprisingly good-quality models.

2 ADVERSARIAL MODEL

We make some relevant security assumptions here: 1) We consider ciphertext-only attacks, i.e., any cipher-plaintext image pair is unknown to the adversary; 2) All infrastructures and communication channels must be secure.

We consider the cloud provider to be an honest-but-curious adversary. We concern with the *privacy of the image datasets and the learned models.* An adversary may be interested in the contents and identification of images that do not belong to it, or the learned models; they may also misuse private models for its own benefits in

the outsourced setting, or launch MIA and GAN attacks to generate pseudo-images that resemble the victim's private data.

3 IMAGE DISGUISING FOR DEEP LEARNING

Assume a user owns a set of images for training, notated as pairs $\{(X_i, y_i)\}$, where X_i is the image pixel matrix and y_i the corresponding label. We formally define the disguising process as follows. Let the disguising mechanism be a transformation T_K, where K is the secret key. By applying image disguising, the training data is transformed to $\{(T(X_i), y_i)\}$, which is used to train a DNN, denoted as a function $\mathbf{D}_T$ that takes disguised images $T(X)$ and outputs a predicted label $\hat{y}$. For any new data X_{new}, the model application is defined as $\mathbf{D}_T(T(X_{new}))$.

Figure 1 shows how the framework works. A data owner disguises her private images before outsourcing them to the cloud for DNN learning. She transforms all of her images using one key. For model application, she transforms new data with the same key.

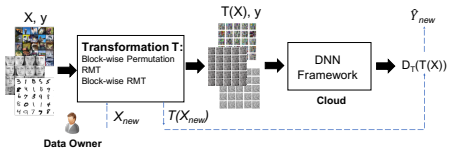

Figure 1: Image disguising framework for DNN learning.

We consider a suite of image disguising mechanisms that can be used individually or layered on top of one another depending on the dataset characteristics and the desired privacy and utility. Candidate mechanisms must hide the *visually identifiable* features in the images, and provide a sufficiently large key space to be resilient to ciphertext-only attacks. As a result, these mechanisms inevitably affect the quality of learned DNNs. Hence, finding the settings that provide both high security and model quality is crucial. We start with the relatively weak block-wise permutation technique and extend to other enhancements.

3.1 Block-wise Permutation

The block-wise permutation simply partitions an image and rearranges image blocks. Let an image $X_{p\times p}$ of p^2 pixels be partitioned into blocks of size $k \times l$ that are labeled sequentially as $v = < 1, 2, 3, 4, \ldots t >$. A pseudorandom permutation of the blocks, $\pi(v)$, shuffles the blocks and reassemble the image. Theoretically, with large t it provides $t!$ candidates, difficult for brute-force attacks. However, such a mechanism is insufficient to hide the image content yet, as the boundary, color, content shape, and texture of the original neighboring blocks provide clues for adversaries to recover the image - imagine the jigsaw puzzle! Figure 2 shows an example. Thus, it has to be combined with other mechanisms.

3.2 Randomized Multidimensional Transformations (RMT)

For an image represented as a pixel matrix X, a general form of randomized multidimensional transformation is defined as $XR + \Delta$, where R can be a random orthogonal (i.e., rotation) or a random projection matrix [12] and Δ is a random additive noise matrix. The matrix R acts as a key across the training data, while Δ is regenerated for each image and drawn uniformly at random from $[0, N]$ where N will be known as the noise level.

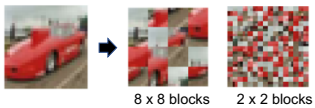

Figure 2: Block-wise Permutation of CIFAR-10 images. The detail on each block can help easily rearrange blocks.

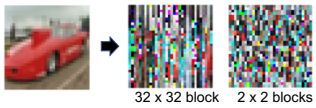

Figure 3: RMT transformation of CIFAR-10 images with orthogonal matrices and 25 noise level. It is difficult to visually detect block-level details and reassemble them.

Block-wise application of RMT. As Figure 3 [1] shows, applying RMT to the entire image may still preserve some visual features, leaving hints to link back the original image. To further strengthen the image privacy, we apply the block-wise RMT. Instead of picking one private R for the entire image, we pick $\{R_1, R_2, \ldots R_t\}$ matrices for the t blocks, respectively. Block-wise RMT can further be combined with block-wise permutation.

4 CALIBRATING IMAGE DISGUISING MECHANISMS

One major issue remains unaddressed: how to tune the parameter settings for the designed mechanisms to meet desired privacy? Our ultimate goal is to design a theoretically justifiable method for evaluating the protection strengths of various disguising mechanisms and their combinations. In our preliminary study, we design a few tools to investigate the effect of different parameter settings. Specifically, we introduce two new concepts: "visual privacy" for quantifying the discernibility of disguised images, and "model misusability" for quantifying the adversarial usability of the developed models on real undisguised data.

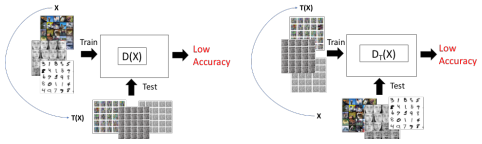

Figure 4: Models trained on transformed images must not perform well on undisguised images and vice versa.

Visual Privacy. The most straightforward approach to visually identifying the disguised images is possibly employing humans to visually examine the images. We move one step further by using a trained DNN for this task as recent studies have shown that a well-trained DNNs are comparable to or even better than human visual recognition. Specifically, we pre-train a "DNN examiner" model on the original image space and measure its accuracy in classifying the transformed images. Let visual privacy be defined as $(1 - accuracy$ of the DNN examiner). We plan to develop more DNN examiners for imitating human examiners' behaviors, i.e., identifying the original neighboring blocks.

Model Mis-usability. Another task is to prevent abusing the learned model, e.g., applying the model on the images captured in public space. Specifically, we assess if the models trained on disguised images also work in classifying undisguised images. Let's

[1] More example figures are uploaded to *https://sites.google.com/site/rmtfordl/*.

define "model mis-usability" as this testing accuracy. The lower the testing accuracy is, the lower the chance of model misuse.

4.1 Resiliency to Model-based Attacks

Model inversion attacks such as GAN and MIA attacks have succeeded in exploiting deep learning models. For a given model, MIA tries to reconstruct a part of training data; GAN attack allows adversarial participant to reconstruct data owners' training data. With the link between the original images and the disguised images hidden from adversaries by our mechanisms, these attacks only reconstruct disguised images, which are useless as the disguised training images are already accessible to adversaries.

5 EXPERIMENTS

We present our experimental findings on 1) model quality, 2) visual privacy and 3) model mis-usability for the block-wise application of RMT. We test the mechanisms in two prevalent DNN benchmarking datasets: MNIST and CIFAR-10.

Table 1: Parameter settings and CNN Architectures.

Datasets	*Mechanisms*	Block size	Noise Level	Architecture
MNIST	block-wise MP + Permutation	$\{7 \times 7\}$	100	Simple
CIFAR-10	block-wise MP	$\{2 \times 2\}$	25	ResNet

Table 1 details the mechanisms, block size, and additive noise level used for the datasets. We used a simple DNN architecture for MNIST [6], and the more powerful ResNet [4] architecture for CIFAR-10 dataset. For MNIST, we set the learning rate to 0.001 and train the network for 1000 iterations. For CIFAR-10, we adapt the learning rate from 0.1 to 0.001 as the model was trained for 350 iterations. Both models are implemented with TensorFlow.

Table 2: Results of applying image disguising mechanisms.

	Model Accuracy			
Datasets	With Disguise	Without Disguise	Visual Privacy	Model Misuability
MNIST	95.6%	96.7%	94.4%	9.2%
CIFAR-10	89.3%	93.4%	89.7%	36.8%

Table 2 shows that the models trained on disguised images perform very close to the optimum accuracy attained by the models trained on undisguised images. Furthermore, we observe high visual privacy for both the datasets and low model mis-usability for MNIST. The model mis-usability of 36.8% for CIFAR-10 36.8% is significantly higher and implies potential risk of model misuse.

The per-record disguising cost for the MNIST dataset with the above setting was less than 1 ms and resulted in images of size 8 KB whereas for the CIFAR-10 the costs were 13 ms and 33 KB.

Figure 5 shows RMT with permutation (w. perm.) over larger block sizes improve model accuracy. Moreover, higher additive noises result in lower quality models. Note: We see a slight improvement in accuracy for MNIST dataset with increasing block sizes and without permutation (w/o perm.).

Figure 6 shows the result of applying a "DNN examiner" on disguised images to assess the preserved visual privacy. The DNN examiners perform consistently low across the board for all block sizes and noise levels for classifying the transformed images, resulting in around 90% preserved visual privacy. Finally, Figure 7 shows that the model mis-usability reduces when the block sizes and the noise level increase and when permutation is applied.

6 CONCLUSION

We propose several image disguising mechanisms to attain practical privacy-preserving deep learning in the outsourced setting. Our preliminary evaluation show highly encouraging results and some

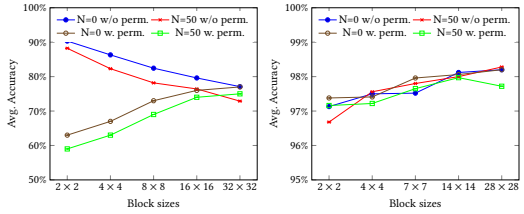

Figure 5: Model quality of DNN models for different block sizes and noise levels (N). CIFAR-10 left, MNIST right

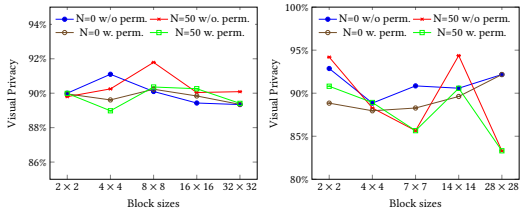

Figure 6: Visual privacy for different block sizes and noise levels (N). CIFAR-10 left, MNIST right

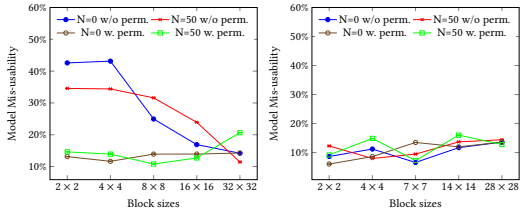

Figure 7: Model mis-usability for different block sizes and noise levels (N). CIFAR-10 left, MNIST right

interesting patterns that are to be further explored. We will extend the evaluation to more datasets, include more image disguising techniques, consider more stringent threat and attack models, and finally establish a theoretical justification of the preserved privacy.

ACKNOWLEDGMENTS

This work is partially supported by the National Science Foundation under Grant 1245847.

REFERENCES

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. 2016.
[2] M. Fredrikson, S. Jha, and T. Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. *Conference on Computer and Communications Security*, page 1322, 2015.
[3] T. Graepel, K. Lauter, and M. Naehrig. Ml confidential: Machine learning on encrypted data. In *Proceedings of the 15th International Conference on Information Security and Cryptology*, ICISC'12, Berlin, Heidelberg, 2013. Springer-Verlag.
[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.
[5] B. Hitaj, G. Ateniese, and F. Pérez-Cruz. Deep models under the GAN: information leakage from collaborative deep learning. *CoRR*, abs/1702.07464, 2017.
[6] T. Jeon. Classifying mnist dataset using cnn. http://yann.lecun.com/exdb/mnist/.
[7] M. Li, L. Lai, N. Suda, V. Chandra, and D. Z. Pan. Privynet: A flexible framework for privacy-preserving deep neural network training with A fine-grained privacy control. *CoRR*, abs/1709.06161, 2017.
[8] P. Mohassel and Y. Zhang. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE Symposium on Security and Privacy (SP)*, 2017.
[9] A. Narayanan. Data privacy: The story of a paradigm shift, 2010.
[10] V. Nikolaenko, U. Weinsberg, S. Ioannidis, M. Joye, D. Boneh, and N. Taft. Privacy-preserving ridge regression on hundreds of millions of records. In *Proceedings of the 2013 IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2013.
[11] R. Shokri and V. Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.
[12] S. S. Vempala. *The Random Projection Method*. American Mathematical Society, 2005.