PRIVACY-PRESERVING DEEP NEURAL NETWORKS WITH PIXEL-BASED IMAGE ENCRYPTION CONSIDERING DATA AUGMENTATION IN THE ENCRYPTED DOMAIN

Warit Sirichotedumrong, Takahiro Maekawa, Yuma Kinoshita and Hitoshi Kiya

Tokyo Metropolitan University, Asahigaoka, Hino-shi, Tokyo, 191-0065, Japan

ABSTRACT

We present a novel privacy-preserving scheme for deep neural networks (DNNs) that enables us not to only apply images without visual information to DNNs for both training and testing but to also consider data augmentation in the encrypted domain for the first time. In this paper, a novel pixel-based image encryption method is first proposed for privacy-preserving DNNs. In addition, a novel adaptation network is considered that reduces the influence of image encryption. In an experiment, the proposed method is applied to a well-known network, ResNet-18, for image classification. The experimental results demonstrate that conventional privacy-preserving machine learning methods including the state-of-the-arts cannot be applied to data augmentation in the encrypted domain and that the proposed method outperforms them in terms of classification accuracy.

***Index Terms*—** Deep learning, deep neural network, image encryption, privacy-preserving

1. INTRODUCTION

The spread of deep neural networks (DNNs) has greatly contributed to solving complex tasks for many applications [1, 2], such as for computer vision, biomedical systems, and information technology. Deep learning utilizes a large amount of data to extract representations of relevant features, so the performance is significantly improved [3, 4]. However, there are security issues when using deep learning in cloud environments to train and test data, such as data privacy, data leakage, and unauthorized data access. Therefore, privacy-preserving DNNs have become an urgent challenge.

In this paper, we propose a novel privacy-preserving method for DNNs that enables us to not only apply images without visual information to DNNs for both training and testing but to also carry out data augmentation in the encrypted domain for the first time. Data augmentation, which is a technique for creating new training data from existing data, is widely used in DNN-based methods because it is easy to implement and is effective. Augmented data can be utilized to improve model robustness and increase accuracy. Since augmentation produces a huge amount of data, it is required that it has to be performed on a cloud server in order to reduce the amount of data traffic. This paper is among the first to discuss data augmentation in the encrypted domain.

Various methods have been proposed for privacy-preserving computation. The methods are classified into two types: perceptual encryption-based [5–16] and homomorphic encryption (HE)-based [17–24]. As described in Section 2, HE-based methods are the most secure options for privacy preserving computation, but they are applied to only limited DNNs [21–24]. Therefore, the HE-based type does not support state-of-the-art DNNs yet. Moreover, data augmentation has to be done before encryption. In contrast, perceptual encryption-based methods have been seeking a trade-off in security to enable other requirements, such as a low processing demand, bitstream compliance, and signal processing in the encrypted domain [5–16]. A few methods were applied to machine learning algorithms in previous works [5, 6]. The first encryption method [9–14] to be proposed for encryption-then-compression (EtC) systems, was demonstrated to be applicable to traditional machine learning algorithms, such as support vector machine (SVM) [5]. However, the block-based encryption method has never been applied to DNNs. Another method [6] was applied to image classification with DNNs, in which an adaption network is added prior to DNNs to avoid the influence of image encryption. However, the accuracy of image classification is lower than that of using plain images, and, moreover, data augmentation in the encrypted domain cannot be applied to the conventional method.

In an experiment, we compare the proposed method with conventional perceptual encryption-based methods. The experimental results show that the proposed methods with DNNs performs better in classification than conventional block-based and pixel-based encryption schemes. In addition, it is proved that the proposed encryption allows us to perform data augmentation in the encrypted domain.

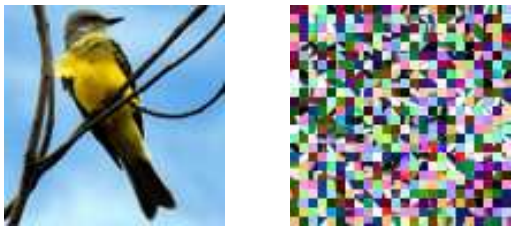
2. RELATED WORKS

2.1. Visual Information Protection

Security mostly refers to protection from adversarial forces. This paper focuses on protecting visual information that allows us to identify an individual, the time, and the location of the taken photograph. Untrusted platforms and unauthorized users are assumed to be adversaries.

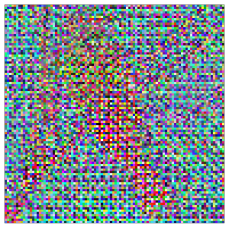
Various perceptual image encryption methods [5–16] have been proposed for protecting the visual information of images. Compared with full encryption with provable security like homomorphic encryption (HE), they generally have a low computational cost and can offer encrypted data robust against various kinds of noise and errors. In addition, some of them aim to consider both security and efficient compression so that they can be adapted to cloud storage and network sharing [9–15]. However, with the exception of a few previous pieces of work, most conventional perceptual encryption methods have never been considered for application to machine learning algorithms [5, 6].

This work was partially supported by Grant-in-Aid for Scientific Research(B), No.17H03267, from the Japan Society for the Promotion Science.

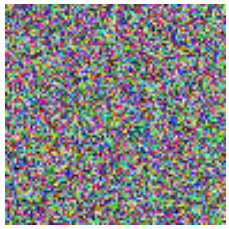


(a) Original image ($X \times Y = 96 \times 96$)

(b) Block-based encryption [9, 10] (Block size=4×4)



(c) Block-based encryption [6] (Block size=4×4)



(d) Pixel-based image encryption [16]

Fig. 1: Examples of images encrypted by conventional schemes

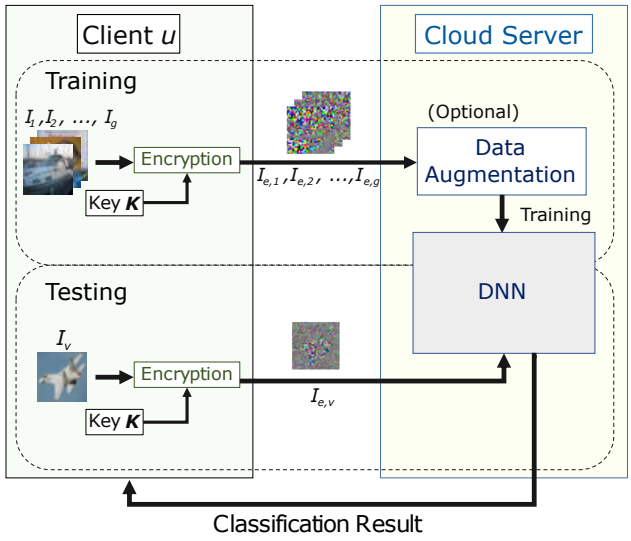


Fig. 2: Scenario

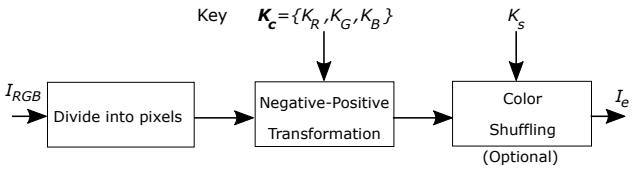


Fig. 3: Proposed image encryption

2.2. Privacy-Preserving Machine Learning

As mentioned above, two perceptual image encryption methods have been studied for privacy-preserving machine learning so far. The first [5–15] is applicable to tradition machine learning algorithms, such as support vector machine (SVM), k-nearest neighbors (KNN), and random forest even under the use of the kernel trick [5]. However, its block-based encryption method has never been applied to DNNs. The other [6] was applied to image classification with DNNs, but the accuracy is lower than that when using plain images, and the influence of data augmentation in the encrypted domain cannot be avoided yet. Examples of encrypted images are shown in Fig. 1.

Alternatively, privacy-preserving machine learning methods with homomorphic encryption (HE) [21–24] have been studied. One is CryptoNet [23], which can apply HE to the influence stage of CNNs. CryptoNet has very high computational complexity, so a dedicated low computer convolution core architecture for CryptoNet was proposed and implemented with a CMOS technology [24]. In CryptoNet, all activation functions and the loss function must be polynomial functions. Therefore, it cannot be applied to state-of-the-art DNNs. Moreover, CryptoNet does not allow us to carry out data augmentation in the encrypted domain, in addition to the high computation complexity.

One approach with HE has been proposed for privacy-preserving weight transmission for multiple owners who wish to apply a machine learning method over combined data sets [21, 22]. However, this approach can not be applied to network training in the encrypted domain.

In this paper, we aim to consider a method that enables not only network training in the encrypted domain but also data augmentation in the encrypted domain. This paper is among the first to discuss data augmentation in the encrypted domain.

3. PROPOSED METHODS

3.1. Overview of Privacy Preserving DNNs

Figure 2 illustrates the scenario used in this paper. In the training process, a client u encrypts all training images $(I_1, I_2, \ldots, I_g)$ to protect the visual information of the training images by using a secret key set, $\boldsymbol{K}$, and sends the encrypted images $(I_{e,1}, I_{e,2}, \ldots, I_{e,g})$ to a cloud server.

In the testing process, the client u encrypts a testing image (I_v) by using a secret key set, $\boldsymbol{K}$, and sends the encrypted image $I_{e,v}$ to a server. The server solves a classification problem with an image classification model trained in advance, and then returns the classification results to the client.

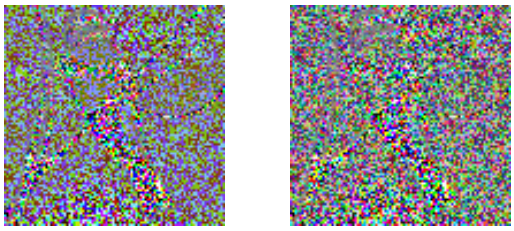
Note that the server has no secret key, so clients are able to control the privacy of images by themselves even when the classification process is done in the server. As shown in Fig. 2, data augmentation is carried out in both the client and the server, or by the server. In other words, data augmentation can be done after encryption. However, for all conventional encryption methods, data augmentation has to be done before encryption.

3.2. Proposed Image Encryption

We present a novel perceptual image encryption method that aims not to only relax the limitations of using encrypted images in DNNs but to also use data augmentation in the encrypted domain. In the block-based encryption [6], the number of feature maps has to be reduced due to block adaptation. In contrast, the proposed encryption is a pixel-based encryption method that enables an adaptation

Table 1: Permutation of color components for random integer. For example, if random integer is equal to 2, red component is replaced by green one, and green component is replaced by red one while blue component is not replaced.

Random Integer	Three Color Channels		
	R	G	B
0	R	G	B
1	R	B	G
2	G	R	B
3	G	B	R
4	B	R	G
5	B	G	R



(a) Negative-positive transformation

(b) Negative-positive transformation and color shuffling

Fig. 4: Examples of images encrypted by proposed method

network with a small number of parameters, so the resolution of an encrypted image can be preserved. Moreover, the proposed encryption method can provide data robust against ciphertext-only attacks compared with the conventional one [6].

To generate an encrypted image (I_e) from a color image, I_{RGB}, the following steps are carried out, as shown in Fig. 3. Note that the color shuffling (Step 3) is an optional encryption step to enhance security.

1) Divide I_{RGB} with $X \times Y$ pixels into pixels.
2) Individually apply negative-positive transformation to each pixel of each color channel, I_R, I_G, and I_B, by using a random binary integer generated by secret keys $\boldsymbol{K_c} = \{K_R, K_G, K_B\}$. In this step, a transformed pixel value of the i-th pixel, p', is calculated using

$$p' = \begin{cases} p & (r(i) = 0) \\ p \oplus (2^L - 1) & (r(i) = 1) \end{cases}, \tag{1}$$

where $r(i)$ is a random binary integer generated by K_c. p is the pixel value of the original image with L bit per pixel. The value of the occurrence probability $P(r(i)) = 0.5$ is used to invert bits randomly [13].
3) (Optional) Shuffle three color components of each pixel by using an integer randomly selected from six integers generated by a key K_s as shown in Table 1.

Examples of images encrypted by using the proposed scheme are shown in Fig. 4, where Fig. 1(a) is the original one. It is proved that the visual information of images was protected as in Fig. 1.

3.3. Data Augmentation

To solve complex tasks, a large amount of data is necessary to train DNNs. Data augmentation aims to enlarge the number of data points used for training and enables us to avoid the overfitting of DNNs. Many data augmentation techniques have already been proposed, e.g., horizontal/vertical flip, random crop, random rotation, cutout, and random erasing [25]. Although data augmentation is very useful for increasing the performance of DNNs, there are no image encryption methods that consider data augmentation. For this reason, data augmentation has to be carried out in each client before encryption when privacy-preserving DNNs are employed.

In this paper, we consider performing augmentation in a cloud server, namely, images are augmented after encryption. Here, the following well-known techniques are utilized for data augmentation:

- *Horizontal/vertical* flip: flips original images horizontally or vertically.
- *Shifting:* shifts pixel locations of original images on both horizontal and vertical axes by number of pixels.

In Section 4, the effect of data augmentation in a cloud server is experimentally analyzed. The results show that the block-based encryption method [6] was heavily affected by carrying out the augmentation in the encrypted domain. In contrast, the proposed encryption maintained a high classification performance, even when the augmentation was carried out after encryption.

3.4. Security Evaluation

Security mostly refers to protection from adversarial forces. Various attacking strategies, such as the known-plaintext attack (KPA) and chosen-plaintext attack (CPA), should be considered [11–13]. Here, we consider brute-force attacks as ciphertext-only attacks.

If an image with $X \times Y$ pixels is divided into pixels, the number of pixels n is given by

$$n = X \times Y. \tag{2}$$

The key spaces of negative-positive transformation (N_{np}) and color component shuffling (N_{col}) are represented by

$$N_{np}(n) = 2^{3n}, N_{col}(n) = \left({}_3P_3\right)^n = 6^n. \tag{3}$$

Consequently, the key space of images encrypted by using the proposed encryption scheme, $N(n)$, is represented by the following.

$$\begin{aligned} N(n) &= N_{np}(n) \cdot N_{col}(n) \\ &= 2^{3n} \cdot 6^n \end{aligned} \tag{4}$$

In contrast, in Tanaka's method [6], I_{RGB} with $X \times Y$ pixels is divided into blocks each with 4×4 pixels, and each block is split into upper 4-bit and lower 4-bit images to generate 6-channel image blocks. Then, the intensities of randomly selected pixels are reversed. Eventually, the pixels in each block are shuffled with the same pattern.

The key space of Tanaka's method [6], N_{tanaka}, is given by

$$N_{tanaka} = 96! \cdot 2^{96}. \tag{5}$$

Therefore, since $N(n) >> N_{tanaka}$, the proposed encryption has a larger key space than Tanaka's method.

3.5. Adaptation Network

We propose a novel adaptation network for DNNs that aims to adapt images encrypted by the proposed encryption method to make the images compatible with DNNs. Since the proposed encryption method is a pixel-based one, the proposed adaptation network consists of simple 1×1-convolutional layers.

Figure 5 illustrates an adaptation network where $C_i^{M_i}$ is the i-th convolutional layer of the network with a kernel size and stride of

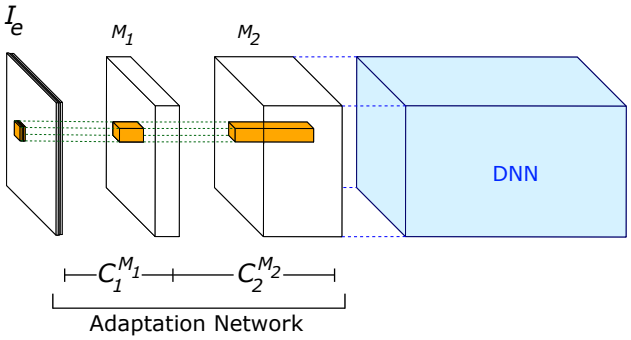


Fig. 5: Proposed adaptation network

Table 2: Image classification accuracy where data augmentation was carried out in cloud server. (ResNet-18)

Encryption	Accuracy (%)
Plain Image	92.98
Proposed (step 2) without Adaptation	85.15
Proposed (step 2) with Adaptation	**86.99**
Proposed (step 2 and 3) without Adaptation	82.51
Proposed (step 2 and 3) with Adaptation	86.16
Tanaka's Scheme [6]	56.41
EtC [9, 10]	69.03
Pixel-based [16]	58.59

(1,1) , and M_i is the number of feature maps of the i-th convolutional layer.

There are 48 possible patterns for each pixel in the proposed encryption, so each pixel has to be adjusted before DNNs are used. $C_i^{M_i}$ learns the patterns of each encrypted pixel, and the feature representations of each encrypted pixel are then obtained. As a result, the output of the adaptation network is applicable to any DNN.

4. EXPERIMENTS AND RESULTS

4.1. Experimental Set-up

To confirm that the proposed scheme is effective, we evaluated the performance in terms of image classification accuracy and compared it with conventional privacy-preserving methods.

We employed CIFAR10, which contains 32×32 pixel color images and consists of 50,000 training images and 10,000 test images in 10 classes [26]. For preprocessing, standard data augmentation (shifting and random horizontal flip) was used.

The network was trained by using stochastic gradient descent (SGD) with momentum for 300 epochs. The learning rate was initially set to 0.1 and was decreased by a factor of 10 at 150 and 225 epochs. We used a weight decay of 0.0005, a momentum of 0.9, and a batch size of 128.

4.2. Experimental Results

The proposed encryption was used to encrypt all training and testing images, and networks were then trained and tested by using the encrypted images, as shown in Fig. 2. In the experiment, the numbers of feature maps, M_1 and M_2, were set to 8 and 32, respectively, and

Table 3: Image classification accuracy where data augmentation was carried out in server or client. (ResNet-18)

Encryption	Data Augmentation	
	In Client	In Cloud
Proposed (Step 2) without Adaptation	**92.94**	85.15
Proposed (Step 2) with Adaptation	90.65	**86.99**
Tanaka's Scheme [6]	85.84	56.41

we evaluated the image classification accuracy of encrypted images under the use of ResNet-18 [27, 28], which consists of 18 layers.

A. Data Augmentation in Cloud Server

Table 2 shows the classification accuracy in the case that the data augmentation was carried out in cloud server. Although the proposed scheme had a lower classification accuracy than that for non-encrypted images, it still maintained a high classification performance compared with the conventional methods. Moreover, the use of the adaptation network was confirmed to improve the classification accuracy for two types of encryption: (Step 2) and (Step 2 and 3).

B. Data Augmentation in Client or Cloud Server

Table 3 shows that the accuracy of the proposed scheme was higher than that of Tanaka's scheme even when the data augmentation was carried out in a client. In addition, the proposed scheme was confirmed to maintain a high classification performance, although Tanaka's scheme was heavily affected by the data augmentation.

C. Effects of Adaptation Network

When the data augmentation was applied to the images encrypted by using the conventional encryption [6], the kernel of the block adaptation layer overlapped with the adjacent encrypted blocks. As a result, the classification accuracy of the conventional encryption was heavily degraded. In comparison, the proposed encryption resulted in superior robustness against augmentation due to the use of pixel-based encryption and a kernel size of (1,1).

5. CONCLUSION

We presented a novel privacy preserving scheme for DNNs that enables us not only to use encrypted images on DNNs but to also use data augmentation in the encrypted domain. This paper was the first to discuss data augmentation in the encrypted domain. Novel pixel-based image encryption was proposed to protect the visual information of images and is available for training and testing DNNs. In addition, we proposed a novel adaptation network that enhances the classification performance of DNNs by obtaining representations of each pixel before passing through state-of-the-art DNNs. The experimental results showed that the proposed scheme performed better in classification than did the conventional block-based and pixel-based encryption schemes. In addition, it was proved that the proposed encryption allows us to use data augmentation in the encrypted domain. As a result, the advantages of data augmentation for enhancing the performance of DNNs can be efficiently utilized even when visual information is protected.

6. REFERENCES

[1] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional ac-

tivation feature for generic visual recognition," in *Proceedings of the 31st International Conference on Machine Learning*, Bejing, China, 22–24 Jun 2014, vol. 32, pp. 647–655.

[2] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, USA, 2012, pp. 1097–1105.

[3] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *2015 IEEE Information Theory Workshop (ITW)*, April 2015, pp. 1–5.

[4] A.M. Saxe, Y. Bansal, J.D., M. Advani, A. Kolchinsky, B.D. Tracey, and D.D. Cox, "On the information bottleneck theory of deep learning," in *International Conference on Learning Representations*, May 2018, pp. 1–27.

[5] T. Maekawa, A. Kawamura, Y. Kinoshita, and H. Kiya, "Privacy-preserving svm computing in the encrypted domain," in *Proceedings of APSIPA Annual Summit and Conference*, 2018, pp. 897–902.

[6] M. Tanaka, "Learnable image encryption," in *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, May 2018, pp. 1–2.

[7] I. Ito and H. Kiya, "One-time key based phase scrambling for phase-only correlation between visually protected images," *EURASIP Journal on Information Security*, vol. 2009, no. 841045, pp. 1–11, 2010.

[8] Z. Tang, X. Zhang, and W. Lan, "Efficient image encryption with block shuffling and chaotic map," *Multimedia Tools Applications*, vol. 74, no. 15, pp. 5429–5448, 2015.

[9] K. Kurihara, S. Shiota, and H. Kiya, "An encryption-then-compression system for jpeg standard," in *Picture Coding Symposium (PCS)*, 2015, pp. 119–123.

[10] K. Kurihara, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for lossless image compression standards," *IEICE Transactions on Information and Systems*, vol. E100-D, no. 1, pp. 52–56, 2017.

[11] T. Chuman, K. Kurihara, and H. Kiya, "On the security of block scrambling-based etc systems against jigsaw puzzle solver attacks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2157–2161.

[12] T. Chuman, K. Kurihara, and H. Kiya, "On the security of block scrambling-based etc systems against extended jigsaw puzzle solver attacks," *IEICE Transactions on Information and Systems*, vol. E101-D, no. 1, 2017.

[13] T. Chuman, W. Sirichotedumrong, and H. Kiya, "Encryption-then-compression systems using grayscale-based image encryption for jpeg images," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1515–1525, 2019.

[14] W. Sirichotedumrong and H. Kiya, "Grayscale-based block scrambling image encryption using YCbCr color space for encryption-then-compression systems," *APSIPA Transactions on Signal and Information Processing*, vol. 8, pp. e7, 2019.

[15] V. Itier, P. Puteaux, and W. Puech, "Recompression of jpeg crypto-compressed images without a key," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2019.

[16] M. T. Gaata and F. F. Hantoosh, "An efficient image encryption technique using chaotic logistic map and rc4 stream cipher," *International Journal of Modern Trends in Engineering and Research*, vol. 3, no. 9, pp. 213–218, 2016.

[17] T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara, "High-throughput semi-honest secure three-party computation with an honest majority," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, CCS '16, pp. 805–817.

[18] T. Araki, A. Barak, J. Furukawa, T. Lichter, Y. Lindell, A. Nof, K. Ohara, A. Watzman, and O. Weinstein, "Optimized honest-majority mpc for malicious adversaries breaking the 1 billion-gate per second barrier," in *2017 IEEE Symposium on Security and Privacy (SP)*, May 2017, pp. 843–862.

[19] W. Lu, S. Kawasaki, and J. Sakuma, "Using fully homomorphic encryption for statistical analysis of categorical, ordinal and numerical data," *IACR Cryptology ePrint Archive*, vol. 2016, pp. 1163, 2016.

[20] Y. Aono, T. Hayashi, L.T. Phong, and L. Wang, "Privacy-preserving logistic regression with distributed data sources via homomorphic encryption," *IEICE Transactions on Information and Systems*, vol. E99.D, no. 8, pp. 2079–2089, 2016.

[21] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1310–1321.

[22] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1333–1345, May 2018.

[23] N. Dowlin, R. Gilad-Bachrach, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *Microsoft Research Techincal Report*, February 2016.

[24] Y. Wang, J. Lin, and Z. Wang, "An efficient convolution core architecture for privacy-preserving deep learning," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2018, pp. 1–5.

[25] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *CoRR*, vol. abs/1708.04896, 2017.

[26] A. Krizhevsky, "Learning multiple layers of features from tiny images," Tech. Rep., 2009.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[28] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, May 2018, pp. 1–13.