# Privacy-Preserving Deep Neural Networks Using Pixel-Based Image Encryption Without Common Security Keys

Warit Sirichotedumrong, 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 but to also consider the use of independent encryption keys, for both training and testing images for the first time. In this paper, a novel pixel-based image encryption method, which considers maintaining the properties of original images, is first proposed for privacy-preserving DNNs. For training, a DNN model is trained with images encrypted by using the proposed method under the use of independent keys. For testing, the model enables us to applied both encrypted images and plain images for image classification. Therefore, there is no need to manage the keys. In an experiment, the proposed method is applied to a well-known network, deep residual networks, for image classification. The experimental results demonstrate that the proposed method with independent encryption keys has robustness against ciphertext-only attack (COA) and can provide almost the same classification performance as that of using plain images. Moreover, the results confirm that the proposed scheme is able to classify plain images as well as encrypted images.**

## I. 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. Moreover, DNNs can be vulnerable if adversaries carry out model inversion attacks [5], [6] to obtain trained data from the trained model. Therefore, privacy-preserving DNNs have become an urgent challenge.

Various methods have been proposed for privacy-preserving computation. The methods are classified into two types: perceptual encryption-based [7]–[20] and homomorphic encryption (HE)-based [21]–[29]. As described in Section II, HE-based methods are the most secure options for privacy preserving computation, but they are applied to only limited DNNs [25]–[29]. 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 [7]–[20]. A few methods were applied to machine learning algorithms in previous works [7], [8]. The first encryption method [11]–[17], which has been proposed for encryption-then-compression (EtC) systems, was demonstrated to be applicable to traditional machine learning algorithms, such as support vector machine (SVM) [7]. However, the block-based encryption method has never been applied to DNNs. Another method [8] 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 classification accuracy is lower than that of plain images, and it is proved that data augmentation in the encrypted domain cannot be applied to Tanaka's scheme [20]. Although a pixel-based image image encryption method [20] was proposed not only to improve the classification performance of the privacy-preserving DNNs but also to consider data augmentation in the encrypted domain, training and testing images are encrypted under only the use of common security keys. In addition, the security level of the encryption method is only evaluated in terms of key space analysis for brute-force attack.

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 consider the use of independent encryption keys, which means that all images are encrypted by using different security keys, for the first time. Moreover, the proposed method provides the availability for clients to classify plain images although DNNs are trained by encrypted images.

In an experiment, we compare the proposed method with conventional perceptual encryption-based methods. The experimental results show that the proposed method with independent encryption keys has robustness against COA so that the visual information cannot be reconstructed. In addition, the proposed methods with DNNs performs better in classification than conventional block-based and pixel-based encryption schemes and can provide almost the same classification performance as plain images. We also confirm that the proposed scheme is able to classify plain images as well as encrypted images.

## II. Related Works

### A. Visual Information Protection

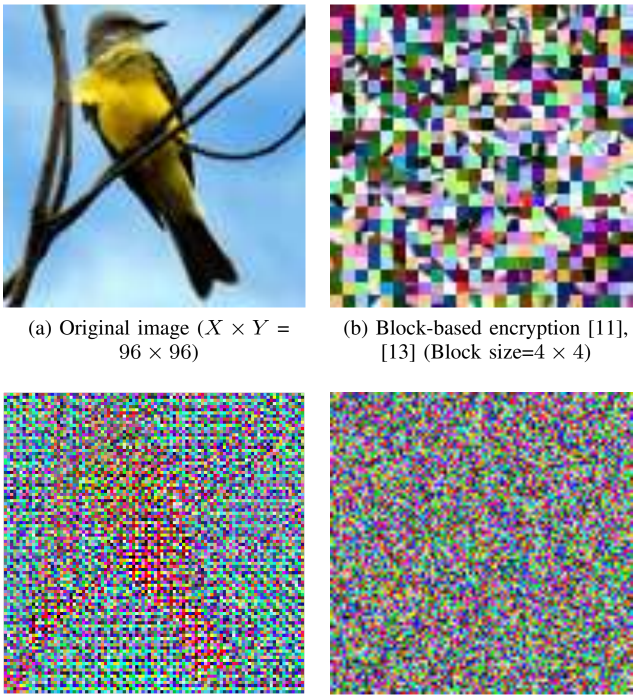
Security mostly refers to protection from adversarial forces. This paper focuses on protecting visual information that allows us to identify an individual, the time, and the location of the taken photograph. Untrusted platforms and unauthorized users are assumed to be adversaries. Moreover, images used for training a DNN model can be reconstructed from the DNN model [5], [6]. If the model is trained with the images without visual information, visual information of the trained images is still protected although the model inversion attacks are carried out.

Various perceptual image encryption methods [7]–[19] 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 [11]–[18]. 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 [7], [8]. Although a pixel-based image image encryption method [20] has been proposed to improve the classification performance of the privacy-preserving DNNs and consider data augmentation in the encrypted domain, training and testing images are encrypted under only the use of common security keys.

### B. Privacy-Preserving Machine Learning

As mentioned above, three perceptual image encryption methods have been studied for privacy-preserving machine learning so far [7]–[18], [20]. The first encryption [7], [9]–[18], which has been proposed for EtC systems, 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 [7]. However, its block-based encryption method has never been applied to DNNs. The other [8] 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. In contrast, the pixel-based image image encryption method [20] has been proposed to improve the classification performance of the privacy-preserving DNNs and consider data augmentation in the encrypted domain. However, the existing privacy-preserving DNNs train the DNNs under the use of common security keys, so the key management is required. Examples of encrypted images are shown in Fig. 1.

Alternatively, privacy-preserving machine learning methods with homomorphic encryption (HE) [25]–[29] have been studied. One is CryptoNet [28], 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 [29]. 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.


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

(b) Block-based encryption [11], [13] (Block size=$4 \times 4$)

(c) Block-based encryption [8] (Block size=$4 \times 4$)

(d) Pixel-based image encryption [19]

Fig. 1: Examples of images encrypted by conventional schemes

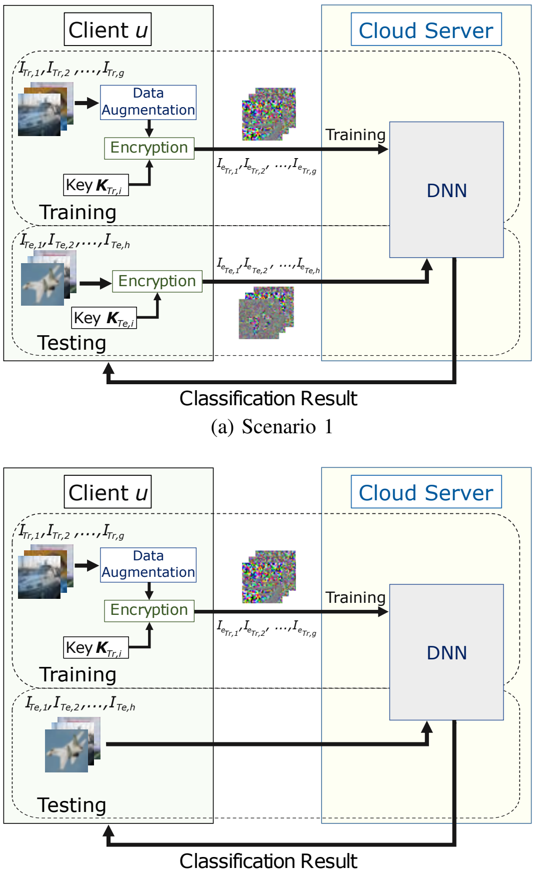
One approach with HE has been proposed for privacy-preserving weight transmission for multiple owners who wish to apply a machine learning method over combined data sets [25]–[27]. However, this approach can not be applied to network training in the encrypted domain.

In this paper, we aim to propose the novel privacy-preserving DNN that enables us to train a DNN model with images encrypted by using the pixel-based image encryption under the use of independent encryption keys. Then, the model is applied with encrypted images or plain images to obtain the classification results. As a result, the key management is not required by the proposed privacy-preserving DNNs.

## III. Proposed method

### A. Overview of Privacy Preserving DNNs

Figure 2 illustrates the scenarios used in this paper. In the training process, data augmentation is first carried out to each training image, $I_{Tr,i}$, $i = 1, 2, \ldots, g$. Then, a client $u$ encrypts the training images to protect the visual information of the


Client $u$
Cloud Server
$I_{Tr,1}, I_{Tr,2}, \dots, I_{Tr,g}$
Data Augmentation
Encryption
Key $\boldsymbol{K}_{Tr,i}$
Training
$I_{e_{Tr,1}}, I_{e_{Tr,2}}, \dots, I_{e_{Tr,g}}$
Training
DNN
$I_{Te,1}, I_{Te,2}, \dots, I_{Te,h}$
Encryption
Key $\boldsymbol{K}_{Te,i}$
$I_{e_{Te,1}}, I_{e_{Te,2}}, \dots, I_{e_{Te,h}}$
Testing
Classification Result

(a) Scenario 1

(b) Scenario 2

Fig. 2: Scenario

training images by using a training secret key set, $\boldsymbol{K_{Tr,i}}$, and sends the encrypted images ($I_{e_{Tr,i}}$) to a cloud server.

There are two testing scenarios used in this paper.

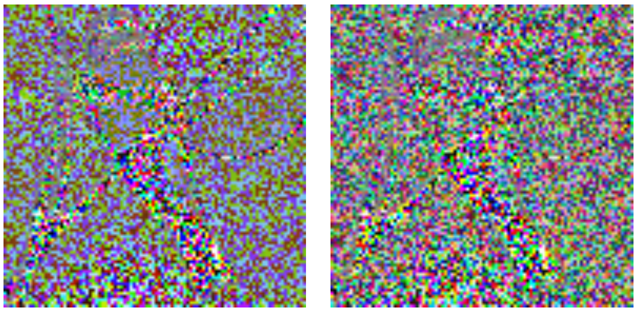
- ***Scenario 1:*** The client $u$ encrypts testing images, $I_{Te,i}$, $i = 1, 2, \dots, h$, by using a testing secret key set, $\boldsymbol{K_{Te,i}}$, and sends the encrypted image $I_{e_{Te,i}}$ to a server, as shown in Fig. 2(a).
- ***Scenario 2:*** The client $u$ sends the plain image $I_{Te,i}$ to a server, as shown in Fig. 2(b).

Then, 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.

Although the conventional privacy-preserving DNNs [20] considers to encrypt training and testing images by using a common security key, we assume that there are two encryption key conditions for generating encrypted images as follows.

- ***Same encryption key:*** Like the conventional method [20], all training and testing images are encrypted by using only one secret key, i.e. $\boldsymbol{K}_{Tr,1} = \boldsymbol{K}_{Tr,2} = \dots = \boldsymbol{K}_{Tr,g} = \boldsymbol{K}_{Te,i} = \boldsymbol{K}$.
- ***Different encryption keys:*** The different secret keys are independently assigned to training and testing images, i.e. $\boldsymbol{K}_{Tr,1} \neq \boldsymbol{K}_{Tr,2} \neq \dots \neq \boldsymbol{K}_{Tr,g} \neq \boldsymbol{K}_{Te,i}$.



(a) Negative-positive transformation (b) Negative-positive transformation and color shuffling

Fig. 3: Examples of images encrypted by the pixel-based image encryption method

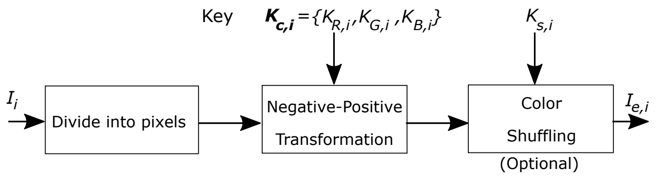

Key
$\boldsymbol{K_{c,i}} = \{K_{R,i}, K_{G,i}, K_{B,i}\}$
$K_{s,i}$
$I_i$
Divide into pixels
Negative-Positive Transformation
Color Shuffling
(Optional)
$I_{e,i}$

Fig. 4: Proposed image encryption

### B. Proposed Image Encryption

In this section, we present a novel perceptual image encryption method that aims not only to relax the limitations of using encrypted images in DNNs but to also enhance security.

To generate an encrypted image ($I_{e,i}$) from a color image, $I_i$, the following steps are carried out, as shown in Fig. 4. Note that the color shuffling (Step 3) is an optional encryption step to enhance security.

1) Divide $I_i$ with $X \times Y$ pixels into pixels.
2) Individually apply negative-positive transformation to each pixel of each color channel, $I_{R,i}$, $I_{G,i}$, and $I_{B,i}$, by using a random binary integer generated by secret keys $\boldsymbol{K_{c,i}} = \{K_{R,i}, K_{G,i}, K_{B,i}\}$. In this step, a transformed pixel value of the $j$-th pixel, $p'$, is calculated using
$$p' = \begin{cases} p & (r(j) = 0) \\ p \oplus (2^L - 1) & (r(j) = 1) \end{cases}, \tag{1}$$
where $r(j)$ is a random binary integer generated by $\boldsymbol{K_{c,i}}$. $p$ is the pixel value of the original image with $L$ bit per pixel. The value of the occurrence probability $P(r(j)) = 0.5$ is used to invert bits randomly [16].
3) (Optional) Shuffle three color components of each pixel by using an integer randomly selected from six integers generated by a key $K_{s,i}$ as shown in Table I.

Images encrypted by using the pixel-based method are illustrated in Fig. 3, where Fig. 1(a) is the original one. It is proved that the visual information of images was protected as in Fig. 1(d).

TABLE I: 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 |

### C. Properties of Encrypted Images

Image encryption methods for privacy-preserving DNNs have to meet the following requirements.

- ***Visual information protection:*** to protect an individual, the time, and the location of the taken photograph.
- ***Security:*** to provide the robustness against COA.
- ***Low damage to DNNs:*** to maintain the performance of DNNs as plain images.

Although block-based encryption methods [7], [9]–[17] can protect the visual information, the encryption causes much damage to DNNs due to the loss of positional information in the spatial domain, so the performance of DNNs is heavily decreased compared with the plain images.

In comparison, the block-based encryption proposed by Tanaka [8] can reduce the damage to DNNs by using an adaptation network. However, the block-based encryption including Tanaka's method [7]–[17] are not secure enough against COA if training and testing images have low resolutions.

The conventional pixel-based encryption [19] can protect the visual information, as shown in Fig. 1(d), and it can provide a higher security level against COA than the block-based encryption even when images have low resolutions. However, it causes much damage to DNNs because the pixel-based method does not consider maintaining the properties of original images.

In contrast, the proposed pixel-based encryption considers maintaining the properties of original images to reduce the damage to DNNs. The encryption steps, negative-positive transformation and color component shuffling, can be expressed as an orthogonal transformation, so the encrypted images can maintain the relation among original images [7]. Moreover, as the proposed encryption is a pixel-based one, high level of security is maintained even if images have low resolutions. Hence, the proposed encryption is expected to outperform the conventional encryption methods.

### D. Robustness against ciphertext-only Attacks

Security mostly refers to protection from adversarial forces. In this paper, we consider brute-force attack and propose a novel DNN-based COA for the pixel-based image encryption.

*1) Brute-force Attack:* If $I_i$ with $X \times Y$ pixels is divided into pixels, the number of pixels $n$ is given by

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

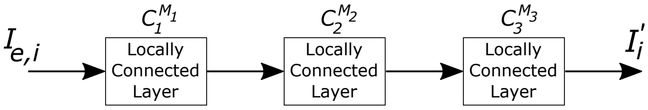


Fig. 5: Proposed DNN-based COA

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 [8], $I_i$ with $X \times Y$ pixels is divided into blocks each with $4 \times 4$ 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 [8], $N_{tanaka}$, is given by

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

$N(n)$ is equal to $N_{tanaka}$ when $n$ is approximately equal to 106.4. Therefore, the proposed encryption has a larger key space than Tanaka's method if $X \times Y = 11 \times 11$ pixels.

*2) DNN-based ciphertext-only Attack:* we propose a novel DNN-based COA that aims to reconstruct the visual information of encrypted images. Since the encryption method is a pixel-based one, the proposed DNN for COA consists of three $1\times1$-locally connected layers, which work similarly to $1\times1$-convolution layer, except that weights are unshared. Figure 5 illustrates the proposed attack where $C_k^{M_k}$ is the $k$-th locally connected layer of the network with a kernel size and stride of (1,1), $M_k$ is the number of feature maps of the $k$-th locally connected layer, $k \in \{1, 2, 3\}$, and $I_i'$ denotes a reconstructed image. The representations of each encrypted pixel are extracted in the first two layers, and then the reconstructed pixels are obtained by the last layer.

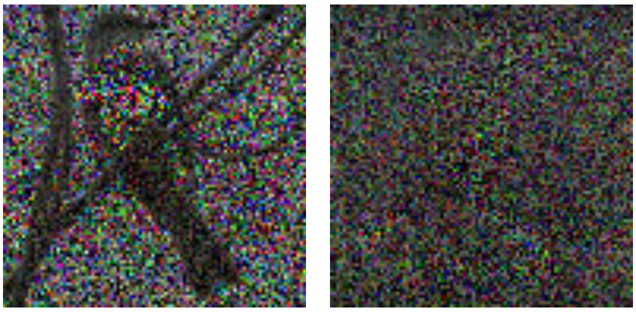
## IV. Experiments

### A. Robustness against ciphertext-only Attack

*1) Experimental Conditions:* We employed STL-10 dataset, which contains $96\times96$ pixel color images and consists of 5,000 training images and 8,000 testing images [30].

In the experiment, the numbers of feature maps, $M_1$, $M_2$, and $M_3$, were set to 8, 32, and 3, respectively.

The network was trained by using stochastic gradient descent (SGD) with momentum for 70 epochs, and used mean squared error (MSE), which compares the differences between the reconstructed images and the original ones, as a loss function. The learning rate was initially set to 0.1 and decreased by a factor of 10 at 40 and 60 epochs. We used a



(a) Same encryption key (b) Different encryption keys

Fig. 6: Examples of reconstructed images from the images encrypted by the negative-positive transformation and color shuffling.

TABLE II: Average SSIM of the reconstructed images compared with the original ones.

| Key Conditions | Encryption | SSIM |
|---|---|---|
| Same encryption key | Step 2 | **0.1732** |
| | Step 2 and 3 | 0.1715 |
| Different encryption keys | Step 2 | 0.0424 |
| | Step 2 and 3 | 0.0425 |

weight decay of 0.0005, a momentum of 0.9, and a batch size of 128.

*2) Results:* Examples of reconstructed images under the use of same and different encryption keys are shown in Fig. 6, where Fig. 1(a) is the original one. The visual information of the reconstructed images was recovered by the proposed scheme if the images are encrypted under same encryption key, as shown in Fig. 6(a). This is because each image was encrypted with only one pattern, so the proposed attack can recognize the pattern and recover the visual information by comparing the difference between reconstructed images and original images. In comparison, the pixel-based encryption method has robustness against COA if the training images are encrypted by using different encryption keys. Therefore, the visual information cannot be recovered, as shown in Fig. 6(b).

Table II shows that the structural similarity (SSIM) values of the encrypted images under the use of same encryption key were much higher than under the use of different encryption keys.

### B. Image Classification

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.

*1) Experimental Conditions:* We employed CIFAR10, which contains $32 \times 32$ pixel color images and consists of 50,000 training images and 10,000 test images in 10 classes [31]. Standard data augmentation (shifting and random horizontal flip) was used.

The network was trained by using 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.

TABLE III: Image classification accuracy when testing with encrypted images $I_{e_{Te,i}}$. (ResNet-18)

| Encryption | Accuracy (%) | |
|---|---|---|
| | Same Key | Different Keys |
| Plain Image | 93.52 | |
| Proposed (step 2) | 93.01 | 92.96 |
| Proposed (steps 2 and 3) | 90.85 | 90.97 |
| Tanaka's Scheme [8] | 90.41 | 32.69 |
| Pixel-based [19] | 72.15 | 67.91 |
| EtC [11], [12] | 82.32 | 52.56 |

TABLE IV: Image classification accuracy when testing with plain images $I_{Te,i}$. (ResNet-18)

| Training Data | | Accuracy (%) |
|---|---|---|
| Plain Image | | 93.52 |
| Proposed (step 2) | Same Key | 91.47 |
| | Different Keys | 92.16 |
| Proposed (steps 2 and 3) | Same Key | 87.62 |
| | Different Keys | 90.46 |
| Tanaka's Scheme [8] | Same Key | 26.68 |
| | Different Keys | 39.39 |

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. We utilized two encryption key conditions to evaluate how different encryption keys affect the image classification performance. We evaluated the image classification accuracy of encrypted images under the use of deep residual networks (ResNet-18) [32], [33], which consists of 18 layers.

*2) Results:* Table III shows that the classification accuracy when the trained model is tested by $I_{e_{Te,i}}$. The results showed that the proposed method outperformed other encryption methods even when images were encrypted under different encryption keys. In comparison, the accuracy of other methods were heavily degraded when images were encrypted under different encryption keys. Therefore, it was proved that the proposed method causes low damage to DNNs because the proposed encryption not only provides the comparable classification accuracy as that of plain images but also enables the use of different encryption keys without any effects to the classification performance.

Table IV shows that the classification accuracy when the trained model is tested by $I_{Te,i}$. It was confirmed that the proposed method under the use of different keys enables us not to only test DNNs with plain images but to also outperform the method under the use of same key. In addition, under the use of different keys, the classification accuracy when testing with $I_{Te,i}$ was almost the same as that when testing with $I_{e_{Te,i}}$. In comparison, the models trained by Tanaka's method [8] are not able to classify $I_{Te,i}$ As a result, it was proved that the proposed method maintains the important properties of plain images even when the visual information is protected.

## V. Conclusion

We presented a novel privacy-preserving scheme for deep neural networks (DNNs) that enables us not to only apply images without visual information to DNNs but to also consider the use of different encryption keys for both training and testing images for the first time. The novel pixel-based image encryption was proposed to protect the visual information of images and be available for training and testing DNNs. In addition, the proposed privacy-preserving scheme for DNNs allowed us to train a DNN model with encrypted images, and then test it with plain images without any classification performance degradation. In an experiment, we evaluated the performance of the proposed method in terms of the robustness against COA, and the classification accuracy. The experimental results demonstrated that the proposed method with different encryption keys has robustness against COA; therefore, the visual information cannot be reconstructed. Moreover, the proposed method can provide almost the same classification performance as that of using plain images even when the training images and testing images are encrypted by using different encryption keys. The results confirmed that the proposed scheme is able to classify plain images without any performance degradation, as well as classifying encrypted images.

## Acknowledgment

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

## References

[1] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, “Decaf: A deep convolutional activation feature for generic visual recognition,” Proceedings of the 31st International Conference on Machine Learning, Proceedings of Machine Learning Research (PMLR), vol.32, Bejing, China, pp.647–655, 2014.

[2] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” Proceedings of the 25th International Conference on Neural Information Processing Systems, pp.1097–1105, 2012.

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

[4] A. Saxe, Y. Bansal, J.D., M. Advani, A. Kolchinsky, B. Tracey, and D. Cox, “On the information bottleneck theory of deep learning,” International Conference on Learning Representations, 2018.

[5] M. Fredrikson, S. Jha, and T. Ristenpart, “Model inversion attacks that exploit confidence information and basic countermeasures,” Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, pp. 1322–1333, 2015.

[6] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, “Model inversion attacks that exploit confidence information and basic countermeasures,” USENIX Security Symposium, pp. 17–32, 2014.

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

[8] M. Tanaka, “Learnable image encryption,” 2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), pp.1–2, May 2018.

[9] 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.

[10] 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.

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

[12] 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.

[13] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, “An encryption-then-compression system for jpeg/motion jpeg standard,” IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.98, no.11, pp.2238–2245, 2015.

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

[15] 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.

[16] 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.

[17] 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, p.e7, 2019.

[18] 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, 2019.

[19] 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.

[20] W. Sirichotedumrong, T. Maekawa, Y. Kinoshita, and H. Kiya, “Privacy-preserving deep neural networks with pixel-based image encryption considering data augmentation in the encrypted domain,” in *IEEE International Conference on Image Processing (ICIP)*, September 2019, to be published. [Online]. Available: http://arxiv.org/abs/1905.01827

[21] T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara, “High-throughput semi-honest secure three-party computation with an honest majority,” Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS ’16, New York, NY, USA, pp.805–817, ACM, 2016.

[22] 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,” 2017 IEEE Symposium on Security and Privacy (SP), pp.843–862, May 2017.

[23] 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, p.1163, 2016.

[24] Y. Aono, T. Hayashi, L. 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.

[25] R. Shokri and V. Shmatikov, “Privacy-preserving deep learning,” Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, CCS ’15, New York, NY, USA, pp.1310–1321, 2015.

[26] 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.

[27] L. Phong and T. Phuong, “Privacy-preserving deep learning for any activation function,” CoRR, vol.abs/1809.03272, 2018.

[28] 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,” tech. rep., February 2016.

[29] Y. Wang, J. Lin, and Z. Wang, “An efficient convolution core archi-

tecture for privacy-preserving deep learning," 2018 IEEE International Symposium on Circuits and Systems (ISCAS), pp.1–5, May 2018.

[30] A. Coates, A. Ng, and H. Lee, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, vol. 15, 2011, pp. 215–223.

[31] A. Krizhevsky, "Learning multiple layers of features from tiny images," tech. rep., 2009.

[32] 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)*, pp. 770–778, June 2016.

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