Access Control Using Spatially Invariant Permutation of Feature Maps for Semantic Segmentation Models

Hiroki Ito, MaungMaung AprilPyone, and Hitoshi Kiya
Tokyo Metropolitan University, Japan

***Abstract*—In this paper, we propose an access control method that uses the spatially invariant permutation of feature maps with a secret key for protecting semantic segmentation models. Segmentation models are trained and tested by permuting selected feature maps with a secret key. The proposed method allows rightful users with the correct key not only to access a model to full capacity but also to degrade the performance for unauthorized users. Conventional access control methods have focused only on image classification tasks, and these methods have never been applied to semantic segmentation tasks. In an experiment, the protected models were demonstrated to allow rightful users to obtain almost the same performance as that of non-protected models but also to be robust against access by unauthorized users without a key. In addition, a conventional method with block-wise transformations was also verified to have degraded performance under semantic segmentation models.**

I. Introduction

Deep neural networks (DNNs) have led to major breakthroughs in computer vision for a wide range of applications. Convolutional neural networks (CNNs) are a type of DNN. Current commercial applications for image recognition, object detection, and semantic segmentation are primarily powered by CNNs [1], [2]. Therefore, CNNs have become the de facto standard for visual recognition systems for many different applications. However, training successful CNNs requires three ingredients: a huge amount of data, GPU-accelerated computing resources, and efficient algorithms, and this is not a trivial task. Therefore, trained CNNs have great business value. Considering the expenses necessary for the expertise, money, and time taken to train a CNN model, a model should be regarded as a kind of intellectual property (IP).

There are two aspects of IP protection for DNN models: ownership verification and access control. The former focuses on identifying the ownership of models, and the latter addresses protecting the functionality of DNN models from unauthorized access. Ownership verification methods are inspired by digital watermarking, and they embed watermarks into DNN models so that the embedded watermarks can be used to verify the ownership of the models in question [3]–[11].

Although the above watermarking methods can facilitate the identification of the ownership of models, in reality, a stolen model can be exploited in many different ways. For example, an attacker can use a model for their own benefit without arousing suspicion, or a stolen model can be used for model inversion attacks [12] and adversarial attacks [13]. Therefore, it is crucial to investigate mechanisms for protecting DNN models from unauthorized access and misuse. In this paper, we focus on protecting a model from misuse when it has been stolen (i.e., access control).

A method for protecting a model against unauthorized access was inspired by adversarial examples [13]–[15] and image encryption [16]–[21], and it was proposed to utilize secret perturbation to control the access of a model [22]. Another study introduced a secret key for protecting a model [23], [24]. The secret key-based protection method [24] uses a key-based transformation that was originally used by an adversarial defense in [25], which was in turn inspired by perceptual image encryption methods [20], [21], [26]–[31]. This model protection method utilizes a secret key in such a way that a stolen model cannot be used to its full capacity without a correct secret key.

However, these methods were evaluated only on image classification tasks, and it is not known how well they perform on other advanced tasks. Therefore, in this paper, we consider protecting semantic segmentation models from unauthorized access for the first time, and we propose a novel access control method that uses the spatially invariant permutation of feature maps on the basis of a secret key. The proposed method allows rightful users with the correct key to access a model to full capacity and degrade the performance for unauthorized users. In experiments on semantic segmentation, we evaluated the access control performance of models trained by using feature map permutation. The results show that the protected models provided almost the same segmentation performance as that of non-protected models against authorized access, while the segmentation accuracy seriously dropped when an incorrect key was given. Furthermore, the conventional method proposed for image classification tasks [24] was demonstrated to have degraded performance under semantic segmentation tasks.

II. Proposed Method

A. Overview

Figure 1 illustrates an overview of access control for protecting semantic segmentation models from unauthorized access. A protected model is prepared by training a network with secret key K. Authorized users input test images into the protected model with correct key K, in which the model provides

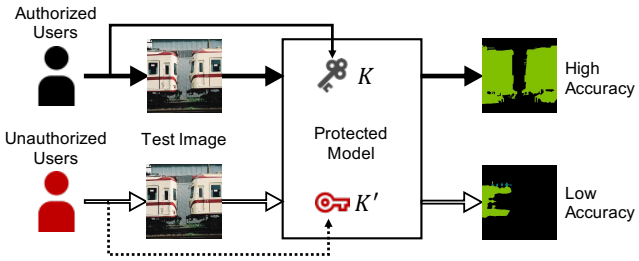

Fig. 1. Overview of access control

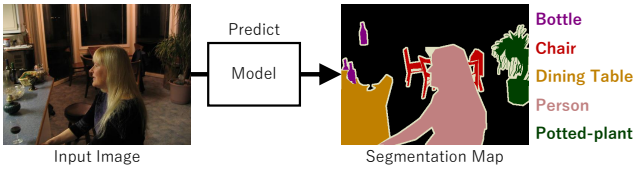

Fig. 2. Example of semantic segmentation

almost the same segmentation map as that predicted by using a non-protected model. In contrast, when unauthorized users who do not know key K input test images into a protected model without any key or with incorrect (estimated) key K', the model provides a degraded map.

B. Semantic Segmentation

The goal of semantic segmentation is to understand what is in an image at the pixel level. Figure 2 shows an example of semantic segmentation. The segmentation model predicts a segmentation map from an input image, where each pixel in the segmentation map represents a class label.

The mean intersection-over-union (mean IoU) [32], [33] is used as a metric for evaluating the segmentation performance. An IoU value is given for each class by

$$\text{IoU} = \frac{TP}{TP + FP + FN} \quad , \tag{1}$$

and the mean IoU is then calculated by averaging IoU values of all classes. TP, FP, and FN mean true positive, false positive, and false negative values calculated from predicted segmentation maps and ground truth ones, respectively. In addition, the metric ranges from zero to one, where a value of one means that predicted segmentation maps are the same as that of the ground truths, and a value of zero indicates they have no overlap.

C. Training Model with Key K

To protect semantic segmentation models, we train models by randomly permuting feature maps with secret key K. The permutation is applied to feature maps in a network as in Fig. 3. In the figure, a fully convolutional network (FCN) [33] with a ResNet-50 [34] backbone is illustrated as an example, although the permutation is not limited to the FCN. There are six feature maps in Fig. 3, and a number of feature maps from the six maps are chosen to be permuted prior to permutation. In this paper, a feature map x with a dimension of $(c \times h \times w)$, where c is the number of channels, h is the height, and w is the width of the feature map, is transformed with key K at each iteration for training a model. There are two steps in the process of transforming a feature map as below (see Fig. 4).

1) Generate a random vector with a size of c such that
$$[\alpha_1, ., \alpha_i, ., \alpha_{i'}, \ldots, \alpha_c], \alpha_i \in \{1, \ldots, c\}, \tag{2}$$
where $\alpha_i \neq \alpha_{i'}$ if $i \neq i'$.
2) Replace all elements of x, $x(i,j,k)$, $i \in \{1,\ldots,c\}$, $j \in \{1,\ldots,h\}$, and $k \in \{1,\ldots,w\}$ with $x(a_i,j,k)$ to produce permuted feature map x', where an element of x', $x'(i,j,k)$ is equal to $x(a_i,j,k)$.

If multiple feature maps are chosen to be permuted, the above steps are applied to each feature map.

The above feature map permutation is spatially invariant as illustrated in Fig. 4. This spatially invariant property is important when the permutation is applied to applications that are required to output images like semantic segmentation. In contrast, an example of spatially variant permutation is given in Fig. 5. The conventional protection method for image classification [24] is carried out by using a spatially variant permutation method, so it is not available for protecting other models such as semantic segmentation models as described later. In this paper, a model protection method for semantic segmentation is discussed for the first time.

D. Applying Queries to Model

As shown in Fig. 1, authorized users have key K, and key K is also used for semantic segmentation. In the proposed method, a query image is applied to a model trained with K, and the model is protected by permuting feature maps with key K as well as for training the model. Protected models are expected to satisfy the following requirements.

1) Providing almost the same performance as that of using unprotected models to authorized users.
2) Degrading performance for unauthorized users even when they estimate key K.

III. Experiments and Results

The effectiveness of the proposed method was evaluated in terms of segmentation performance and robustness against unauthorized uses.

A. Experimental Setup

We used a FCN with a ResNet-50 backbone (see Fig. 3) for semantic segmentation, and the backbone was pretrained with the 1000-class ImageNet dataset [35]. Segmentation models were trained by using the PASCAL visual object classes dataset released in 2012 [36] for semantic segmentation, where the dataset has a training set with 1464 pairs (i.e., images and corresponding ground truths) and a validation set with 1449 pairs. We split the training set into training and development sets; 1318 samples were used for training, and 146 samples were used for development when training models. The whole validation set was used for testing the models. The conventional method using block-wise transformations [24]

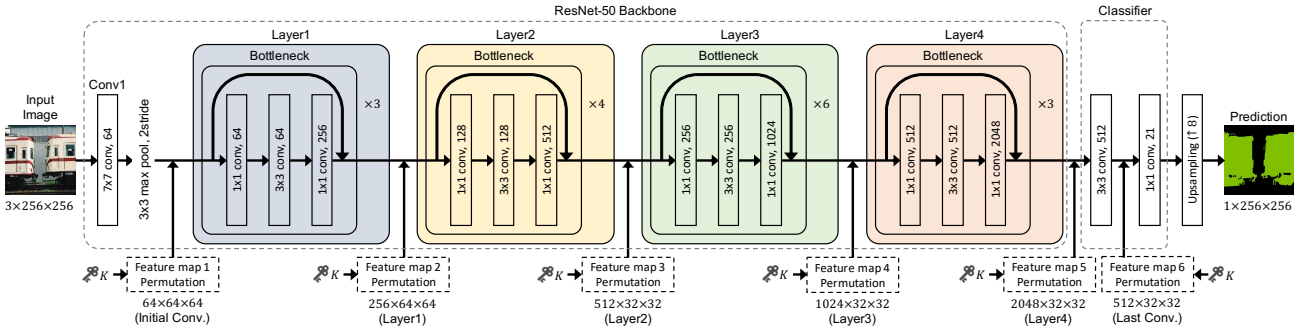

Fig. 3. Semantic segmentation model (FCN with ResNet-50 backbone) with feature map permutation

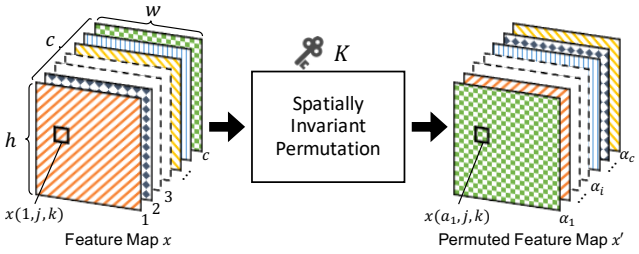

Fig. 4. Spatially invariant feature map permutation

TABLE I
SEGMENTATION ACCURACY (MEAN IOU) OF PROTECTED MODELS. BEST ACCURACIES ARE SHOWN IN BOLD.

Selected feature map	Correct (K)	No-perm	Incorrect (K')
1 (Model-1)	0.4645	0.1178	0.0598
2 (Model-2)	0.4279	0.2791	0.0367
3 (Model-3)	0.3591	0.0966	**0.0349**
4 (Model-4)	0.4973	0.0397	0.0377
5 (Model-5)	**0.5778**	0.0373	0.0397
6 (Model-6)	0.5768	**0.0349**	**0.0349**
Baseline		0.5752	

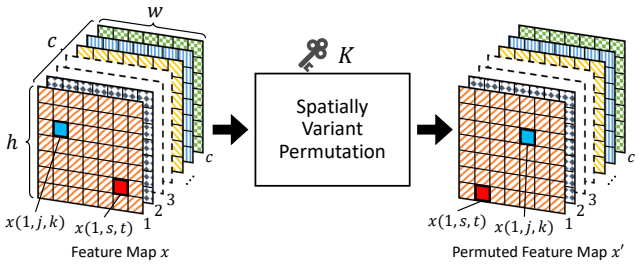

Fig. 5. Spatially variant feature map permutation

requires a fixed input image size. Therefore, to compare the conventional method with the proposed one, all input images and ground truths were resized to 256×256. In addition, standard data-augmentation methods, i.e., random resized crop and horizontal flip, were performed in the training.

All networks were trained for 30 epochs by using the stochastic gradient descent (SGD) optimizer with a weight decay of 0.0001 and a momentum of 0.9. The learning rate (lr) was initially set to 0.02, and it was decayed in each iteration as

$$lr = 0.02 \times \left(1 - \frac{x}{30 \times 42}\right)^{0.9}, \tag{3}$$

where x is the current iteration number. The batch size was 32. We used cross-entropy loss to calculate loss. After the training, we selected the model that provided the lowest loss value under the validation.

B. Performance Evaluation under Correct Key K

In this experiment, one feature map was chosen from six feature maps in the network, and segmentation models were then trained by permuting the chosen feature map with key K. The trained models were evaluated for authorized users with K. "Correct (K)" in Table I shows the result under this condition, where the model trained by permuting feature map 1 is referred to as Model-1 as an example, and "Baseline" denotes that the model was trained and tested without any feature map permutation.

From Correct (K) in Table I, several models such as Model-5 and Model-6 had a high segmentation accuracy, which was almost the same as that of the baseline, although a couple of models had a slightly degraded accuracy. Figure 6 also shows an example of prediction results. From this figure, the proposed model was demonstrated to maintain prediction results similar to the baseline.

C. Robustness against Unauthorized Access

We assume that unauthorized users have no key K and that they know both the method for protecting models and the permuted feature maps. To evaluate robustness against unauthorized access, we also evaluated the six protected models under two key conditions: No-perm and Incorrect, as shown in Table I. "No-perm" indicates that protected models were tested without any feature map permutation. "Incorrect" denotes that protected models were tested by permuting a feature map used in training with incorrect (randomly generated) key K'.

Table I shows the results under these conditions, where the results for Incorrect were averaged over 100 incorrect

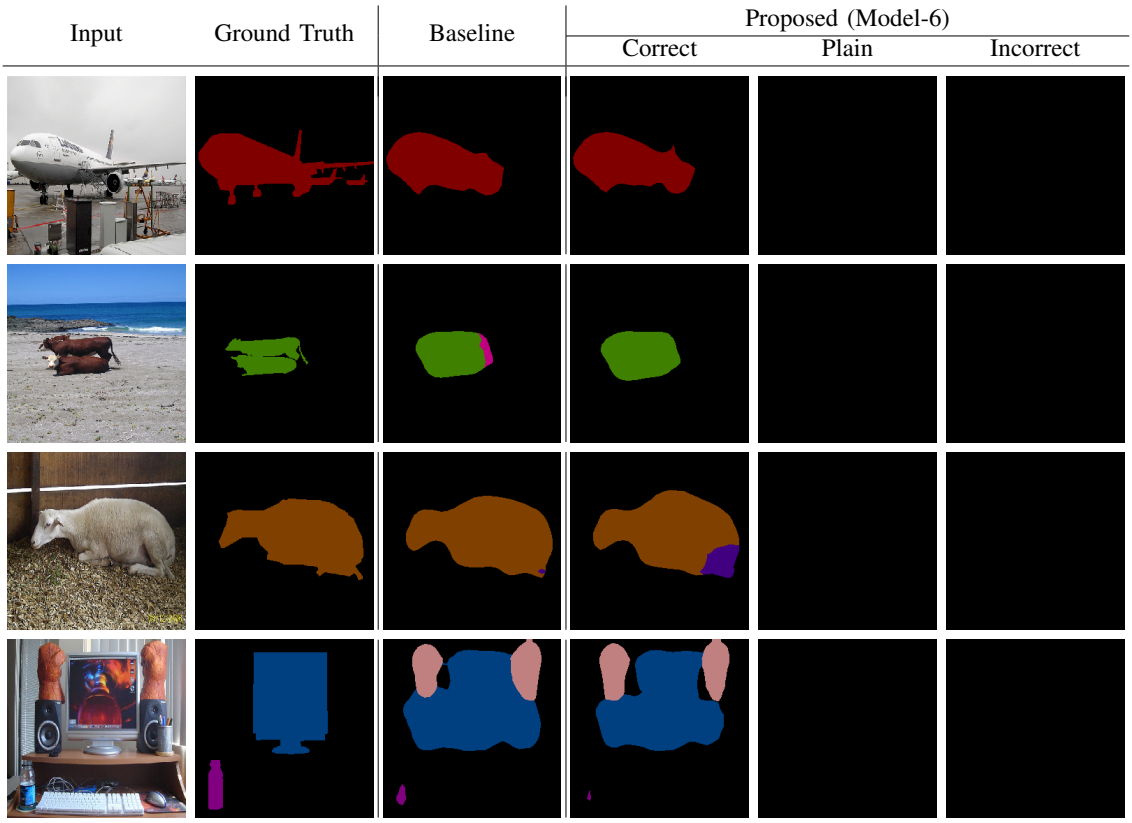

Fig. 6. Example of prediction results

keys. From the table, the protected models provided a low segmentation accuracy, so the proposed models were robust against these attacks. An example of prediction results of using Model-6 is also illustrated in Fig. 6. The robustness of the proposed models can be visually confirmed from the figure as well.

D. Comparison with State-of-the-art Method

A method for model protection was proposed in [24]. In the method, input images are transformed by using three block-wise transformations with a secret key: pixel shuffling (SHF), negative/positive transformation (NP), and format-preserving Feistel-based encryption (FFX) [37]. Although this conventional method can achieve high performance in image classification models, it has never been applied to other models such as semantic segmentation ones. To be compared with the proposed method, it was also applied to semantic segmentation models.

We trained segmentation models with different block sizes and tested the models with three key conditions, i.e., Correct, Plain, and Incorrect, where "Plain" used plain images as input ones, and "Incorrect" used images encrypted by using an incorrect key. As shown in Table II, the performance of all transformations heavily decreased compared with the proposed method. In particular, when using a large block size, the accuracy for the correct key was low. In contrast, the accuracy for the plain and incorrect keys was high, so the performance was confirmed to be poor for protecting semantic segmentation models.

Semantic segmentation models are required to output visual information as an image, so transformations applied to images or feature maps for training and testing models have to be spatially invariant, but the conventional block-wise transformations are not spatially invariant. That is why the performance of the block-wise transformations was poor for semantic segmentation models.

IV. Conclusion

We proposed an access control method that uses the spatially invariant permutation of feature maps for protecting semantic segmentation models for the first time. Semantic segmentation models are required to output visual information as an image, so transformations for model protection have to be spatially invariant, but conventional transformations are not spatially invariant. The proposed method allows us not only to obtain a high segmentation accuracy but also for there to be robustness against various attacks by unauthorized users. In experiments, the effectiveness of the proposed method was verified in terms of segmentation performance and robustness against unauthorized access. In contrast, the conventional protection method with block-wise transformations that was proposed for image classification models was demonstrated to

TABLE II
SEGMENTATION ACCURACY (MEAN IOU) OF TRANSFORMATIONS OF CONVENTIONAL METHOD

Block size	SHF			NP			FFX		
	Correct	Plain	Incorrect	Correct	Plain	Incorrect	Correct	Plain	Incorrect
2	0.5062	0.4518	0.4556	0.5132	0.4904	0.1398	0.3794	0.0346	0.0357
4	0.4560	0.4470	0.3865	0.4131	0.1762	0.1064	0.3251	0.0349	0.0371
8	0.3154	0.3143	0.2568	0.2421	0.0925	0.0985	0.2660	0.0349	0.0403
16	0.1893	0.1544	0.1370	0.2568	0.0745	0.1063	0.2216	0.0349	0.0690
32	0.0847	0.0493	0.0745	0.1677	0.0463	0.0937	0.1788	0.0349	0.1199
Model-6 (Proposed)	0.5768 (Correct)			0.0349 (Plain)			0.0349 (Incorrect)		
Baseline (non-protected)	0.5752								

not be applicable to segmentation models. As future work, we plan to evaluate the robustness against more diverse attacks such as key estimation attacks.

ACKNOWLEDGMENT

This study was partially supported by JSPS KAKENHI (Grant Number JP21H01327) and Support Center for Advanced Telecommunications Technology Research, Foundation (SCAT).

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[2] X. Liu, Z. Deng, and Y. Yang, "Recent progress in semantic image segmentation," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 1089–1106, Aug. 2019.

[3] M. Xue, J. Wang, and W. Liu, "DNN intellectual property protection: Taxonomy, attacks and evaluations (invited paper)," in *Proceedings of the 2021 on Great Lakes Symposium on VLSI*, Jun. 2021, pp. 455–460.

[4] Y. Uchida, Y. Nagai, S. Sakazawa, and S. Satoh, "Embedding watermarks into deep neural networks," in *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, 2017, pp. 269–277.

[5] J. Zhang, Z. Gu, J. Jang, H. Wu, M. P. Stoecklin, H. Huang, and I. Molloy, "Protecting intellectual property of deep neural networks with watermarking," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, 2018, pp. 159–172.

[6] B. Darvish Rouhani, H. Chen, and F. Koushanfar, "Deepsigns: An end-to-end watermarking framework for ownership protection of deep neural networks," in *Proceedings of the Twenty-Fourth International Conference on ASPLOS*, 2019, pp. 485–497.

[7] E. Le Merrer, P. Pérez, and G. Trédan, "Adversarial frontier stitching for remote neural network watermarking," *Neural Computing and Applications*, vol. 32, no. 13, pp. 9233–9244, Jul. 2020.

[8] L. Fan, K. W. Ng, and C. S. Chan, "Rethinking deep neural network ownership verification: Embedding passports to defeat ambiguity attacks," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 4716–4725.

[9] S. Sakazawa, E. Myodo, K. Tasaka, and H. Yanagihara, "Visual decoding of hidden watermark in trained deep neural network," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval*, 2019, pp. 371–374.

[10] H. Chen, B. D. Rouhani, C. Fu, J. Zhao, and F. Koushanfar, "Deepmarks: A secure fingerprinting framework for digital rights management of deep learning models," in *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, 2019, pp. 105–113.

[11] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: Watermarking deep neural networks by backdooring," in *27th USENIX Security Symposium (USENIX Security 18)*, Aug. 2018, pp. 1615–1631.

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

[13] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *2nd International Conference on Learning Representations*, 2014.

[14] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *3rd International Conference on Learning Representations*, 2015.

[15] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, 2017, pp. 506–519.

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

[17] K. Madono, M. Tanaka, M. Onishi, and T. Ogawa, "Block-wise scrambled image recognition using adaptation network," in *Artificial Intelligence of Things (AIoT), Workshop on AAAI conference Artificial Intellignece, (AAAI-WS)*, 2020.

[18] W. Sirichotedumrong and H. Kiya, "A gan-based image transformation scheme for privacy-preserving deep neural networks," in *28th European Signal Processing Conference, EUSIPCO*, 2020, pp. 745–749.

[19] H. Ito, Y. Kinoshita, M. Aprilpyone, and H. Kiya, "Image to perturbation: An image transformation network for generating visually protected images for privacy-preserving deep neural networks," *IEEE Access*, vol. 9, pp. 64 629–64 638, 2021.

[20] W. Sirichotedumrong, Y. Kinoshita, and H. Kiya, "Pixel-based image encryption without key management for privacy-preserving deep neural networks," *IEEE Access*, vol. 7, pp. 177 844–177 855, 2019.

[21] 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 *2019 IEEE International Conference on Image Processing*, 2019, pp. 674–678.

[22] M. Chen and M. Wu, "Protect your deep neural networks from piracy," in *2018 IEEE International Workshop on Information Forensics and Security*, 2018, pp. 1–7.

[23] M. AprilPyone and H. Kiya, "Training dnn model with secret key for model protection," in *2020 IEEE 9th Global Conference on Consumer Electronics*, 2020, pp. 818–821.

[24] ——, "A protection method of trained CNN model with a secret key from unauthorized access," *APSIPA Transactions on Signal and Information Processing*, vol. 10, p. e10, 2021.

[25] ——, "Block-wise image transformation with secret key for adversarially robust defense," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2709–2723, 2021.

[26] 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.

[27] 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.

[28] 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.

[29] T. Chuman, K. Kurihara, and H. Kiya, "Security evaluation for block scrambling-based etc systems against extended jigsaw puzzle solver attacks," in *2017 IEEE International Conference on Multimedia and Expo*, 2017, pp. 229–234.

[30] A. P. M. Maung and H. Kiya, "Training DNN model with secret key for model protection," in *2020 IEEE 9th Global Conference on Consumer Electronics*, 2020, pp. 818–821.
[31] M. Swanson, M. Kobayashi, and A. Tewfik, "Multimedia data-embedding and watermarking technologies," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1064–1087, 1998.
[32] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 658–666.
[33] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.
[34] 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*, 2016, pp. 770–778.
[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
[36] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, Jan. 2015.
[37] M. Bellare, P. Rogaway, and T. Spies, "Addendum to "the ffx mode of operation for format-preserving encryption" a parameter collection for enciphering strings of arbitrary radix and length," in *Draft 1.0*, 2010. [Online]. Available: https://csrc.nist.rip/groups/ST/toolkit/BCM/documents/proposedmodes/ffx/ffx-spec2.pdf