APSIPA Transactions on Signal and Information Processing, 2022, 11, e24
This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (http://creativecommons.org/licenses/by-nc/4.0/), which permits unrestricted re-use, distribution, and reproduction in any medium, for non-commercial use, provided the original work is properly cited.

Original Paper

Access Control of Semantic Segmentation Models Using Encrypted Feature Maps

Hiroki Ito, MaungMaung AprilPyone, Sayaka Shiota and Hitoshi Kiya*

Department of Computer Science, Tokyo Metropolitan University, Tokyo 191-0065, Japan

ABSTRACT

In this paper, we propose an access control method with a secret key for semantic segmentation models for the first time so that unauthorized users without a secret key cannot benefit from the performance of trained models. The method enables us not only to provide a high segmentation performance to authorized users, but also to degrade the performance for unauthorized users. We first point out that, for the application of semantic segmentation, conventional access control methods which use encrypted images for classification tasks are not directly applicable due to performance degradation. Accordingly, in this paper, selected feature maps are encrypted with a secret key for training and testing models, instead of input images. In an experiment, the protected models allowed authorized users to obtain almost the same performance as that of non-protected models but also with robustness against unauthorized access without a key.

Keywords: Access control, deep learning, semantic segmentation, model protection.

*Corresponding author: Hitoshi Kiya, kiya@tmu.ac.jp. This study was partially supported by JSPS KAKENHI (grant number JP21H01327) and JST CREST (grant number JPMJCR20D3).

Received 15 March 2022; Revised 10 June 2022

ISSN 2048-7703; DOI 10.1561/116.00000013
© 2022 H. Ito, M. AprilPyone, S. Shiota and H. Kiya

1 Introduction

Deep neural networks (DNNs) and convolutional neural networks (CNNs) have been deployed in many applications such as biometric authentication, automated driving, and medical image analysis [17, 18]. However, training successful CNNs requires three ingredients: a huge amount of data, GPU-accelerated computing resources, and efficient algorithms, and it is not a trivial task. In fact, collecting images and labeling them is also costly and will also consume a massive amount of resources. Moreover, algorithms used in training a model may be patented or have restricted licenses. Therefore, trained DNNs and CNNs have great business value. Considering the expenses necessary for the expertise, money, and time taken to train a 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 [15]. The former focuses on identifying the ownership of the models, and the latter addresses protecting the functionality of the models from unauthorized access. Ownership verification methods were inspired by digital watermarking [30] and embed watermarks into models so that the embedded watermarks can be used to verify the ownership of the models in question [1, 4, 9, 11, 16, 21, 25, 33–35].

Although the above watermarking methods can facilitate in identifying 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, 22, 31]. Therefore, it is crucial to investigate mechanisms to protect 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 models against unauthorized access was inspired by adversarial examples and proposed to utilize secret perturbation to control the access of models [6]. In addition, another study introduced a secret key to protect models [3], and it was shown to outperform the other methods. The secret key-based protection method uses a key-based transformation that was originally used by an adversarial defense in AprilPyone and Kiya [2], which was in turn inspired by perceptual image encryption methods [7, 8, 20, 26–29, 32]. This block-wise 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. These existing methods provide a good access control performance, but they all focus on the access control of image classification models. In this paper, we point out that conventional access control methods with encrypted images for classification models are not directly applicable to segmentation models.

Therefore, for the first time, in this paper, we propose a model protection method for semantic segmentation models by applying a key-based transfor-

mation to feature maps. The method not only achieves a high classification accuracy (i.e., almost the same accuracy as in the non-protected case), but also increases the key space substantially. Our contribution in this paper is to propose an access control method with a secret key for semantic segmentation models for the first time, which enables us not only to maintain a high segmentation accuracy but also to increase the key space. To evaluate the proposed method, we conduct relevant attacks. In experiments, the proposed model-protection method is confirmed to outperform previous such methods.

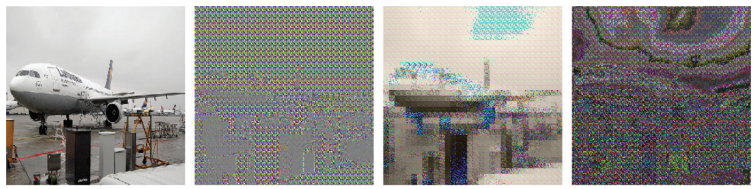
2 Related Work

There are two approaches to protecting trained models: ownership verification and access control. The former focuses on identifying the ownership of trained models. The latter addresses protecting the functionality of trained models. The former aims for only ownership verification. Therefore, stolen models can be directly used by unauthorized users, so we focus on access control to protect trained models from unauthorized access even if the models are stolen.

The first access control method, which was inspired by adversarial examples [13, 22, 31], was proposed for image classification models in Chen and Wu [6]. In this method, authorized users add a secret perturbation generated by an anti-piracy transform module to input images, and the processed input images are fed to a protected model. Therefore, this method needs additional resources to train the module. In addition, the method focuses on protecting image classification models.

The second method is to extend the passport-based ownership verification method [11] as an access control method. However, the passport in Fan *et al.* [11] is a set of extracted features of a secret image/images or equivalent random patterns from a pre-trained model. In addition, a network has to be modified with additional passport layers to use passports. Therefore, there are significant overhead costs in both the training and inference phases. Moreover, the effectiveness of the passport-based method has never been confirmed under the use of semantic segmentation models.

The third is a block-wise image transformation method with a secret key [3], which is inspired by learnable image encryption [2, 20, 26, 27, 29, 32]. In this method, input images are encrypted with a key, for which three types of encryption methods: negative/positive transformation (NP), pixel shuffling (SHF), and format-preserving Feistel-based encryption (FFX), were proposed as illustrated in Figure 1. Figure 2 shows the framework of the block-wise method. In the framework, an owner transforms all training images with secret key K, and a model is trained to protect the model by using the transformed images and corresponding ground truths. An authorized user with key K transforms



(a) Original (b) NP (c) SHF (d) FFX

Figure 1: Images transformed by block-wise transformations.

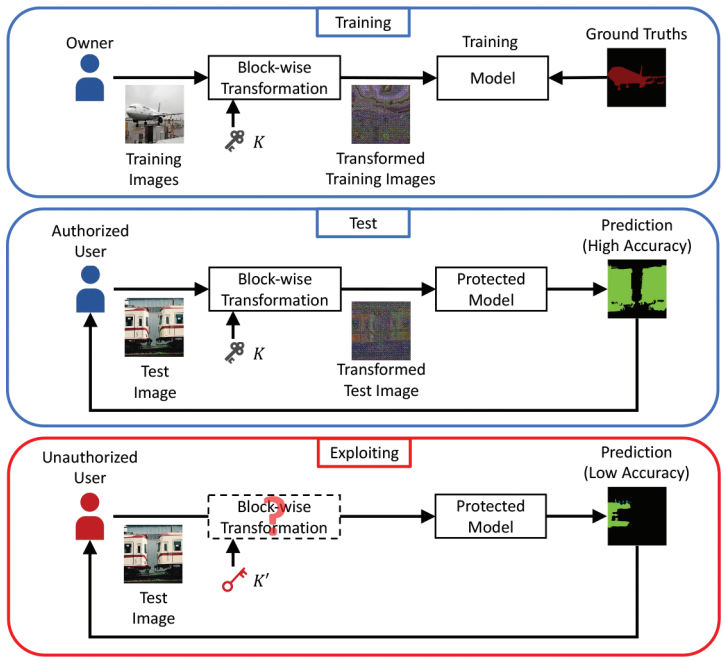


Figure 2: Access control framework with encrypted input images.

a test image with K and feeds it to the protected model to get a prediction result with high accuracy. In contrast, an unauthorized user without key K cannot obtain a prediction result with high accuracy, even if the unauthorized user knows the framework and the encryption algorithm. In addition, the method with key K does not need any network modification or incur significant overhead costs. However, the use of the block-wise transformation is limited to image classification tasks.

Accordingly, in this paper, we propose a novel access control method for semantic segmentation tasks for the first time. The proposed method

does not need any network modification or incur significant overhead costs as well.

3 Access Control with Encrypted Feature Maps

Access control of semantic segmentation models with encrypted feature maps is proposed here.

3.1 Overview

Protected models for access control should satisfy the following requirements. The protected models should provide prediction results with a high accuracy to authorized users but not provide such high-accuracy results to unauthorized users. To meet these requirements, encrypted feature maps are used as shown in Figure 3.

In the framework with encrypted feature maps, an owner trains a model by using plain training images and corresponding ground truths, where selected feature maps in the network are encrypted by using a secret key K at each

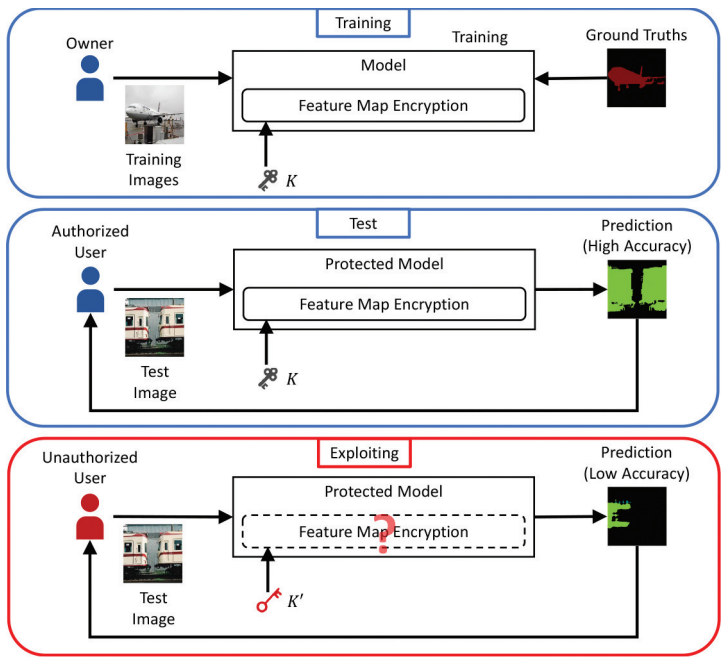
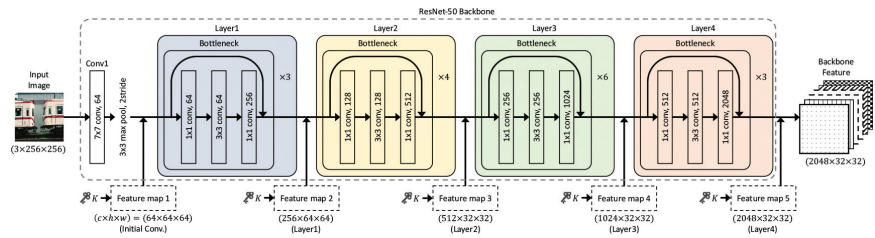


Figure 3: Access control framework with encrypted feature maps.

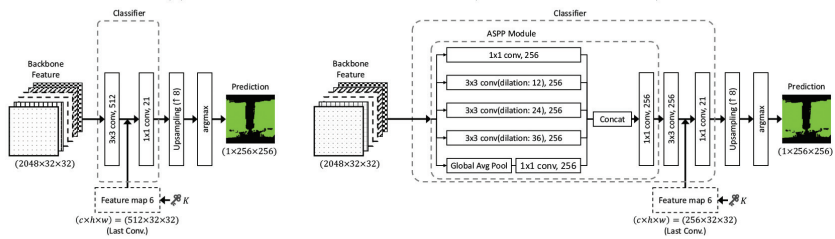
training iteration in accordance with the proposed method. For testing, an authorized user with key K feeds a test image to the trained model to obtain a prediction result with high accuracy. In contrast, when an unauthorized user without key K inputs a test image to the trained model without any key or with an estimated key K', the unauthorized user cannot benefit from the performance of the trained model.

3.2 Feature Map

In the proposed method, one or more feature maps in a network are selected, and then the selected feature maps are encrypted with a secret key. We illustrate semantic segmentation architectures in Figure 4 as an example, in which there are six feature maps (feature maps 1-6) where two classifiers correspond to a fully convolutional network (FCN) [19] and a network using atrous convolution (DeepLabv3) [5], respectively. Both networks consist of one backbone and one classifier, and ResNet-50 is commonly used as the backbone. Input images are fed to the backbone, and the classifier gets features from the backbone. Finally, the classifier outputs a prediction result.



(a) Backbone for FCN and DeepLabv3 (ResNet-50 architecture)



(b) FCN classifier (c) DeepLabv3 classifier

Figure 4: Segmentation models using encrypted feature maps.

3.3 Feature Map Encryption

A feature map is an intermediate output in a convolutional network. Unlike weights which are learned by using all input images in a model, a feature

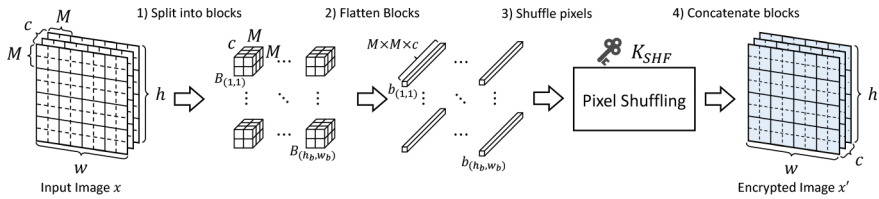


Figure 5: Block-wise pixel shuffling process for input image encryption.

map is decided by using each input image. Therefore, a selected feature map $x \in \mathbb{R}^{c \times h \times w}$ is transformed with key K at each iteration for training a model, where c is the number of channels, h is the height, and w is the width of the feature map. To transform feature maps, we address two methods: pixel shuffling and channel permutation (CP) as follows.

3.3.1 Block-Wise Pixel Shuffling

A block-wise pixel shuffling method, referred to as pixel shuffling (SHF), was investigated as a method for encrypting input images in [2, 3]. In this paper, SHF is extended for the access control of semantic segmentation models.

Below is the encryption procedure of the conventional SHF (see Figure 5), where x is an input image.

1. Divide x into blocks with a size of $M \times M$ as

$$\{B_{(1,1)}, \ldots, B_{(l,m)}, \ldots, B_{(h_b,w_b)}\}, \tag{1}$$

   where $h_b \times w_b$ denotes the number of blocks, and each block has a shape of (c, M, M).

2. Flatten each block $B_{(l,m)}$ as a vector

$$b_{(l,m)} = [b_{(l,m)}(1), \ldots, b_{(l,m)}(L)], \tag{2}$$

   where the length of the flattened vector is $L = c \times M \times M$.

3. Shuffle pixels: First, generate secret key K_{SHF} as

$$K_{\mathrm{SHF}} = [\alpha_1, \ldots, \alpha_i, \ldots, \alpha_{i'}, \ldots, \alpha_L], \tag{3}$$

   where $\alpha_i \in \{1, \ldots, L\}$, and $\alpha_i \neq \alpha_{i'}$ if $i \neq i'$. Second, shuffle each vector $b_{(l,m)}$ with K_{SHF} such that

$$b'_{(l,m)}(i) = b_{(l,m)}(\alpha_i), \tag{4}$$

and a shuffled vector is given by

$$b'_{(l,m)} = [b'_{(l,m)}(1), \ldots, b'_{(l,m)}(L)]. \tag{5}$$

All vectors are converted with the same key.

4. Concatenate blocks: The shuffled vectors are integrated to obtain transformed input image x' with a dimension of (c, h, w).

3.3.2 Channel Permutation

SHF is extended for application to semantic segmentation models in terms of two points: the use of feature maps and a block size of $M = 1$. The extended encryption is called CP. Accordingly, CP is a pixel-wise transformation, where a feature map is permuted only along the channel dimension.

The following is the procedure of CP.

1. Select a feature map x to be encrypted.
2. Generate secret key K_{CP} with a size of c as

$$K_{\mathrm{CP}} = [\beta_1, \ldots, \beta_j, \ldots, \beta_{j'}, \ldots, \beta_c], \tag{6}$$

where $\beta_j \in \{1, \ldots, c\}$, and $\beta_j \neq \beta_{j'}$ if $j \neq j'$.

3. Replace all elements of x, $x(j, p, q)$, $p \in \{1, \ldots, h\}$, and $q \in \{1, \ldots, w\}$ as

$$x'(j, p, q) = x(k_j, p, q), \tag{7}$$

and permuted feature map $x' \in \mathbb{R}^{c \times h \times w}$ is obtained.

As shown in Figure 6, CP is a spatially-invariant transformation, so it can support a pixel-level resolution, which is important for semantic segmentation, even though SHF supports a block-level one.

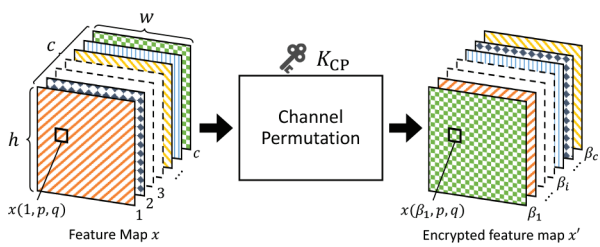


Figure 6: Channel permutation process.

Table 1: Key space of access control methods.

Method	Key space	Remark
SHF	$(c \times M \times M)!$	$c = 3$ (input image)
CP	$c!$	$c \gg 3$ (feature map)

3.4 Difference between SHF and CP

The differences between SHF and CP are summarized below:

(a) CP is a pixel-wise transformation.

(b) CP is applied to a feature map.

(c) The number of feature map channels is larger than that of input image channels.

Difference (a) allows us to obtain results with a pixel-level resolution, but pixel-wise transformations are not robust against various attacks if the transformation is applied to input images as discussed in [3] because the number of input image channels c is small (i.e., RGB images have $c = 3$). To improve on this, we propose encrypting feature maps that have a larger number of channels such as $c = 2048$ as shown in Figure 4. In addition, the use of feature maps enables us to maintain a high accuracy as described later.

3.5 Threat Models

A threat model includes a set of assumptions such as an attacker's goals, knowledge, and capabilities. Users without secret key K are assumed to be the adversary. Attackers may steal a model to achieve different goals for profit. In this paper, we consider the attacker's goal is to be able to make use of a stolen model. This paper considers brute-force, random key, and fine-tuning attacks as ciphertext-only attacks. Therefore, the following possible attacks done with the intent of stealing a model are discussed to evaluate the robustness of the proposed model-protection method. In experiments, the method will be demonstrated to be robust against attacks.

3.5.1 Brute-Force Attack

A simple attack to decrypt an encrypted input image or feature map is a brute-force attack. This attack systematically checks all possible passwords until the correct one is found. Therefore, the encryption method must have a large enough key space. The key space of each method is summarized in Table 1.

The key space is decided by block size M and channel size c. For example, if SHF with $M = 2$ is applied to an input image, the key space is $(3 \times 2 \times 2)! \approx 4.79 \times 10^8 < 2^{29}$. In contrast, when a feature map with $c = 256$ is encrypted by using CP, the key space is $256! \approx 2^{1684}$. In general, the channel number of feature maps is much larger than that of input images, so CP has a large key space even when $M = 1$ is selected, compared with SHF. In addition, when using CP, the attacker has to know or estimate the location of the transformed feature map, which cannot be known from the model itself.

3.5.2 Random Key Attack

In reality, the random attack is hard to carry out for the proposed model protection because there are many layers in a conventional CNN architecture, and the location of the transformed feature map cannot be known from the model itself. To be practical, the cost of an attack should always be lower than that of training a new model. We will consider a worst-case scenario in which an attacker obtains additional information about the transformed feature map and the transformation process except for the secret key, in an experiment.

3.5.3 Fine-Tuning Attack

Fine-tuning is a process that takes a trained model and then tunes the model to make it perform some purpose (e.g., to process a similar task). An attacker may use fine-tuning as an attack to override model protection so that the attacker can utilize a protected model without a secret key. This attack aims to disable the key by retraining a protected model with a small subset of a dataset. We assume an attacker has the model weights and a small dataset D' for this attack.

4 Experimental Results

To verify the effectiveness of the proposed method, the method was evaluated in terms of access control and robustness against attacks. All experiments were conducted with the PyTorch library [23] in Python.

4.1 Setup

4.1.1 Dataset

Semantic segmentation models were trained by using a dataset released for the segmentation competition of Visual Object Classes Challenge 2012 (VOC2012) [10]. The dataset consists of a training set with 1464 pairs (i.e., images and corresponding ground truths) and a development set with 1449 pairs. In

addition, a test set is also available only on the evaluation server, but it was not used in the experiment due to some constraints.

The training set was divided into 1318 samples for training models and 146 samples for validating the loss of models during the training, and we selected the model that provided the lowest loss value after the training. The performance of the trained models was evaluated by using the development set with 1449 pairs.

All input images and ground truths were resized to a size of 256×256 because block-wise transformation requires images with a fixed size. In addition, standard data-augmentation methods, i.e., random resized crop and horizontal flip, were performed in training models.

4.1.2 Networks

We used a FCN [19] and a network with atrous convolution (DeepLabv3) [5] for semantic segmentation, as shown in Figure 4. In the experiments, a deep residual network with 50 layers (ResNet-50) [14] was used as a backbone for both networks, where only the backbone was pre-trained on a dataset used in ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) [24], and the pre-trained weights were provided on PyTorch. All networks were trained for 30 epochs by using a stochastic gradient descent (SGD) optimizer, where an initial learning rate (lr) of 0.02, a weight decay of 0.0001, and a momentum of 0.9 were selected as the hyperparameters of the optimizer. The learning rate was decayed in each iteration as

$$lr = 0.02 \times \left(1 - \frac{n}{30 \times 42}\right)^{0.9}, \tag{8}$$

where n is the current iteration number. The batch size was 32, and the standard pixel-wise cross-entropy loss without weight rebalancing was used.

4.2 Performance Evaluation

In this experiment, the segmentation performance of the protected models was evaluated with the mean intersection-over-union (mean IoU), which is a common evaluation metric for semantic segmentation. An IoU value is given for each class by

$$\text{IoU} = \frac{TP}{TP + FP + FN}, \tag{9}$$

and the mean IoU is then calculated by averaging the 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

Table 2: Segmentation accuracy (mean IoU) of proposed method (CP). Best accuracies are shown in bold.

Network		FCN		DeepLabv3	
Key condition		Correct	No-enc	Correct	No-enc
Selected feature map	1	46.35	14.71	54.79	15.82
	2	43.86	29.43	51.38	37.82
	3	34.18	7.76	38.72	10.06
	4	50.79	3.79	55.33	3.93
	5	57.19	3.80	64.75	3.57
	6	**58.52**	**3.49**	**65.15**	**3.49**
Baseline		58.89 (non-protected)		65.77 (non-protected)	

one means that the predicted segmentation maps are the same as those of the ground truths, and a value of zero indicates that they have no overlap.

4.2.1 Model Trained with Encrypted Feature Map

In this experiment, a CP was applied to a selected feature map in a network for semantic segmentation. Table 2 shows the results under two classifiers: FCN and DeepLabv3, where one feature map was selected to be encrypted from six feature maps in each network (see Figure 4). In the table, the segmentation accuracy was calculated by using 1449 pairs under two conditions: Correct and No-enc, where "Correct" means the use of test images encrypted with correct key K, and "No-enc" indicates the use of plain test images. An example of the results with DeepLabv3 is also shown in Figure 7.

From the table, CP was confirmed to achieve almost the same accuracy as that of the baselines under the use of the correct key when feature map 5 or 6 was selected. In contrast, CP provided a low accuracy to unauthorized users without the key (No-enc). Note that the segmentation performance slightly varies in general due to the initial weights of a model and the key. We carried out the experiment 10 times with different initial weights and keys under each condition. Average results were presented in Table 2. From the experiments, we confirmed that the proposed access control method with a selected feature map encryption can achieve almost the same performance as the baseline (non-protected) model.

From Figure 7, the prediction results were confirmed to be similar to the corresponding ground truths under the use of the correct key. In contrast, the results estimated from plain images had only a background label. Accordingly, CP with encrypted feature maps was effective in the access control of semantic segmentation models.

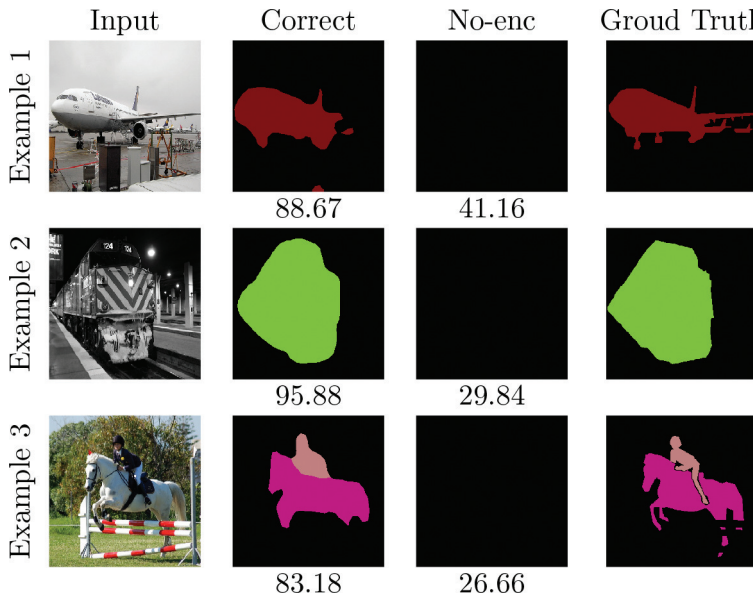


Figure 7: Example of prediction results for CP (DeepLabv3). Mean IoU values are given under predictions.

4.2.2 Selection of Feature Maps

As shown in Table 2, the performance of the models trained with encrypted feature maps depended on the selection of feature maps. In the experiment, when the encryption was applied to a feature map at positions 2 to 4, the segmentation accuracy was lower than that of models 5 and 6. The difference in segmentation accuracy among the selected feature maps was caused by a residual connection in the ResNet-50 backbone in Figure 4. From Figure 4, feature maps 2-4 had residual connections on both the front and back of each feature map. In contrast, in feature maps 1, 5, and 6, the influence of CP can be easily canceled out by a convolutional layer because there is no residual connection either in front or behind. Accordingly, feature map 6 is recommended as an encrypted feature map.

Although the access control performance of the models trained with encrypted feature maps depend upon the selection of feature maps, the selection of feature maps is independent of the type of datasets. Accordingly, we can experimentally select a feature map to be encrypted under the use of a dataset. In principle, one or more feature maps can be encrypted in the proposed access control method. However, when unsuitable feature maps are encrypted, it degrades the performance of models as shown in Table 2. Our experiments confirmed that encrypting only one feature map has already provided a good access control

performance, and encrypting two or more feature maps does not have any significant advantage. Therefore, only one feature map was encrypted in experiments.

4.2.3 Model Trained with Encrypted Input Images

Input images were encrypted in accordance with SHF under various block sizes (i.e., $M \in \{1, 2, 4, 8, 16, 32\}$) for comparison with the proposed method (CP). SHF was already demonstrated to achieve a high access control performance in image classification tasks in [3], but it has never been applied to semantic segmentation ones.

Table 3 shows the segmentation accuracy of the protected models calculated from 1449 pairs. From the results, even when correct key K was used, the segmentation accuracy decreased significantly as block size M increased in both networks. In contrast, when the block size was small, the protected model achieved a segmentation accuracy close to the baseline. However, the accuracy without the encryption (i.e., No-enc) was almost the same as that of "Correct," so the access control was weak under the use of a small block size.

Table 3: Segmentation accuracy (mean IoU) of conventional method with encrypted input images (SHF).

Network		FCN		DeepLabv3	
Key condition		Correct	No-enc	Correct	No-enc
Block size M	1	56.55	56.15	64.76	62.88
	2	51.54	47.58	59.74	56.67
	4	48.37	46.72	50.82	51.96
	8	34.25	34.68	37.70	35.95
	16	18.05	13.42	20.91	15.83
	32	7.68	5.21	11.14	5.58
Baseline		58.89 (non-protected)		65.77 (non-protected)	

From Figure 8, we also confirmed that the prediction results for Correct were similar to those for No-enc when a small block size was used. In addition, the prediction result for $M = 8$ was significantly degraded compared with the ground truth. Therefore, applying SHF to input images is not suitable for the access control of semantic segmentation models, even though it is suitable for image classification tasks.

4.3 Robustness against Random Key Attack

In this experiment, CP was evaluated in terms of robustness against the random key attack described in Section 3.5.2, where models were protected by

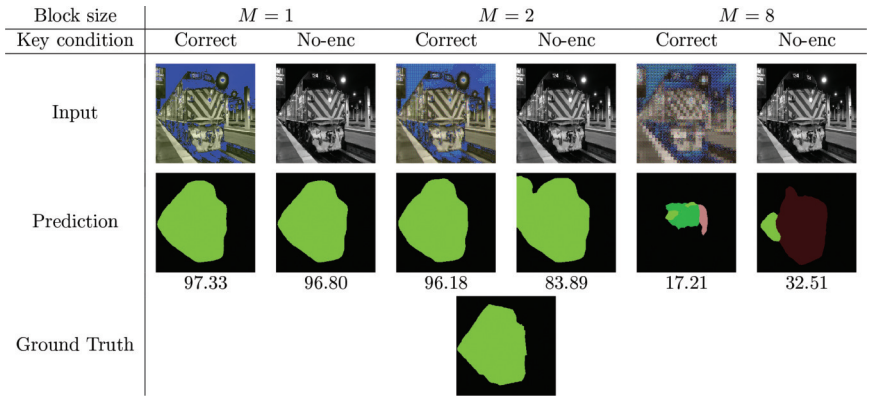
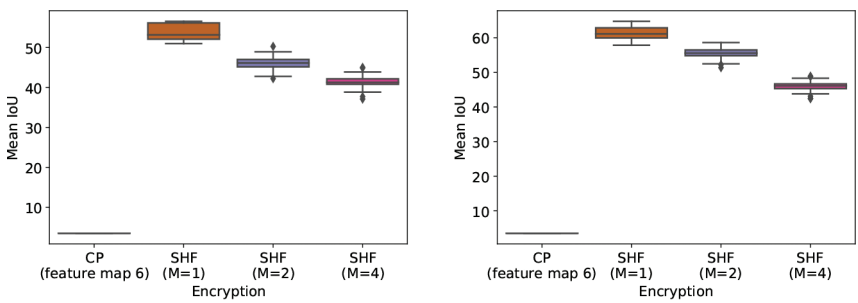


Figure 8: Example of prediction results for SHF (DeepLabv3). Mean IoU values are given under predictions.

encrypting feature map 6. An evaluation was carried out on robustness with 100 incorrect keys that were randomly generated.

Figure 9 shows the segmentation performance of the protected models under the use of the incorrect keys on the development set of VOC2012. From the results of using CP, the mean IoU values were significantly low for both models, which means that the models were robust enough against this attack. However, the mean IoU values of using SHF increased as the block size decreased. Therefore, the proposed method (CP) outperformed the conventional method (SHF) in terms of robustness against the random key attack.



(a) FCN (b) DeepLabv3

Figure 9: Mean IoU values of protected models with 100 incorrect keys. Boxes span from first to third quartile, referred to as Q_1 and Q_3, and whiskers show maximum and minimum values in range of $[Q_1 - 1.5(Q_3 - Q_1), Q_3 + 1.5(Q_3 - Q_1)]$. Band inside box indicates median. Outliers are indicated as dots.

4.4 Robustness against Fine-Tuning Attack

We ran an experiment with different sizes for an attacker's small dataset (i.e., $|D'| \in \{5\%, 10\%\}$ of the training data). Models protected by encrypting feature map 6 were retrained by using D' to disable the key. Table 4 shows the results of the fine-tuning attack for both networks.

Although the accuracy of the fine-tuned models was higher when the size of D' was larger, it was still lower than the accuracy of the original protected models (i.e., "Protected" in Table 4). Therefore, the attacker was not able to use the models to full capacity even when preparing a small dataset.

Table 4: Segmentation accuracy (mean IoU) of fine-tuned models.

Network			FCN	DeepLabv3
Fine-tuned (test without key)	D'	0%	3.49	3.49
		5%	27.02	29.78
		10%	41.70	46.19
No fine-tuned (test with key)			59.24	65.43

5 Conclusion and Future Work

In this paper, we proposed an access control method for semantic segmentation models for the first time. The method is carried out by encrypting selected feature maps with a secret key called CP, while input images are encrypted by using a block-wise encryption method in conventional methods. The use of CP allows us not only to obtain a pixel-level accuracy that is required for semantic segmentation but also to maintain a wide key space even when a pixel-wise permutation is used. As a result, the proposed method can maintain both a high accuracy and robustness against attacks. In experiments, the conventional method with encrypted input images was not effective in the access control of semantic segmentation models, and the effectiveness of the proposed method was demonstrated in terms of segmentation accuracy.

As for future work, we shall generalize the proposed method to other models such as object detection models and generative models. In addition, if the key is compromised, the proposed method in its current form needs to repeat the whole training to update the key in the same way that existing key-based access control methods such as the use of encrypted input images for image classification need to repeat the training. To overcome this limitation, we shall explore possible options in our future work. We shall also identify other potential threats to the access control of the models.

Biographies

Hiroki Ito received his B.E and M.E. degrees from Tokyo Metropolitan University in 2020 and 2022, respectively. He received the Student Best Paper Award at SISA 2021. His research interests are in the area of machine learning and image processing.

MaungMaung AprilPyone received a BCS degree from the International Islamic University Malaysia in 2013 under the Albukhary Foundation Scholarship, MCS degree from the University of Malaya in 2018 under the International Graduate Research Assistantship Scheme, and Ph.D. degree from the Tokyo Metropolitan University in 2022 under the Tokyo Human Resources Fund for City Diplomacy Scholarship. He is currently working as a Project Assistant Professor in the Tokyo Metropolitan University and as a researcher in rinna Co. Ltd. He received an IEEE ICCE-TW Best Paper Award in 2016. His research interests are in the area of adversarial machine learning and information security. He is a member of IEEE.

Sayaka Shiota received her B.E., M.E., and Ph.D. degrees in intelligence and computer science, engineering, and engineering simulation from the Nagoya Institute of Technology, Nagoya, Japan, in 2007, 2009, and 2012, respectively. From February 2013 to March 2014, she worked at the Institute of Statistical Mathematics as a Project Assistant Professor. In April of 2014, she joined Tokyo Metropolitan University as an Assistant Professor. Her research interests include statistical speech recognition and speaker verification. She is a member of ASJ, IPSJ, IEICE, APSIPA, ISCA, and IEEE.

Hitoshi Kiya received B.E. and M.E. degrees from the Nagaoka University of Technology, Japan, in 1980 and 1982, respectively, and a Dr.Eng. degree from Tokyo Metropolitan University in 1987. In 1982, he joined Tokyo Metropolitan University, where he became a Full Professor in 2000. From 1995 to 1996, he attended The University of Sydney, Australia, as a Visiting Fellow. He is a fellow of IEEE, IEICE, and ITE. He served as the President of APSIPA from 2019 to 2020 and the Regional Director-at-Large for Region 10 of the IEEE Signal Processing Society from 2016 to 2017. He was also the President of the IEICE Engineering Sciences Society from 2011 to 2012. He has been an editorial board member of eight journals, including IEEE TIP, IEEE TSP, and IEEE TIFS. He has organized a lot of international conferences in such roles as the TPC Chair of IEEE ICASSP 2012 and as the General Co-Chair of IEEE ISCAS 2019.

References

[1] 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)*, August 2018, 1615–31.

[2] M. AprilPyone and H. Kiya, "Block-wise Image Transformation with Secret Key for Adversarially Robust Defense", *IEEE Transactions on Information Forensics and Security*, 16, 2021, 2709–23.

[3] M. AprilPyone and H. Kiya, "A Protection Method of Trained CNN Model with a Secret Key from Unauthorized Access", *APSIPA Transactions on Signal and Information Processing*, 10, 2021, e10, DOI: 10.1017/ATSIP.2021.9.

[4] 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*, New York, NY, USA: Association for Computing Machinery, 2019, 105–13, DOI: 10.1145/3323873.3325042.

[5] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation", *arXiv:1706.05587*, 2017, https://arxiv.org/abs/1706.05587.

[6] M. Chen and M. Wu, "Protect Your Deep Neural Networks from Piracy", in *2018 IEEE International Workshop on Information Forensics and Security*, 2018, 1–7, DOI: 10.1109/WIFS.2018.8630791.

[7] 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*, E101.D(1), 2018, 37–44, DOI: 10.1587/transinf.2017MUP0001.

[8] 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*, 14(6), 2019, 1515–25, DOI: 10.1109/TIFS.2018.2881677.

[9] 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*, Association for Computing Machinery, 2019, 485–97, DOI: 10.1145/3297858.3304051.

[10] 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*, 111(1), 2015, 98–136.

[11] 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, 4716–25.

[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, 1322–33, DOI: 10.1145/2810103.2813677.

[13] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples", in *3rd International Conference on Learning Representations*, 2015.

[14] 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, 770–8, DOI: 10.1109/CVPR.2016.90.

[15] H. Kiya, M. AprilPyone, Y. Kinoshita, S. Imaizumi, and S. Shiota, "An Overview of Compressible and Learnable Image Transformation with Secret Key and its Applications", *APSIPA Transactions on Signal and Information Processing*, 11(1), 2022, e11, DOI: 10.1561/116.00000048.

[16] E. Le Merrer, P. Pérez, and G. Trédan, "Adversarial Frontier Stitching for Remote Neural Network Watermarking", *Neural Computing and Applications*, 32(13), 2020, 9233–44, DOI: 10.1007/s00521-019-04434-z.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning", *Nature*, 521(7553), 2015, 436–44, DOI: 10.1038/nature14539, https://doi.org/10.1038/nature14539.

[18] X. Liu, Z. Deng, and Y. Yang, "Recent Progress in Semantic Image Segmentation", *Artificial Intelligence Review*, 52(2), 2019, 1089–106, DOI: 10.1007/s10462-018-9641-3.

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation", in *2015 IEEE Conference on Computer Vision and Pattern Recognition*, 2015, 3431–40, DOI: 10.1109/CVPR.2015.7298965.

[20] 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.

[21] A. MaungMaung and H. Kiya, "Piracy-Resistant DNN Watermarking by Block-Wise Image Transformation with Secret Key", in *Proceedings of the 2021 ACM Workshop on Information Hiding and Multimedia Security*, Association for Computing Machinery, 2021, 159–64, DOI: 10.1145/3437880.3460398.

[22] 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*, Association for Computing Machinery, 2017, 506–19, DOI: 10.1145/3052973.3053009.

[23] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An Imperative Style, High-Performance Deep Learning Library", in *Advances in Neural Information Processing Systems 32*, 2019, 8024–35.

[24] 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*, 115(3), 2015, 211–52, DOI: 10.1007/s11263-015-0816-y.

[25] 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, 371–4, DOI: 10.1109/MIPR.2019.00073.

[26] W. Sirichotedumrong, Y. Kinoshita, and H. Kiya, "Pixel-Based Image Encryption Without Key Management for Privacy-Preserving Deep Neural Networks", *IEEE Access*, 7, 2019, 177844–55, DOI: 10.1109/ACCESS.2019.2959017.

[27] 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, 745–9, DOI: 10.23919/Eusipco47968.2020.9287532.

[28] 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*, 8, 2019, e7, DOI: 10.1017/ATSIP.2018.33.

[29] 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, 674–8, DOI: 10.1109/ICIP.2019.8804201.

[30] M. Swanson, M. Kobayashi, and A. Tewfik, "Multimedia Data-embedding and Watermarking Technologies", *Proceedings of the IEEE*, 86(6), 1998, 1064–87, DOI: 10.1109/5.687830.

[31] 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.

[32] M. Tanaka, "Learnable Image Encryption", in *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, 2018, 1–2, DOI: 10.1109/ICCE-China.2018.8448772.

[33] 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, 269–77, DOI: 10.1145/3078971.3078974.

[34] 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*, Association for Computing Machinery, June 2021, 455–60, DOI: 10.1145/3453688.3461752.

[35] 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, 159–72, DOI: 10.1145/3196494.3196550.