IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 25, 2021, accepted September 5, 2021, date of publication September 14, 2021,
date of current version September 27, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3112684

SIA-GAN: Scrambling Inversion Attack Using Generative Adversarial Network

KOKI MADONO[1,2], MASAYUKI TANAKA[2,3], (Member, IEEE), MASAKI ONISHI[2], (Member, IEEE), AND TETSUJI OGAWA[1,2], (Member, IEEE)

[1]Department of Communications and Computer Engineering, Waseda University, Tokyo 169-8050, Japan
[2]Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology, Tokyo 135-0064, Japan
[3]Department of Systems and Control Engineering, School of Engineering, Tokyo Institute of Technology, Tokyo 152-8550, Japan

Corresponding author: Koki Madono (madonomadonorunning@gmail.com)

This work was supported by JST CREST under Grant JPMJCR19F5.

ABSTRACT This paper presents a scrambling inversion attack using a generative adversarial network (SIA-GAN). This method aims to evaluate the privacy protection level achieved by image scrambling method. For privacy-preserving machine learning, scrambled images are often used to protect visual information, assuming that searching the scramble parameters is highly difficult for an attacker due to the application of complex image scrambling operations. However, the security of such methods has not been thoroughly investigated. SIA-GAN learns the mapping between pairs of scrambled images and original images, then attempts to invert image scrambling. Therefore, the attacker is assumed to have real images whose domain is the same as that of scrambled images. Experimental results demonstrate that scrambled images cannot be recovered if block shuffling is applied as a scrambling operation. The experimental code of SIA-GAN is available at https://github.com/MADONOKOUKI/SIA-GAN.

INDEX TERMS Artificial intelligence, machine learning, computer vision, visual information hiding, image scrambling.

I. INTRODUCTION

Deep neural networks (DNNs) have produced impressive results for various computer vision tasks [1]–[3] owing to the rapid advancement of neural networks. DNNs can satisfy user demands for training on personal data. However, personal images often contain sensitive information (e.g., faces, addresses, social relationships). Therefore, privacy protection is important to develop DNN solutions. Image scrambling [4]–[7] has been introduced to protect the privacy of visual information, and it enables privacy-preserving DNN training in external computing environments (e.g., cloud, shared server). In addition, image scrambling has low memory requirements and a low computational cost compared with other methods such as homomorphic encryption [8]–[13]. Therefore, it is suitable for privacy-preserving machine learning. Figure 1 shows the diagram of image scrambling using extended learnable encryption (ELE) [6], which simply performs pixel shuffling (i.e., block shuffling and a block-wise pixel operation). By applying pixel shuffling, representative features for classification can be extracted from scrambled images while hiding visual information [8]–[13].

The development in [6] is a basis for our study. It shows that an adaptation network notably contributes to maintain the classification performance because features should be extracted from scrambled images. Block shuffling, an image scrambling method, effectively hides visual information, and an adaptation network is required to obtain high classification performance for model inversion. Therefore, a scrambled image classification framework can provide a secure solution for machine learning in external computing environments. Nevertheless, image scrambling should be further investigated regarding security against unexpected attacks for its deployment in real settings.

Cryptanalysis methods [17]–[19] are proposed to evaluate the security level of permuted images. These works use the correlation and histogram of images for inversion. Those cryptanalysis methods are only for gray-scale images permuted pixel location. For that reason, those methods cannot be directly applicable to the scramble image approaches [4]–[7] because the scramble image approaches are for the RGB images, and they apply not only pixel location permutation but also channel-wise shuffling and negative-positive transformation.

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

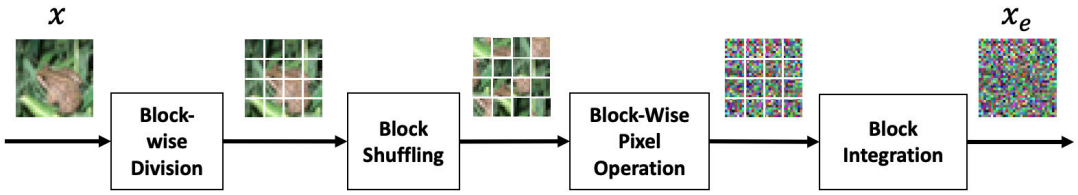


FIGURE 1. Diagram of ELE [6] for image scrambling. It comprises two main operations, block shuffling and a block-wise pixel operation.

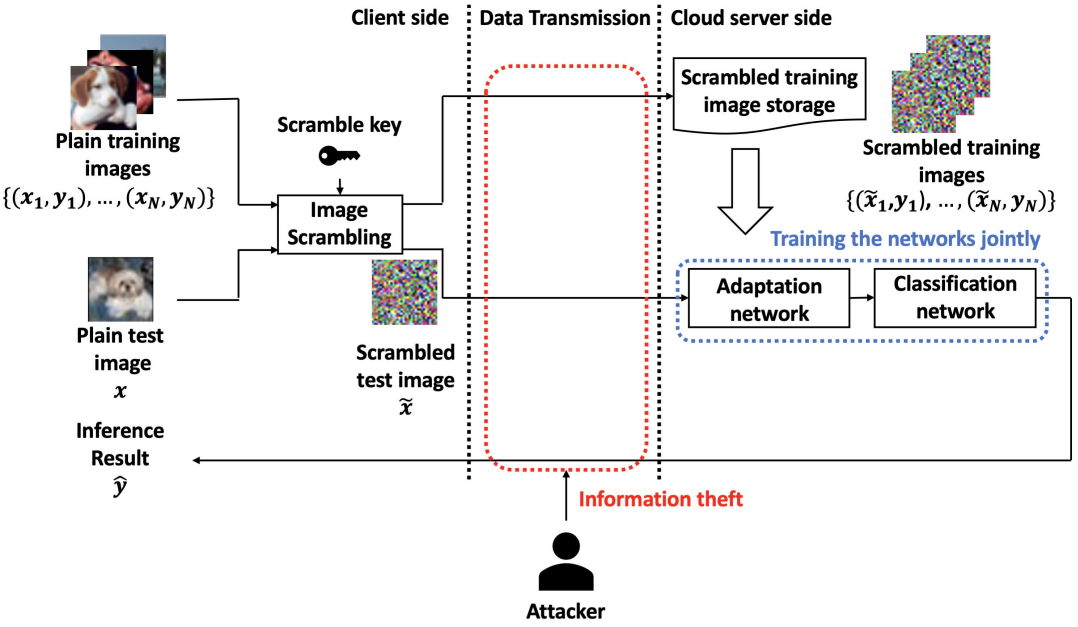


FIGURE 2. Scenario of machine learning using image scrambling under compromised cloud and data leaked to attacker. During training, the user sends pairs of images and labels to the cloud. During inference, a posterior probability is calculated from a scrambled image in the cloud. The attacker can steal a pair of images and labels during training and images during inference.

Recently, generative adversarial network (GANs) [20] have been applied to model inversion attacks [21]. Such attacks aims to estimate sensitive information in the training data of machine learning models. In addition, they use partial public information, which can be very generic, to learn the prior distribution of a real image through a GAN for guiding the inversion process, thus avoiding the reconstruction of private training data from scratch. Using blurred or corrupted private images, the original private images can be mostly recovered through a model inversion attack. However, attacking scrambled images is challenging because they completely hide visual information, outperforming blurred or corrupted images.

We propose a scrambling inversion attack using a GAN (SIA-GAN) that targets scrambled images using an adaptation network, to recover semantic information. The proposed SIA-GAN can be used to evaluate the effectiveness of image scrambling methods. SIA-GAN minimizes the divergence in feature extraction from scrambled images and real images, whose contents are similar to those of images before scrambling.

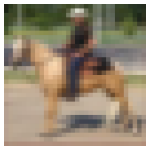
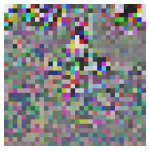
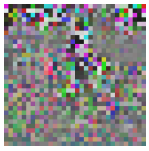
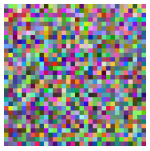
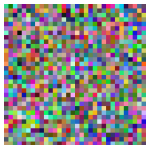
Experiments confirmed that SIA-GAN effectively attacks scrambled images, and our novel adaptation network provides a high performance for GAN-based attacks. Furthermore, SIA-GAN can attack reference images whose distribution is different from that of the target images to be revealed. The experimental results show that block shuffling is one of the most secure methods to generate robust scrambled images.

This study provides contributions in the following aspects:

- Introduction of a simple and efficient method, SIA-GAN to evaluate the security of scrambled images,
- Effective training by integrating an adaptation network into a GAN,
- Qualitative evaluations of visual information hiding in terms of learned perceptual image patch similarity (LPIPS).

The remainder of this paper is organized as follows. The scenario considered in this study for image scrambling machine learning is presented in Section II. The evaluated image scrambling methods are described in Section III-A. The proposed SIA-GAN is detailed in Section IV. The evaluation results of SIA-GAN are reported in Section V. Finally,

TABLE 1. Existing image scrambling methods. A block has $B \times B$ pixels, and an image contains N blocks.

Dataset	Plain image	PE [14]	Random PE [15]	LE [4]	ELE [6]	EtC [16]
Sample image						
Block division				✓	✓	✓
Block key				Common	Different	Different
Pixel key		Different	Different			
Random key generation			✓			
Block shuffling					✓	✓
(Blockwise) pixel operation		Color component shuffling, negative–positive transform	Color component shuffling, negative–positive transform	Pixel shuffling, negative–positive transform	Pixel shuffling, negative–positive transform	Block rotation & inversion, color component shuffling, negative–positive transform

Section VI concludes this paper. SIA-GAN allows to verify the privacy protection level of scrambled images, and block shuffling can effectively protect sensitive information in scrambled images.

II. EVALUATION SCENARIO

Figure 2 shows the scenario for model training considered in this study. During training, the user prepares a classification network and image storage in the cloud. When the users sends an image–label pair of training data, image scrambling is applied to protect conceal visual information. The classification network is, then, trained using the labeled scrambled images. During inference, the user sends a scrambled test image, and a posterior probability is inferred by the classification network in the cloud and retrieved to the user.

When data are transmitted in a public computing environment, an attacker may steal data from training or testing. In this case, the attacker can obtain the following information:

- Scrambled images used for training or testing
- The block size of scrambled images
- Class information (e.g., dog, cat)

Scrambled images can be intercepted when the user sends them to the cloud. In addition, the block size can be obtained when the user constructs an adaptation network, and class information can be obtained from the labels. Then, real images can be crawled using both the class information and intercepted images through methods such as applying a GAN for image scrambling inversion.

III. RELATED WORKS

A. IMAGE SCRAMBLING METHODS

Various image scrambling methods have been proposed to prevent cyberattacks to machine learning methods implemented in external computing environments [4]–[7]. Image scramble relies on diverse operations to render visual image features imperceptible to humans.

Figure 1 shows an overview of ELE image scrambling [6]. First, an input image is divided into blocks. Then, the block locations are shuffled. In each block, the intensity location is shuffled. Finally, the scrambled image is obtained by integrating the blocks.

Table 1 lists the characteristics of existing image scrambling methods. Pixel-based image encryption (PE) [14] uses the negative–positive transform and color component shuffling with a unique key per pixel. Random PE [15] also uses the negative–positive transform and color component shuffling with a key pixel, but the keys are generated at every execution of scrambling. Therefore, the scramble key is not restored. Learnable encryption (LE) [4] shuffles pixels and applies negative–positive transforms with the same key for every block. Then, each block is split into eight bits, four upper and four lower bits, to generate from three to six block channels. Subsequently, the pixels are randomly shuffled, yielding diverse combinations. The intensity of the randomly selected pixels is then reversed, yielding more combinations. ELE [6] uses block-wise pixel shuffling with a unique key per block and block location shuffling. Encryption

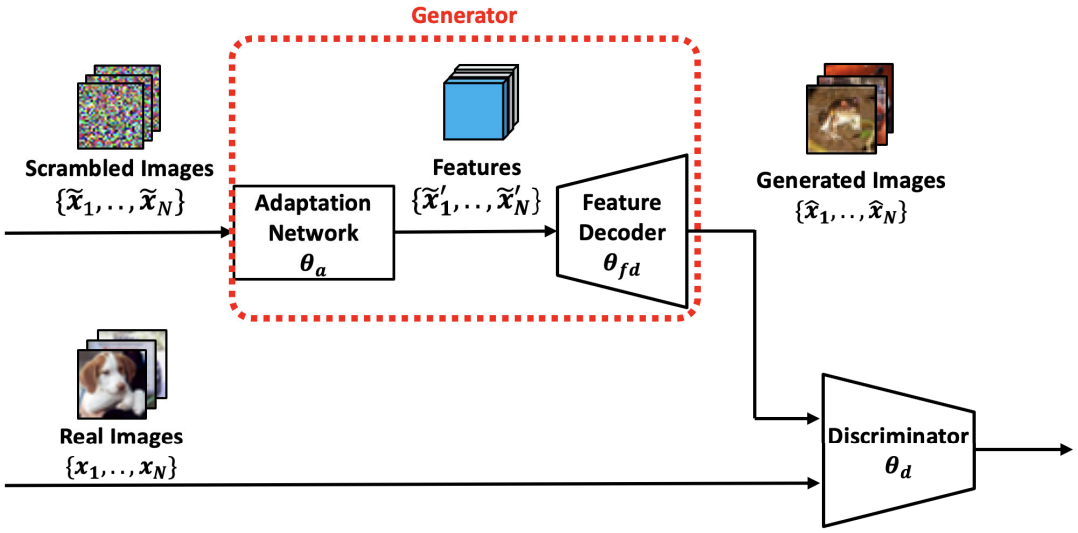


FIGURE 3. Overview of proposed SIA-GAN. The combination of adaptation network and feature decoder is intended to demonstrate feature extraction from a block-wise scrambled image.

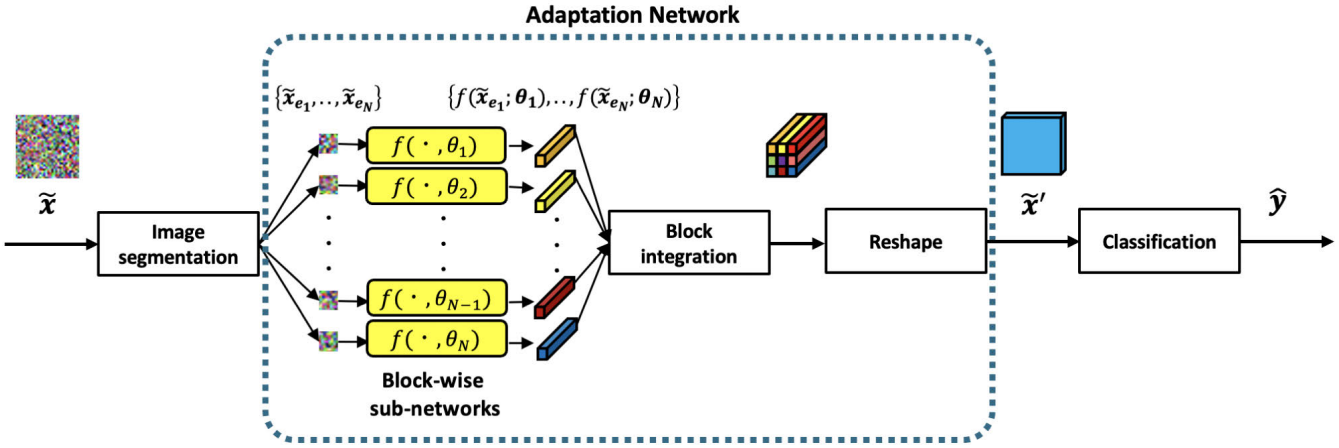


FIGURE 4. Architecture of adaptation network in SIA-GAN. A scrambled image is segmented into image blocks, which are input to block-wise subnetworks. Each feature is integrated in its original location, and the combined features are reshaped to adjust the input size. The resulting map is input to a generator. (The colored boxes represent trainable processing).

then compression (EtC) [16] uses block rotation followed by inversion, negative–positive transform, color component shuffling, and block location shuffling with the same key for every block.

The abovementioned image scrambling methods hide sensitive information in an image because their underlying operations completely distort visual information in an image. In this study, we confirmed that these methods are less secure under a GAN-based attack if the partial public information related to the scrambled images is obtained. Although restoring the original state from a scrambled image is an ill-posed problem, the proposed SIA-GAN may solve it.

B. CRYPTANALYSIS

Cryptanalysis methods [17]–[19] have been to evaluate the security of permuted images that aims at ensuring security. These works invert the original state of permuted images using a greedy searching way. They use gray-scale images that only permute the pixel values at the spatial space. The abovementioned cryptanalysis methods use the correlation and histogram of images for inversion. In the case of scrambled images, these methods are difficult to apply since the scrambled image is an RGB image and has other shuffling operations such as channel-wise shuffling and negative-positive transformation. Considering this point, our proposed attack aims at inverting the original state from such a scramble image. Since we use a data-driven approach, the proposed SIA-GAN can naturally learn the original state from the image distribution. In addition, we can quickly invert the original state of scrambled images since the proposed SIA-GAN only needs a forwarding time.

IV. PROPOSED SIA-GAN

This section introduces the proposed SIA-GAN, which aims to unveil vulnerabilities in image scrambling methods. To successfully attack a scrambled image using a GAN, we use an adaptation network in the generator to learn the image scrambling operations SIA-GAN aims at demonstrating the dangers of scrambled images. To successfully attack the scrambled images, we use an adaptation network in generator that can learn the way of image scrambling.

A. OVERVIEW

SIA-GAN is proposed to attack against the scrambled images by restoring the original information through model inversion. Figure 3 shows an overview of SIA-GAN, which consists of a generator and a discriminator. The generator comprises an adaptation network and a feature decoder. The details of the adaptation network, feature decoder, and discriminator are listed in Tables 4, 5, and 6, respectively. SIA-GAN is trained to minimize the difference between pairs of the scrambled images and real images and thus determine find the mapping applied for image scrambling. The adaptation network is essential to extract semantic features from the scrambled images.

B. GENERATOR

1) ADAPTATION NETWORK

Figure 4 illustrates the architecture of the adaptation network, which consists of block-wise subnetworks,

In block decomposition, scrambled image $\boldsymbol{x}_e$ is segmented into N blocks of $B \times B$ pixels, $\{\boldsymbol{x}_{e_1}, \boldsymbol{x}_{e_2}, \cdots, \boldsymbol{x}_{e_N}\}$, where $\boldsymbol{x}_{e_b}$ represents a block (i.e., segmented image). Each block is transformed by the corresponding block-wise subnetwork, $f(\boldsymbol{x}_{e_b}; \boldsymbol{\theta}_b)$. Then, the adaptation network is individually trained on each block to handle images processed by block-wise scrambling with different keys. Subsequently, the extracted features are combined on block composition. To input the features to the generator, the combined features are reshaped. In this study, we used a pixel shuffling layer [22] for reshaping.

2) FEATURE DECODER

We propose feature decoder $\boldsymbol{\theta}_{fd}$ to convert features $\{\widetilde{\boldsymbol{x}}'_{\boldsymbol{1}}, \ldots, \widetilde{\boldsymbol{x}}'_{\boldsymbol{N}}\}$ into real images $\{\hat{\boldsymbol{x}}_{\boldsymbol{1}}, \ldots, \hat{\boldsymbol{x}}_{\boldsymbol{N}}\}$. During training, the feature decoder aims to recover the original state from the scrambled images. Table 5 lists the architecture details of the feature decoder. We adapt spectral normalization for GAN [23] in the proposed feature decoder to generate high-quality images.

C. DISCRIMINATOR

We introduce discriminator $\boldsymbol{\theta}_d$ to determine whether input images are real or synthetic. During training, the discriminator guides the generator to produce more realistic images. Table 6 lists the architecture details of the discriminator. Again, we adapt spectral normalization for GAN [23] in the proposed discriminator for consistency with the generator.

D. TRAINING

During training, we jointly train the discriminator and generator, whose adaptation network and feature decoder are simultaneously updated.

Scrambled image $\{\widetilde{\boldsymbol{x}}_{\boldsymbol{1}}, \ldots, \widetilde{\boldsymbol{x}}_{\boldsymbol{N}}\}$ is input to adaptation network $\boldsymbol{\theta}_a$ for feature extraction. Output $\{\widetilde{\boldsymbol{x}}'_{\boldsymbol{1}}, \ldots, \widetilde{\boldsymbol{x}}'_{\boldsymbol{N}}\}$ with rich information about the original image is fed to the generator to recover the original state of the scrambled image, $\{\hat{\boldsymbol{x}}_{\boldsymbol{1}}, \ldots, \hat{\boldsymbol{x}}_{\boldsymbol{N}}\}$. Finally, output $\{G(\hat{\boldsymbol{x}}_{\boldsymbol{1}}), \ldots, G(\hat{\boldsymbol{x}}_{\boldsymbol{N}})\}$ is fed to the discriminator to learn the mapping onto the real image.

The loss function for training the adaptation network, generator, and discriminator is given by

$$\begin{aligned} L_G &= L_{adv}(G, \hat{D}) \\ L_D &= L_{adv}(\hat{G}, D) \end{aligned} \tag{1}$$

where G denotes the generator, D denotes the discriminator, $\hat{G}$ denotes the generator with parameter freezing, $\hat{D}$ denotes the discriminator with parameter freezing, and L_{adv} denotes the adversarial loss in spectral normalization for GAN [23]. The adversarial loss is aimed at transforming the scrambled image into the original real image through adversarial training. Adversarial loss $L_{adv}(G, D)$ is given by

$$L_{adv}(G, D) = \mathbb{E}[\log D(\boldsymbol{x})] + \mathbb{E}[\log(1 - D(G(\tilde{\boldsymbol{x}})))] \tag{2}$$

where $\boldsymbol{x}$ denotes the real image, $\tilde{\boldsymbol{x}}$ denotes the scrambled image, and $G(\tilde{\boldsymbol{x}})$ denotes the generator output. Using Eq. 2, the parameters of generator and discriminator are updated alternately. In this study, we used the hinge loss to compute the adversarial loss. The adversarial loss of the discriminator is computed as

$$\begin{aligned} L_{adv}(\hat{G}, D) = \mathbb{E}[\min(0, -1 + D(\boldsymbol{x})] \\ + \mathbb{E}[\min(0, -1 - D(\hat{G}(\tilde{\boldsymbol{x}})))] \end{aligned} \tag{3}$$

where $L_{adv}(\hat{G}, D)$ denotes the update of the discriminator with parameter freezing of the adaptation network and generator.

The adversarial loss of the generator is computed as

$$L_{adv}(G, \hat{D}) = \mathbb{E}[\hat{D}(G(\tilde{\boldsymbol{x}}))] \tag{4}$$

where $L_{adv}(G, \hat{D})$ denotes the update of the adaptation network and generator with parameter freezing of the discriminator. This adversarial loss aims at updating the generator parameters.

E. ATTACK

During the application of SIA-GAN, we assumed that the image scrambling method from the test data was the same as that from the training data. The inference operation is given by

$$x = G(\hat{x}) \tag{5}$$

where x denotes the generator output, G denotes the generator, and $\hat{x}$ denotes the scrambled image in test data.

TABLE 2. Images generated by SIA-GAN from scrambled images.

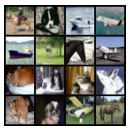
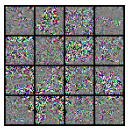
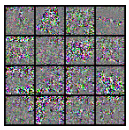
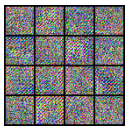
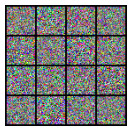
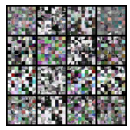
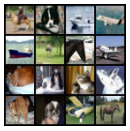
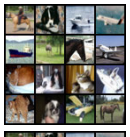
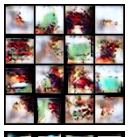
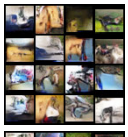
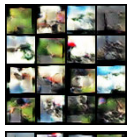
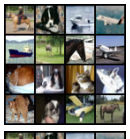
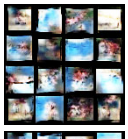
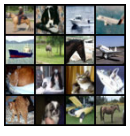
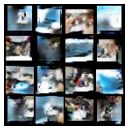
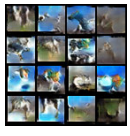
	Adaptation network	Real image	Answer (CIFAR-10)	PE [14]	Random PE [15]	LE [4]	ELE [6]	EtC [16]
Scrambled image								
SIA-GAN		CIFAR-100						
SIA-GAN	✓	CIFAR-100						
SIA-GAN		CIFAR-10						
SIA-GAN	✓	CIFAR-10						

TABLE 3. The original LPIPS scores of scrambled images are shown for comparison, and the other LPIPS scores are those from images generated by applying SIA-GAN.

	Adaptation	Real	PE [14]	Random PE [15]	LE [4]	ELE [6]	EtC [16]
Scrambled image			0.3255	0.3231	0.3035	0.2947	0.2074
GAN-based attack		Diff.	0.1845	0.1954	0.1069	0.1589	0.2628
GAN-based attack	✓	Diff.	0.1632	0.1960	0.1565	0.1483	0.2116
GAN-based attack		Same	0.1964	0.1509	0.1451	0.1897	0.2067
GAN-based attack	✓	Same	0.1719	0.1559	0.0914	0.1512	0.2090

TABLE 4. Architecture of adaptation network. Batch normalization is applied before activation. B denotes the block size of image scrambling and H is equal to the 32/B (ReLU, rectified linear unit).

Layer type	Kernel size	Spectral normalization	Batch normalization	Activation function	Output
Input	-	-	-	-	$3 \times 32 \times 32$
Convolution	$B \times B$	✓	✓	Leaky ReLU	$\{16 \times B \times B\} \times H \times H$
Pixel shuffle [22]	-	-	-	-	$16 \times 32 \times 32$

V. EXPERIMENTAL VALIDATION

The performance of SIA-GAN was evaluated using the adaptation network to confirm the effectiveness of feature extraction. The CIFAR-10 and CIFAR-100 datasets were used for both qualitative and qualitative evaluations. For the qualitative evaluation, images were generated by SIA-GAN to understand the attack results regarding human perception. For the quantitative evaluation, we used the reliable LPIPS score [24]. LPIPS was recently proposed to evaluate the perceptual similarity of generated images and original images.

We evaluated original image and scrambled images obtained by applying PE, random PE, LE, ELE, and EtC (Table 1). The images were converted into block-wise or pixelwise scrambled images to be used as inputs to the adaptation network. Data augmentation was applied before block-wise or pixelwise scrambling.

For evaluation, we used the adaptation network, generator, and discriminator as detailed in Tables 4, 5, 6, respectively. In adaptation network, the kernel size was adjusted according to the scrambled images. For PE and random PE, the kernel size was 1×1 because these methods use pixel-wise scrambling operations. For LE, ELE, and EtC, the kernel size was 4×4 given the block size of 4×4 pixels.

TABLE 5. Architecture of feature decoder. Spectral normalization is conducted before batch normalization, which is followed by activation.

Layer type	Kernel size	Spectral normalization	Batch normalization	Activation function	Output
Input	-	-	-	-	$16 \times 32 \times 32$
Convolution	3×3	✓	✓	Leaky ReLU	$64 \times 32 \times 32$
ResBlock (Figure 5)	-	-	-	Leaky ReLU	$64 \times 32 \times 32$
ResBlock (Figure 5)	-	-	-	Leaky ReLU	$64 \times 32 \times 32$
ResBlock (Figure 5)	-	-	-	Leaky ReLU	$64 \times 32 \times 32$
Convolution	3×3	✓	✓	Leaky ReLU	$64 \times 32 \times 32$
Convolution	3×3	✓	✓	Tanh	$3 \times 32 \times 32$

TABLE 6. Architecture of discriminator. Spectral normalization is applied before batch normalization, which is followed by activation.

Layer type	Kernel size	Spectral normalization	Batch normalization	Activation function	Output
Input	-	-	-	-	$3 \times 32 \times 32$
Convolution	3×3	✓	-	Leaky ReLU	$128 \times 32 \times 32$
Convolution	4×4	✓	-	Leaky ReLU	$128 \times 16 \times 16$
Convolution	3×3	✓	-	Leaky ReLU	$256 \times 16 \times 16$
Convolution	4×4	✓	-	Leaky ReLU	$256 \times 8 \times 8$
Convolution	3×3	✓	-	Leaky ReLU	$512 \times 8 \times 8$
Convolution	4×4	✓	-	Leaky ReLU	$512 \times 4 \times 4$
Self attention [25]	-	-	-	-	$512 \times 4 \times 4$
Convolution	3×3	✓	-	Leaky ReLU	$1024 \times 4 \times 4$
Self attention [25]	-	-	-	-	$1024 \times 4 \times 4$
Flatten	-	-	-	-	16384
Linear	-	-	-	-	1

A. SIA-GAN APPLIED TO SCRAMBLED IMAGES

1) EXPERIMENTAL SETUP

The minibatch size was set to 64 during training and testing. Adam optimizer was applied with $beta1 = 0.0$, $beta2 = 0.9$. A learning rate of both the adaptation network, and generator is set to 1e-4, and discriminator is set to 4e-4. The training model was updated after every iteration, the network was trained for 100 epochs. Although GAN training generally requires several epochs, 100 epochs provided satisfactory results to verify the effectiveness of the proposed SIA-GAN in the evaluations. The 50,000 images in the CIFAR-10 dataset were used for training the model. All training images were converted into scrambled images to resemble an attacker crawling those images. For the real images, we considered the CIFAR-10 and CIFAR-100 datasets to represent a scenario in which the attacker does not know the original dataset although similar datasets can be collected.

2) EXPERIMENTAL RESULTS

Table 2 lists the images generated by the proposed SIA-GAN. If block shuffling is applied, as in ELE or EtC, the scrambled imagescannot be converted back into the original images. Therefore, block shuffling seems suitable to protect scrambled images from GAN-based attacks.

For the other types of scrambled images, SIA-GAN recovered the original images from the scrambled images. Although the quality of PE and random PE aren't good enough, the structure of content can be confirmed. Although the quality of images recovered after the application of PE and random PE is poor, the structure of their contents can be inferred.

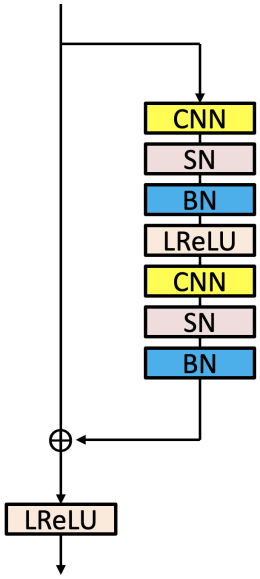


FIGURE 5. Architecture of ResBlocks (residual blocks) in generator. (CNN; convolutional layer with kernel of 3 × 3; SN, spectral normaliztion layer; BN, batch normalization; LReLU, leaky ReLU layer).

Although random PE applies scrambling with random parameters, SIA-GAN can suitably reconstruct the original state. Regarding the real images, we confirmed that the attack results are similar, indicating that using natural images is important to attack scrambled images.

Overall, we experimentally confirmed that block shuffling in the scrambled image is important in image scrambling to achieve robust protection against model-inversion attacks that minimize the classification error.

Table 3 lists the image quality of scrambled images and recovered images in term of LPIPS. If the proposed attack can reconstruct the original state, the corresponding LIPIS score is high.

These results indicate that SIA-GAN can reconstruct the original state if block shuffling is not applied for scrambling. The LPIPS scores quantitatively show reasonable reconstruction when applying the proposed SIA-GAN.

VI. CONCLUSION

We propose SIA-GAN to evaluate the privacy protection level provided by various types of image scrambling methods. SIA-GAN integrates an adaptation network into a conventional GAN to extract visual information from scrambled images. This type of attack processes scrambled images using model inversion. For SIA-GAN, we consider more advantageous settings for malicious attacks than using real images whose domain is the same as that of scrambled images. Indeed, we consider pairs of real images and scrambled images. Then, the GAN minimizes the distribution difference between scrambled images and real images to learn the image scrambling operations. Experimental results indicate that block shuffling effectively protects scrambled images from the GAN-based attack. In practice, block shuffling degrades the classification performance, as described in our previous work [6]. For privacy-preserving machine learning, block shuffling should be applied during image scrambling.

APPENDIX
MODEL ARCHITERCURE

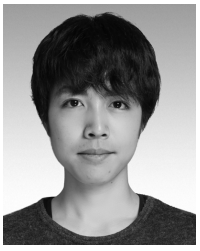
See Figure 5 and Tables 4–6.

ACKNOWLEDGMENT

This work was supported by JST CREST Grant Number JPMJCR19F5.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
[2] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 1150–1157.
[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[4] M. Tanaka, "Learnable image encryption," in *Proc. IEEE Int. Conf. Consum. Electron.-Taiwan (ICCE-TW)*, May 2018, pp. 1–2.
[5] T. Chuman, W. Sirichotedumrong, and H. Kiya, "Encryption-then-compression systems using grayscale-based image encryption for JPEG images," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 6, pp. 1515–1525, Jun. 2019.
[6] K. Madono, M. Tanaka, M. Onishi, and T. Ogawa, "Block-wise scrambled image recognition using adaptation network," 2020, *arXiv:2001.07761*. [Online]. Available: http://arxiv.org/abs/2001.07761
[7] 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 *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 674–678.
[8] S. Singh, Y.-S. Jeong, and J. H. Park, "A survey on cloud computing security: Issues, threats, and solutions," *J. Netw. Comput. Appl.*, vol. 75, pp. 200–222, Nov. 2016.
[9] Z. Shan, K. Ren, M. Blanton, and C. Wang, "Practical secure computation outsourcing: A survey," *ACM Comput. Surv.*, vol. 51, no. 2, p. 31, 2018.
[10] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. E. Lauter, M. Naehrig, and J. R. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proc. ICML*, 2016, pp. 201–210.
[11] T. van Elsloo, G. Patrini, and H. Ivey-Law, "SEALion: A framework for neural network inference on encrypted data," 2019, *arXiv:1904.12840*. [Online]. Available: http://arxiv.org/abs/1904.12840
[12] P. Xie, B. Wu, and G. Sun, "BAYHENN: Combining Bayesian deep learning and homomorphic encryption for secure DNN inference," in *Proc. IJCAI*, 2019, pp. 1–7.
[13] Q. Lou, B. Feng, G. C. Fox, and L. Jiang, "Glyph: Fast and accurately training deep neural networks on encrypted data," 2019, *arXiv:1911.07101*. [Online]. Available: http://arxiv.org/abs/1911.07101
[14] T. Chuman, K. Kurihara, and H. Kiya, "On the security of block scrambling-based EtC systems against extended jigsaw puzzle solver attacks," *IEICE Trans. Inf. Syst.*, vol. E101.D, no. 1, pp. 37–44, 2018.
[15] 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. 177844–177855, 2019.
[16] 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," 2019, *arXiv:1905.01827*. [Online]. Available: http://arxiv.org/abs/1905.01827
[17] M. Li, D. D. Lu, Y. Xiang, Y. Zhang, and H. Ren, "Cryptanalysis and improvement in a chaotic image cipher using two-round permutation and diffusion," *Nonlinear Dyn.*, vol. 96, no. 1, pp. 31–47, 2019.
[18] A. Jolfaei, X.-W. Wu, and V. Muthukkumarasamy, "On the security of permutation-only image encryption schemes," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 2, pp. 235–246, Feb. 2016.
[19] W. Song, X. Liao, D. Weng, Y. Zheng, Y. Liu, and Y. Wang, "Cryptanalysis of phase information based on a double random-phase encryption method," *Opt. Commun.*, vol. 497, Oct. 2021, Art. no. 127172.
[20] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, and D. Warde-Farley, "Generative adversarial networks," *CoRR*, vol. abs/1406.2661, pp. 1–25, Nov. 2014.
[21] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D. Song, "The secret revealer: Generative model-inversion attacks against deep neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 250–258.
[22] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874–1883.
[23] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," 2018, *arXiv:1802.05957*. [Online]. Available: http://arxiv.org/abs/1802.05957
[24] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.
[25] H. Zhang, J. I. Goodfellow, N. D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," *CoRR*, vol. abs/1805.08318, pp. 1–10, May 2018.



KOKI MADONO received the B.S. and M.S. degrees in fundamental science and engineering from Waseda University, Tokyo, Japan, in 2019 and 2021, respectively, where he is currently pursuing the Ph.D. degree with the Department of Fundamental Science and Engineering. He has been a Research Assistant with the Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology, since 2019. His research interests include artificial intelligence, machine learning, image processing, privacy, and security.

MASAYUKI TANAKA (Member, IEEE) received the B.S. and M.S. degrees in control engineering and the Ph.D. degree from Tokyo Institute of Technology, in 1998, 2000, and 2003, respectively. He was a Software Engineer at Agilent Technology, from 2003 to 2004, a Research Scientist with Tokyo Institute of Technology, from 2004 to 2008, and an Associate Professor with the Graduate School of Science and Engineering, Tokyo Institute of Technology, from 2008 to 2016. In addition, he was a Visiting Scholar with the Department of Psychology, Stanford University, from 2013 to 2014, and an Associate Professor with the School of Engineering, Tokyo Institute of Technology, from 2016 to 2017. Since 2017, he has been a Senior Researcher with the National Institute of Advanced Industrial Science and Technology.

MASAKI ONISHI (Member, IEEE) received the M.Eng. and Dr.Eng. degrees from Osaka Prefecture University, in 1999 and 2002, respectively. From 2002 to 2006, he was a Research Scientist with the Bio-Mimetic Control Research Center, RIKEN. Since 2006, he has been a Research Scientist with the National Institute of Advanced Industrial Science and Technology (AIST). His research interests include computer vision, video surveillance, and human–robot interaction.

TETSUJI OGAWA (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Waseda University, Tokyo, Japan, in 2000, 2002, and 2005, respectively. He was a Research Associate, from 2004 to 2007, and a Visiting Lecturer with Waseda University, in 2007. From 2007 to 2012, he was an Assistant Professor with the Waseda Institute for Advanced Study. Since 2012, he has been an Associate Professor with Waseda University and Egypt—Japan University of Science and Technology. He was a Visiting Scholar with the Center for Language and Speech Processing, Johns Hopkins University, USA, from June 2012 to September 2012 and from June 2013 to August 2013. He was a Visiting Scholar with the Speech Processing Group and the Faculty of Information Technology, Brno University of Technology, Czech Republic, from June 2014 to July 2014 and from May 2015 to August 2015. His research interests include stochastic modeling for pattern recognition, speech enhancement, and speech and speaker recognition. He is a member of the Information Processing Society of Japan and Acoustic Society of Japan.