PAPER *Special Section on Enriched Multimedia—Application of Multimedia Technology and Its Security—*

Image Identification of Encrypted JPEG Images for Privacy-Preserving Photo Sharing Services

Kenta IIDA[†], ***Member and*** **Hitoshi KIYA**[†a)], ***Fellow***

SUMMARY We propose an image identification scheme for double-compressed encrypted JPEG images that aims to identify encrypted JPEG images that are generated from an original JPEG image. To store images without any visual sensitive information on photo sharing services, encrypted JPEG images are generated by using a block-scrambling-based encryption method that has been proposed for Encryption-then-Compression systems with JPEG compression. In addition, feature vectors robust against JPEG compression are extracted from encrypted JPEG images. The use of the image encryption and feature vectors allows us to identify encrypted images recompressed multiple times. Moreover, the proposed scheme is designed to identify images re-encrypted with different keys. The results of a simulation show that the identification performance of the scheme is high even when images are recompressed and re-encrypted.

key words: *image identification, JPEG, Encryption-then-Compression system, privacy-preserving*

1. Introduction

With the rapid growth of social networking services (SNSs) and cloud computing, photo sharing via various services has greatly increased. Generally, images are uploaded and stored in a compressed form to reduce the amount of data. In the uploading process for these SNSs, it is known that service providers employ manipulation, such as recompression. In addition, most of the content includes sensitive information, such as personal data and copyrights. Thus, it is required that images on photo-sharing services be prevented from leakage and unauthorized use by service providers.

For privacy-preserving photo sharing on these services, three requirements need to be satisfied: 1) protection of visual information, 2) tolerance for recompression after encryption, and 3) identification of encrypted images. In terms of requirement 1, a lot of studies on secure and efficient communications have been reported [1]–[20], [26]. The encryption methods [10]–[16] are based on the random operation of DCT coefficients in JPEG images, and some of them [12]–[16] satisfy requirements 1 and 3. However, they do not consider the influence of recompression, i.e. requirement 2. The searchable encryption method with a (k, n)-threshold secret sharing scheme [17] has a high security level when the number of colluders is less than $n-1$, but this method does not satisfy requirement 2. The pixel-based encryption methods [18]–[20] aim to reduce the correlation between the adjacent pixels to enhance the security. Therefore, image compression is not applicable to these methods. As systems that satisfy both requirements 1 and 2, Encryption-then-Compression (EtC) systems have been developed [2], [3], [6]–[9], [26]. In this paper, we focus on a block-scrambling-based image encryption method that has been proposed for EtC systems.

Regarding requirement 3, image retrieval and the identification of encrypted images have never been considered for EtC systems, although image identification and retrieval schemes that are robust against JPEG compression have been proposed for unencrypted images [21]–[25]. However, the performance of these schemes is degraded in the case of identification between encrypted images and corresponding re-encrypted images, so, to satisfy all requirements, novel image identification methods are required.

Thus, we propose a novel image identification scheme for encrypted JPEG images compressed under various coding conditions. Image encryption is carried out by extending the method based on block-scrambling for EtC systems [2], [3], [6]–[9], [26]. The extended method has steps for two-layer encryption. Moreover, the feature vector used for identification is designed for images encrypted by the extended method. The use of the two-layer encryption and features allows us to identify re-encrypted images without re-calculating the features. Simulation results show that the proposed scheme has a high identification performance, even when images are recompressed and re-encrypted.

2. Preliminaries

2.1 EtC Image

We focus on EtC images which have been proposed for Encryption-then-Compression (EtC) systems with JPEG compression [2], [3], [6]–[9], [26]. EtC images not only have almost the same compression performance as that of unencrypted images, but also enough robustness against various ciphertext-only attacks including jigsaw puzzle solver attacks [9]. The procedure for generating EtC images is conducted as below (see Figs. 1 and 2) [7].

1) Divide an image with $X \times Y$ pixels into non over-lapping 8×8 blocks.
2) Permute randomly M divided blocks by using a random integer secret key K_1, where $M = \lfloor \frac{X}{8} \rfloor \times \lfloor \frac{Y}{8} \rfloor$.
3) Rotate and invert randomly each divided block by using a random integer secret key K_2.

Manuscript received March 25, 2019.
Manuscript revised August 7, 2019.
Manuscript publicized October 25, 2019.

[†] The authors are with the Tokyo Metropolitan University, Hino-shi, 191–0065 Japan.
a) E-mail: kiya@tmu.ac.jp

DOI: 10.1587/transinf.2019MUP0006

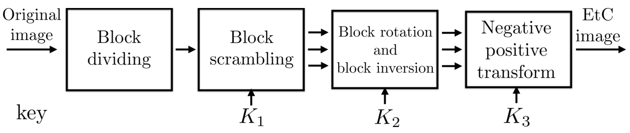
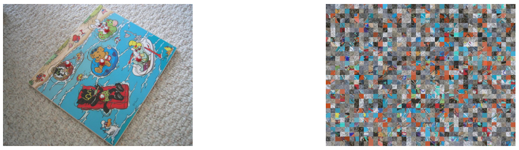


Fig. 1 Generating EtC image.



(a) Original image

(b) Encrypted image

Fig. 2 Example of original image and encrypted image.

4) Apply a negative-positive transform to each divided block by using a random integer secret key K_3. In this step, transformed pixel value at the position (x, y) in a block $I_{np}(x, y)$ is computed from the pixel value at the same position $I(x, y)$ in a 8 × 8 block of an image applied step from 1 to 3 to by

$$I_{np}(x, y) = \begin{cases} I(x, y), \ b = 0, \\ 255 - I(x, y), \ b = 1, \end{cases} \tag{1}$$

where b is a random binary value generated by K_3 under the probability $P(b) = 0.5$ and $0 \leq I(x, y)$, $I_{np}(x, y) \leq 255$.

In this paper, images encrypted by using these steps are referred to "EtC images".

2.2 Image Identification for JPEG Images

Let us consider a situation in which there are two or more compressed images generated under different or the same coding conditions. They originated from the same image and were compressed under various coding conditions. We refer to the identification of these images as "image identification." Note that the aim of the image identification is not to retrieve visually similar images.

The JPEG standard is the most widely used image compression standard. In the usual coding procedure, after color transformation from RGB space to YC_bC_r space and subsamples C_b and C_r, an image is divided into non-overlapping consecutive 8 × 8-blocks. All pixel values in each block are shifted from $[0, 255]$ to $[-128, 127]$ by subtracting 128, and DCT is then applied to each block to obtain 8 × 8 DCT coefficients. After that, the DCT coefficients are quantized, and the quantized coefficients are entropy-coded.

A DC coefficient DC in each block is obtained by using the following equation, where $I(x, y)$ represents a pixel value at the position (x, y) in a block.

$$DC = \frac{1}{8} \sum_{x=0}^{7} \sum_{y=0}^{7} (I(x, y) - 128) \tag{2}$$

The range of the DC coefficients is $[-1024, 1016]$. It has been reported that the features extracted from DC coefficients are effective for recompression in the conventional schemes [21], [22]. Therefore, the DC coefficients are used for the identification in this paper.

Table 1 Notations used in this paper.

O_i	ith original JPEG image
$E_i^{(j,k_0,k)}$	ith encrypted JPEG image generated from O_i with seeds k_0 and k and compressed j times (secret keys K_0 and $\mathbf{K}$ are generated from k_0 and k)
M	number of 8 × 8-blocks in image
$O_i(m)$	DC coefficient of mth block in image O_i $(0 \leq m < M)$
$E_i^{(j,k_0,k)}(m)$	DC coefficient of luminance in mth block of image $E_i^{(j,k_0,k)}$ $(0 \leq m < M)$

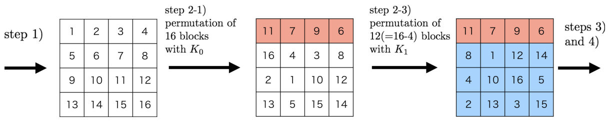


Fig. 3 Example of two-layer image encryption under $M = 16$ and $N = 4$.

3. Proposed Scheme

In this section, a novel two-layer image encryption method and the proposed identification scheme are explained. The combination of them enables us to avoid the effects of not only recompression but also re-encryption. The notations used in the following sections are shown in Table 1.

3.1 Two-Layer Image Encryption

In the proposed scheme, the novel image encryption method, which is an extension of the method mentioned in Sect. 2.1 for the proposed identification scheme, is conducted. The permutation process in step 2 is divided into two layers for identification. After dividing an image into 8 × 8 blocks, the following processes are performed, instead of step 2.

2-1) Permute randomly M divided blocks using a random integer secret key K_0.
2-2) Select a positive integer value N.
2-3) Permute randomly the last $M - N$ blocks using a random integer secret key K_1 again.

After that, steps 3 and 4 are carried out.

An example of the encryption process under $M = 16$ and $N = 4$ is shown in Fig. 3. It can be confirmed from Fig. 3 that the first four blocks are not permuted in step 2-3. Thus, the process in these steps does not change the positions of the first N blocks under the same K_0, even if K_1 is changed. This property will play an important role in the proposed identification scheme. Note that the permutation range in step 2-3 decreases, when the value of N increases. In contrast, it is expected that the use of large N values improves the accuracy of identification as shown later.

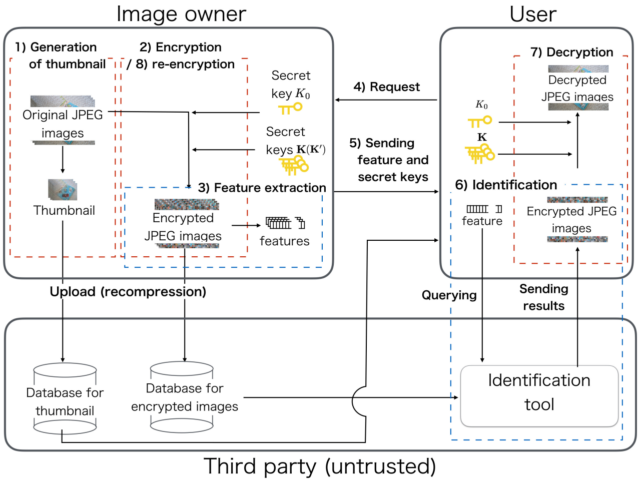


Fig. 4 Privacy-preserving photo-sharing.

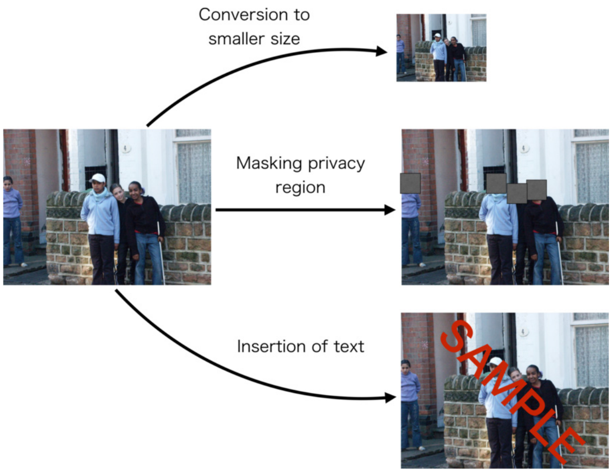


Fig. 5 Examples of thumbnail images.

3.2 Overview of Photo-Sharing Services

So that an image owner share JPEG images on an untrusted service without publishing the sensitive information of the original quality images, a scenario of the proposed scheme is illustrated in Fig. 4.

1) An image owner generates thumbnail JPEG images from original JPEG images, and the thumbnail images are then uploaded to a third party.
2) The image owner encrypts the original JPEG images with secret keys K_0 and $\mathbf{K} = [K_1, K_2, K_3]$ according to the two-layer image encryption. As shown in Fig. 3, the blocks of original JPEG images are permuted with the secret key K_0, and encryption with $\mathbf{K}$ is then performed. After that, the compressed EtC images are uploaded to the third party. In this uploading process, these images may be recompressed.
3) The image owner extracts features from the encrypted JPEG images. The features are related to the uploaded thumbnail images.
4) A user selects a thumbnail image from those stored on the third party's storage, and then sends the selected images to the image owner.
5) The image owner sends the corresponding feature and the secret keys K_0 and $\mathbf{K}$ to the user.
6) The third party identifies the encrypted images corresponding to the feature received from the user, and sends the identified image to the user.
7) The user decrypts the encrypted image with K_0 and $\mathbf{K}$.
8) The image owner re-encrypts the identified image with K_0 and $\mathbf{K}' = [K'_1, K'_2, K'_3]$ where K'_1, K'_2 and K'_3 are different from K_1, K_2 and K_3.

Examples of thumbnail images are illustrated in Fig. 5. To offer part of visual information to users, thumbnail images are generated by the image owner. The third party may apply some manipulations to the thumbnail images.

It is difficult for users and image owners to confirm whether the third party are trusted or not, namely called semi-trusted or semi-honest. Therefore, thumbnail images are not tied to corresponding encrypted images on the services because of the following two reasons. One is to prevent jigsaw solver attacks done by using visual information of a thumbnail image, which the third party uses to decrypt the corresponding encrypted image. The other is to prevent the unauthorized use of user data. The third party makes it possible to collect the data on users, such as their preferences, from a thumbnail image corresponding to an encrypted image. Therefore, identification using encrypted images without any visual sensitive information is required for the privacy-preserving communications. In contrast, it is possible for users to easily confirm whether the result of identification sent by the third party is correct, by decrypting images. Thus, we do not focus on the correctness of identification results.

It is known that service providers usually employ manipulation such as recompression to uploaded images. Therefore, recompression is assumed in steps 1 and 2 of this scenario.

The feature used in the proposed identification scheme is designed to identify images encrypted with the different key sets $\mathbf{K}$ and $\mathbf{K}'$ under the same K_0. Thus, recalculation of features is not needed, even when images are re-encrypted.

In the scenario, four secret keys K_0, $K_{1,i}$, $K_{2,i}$ and $K_{3,i}$ for image O_i are needed. To efficiently manage these keys, a method using a hash function was proposed as an efficient key generation and management method [26]–[28]. Thus, the use of the method allows us to manage only one key as follows (see also Fig. 6). At first, a master secret key K_M is selected. Next, a seed value for each image O_i is generated from K_M by applying a certain operation with the information on each image α_i (e.g. filename and identifier). After that, hash values are calculated with the seed value to obtain $K_{1,i}$, $K_{2,i}$ and $K_{3,i}$. As well, K_0 and keys used for re-encryption are generated by following this method, where α indicates data for the generation of K_0. Therefore, an image owner manages only one key K_M for all images.

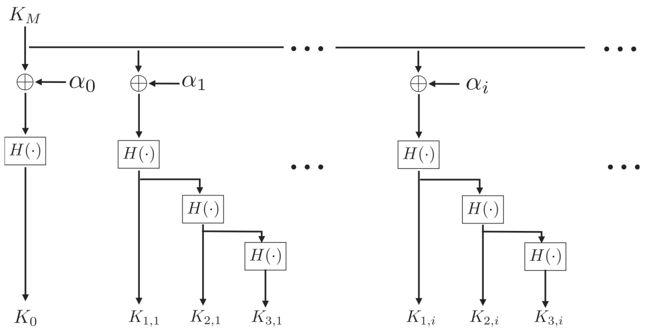


Fig. 6 Efficient key management with a hash function $H(\cdot)$, where α indicates data for the generation of K_0.

3.3 Attack Model

In the scenario, there are three roles: image owner, third party, user. It is not guaranteed that the third party is trusted, namely called semi-trusted or semi-honest. Thus, it is needed that the system satisfies two requirements. The first one is that the third party can not obtain high quality images, and the second is that the third party can not collect user data (such as preferences). To satisfy the second one, the information on the relation between each encrypted image and the corresponding thumbnail one is not given to the third party.

In contrast, to satisfy the first one, it is required that high quality images are not reconstructed from the encrypted images. In the other words, the encryption method is required to have robustness against ciphertext-only attacks. The robustness against ciphertext-only attacks including the brute-force and jigsaw-solver attacks was discussed in the previous researches [6]–[9], [26]. In the two-layer image encryption, most of blocks in each image are encrypted with the different keys from other images, so the two-layer encryption makes the ciphertext-only attacks difficult.

In the scenario, there is only one third party. So, the third party can not conduct collusion attacks. However, if the third party leaks the encrypted images to attackers, they enable to conduct collusion attacks. In order to conduct these attacks, the attackers are required to reconstruct the high quality image from each encrypted image. As a result, the attackers try to do ciphertext-only attacks, against which the proposed scheme has robustness.

In addition to the encrypted images, the third party has thumbnail images and feature vectors, and there are a possibility to leak them to the attackers. Thumbnail images are generated to publish the information for users, so the leakage of the thumbnail is not a problem. In the case that feature vectors are leaked, the attackers enables to obtain the encrypted images to send the feature vectors to the third party. However, it is difficult for the attackers to obtain high quality images because they do not have the keys.

3.4 Proposed Identification Scheme

In the proposed identification scheme, the feature vectors extracted from the first N DC coefficients of the luminance component Y in the encrypted images are used. The use of these features allows us to robustly identify images against recompression and re-encryption. Here, the feature extraction and the identification processes are explained.

3.4.1 Feature Extraction Process

To extract the feature vector of $E_i^{(1,k_0,k)}$, the following process is performed.

(a) Set N.
(b) Set $n := 0$.
(c) Extract the feature vector $v_{E_i^{(1,k_0,k)}}$ from the nth DC coefficient of the Y component in $E_i^{(1,k_0,k)}$ as below.

$$v_{E_i^{(1,k_0,k)}}(n) = |E_i^{(1,k_0,k)}(n)|, \tag{3}$$

(d) Set $n := n + 1$. If $n < N$, return to step (c). Otherwise, the image owner halts the process for $E_i^{(1,k_0,k)}$.

In step (c), the feature vector is extracted from the the absolute values of the DC coefficients of the Y component. It is known that block rotation and inversion in the DCT domain do not change the value of the DC coefficient in each block [29]. In addition, from Eqs. (1) and (2), the absolute value of a DC coefficient is not greatly changed by a negative-positive transform as below.

$$\begin{aligned} |DC_{np}| &= \left| \frac{1}{8} \sum_{x=0}^{7} \sum_{y=0}^{7} ((255 - I(x,y)) - 128) \right| \\ &= \left| \frac{1}{8} \sum_{x=0}^{7} \sum_{y=0}^{7} (128 - I(x,y) - 1) \right| \\ &= |-DC - 8|, \end{aligned} \tag{4}$$

where DC_{np} is a DC coefficient of a negative-positive transformed block. Thus, the use of DC coefficients allows us to avoid not only the effect of the recompression but also that of the encryption. In the proposed scheme, when K_0 and N are not changed, it is expected that the absolute values of DC coefficients in the first N blocks are close to those in the images encrypted with the different key sets $\mathbf{K} \neq \mathbf{K}'$. This allows us to identify re-encrypted images without recalculating features.

3.4.2 Identification Process

As shown in Fig. 4, a user obtains the feature vector corresponding to a thumbnail image $v_{E_i^{(1,k_0,k)}}$ from the image owner, and then sends the vector to the third party. The third party performs the following process to identify $E_{i'}^{(j,k_0,k')}$ with $E_i^{(1,k_0,k)}$, after extracting $v_{E_{i'}^{(j,k_0,k')}}$ from $E_{i'}^{(j,k_0,k')}$.

(a) No.000 (b) No.001 (c) No.113 (d) No.114

Fig. 7 Examples of images in UKbench

(a) Set N and d, where d is a parameter that determines the acceptance error.
(b) Set $n := 0$ and $i' = 0$.
(c) If $|v_{E_i^{(1,k_0,k)}}(n) - v_{E_{i'}^{(j,k_0,k')}}(n)| \leq d$, proceed to step (d). Otherwise, the third party judges that $E_{i'}^{(j,k_0,k')}$ is not generated from O_i and proceed to step (e).
(d) Set $n := n + 1$. If $n < N$, return to step (c). Otherwise, the third party judges that $E_{i'}^{(j,k_0,k')}$ is generated from the same original image as that of $E_i^{(1,k_0,k)}$, i.e., O_i and the process for $E_i^{(1,k_0,k)}$ is halted.
(e) Set $i' := i' + 1$ and $n := 0$. If i' is equal to the number of images stored in the database of the third party, return to step (c). Otherwise, third party judges that there are no image corresponding to $E_i^{(1,k_0,k)}$.

The distance between the absolute values of the features is used for identification under the acceptance error d in step (c). As well as the conventional schemes, this value was experimentally determined in a pre-experiment.

A large value of N is required to maintain high identification accuracy because many blocks can be used for the identification, although the number of blocks permuted in step 2-3 decreases. In other words, there is the trade-off relation between the number of blocks permuted in step 2-3 and the identification accuracy. In an experiment, it will be demonstrated that the proposed scheme has a reasonable performance in terms of the accuracy and the protection of visual information.

4. Simulation

A number of simulations were conducted to evaluate the performance of the proposed identification scheme. We used images (size of 480 × 640) in UKbench dataset [30] and an encoder and a decoder from the IJG (Independent JPEG Group) [31] in the simulations. The dataset consists of 10,200 images (4 images per 2,520 objects), and 500 images from No.000 to No.499 were chosen from the 10,200 images (see Fig. 7).

4.1 Simulation Conditions

Table 2 summarizes the conditions for generating original JPEG images and encrypted ones, where QF_I indicates the quality factor used for JPEG image I, $I \in \{E_i^{(1,k_0,k)}, E_i^{(2,k_0,k)}, E_i^{(2,k_0,k')}\}$. Note that the sampling ratios for all compressions were 4 : 4 : 4. For instance, in the case of condition (1), 500 original JPEG images were generated with $QF_{O_i} = 95$ from the 500 UKbench images first. To generate $E_i^{(1,k_0,k)}$, the 500 original JPEG images were encrypted and compressed with $QF_{E_i^{(1,k_0,k)}} = 95$. Next, these 500 single-compressed encrypted images were recompressed with $QF_{E_i^{(2,k_0,k')}} = 85, 80, 75, 70$. As a result, 500 single-compressed and 2,000 double-compressed images encrypted with k were generated under each condition. As well as the generation of images encrypted with k, there were 500 single-compressed and 2,000 double-compressed JPEG images encrypted with k', $QF_{E_i^{(1,k_0,k')}} = 95$ and $QF_{E_i^{(2,k_0,k')}} = 85, 80, 75, 70$. In the simulations, the identification performances between 500 single-compressed and 2,000 double-compressed images encrypted with k were evaluated for each condition first. Also, identification between 500 single-compressed images with k and 2,000 double-compressed ones with k' was performed.

Table 2 Condition to generate original and encrypted JPEG images, where QF_I indicates the quality factor generated for the JPEG image I, $I \in \{E_i^{(1,k_0,k)}, E_i^{(2,k_0,k)}, E_i^{(2,k_0,k')}\}$.

Condition	$QF_{O_i} =$	$QF_{E_i^{(1,k_0,k)}} =$	$QF_{E_i^{(1,k_0,k')}} =$ $QF_{E_i^{(1,k_0,k')}} =$	$QF_{E_i^{(2,k_0,k)}} =$ $QF_{E_i^{(2,k_0,k')}} =$
(1)	95	95	95	71, 75, 80, 85
(2)	85	85	85	
(3)	75	75	75	

As the parameter, $N = 480$ was selected. UKbench images were divided into 4800 8 × 8 blocks, i.e. $M = 4800$, so that $N = 480$ was selected as 10% of all blocks.

The proposed scheme was compared with four identification schemes (DC signs-based one [22], sparse coding-based [23], quaternion-based [24] and iterative quantization (ITQ)-based ones [25]). In the schemes [23]–[25], the hamming distances between the hash values of the encrypted images were calculated, and images that had the smallest distance were then chosen as the images generated from an original image, after all images were decompressed.

4.2 Parameter Selection

To determine a value of d, a pre-experiment using 885 images in Uncompressed Color Image Database (UCID) [32] was conducted as below.

a) Encrypt an image with k and k_0, and then compress the encrypted image with $QF_{E_i^{(1,k_0,k)}} = 85$.
b) Encrypt the image selected in step a) by using k' and k_0. After that, double-compress the encrypted image with $QF_{E_i^{(1,k_0,k')}} = 85$ and $QF_{E_i^{(2,k_0,k')}} = 85$.
c) Compare the DC coefficients of the first N blocks in the image generated in step a) with those of one generated in step b). The difference of the absolute values of the coefficients at each block position is calculated. After that, the maximum absolute value of the differences in N positions is chosen, and then the value is stored.
d) Repeat steps from a) to c) for 885 images. When all

Table 3 Identification performance for double-compressed encrypted images ($N = 480$, $d = 150$).

Scheme	Condition	$k = k'$		$k \neq k'$	
		$p[\%]$	$r[\%]$	$p[\%]$	$r[\%]$
Proposed ($N = 480$, $d = 150$)	(1)	100	100	100	100
	(2)	100	100	100	100
	(3)	100	100	100	100
DC sign [22]	(1)	100	100	0	0
	(2)	100	100	0	0
	(3)	100	100	0	0
ITQ [25]	(1)	100	100	3.45	3.45
	(2)	100	100	3.25	3.25
	(3)	100	100	3.6	3.6
Sparse coding [23]	(1)	99.95	100	0.09	0.15
	(2)	100	100	0.33	0.55
	(3)	100	100	0.06	0.1
Quaternion [24]	(1)	100	100	0.31	0.5
	(2)	100	100	0.24	0.4
	(3)	100	100	0.56	0.95

885 images are selected, the largest value in 885 images is determined as d.

From the result, d was determined as 150.

4.3 Identification Performance

At first, the identification performance of the proposed scheme with $d = 150$ was compared with those of other schemes under $N = 480$. After that, the identification performance was evaluated under the use of various N and d values.

4.3.1 Comparison with Other Schemes

Table 3 shows the precision value p and recall value r, defined by

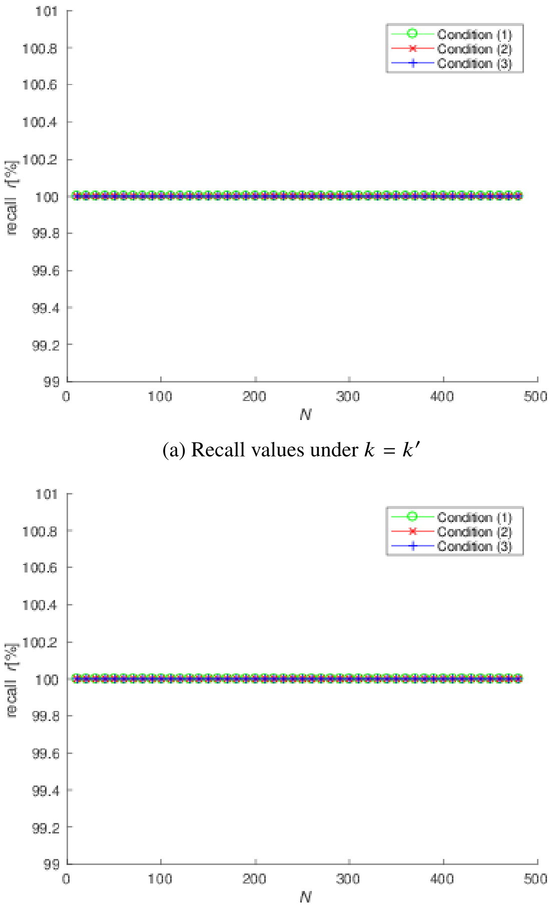
$$p = \frac{TP}{TP + FP}, \quad r = \frac{TP}{TP + FN}, \tag{5}$$

where TP, FP and FN represent the number of true positive, false positive and false negative matches respectively. Note that $r = 100[\%]$ means that there were no false negative matches, and $p = 100[\%]$ means that there were no false positive matches.

It was confirmed that all schemes had high identification performances under $k = k'$. However, under all three conditions with $k \neq k'$, all schemes except for the proposed scheme did not achieve $r = 100\%$ and $p = 100\%$.

4.3.2 Identification Performance with $N \neq 480$

Next, identification performances under $10 \leq N \leq 480$ and $d = 150$ are discussed. Figures 8 and 9 show the results. There was no degradation in the recall values, although the precision values were not 100[%] when $N < 160$. The selection of smaller N means that the length of the feature vector is shorten, so the number of true positive matches does not increase in principle, while false positive matches may do. Thus, only the precision values were degraded according to the decrease of N.



(a) Recall values under $k = k'$

(b) Recall values under $k \neq k'$

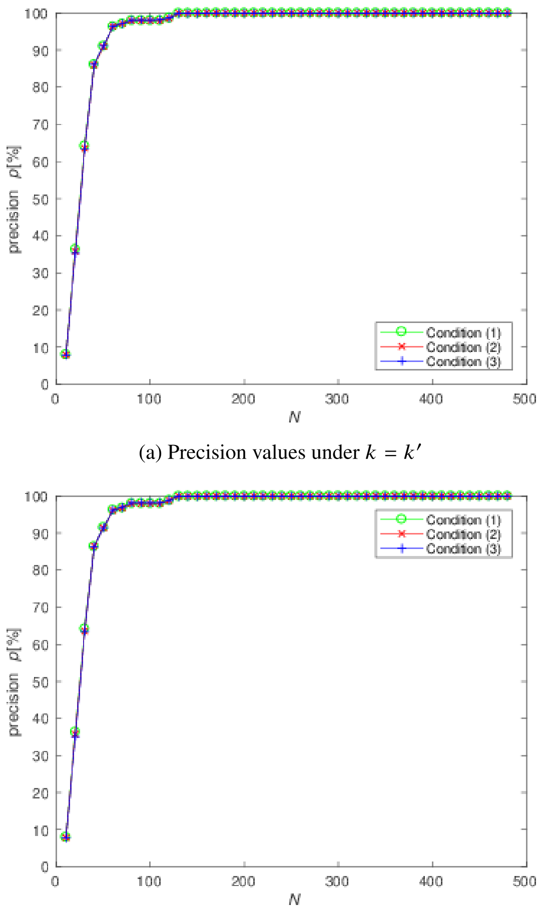
Fig. 8 Recall values under $10 \leq N \leq 480$ and $d = 150$.

The number of combinations of the blocks permuted in step 2-3 was 4640! when $N = 160$. This value is much larger than 2^{256}, which is the key space of 256 bit keys, so the proposed scheme enables to achieve a reasonable performance in terms of the number of blocks permuted in step 2-3 and the precision of the identification. Moreover, higher resolution images than ones used in the simulation are used in practice, so the number of blocks will further increase.

The performance under $k = k'$ is similar to the performance under $k \neq k'$, as shown in the figures. Therefore, the proposed scheme enables us to avoid the both effects of recompression and re-encryption by selecting proper parameter values.

4.3.3 Identification Performance with $d \neq 150$

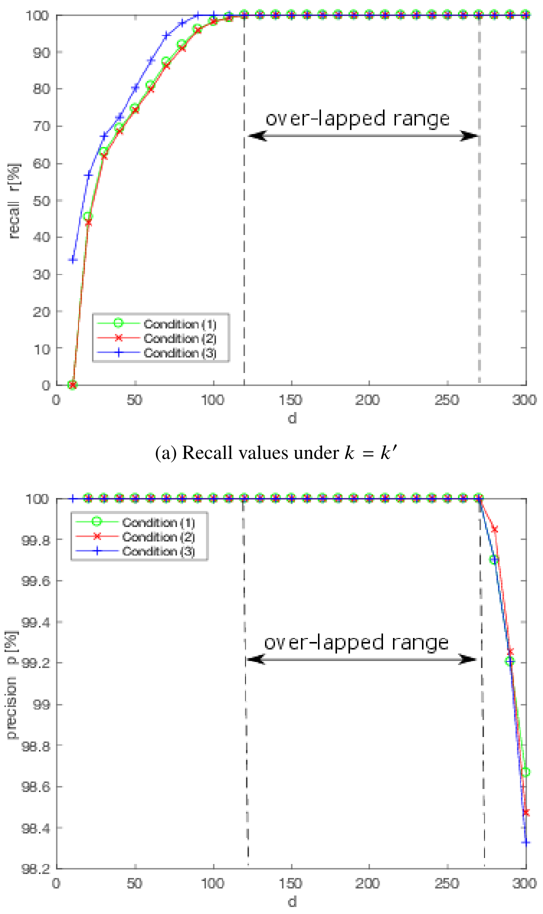
Figure 10 shows the performances under $10 \leq d \leq 300$ and $N = 480$. It was confirmed that the identification performances were perfect in the range of $120 \leq d \leq 270$ under all conditions. When $d < 120$, the effect of the errors caused by recompression and encryption can not be avoided, so that the recall values decreased. As shown in the results, the parameter range that allows us to achieve perfect identification



(a) Precision values under $k = k'$

(b) Precision values under $k \neq k'$

Fig. 9 Precision values under $10 \leq N \leq 480$ and $d = 150$.



(a) Recall values under $k = k'$

(b) Precision values under $k = k'$

Fig. 10 Recall and precision values under $10 \leq d \leq 300$ and $N = 480$.

is wide, so there are many d values to offer a good identification performance.

5. Conclusion

In this paper, an novel identification scheme for encrypted images was proposed. The image encryption is based on a block-scrambling method, and two-layer block permutation is performed in the encrypted process. For the identification, the feature vector used in the proposed scheme is extracted from the DC coefficients of luminance. The use of the image encryption method and feature vectors allows us to avoid not only the effect of recompression but also that of re-encryption with different keys. Simulation results showed the effectiveness of the proposed scheme, even when images were recompressed and re-encrypted by different keys for the second layer. In this paper, robustness against ciphertext-only attacks by a third party was mainly discussed. In the future work, assuming that other users and image owners may become attackers, robustness against more various attacks such as known-plaintext attacks will be considered.

Acknowledgments

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

References

[1] R. Caldelli, R. Becarelli, and I. Amerini, "Image origin classification based on social network provenance," IEEE Trans. Inf. Forensics Security, vol.12, no.6, pp.1299–1308, 2017.

[2] T. Chuman, K. Iida, W. Sirichotedumrong, and H. Kiya, "Image manipulation specifications on social networking services for encryption-then-compression systems," IEICE Trans. Inf. & Syst., vol.E102-D, no.1, pp.11–18, Jan. 2019.

[3] T. Chuman, K. Iida, and H. Kiya, "Image manipulation on social media for encryption-then-compression systems," Proc. APSIPA Annual Summit and Conference, pp.858–863, 2017.

[4] C.-T. Huang, L. Huang, Z. Qin, H. Yuan, L. Zhou, V. Varadharajan, and C.-C.J. Kuo, "Survey on securing data storage in the cloud," APSIPA Trans. Signal and Information Processing, vol.3, 2014.

[5] R.L. Lagendijk, Z. Erkin, and M. Barni, "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation," IEEE Signal Process. Mag., vol.30, no.1, pp.82–105, 2013.

[6] J. Zhou, X. Liu, O.C. Au, and Y.Y. Tang, “Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation.,” IEEE Trans. Inf. Forensics Security, vol.9, no.1, pp.39–50, 2014.
[7] 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, June 2019.
[8] W. Sirichotedumrong and H. Kiya, “Grayscale-based block scrambling image encryption using YCbCr color space for encryption-then-compression systems,” APSIPA Trans. Signal and Information Processing, e7, vol.8, 2019.
[9] T. Chuman and H. Kiya, “Security evaluation for block scrambling-based image encryption including JPEG distortion against jigsaw puzzle solver attacks,” IEICE Trans. Fundamentals, vol.E101-A, no.12, pp.2405–2408, Dec. 2018.
[10] P. Li and K.-T. Lo, “Joint image encryption and compression schemes based on 16 × 16 DCT,” Journal of Visual Communication and Image Representation, vol.58, pp.12–24, 2019.
[11] V. Itier, P. Puteaux, and W. Puech, “Recompression of JPEG crypto-compressed images without a key,” IEEE Trans. Circuits Syst. Video Technol., https://www.doi.org/10.1109/TCSVT.2019.2894520, 2019.
[12] M. Ra, R. Govindan, and A. Ortega, “P3: Toward privacy-preserving photo sharing,” Proc. USENIX Symposium on Networked Systems Design and Implementation, pp.515–528, 2013.
[13] K. Munadi, F. Arnia, M. Syaryadhi, and H. Kiya, “A content-based image retrieval system for visually protected image databases,” Proc. Asia Pacific Conference on Multimedia and Broadcasting, pp.1–6, 2015.
[14] H. Cheng, X. Zhang, and J. Yu, “AC-coefficient histogram-based retrieval for encrypted JPEG images,” Multimedia Tools and Applications, vol.75, no.21, pp.13791–13803, 2016.
[15] Y. Xu, J. Gong, L. Xiong, Z. Xu, J. Wang, and Y. Shi, “A privacy-preserving content-based image retrieval method in cloud environment,” Journal of Visual Communication and Image Representation, vol.43, pp.164–172, 2017.
[16] H. Cheng, X. Zhang, J. Yu, and F. Li, “Markov process-based retrieval for encrypted JPEG images,” EURASIP Journal on Information Security, vol.2016, Aritcle Number 1, 2016.
[17] A.A.A.M. Kamal, K. Iwamura, and H. Kang, “Searchable encryption of image based on secret sharing scheme,” Proc. APSIPA Annual Summit and Conference, pp.1495–1503, 2017.
[18] Z. Xia, N.N. Xiong, A.V. Vasilakos, and X. Sun, “EPCBIR: An efficient and privacy-preserving content-based image retrieval scheme in cloud computing,” Information Sciences, vol.387, pp.195–204, 2017.
[19] J. Qin, H. Li, X. Xiang, Y. Tan, W. Pan, W. Ma, and N.N. Xiong, “An encrypted image retrieval method based on Harris corner optimization and LSH in cloud computing,” IEEE Access, vol.7, pp.24626–24633, 2019.
[20] Y. Su, Y. Wo, and G. Han, “Reversible cellular automata image encryption for similarity search,” Signal Processing: Image Communication, vol.72, pp.134–147, 2019.
[21] K. Iida and H. Kiya, “Robust image identification with DC coefficients for double-compressed JPEG images,” IEICE Trans. Inf. & Syst., vol.E102-D, no.1, pp.2–10, Jan. 2019.
[22] K. Iida and H. Kiya, “Robust image identification for double-compressed and resized JPEG images,” Proc. APSIPA Annual Summit and Conference, pp.1968–1974, 2018.
[23] Y. Li and P. Wang, “Robust image hashing based on low-rank and sparse decomposition,” Proc. IEEE Int'l Conf. on Acoustics, Speech and Signal Processing, pp.2154–2158, 2016.
[24] Y.N. Li, P. Wang, and Y.T. Su, “Robust image hashing based on selective quaternion invariance,” IEEE Signal Process. Lett., vol.22, no.12, pp.2396–2400, 2015.
[25] Y. Gong, S. Lazebnik, A. Gordo, and F. Perronnin, “Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval,” IEEE Trans. Pattern Anal. Mach. Intell., vol.35, no.12, pp.2916–2929, 2013.
[26] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, “An encryption-then-compression system for JPEG/motion JPEG standard,” IEICE Trans. Fundamentals, vol.E98-A, no.11, pp.2238–2245, Nov. 2015.
[27] S. Imaizumi, M. Fujiyoshi, Y. Abe, and H. Kiya, “Collusion attack-resilient hierarchical encryption of JPEG 2000 codestreams with scalable access control,” Proc. IEEE Int'l Conf. on Image Processing, pp.137–140, 2007.
[28] S. Imaizumi, M. Fujiyoshi, and H. Kiya, “Efficient collusion attack-free access control for multidimensionally hierarchical scalability content,” Proc. IEEE Int'l Symposium on Circuits and Systems, pp.505–508, 2009.
[29] R.L. de Queiroz, “Processing JPEG-compressed images and documents,” IEEE Trans. Image Process., vol.7, no.12, pp.1661–1672, 1998.
[30] “Ukbench dataset,” https://archive.org/details/ukbench
[31] “The independent JPEG group software JPEG codec,” http://www.ijg.org/
[32] G. Schaefer and M. Stich, “UCID: An uncompressed color image database,” Proc. SPIE 5307, Storage and Retrieval Methods and Applications for Multimedia 2004, pp.472–480, 2003.

Kenta Iida received his B.Eng. and M.Eng. degrees from Tokyo Metropolitan University, Japan in 2016 and 2018 respectively. He is a Doctor course student at Tokyo Metropolitan University, Japan. His research interests include image processing, biometrics, and multimedia security. He is a member of IEICE and a student member of IEEE.

Hitoshi Kiya received his B.E and M.E. degrees from Nagaoka University of Technology, in 1980 and 1982 respectively, and his 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 currently serves as President-Elect of APSIPA, and he served as Inaugural Vice President (Technical Activities) of APSIPA from 2009 to 2013, and as Regional Director-at-Large for Region 10 of the IEEE Signal Processing Society from 2016 to 2017. He was also President of the IEICE Engineering Sciences Society from 2011 to 2012, and he served there as a Vice President and Editor-in-Chief for IEICE Society Magazine and Society Publications. He was Editorial Board Member of eight journals, including IEEE Trans. on Signal Processing, Image Processing, and Information Forensics and Security, Chair of two technical committees and Member of nine technical committees including APSIPA Image, Video, and Multimedia Technical Committee (TC), and IEEE Information Forensics and Security TC. He has organized a lot of international conferences, in such roles as TPC Chair of IEEE ICASSP 2012 and as General Co-Chair of IEEE ISCAS 2019. He has received numerous awards, including nine Best Paper Awards.