A PRIVACY-PRESERVING CONTENT-BASED IMAGE RETRIEVAL SCHEME ALLOWING MIXED USE OF ENCRYPTED AND PLAIN IMAGES

Kenta Iida* and Hitoshi Kiya*
* Tokyo Metropolitan University
6-6, Asahigaoka, Hino-shi, Tokyo, Japan
E-mail: iida-kenta2@ed.tmu.ac.jp, kiya@tmu.ac.jp

***Abstract*—In this paper, we propose a novel content based-image retrieval scheme allowing the mixed use of encrypted and plain images for the first time. In the proposed scheme, images are encrypted by a block-scrambling method developed for encryption-then-compression (EtC) systems. The encrypted images, referred to as EtC images, can be compressed with JPEG, as well as for plain images. Image descriptors used for the proposed retrieval is designed to avoid the effect of image encryption. As a result, the use of EtC images and the descriptors allows us to carry out retrieval of both encrypted images and plain ones. In an experiment, the proposed scheme is demonstrated to have the same performance as conventional retrieval methods with plain images, even under the mixed use of plain images and EtC ones.**

I. Introduction

With the rapid growth of cloud computing, outsourcing images to cloud storage services and sharing photos have greatly increased. Generally, images are uploaded and stored in a compressed form to reduce the amount of data. In addition, most im†ages include sensitive information, such as personal data and copyright information [1], [2]. However, cloud providers are not trusted in general, so there is the possibility of data leakage and unauthorized use in cloud environments. Therefore, various privacy-preserving image identification, retrieval, and processing schemes have been studied for untrusted cloud environments [3]–[21].

For the above reasons, privacy-preserving image-retrieval methods should satisfy generally three requirements: 1) protecting visual information on plain images, 2) having a high retrieval performance for encrypted images, and 3) being applicable to compressible encrypted images. In regards to requirement 1), it is classified into two requirements: 1-a) protecting images stored in databases of cloud providers, and 1-b) protecting query images uploaded by users. Requirement 1-a) should be always satisfied for preserving the privacy of stored images and unauthorized use. In contrast, requirement 1-b) is needed under the assumption that cloud providers use query images on unauthorized purposes, such as collection of user's preferences. Thus, when users do not care about the unauthorized use of query images, the satisfaction of requirement 1-b) is not needed.

In this paper, to satisfy above all requirements, we propose a novel content-based image retrieval scheme (CBIR) for the first time. In the proposed scheme, images are encrypted by a block-scrambling method developed for encryption-then-compression (EtC) systems [22]–[26]. The encrypted images, referred to as EtC images, can be compressed with JPEG. In addition, extended SIMPLE descriptors are combined with scalable color descriptor (SCD) to avoid the influence of image encryption. Simulation results show that retrieval performances are the same as that of using plain images, even when the mixed use of plain images and EtC ones.

II. Related work

A. Image retrieval in encrypted domain

Image retrieval methods in the encrypted domain are classified into two classes as shown below.

1) Generating descriptors from plain images

In this class, descriptors are calculated from plain images. For instance, SIFT-based [4], SURF-based [5], ORB [6], MPEG-7 [7], [8], and CNN-based [9]–[11] descriptors are used in conventional schemes corresponding to this class. To protect the content of plain images, not only plain images but also the descriptors are encrypted by a data owner. After that, the encrypted descriptors and images are sent to a cloud server.

In this approach, data owners are required to extract descriptors and encrypt both the descriptors and plain images by themselves. Moreover, data owners and users have to share a common key in some schemes [9], [11].

2) Generating descriptors from encrypted images

In the second approach, descriptors are directly extracted from encrypted images by a cloud provider as well as for plain images [12]–[17], where each data owner only encrypts images. In addition, data owners and users do not need to share keys. The proposed retrieval scheme corresponds to this approach.

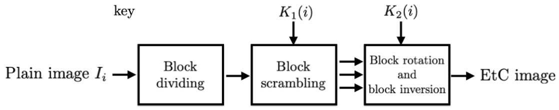


Fig. 1: Generation of EtC images

(a) Plain image (b) EtC image

Fig. 2: Example of plain image and encrypted one

Schemes [12]–[17] are carried out the basis on this approach. Some schemes [15], [16] consider compressing images with JPEG, but their retrieval performance will be degraded due to the difference in coding parameters used for JPEG compression. On the other hand, a scheme using extended SIMPLE descriptors extracted from EtC images does not have the limitation [17]. However, the mixed use of plain images and encrypted ones have never considered yet in this scheme.

B. EtC image

We focus on EtC images, which have been proposed for encryption-then-compression (EtC) systems with JPEG compression [22]–[27]. EtC images not only have almost the same compression performance as that of plain images but also enough robustness against various ciphertext-only attacks including jigsaw puzzle solver attacks [23], [24], [26]. The procedure for generating EtC images is conducted as below (see Figs. 1 and 2) [22].

(a) Divide image I_i with $X \times Y$ pixels into non-over-lapping 16×16 blocks.
(b) Permute randomly $\lfloor \frac{X}{16} \rfloor \times \lfloor \frac{Y}{16} \rfloor$ divided blocks by using a random integer generated by secret key $K_1(i)$.
(c) Rotate and invert randomly each divided block by using a random integer generated by secret key $K_2(i)$.

In this paper, images encrypted by using these steps are referred to as "EtC images". For the retrieval of EtC images, the proposed scheme is designed to avoid the effects of two encryption operations in steps (b) and(c): block scrambling, and block rotation and block inversion.

$K_1(i)$ and $K_2(i)$ are stored as a key set $\mathbf{K}_i = [K_1(i), K_2(i)]$, which is used for the encryption of I_i.

C. Weighted SIMPLE descriptors

It is well-known that weighted SIMPLE descriptors outperform SIMPLE descriptors [28]. Thus, the proposed scheme is designed on the basis of weighted SIMPLE descriptors. The extraction weighted SIMPLE descriptors from plain images is carried out by the following steps.

a) Extract a patch descriptor from each patch of images after selecting the positions and sizes of patches from every image.
b) Generate a codebook with a size of M from the extracted patch descriptors.
c) Calculate a histogram as a SIMPLE descriptor of each image by using the codebook and patch descriptors extracted from the image.
d) Obtain weighted SIMPLE descriptors by weighting the SIMPLE descriptors.

As mentioned above, this procedure consists of four steps: patch descriptors generation, codebook generation, SIMPLE descriptors generation, and weighting SIMPLE descriptors. Here, the operation of each step is summarized.

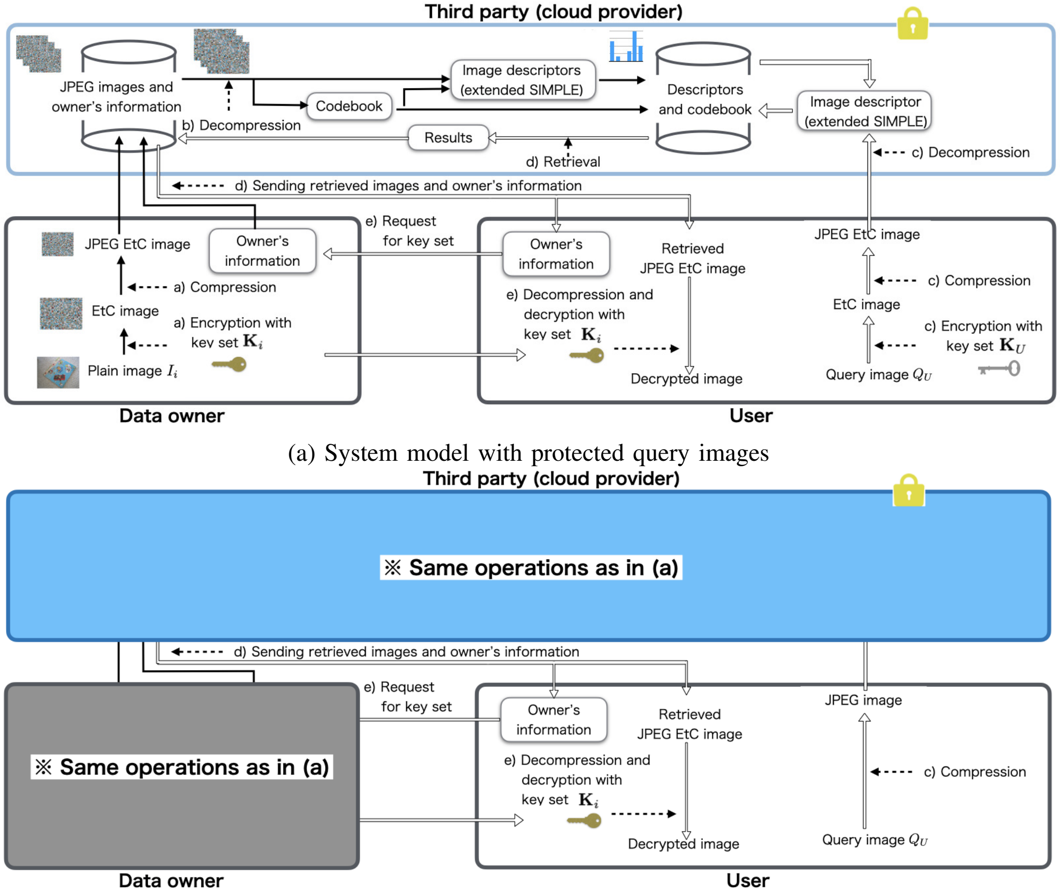
- Patch descriptors are extracted to generate the SIMPLE descriptor of each image and the codebook in step a). At first, the positions and sizes of patches are determined by using a detector such as the SURF detector or random sampling. In the case of using the SURF detector, those of each patch are determined from detected feature points and scales, respectively. In contrast, they are randomly selected, if random sampling is used. After that, the type of patch descriptors extracted from selected patches is selected from MPEG-7 descriptors or MPEG-7-like descriptors [28], which are represented as vectors.
- The codebook, which is used for calculating SIMPLE descriptors, is generated from patch descriptors by using k-means clustering in step b). The codebook consists of visual words, where a visual word is a vector that represents the center of each class. Note that the size of the codebook M corresponds to the number of classes in k-means clustering.
- A SIMPLE descriptor of an image is calculated from patch descriptors of the image and the codebook in step c). After classifying each patch descriptor in the image into a visual word in the codebook, and the number of patch descriptors classified into the mth visual word corresponds to the mth bin in the histogram of the image, where $0 \leq m < M$.
- Weighting SIMPLE descriptors is achieved by the modification of each component in SIMPLE descriptors and the normalization of the modified components. When N SIMPLE descriptors are generated, the mth component of the nth weighted SIMPLE descriptor $v_n(m)$, $0 \leq m < M$, $0 \leq n < N$, is calculated as below in this paper.

$$v_n(m) = (1 + log(tf_{(m,n)})) \times log\frac{N}{df_{(m)}}, \quad (1)$$

where $tf(m,n)$ represents the frequency of the mth visual word in the nth descriptor, and $df(m)$ denotes the number of SIMPLE descriptors containing the mth visual word in the N SIMPLE descriptors. After that, l_2 normalization is applied to every SIMPLE descriptor.

III. Proposed Scheme

A novel content-based image retrieval scheme using EtC images is proposed here. For carrying out the image retrieval in the encrypted domain, weighted SIMPLE descriptors is modified, and SCD is used as a patch descriptor of each patch.



(b) System model with plain query images

Fig. 3: System models, where image descriptors correspond to extended SIMPLE descriptors

A. System model

Two frameworks used in the proposed scheme are shown in Figs. 3 (a) and (b).

1) System model with protected query images

Each operation in the system model with protected query images, which is shown in Fig. 3 (a), is explained as follows.

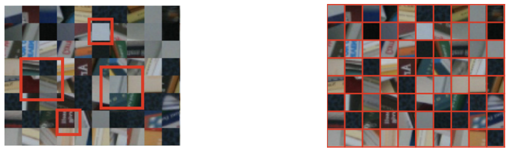
a) A data owner generates an EtC image from plain image I_i by using secret key set $\mathbf{K}_i$, and then the EtC image is compressed by the JPEG standard. After that, the JPEG EtC image is uploaded with the owner's information to a third party.
b) The third party generates a codebook from the uploaded EtC images after decompression, and image descriptors are then calculated by using the codebook. Finally, the codebook and the image descriptors are stored in a database.
c) A user encrypts query image Q_U with key set $\mathbf{K}_U$, where $\mathbf{K}_U$ can be prepared by the user. After compressing the EtC image, the user sends the JPEG EtC image as query one to the third party.
d) The third party calculates the image descriptor of query image, and retrieves EtC images similar to the query image in the encrypted domain. The retrieved images and the owner's information are returned to the user.
e) The user requests key set $\mathbf{K}_i$ from the data owner by basing on the received owner's information. By using key set $\mathbf{K}_i$ received from the data owner, the user decrypts the EtC images received from the third party.

The third party not only has no visual information on images but also no secret keys in this model. Moreover, each image can be encrypted by using different keys. In the other words, EtC images are able to be retrieved in this system, even when $\mathbf{K}_i \neq \mathbf{K}_U$.

2) System model with plain query images

Figure 3 (b) shows a system model with plain query images. Except for operation c), operations in this model are the same as those in the model with protected images. Operation c) in this model is carried out as below.

c) A user sends query image Q_U without image encryption to the third party, after applying the JPEG compression into Q_U.



(a) Random sampling (b) Extended SIMPLE

Fig. 4: Examples of selected patches

The proposed scheme is designed by considering both the system models.

B. Proposed image retrieval scheme

To maintain the retrieval performance that using plain images can achieve, an extension of weighted SIMPLE descriptors, referred to as extended SIMPLE descriptors, is proposed to avoid the influence of block scrambling.

1) Extended SIMPLE descriptors

To generate extended SIMPLE descriptors, step a) in Sec. II-C is replaced with step a') as below.

- a') Divide each image into non-overlapping 16×16-blocks and use each 16×16-block as a patch, where 16×16 corresponds to the block size of EtC images (see Fig.4 (b)). After that, SCD is extracted as a patch descriptor from every patch.

The use of non-overlapping 16×16-blocks allows us to avoid the influence of block scrambling. Assuming that patches are selected by random sampling and SURF detector, boundaries generated by the block scrambling operation will be included as shown in Fig.5 (a). In contrast, every patch selected by conducting step a') does not include these boundaries. In addition, since extended SIMPLE descriptors are calculated by using a histogram of visual words contained in an image, block permutation in the block scrambling operation does not give any influence to the descriptors.

In addition, to be robust against block rotation and block inversion, SCD is used as a patch descriptor extracted from each 16×16 blocks in step b). SCD is represented as a vector of coefficients obtained by applying a Haar transform to a color histogram of a patch in the HSV color space. Thus, SCD has no influence of using these operations in principle because block rotation and block inversion operations do not affect the color histogram in a patch. In this paper, 256 coefficients are extracted from each 16×16 blocks.

2) Retrieval process for query image

When the third party receives a query image from an user, the retrieval process for a query image is performed as below.

- a) Select the 16×16-blocks in the query EtC image as patches after dividing the EtC image into non-overlapping 16×16-blocks, and then SCD is extracted from each patch.
- b) Calculate an extended SIMPLE descriptor of the query image by following steps c) and d) in Sec. II-C.
- c) Compute l_2 distance between every descriptor stored in the database and the query descriptor, and then decide similar images.
- d) Send the retrieved images with owner's information to the user.

Note that these steps are the same as ones for plain images, except for obtaining patches from EtC images.

IV. Experiment

A. Experiment setup

To evaluate the performance of the proposed image retrieval, Lucene Image Retrieval (LIRE) [29], which is an open-source Java library for CBIR and supports various image descriptors, was used in this experiment. For the comparison with the proposed scheme, five image descriptors were used: SCD [30], color and edge directivity descriptor (CEDD) [31], SURF [32], weighted SIMPLE descriptor with random sampling and weighted SIMPLE descriptor with SURF detector. In the case of using SURF descriptor for retrieval, the bag-of-visual words model and weighting term frequencies were used.

The performance was evaluated in terms of mean average precision (mAP). To obtain mAP scores, the average precision value of each query image was calculated. When N images are stored in the database, the average precision of the qth query image AP_q is calculated as,

$$AP_q = \frac{1}{G}\sum_{n=1}^{N}\frac{TP@n}{n}\times f(n), \quad (2)$$

where G is the number of ground truth images, and $TP@n$ represents the number of the true positive matches at the rank n and $f(n)$ is defined as below.

$$f(n) = \begin{cases} 1, & \text{if the } n\text{th image is a ground truth image,} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

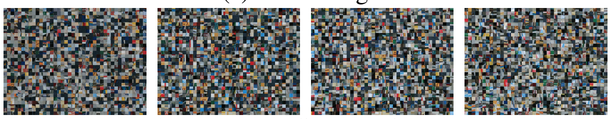
After calculating average precision values for all Q query images, mAP score was calculated as

$$mAP = \frac{\sum_{q=0}^{Q-1} AP_q}{Q}. \quad (4)$$

In this experiment, we used two image dataset: UKbench dataset [33] and INRIA Holidays dataset [34] (see Figs. 5 and 6). UKbench dataset consists of 10,200 images with a size of 640 × 480, and the images are classified into 2,250 groups. Each group has four images containing a single object captured from different viewpoints and lighting conditions. 1,000 images from No. 00000 to No. 00999 were chosen from the data set (see Fig. 5) in this experiment. $N = 1,000$ images were uploaded to a third party by a data owner, where the number of groups was 250, and each group consisted of 4 images, i.e., $G = 4$. As query images, $Q = 250$ images were selected from the first image of each group.

In contrast, INRIA Holidays dataset contains 1,491 images, which are classified into 500 groups, and each group has at least 2 images ($2 \le G \le 13$) (see Fig. 6). In the case of using

(a) Plain images



(b) Corresponding EtC images

Fig. 5: Image examples in group (UKbench dataset)

Fig. 6: Image examples in group (INRIA Holidays dataset)

TABLE I: Relation between types of stored and query images

Notation	Type of stored images	Type of query images
"plain images vs plain images"	plain	plain
"EtC images vs plain images"	plain	EtC
"EtC images vs EtC images"	EtC	EtC

this dataset, $N = 1,491$ images were stored in the third party, and $Q = 500$ images were used as query ones.

B. Experiment results

1) Performance for UKbench dataset

Figure 7 shows the retrieval performances for images in UKbench dataset, where the abbreviated names shown in Tab. I were used such as "EtC images vs plain images". For instance, "EtC images vs plain images" indicates that EtC images were stored in the database and plain images were used as query ones.

Regarding retrieval using plain images as query images, the performance was almost the same as that under "EtC images vs EtC images". In addition, the retrieval performances under "EtC images vs EtC images" had the same mAP scores as ones under "EtC images vs plain images", even when different secret keys were used. Moreover, Table II shows that the performances of the proposed scheme were compared with those of conventional CBIR schemes with plain images. It was confirmed that the proposed scheme had a higher retrieval performance than the typical conventional CBIR schemes with plain images.

2) Performance for INRIA Holidays dataset

The retrieval performances for the INRIA Holidays dataset were also evaluated, as shown in Fig. 8. As well as for the UKbench dataset, the performances under "EtC images vs EtC images" had the same trend as ones under "EtC images vs plain images". It was also confirmed from Table III that the proposed scheme had a higher retrieval performance than the conventional CBIR schemes with plain images as well.

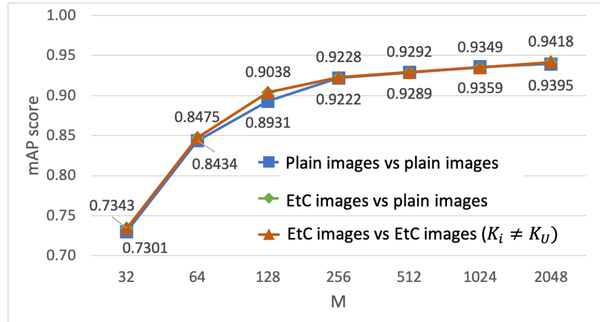


Fig. 7: Retrieval performance for UKbench dataset.

TABLE II: Comparison with conventional CBIR methods using plain images (UKbench dataset)

Descriptor		$M =$	mAP score
SCD [30] (plain)		-	0.9179
CEDD [31] (plain)		-	0.8806
SURF [32] (plain)		256	0.8304
		512	0.8355
Weighted SIMPLE with random sampling (plain)		256	0.9110
		512	0.9262
Weighted SIMPLE with SURF detector (plain)		256	0.8949
		512	0.9109
Proposed	Extended SIMPLE (encrypted, $\mathbf{K}_i \neq \mathbf{K}_U$)	256	0.9228
		512	0.9292

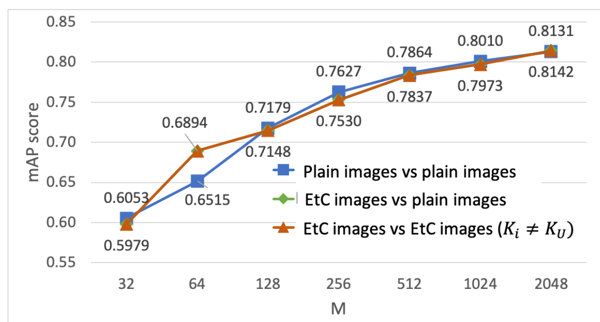


Fig. 8: Retrieval performance for INRIA Holidays dataset.

Therefore, the proposed scheme is effective for privacy-preserving CBIR, and it allows the mixed use of encrypted and plain images as query images. We also confirmed that the performance for compressing EtC images with JPEG had the same trend as one for compressing plain images with JPEG.

V. Conclusion

A novel content-based image retrieval scheme allowing the mixed use of plain images and encrypted images was proposed. To achieve a high retrieval performance, the proposed scheme was designed on the basis of weighted SIMPLE descriptors. In addition, to avoid the influence of encrypted compressible images, called EtC images, weighted SIMPLE descriptors was also extended as extended SIMPLE descriptors under the use of SCD. Experiment results showed that the proposed scheme enables us not only to avoid the influence

TABLE III: Comparison with conventional CBIR methods (INRIA Holidays dataset)

Descriptor	$M =$	mAP score
SCD [30] (plain)	-	0.7524
CEDD [31] (plain)	-	0.7247
SURF [32] (plain)	256	0.6858
	512	0.6986
Weighted SIMPLE with random sampling (plain)	256	0.7716
	512	0.7918
Weighted SIMPLE with SURF detector (plain)	256	0.7156
	512	0.7338
Proposed: Extended SIMPLE (encrypted, $\mathbf{K}_i \neq \mathbf{K}_U$)	256	0.7530
	512	0.7837

of the image encryption, but also to outperform conventional CBIR schemes with plain images.

REFERENCES

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

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

[3] V. Itier, P. Puteaux, and W. Puech, "Recompression of jpeg crypto-compressed images without a key," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2019.

[4] Y. Xu, X. Zhao, and J. Gong, "A large-scale secure image retrieval method in cloud environment," *IEEE Access*, vol. 7, pp. 160082–160090, 2019.

[5] 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.

[6] Zongye Zhang, Fucai Zhou, Shiyue Qin, Qiang Jia, and Zifeng Xu, "Privacy-preserving image retrieval and sharing in social multimedia applications," *IEEE Access*, vol. 8, pp. 66828–66838, 2020.

[7] 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," *Elsevier Information Sciences*, vol. 387, pp. 195–204, 2017.

[8] Z. Xia, X. Wang, L. Zhang, Z. Qin, X. Sun, and K. Ren, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE transactions on information forensics and security*, vol. 11, no. 11, pp. 2594–2608, 2016.

[9] Z. Huang, M. Zhang, and Y. Zhang, "Toward efficient encrypted image retrieval in cloud environment," *IEEE Access*, vol. 7, pp. 174541–174550, 2019.

[10] Chengyuan Zhang, Lei Zhu, Shichao Zhang, and Weiren Yu, "Tdhppir: An efficient deep hashing based privacy-preserving image retrieval method," *Neurocomputing*, 2020.

[11] X. Li, Q. Xue, and M. C. Chuah, "Casheirs: Cloud assisted scalable hierarchical encrypted based image retrieval system," *in Proc. IEEE Conference on Computer Communications*, pp. 1–9, 2017.

[12] H. Wang, Z. Xia, J. Fei, and F. Xiao, "An aes-based secure image retrieval scheme using random mapping and bow in cloud computing," *IEEE Access*, 2020.

[13] Z. Xia, L. Jiang, D. Liu, L. Lu, and B. Jeon, "Boew: A content-based image retrieval scheme using bag-of-encrypted-words in cloud computing," *IEEE Trans. on Services Computing*, pp. 1–1, 2019.

[14] B. Ferreira, J. Rodrigues, J. Leitao, and H. Domingos, "Practical privacy-preserving content-based retrieval in cloud image repositories," *IEEE Transactions on Cloud Computing*, 2017.

[15] H. Cheng, X. Zhang, J. Yu, and F. Li, "Markov process-based retrieval for encrypted jpeg images," *EURASIP Journal on Information Security*, vol. 2016, no. 1, pp. 1, 2016.

[16] H. Cheng, X. Zhang, and J. Yu, "Ac-coefficient histogram-based retrieval for encrypted jpeg images," *Springer Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13791–13803, 2016.

[17] K. Iida and H. Kiya, "A content-based image retrieval scheme using compressible encrypted images," *in Proc. European Signal Processing Conf.*, pp. 730–734, 2020.

[18] 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 International Conf. on Image Processing*, pp. 674–678, 2019.

[19] 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.

[20] K. Iida and H. Kiya, "An image identification scheme of encrypted jpeg images for privacy-preserving photo sharing services," *Proc. IEEE International Conf. on Image Processing*, pp. 4564–4568, 2019.

[21] F. Arnia, I. Iizuka, M. Fujiyoshi, and H. Kiya, "Fast and robust identification methods for jpeg images with various compression ratios," *in Proc. IEEE Int'l Conf. Acoustics Speech and Signal Processing Proceedings*, vol. 2, pp. II–II, 2006.

[22] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for jpeg/motion jpeg standard," *IEICE Trans. on Fundamentals.*, vol. 98, no. 11, pp. 2238–2245, 2015.

[23] 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, 2018.

[24] T. Chuman, W. Sirichotedumrong, and H. Kiya, "Encryption-then-compression systems using grayscale-based image encryption for jpeg images," *IEEE Trans. on Information Forensics and security*, vol. 14, no. 6, pp. 1515–1525, 2019.

[25] 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*, vol. 8, 2019.

[26] T. Chuman, K. Kurihara, and H. Kiya, "Security evaluation for block scrambling-based etc systems against extended jigsaw puzzle solver attacks," *in Proc. IEEE Int'l Conf. Multimedia and Expo*, pp. 229–234, 2017.

[27] 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. information forensics and security*, vol. 9, no. 1, pp. 39–50, 2014.

[28] C. Iakovidou, N. Anagnostopoulos, A. Kapoutsis, Y. Boutalis, M. Lux, and Savvas A. Chatzichristofis, "Localizing global descriptors for content-based image retrieval," *EURASIP Journal on Advances in Signal Processing*, vol. 2015, no. 1, pp. 80, 2015.

[29] M. Lux and Savvas A Chatzichristofis, "Lire: lucene image retrieval: an extensible java cbir library," *in Proc. ACM international Conf. on Multimedia*, pp. 1085–1088, 2008.

[30] T. Sikora, "The mpeg-7 visual standard for content description an overview," *IEEE Trans. on circuits and systems for video technology*, vol. 11, no. 6, pp. 696–702, 2001.

[31] Savvas A Chatzichristofis and Yiannis S Boutalis, "Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval," *in Proc. Springer International Conf. on Computer Vision Systems*, pp. 312–322, 2008.

[32] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Elsevier Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[33] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," *in Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 2161–2168, 2006.

[34] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," *in Proc. Springer European conference on computer vision*, pp. 304–317, 2008.