IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 20, 2020, accepted May 26, 2020, date of publication June 10, 2020, date of current version June 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3000683

Secure Outsourcing Algorithm of BTC Feature Extraction in Cloud Computing

MINGFANG JIANG AND HENGFU YANG
Department of Information Science and Engineering, Hunan First Normal University, Changsha 410205, China

Corresponding author: Hengfu Yang (hengfuyang@hotmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61872408, in part the Social Science Fund of Hunan Province under Grant 16YBA102, and in part by the Research Fund of Hunan Provincial Key Laboratory of Informationization Technology for Basic Education under Grant 2015TP1017.

ABSTRACT Advances in cloud computing have aroused many researchers' interest in privacy-preserving feature extraction over outsourced multimedia data, especially private image data. Since block truncation coding (BTC) is known as a simple and efficient technology for image compression, this paper focuses on privacy-preserving feature extraction in BTC compressed domain. We propose a privacy-preserving computation of BTC feature extraction over massive encrypted images (also called PPBTC). First, all images are uploaded to the cloud after encryption. The privacy-preserving image encryption process consists of block permutation, pixel diffusion, and a bit-plane random shift. BTC features remain unchanged after encryption and the cloud server can directly extract BTC features from the encrypted images. Some analyses and experimental results demonstrate that the proposed privacy-preserving feature extraction scheme for BTC-compressed images is efficient and secure, and it can be applied to secure image computation applications in cloud computing.

INDEX TERMS Feature extraction, privacy-preserving, block truncation coding, homomorphic encryption.

I. INTRODUCTION

Nowadays, as one of the most popular multimedia contents, digital images are playing more and more roles in almost every aspect of people's daily life. With the development of cloud computing, Internet of Things and big data, data owners are inclined to store and backup their sensitive data to cloud server for its cost-saving and convenience since the cloud server provides huge storage space and powerful computing capacity. By outsourcing image data and image computation to the cloud server, the cloud services can help the data owners relieve himself/herself from the large storage cost, heavy computational burden and complex data management of the image data. While the outsourcing services of the data storage and image computation bring many benefits, it also results in data privacy concerns because there exist some private contents in the multimedia data. A traditional method to protect privacy is encryption, but it will sacrifice the usability and accessibility of the multimedia data stored online since it cannot provide users common data management services such as search, classification, and data analysis over encrypted multimedia data. Secure image feature extraction, privacy-preserving information retrieval, or searchable encryption are a promising solution to privacy protection in outsourcing data management [1]–[7]. In these schemes, to protect data privacy, sensitive user data will be encrypted before outsourcing to the cloud server. Sever can perform computing including image feature detection and similarity search over encrypted data without decryption. Thus, they can provide efficient retrieval while maintaining privacy requirements. In the cloud computing era, privacy-preserving image computing and searchable encryption are playing an important role in online multimedia data management [8]–[12].

Searchable encryption can be dated back to 2000 with work by Song and Perrig [1]. Since then, there are many different types of searchable encryption schemes are reported in the literature. Prior work in the area of searchable encryption focused on text documents [1], [2], [4], [5], [13]. In addition to text data, there is a great number of multimedia data such as image data and video data stored on the cloud server. There is an increasing need for secure image data sharing in cloud computing. Numerous previous works focused on secure image feature extraction from encrypted

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

images such as speeded up robust feature (SURF) [14], scale-invariant feature transform (SIFT) [15]–[18], the histogram of oriented gradient (HOG) [19] and local binary pattern (LBP) [20]–[22]. To save storage space and communication bandwidth, digital images are often compressed before uploading on the cloud server. Considering the simplicity and low computation cost of block truncation coding (BTC), we propose a privacy-preserving BTC feature extraction method to ensure the secure sharing of compressed images in the cloud server in this paper.

The main contributions of secure BTC feature extraction are as follows.

(1) We developed a new secure feature extraction scheme suitable for compressed images by combining the BTC image compression method. Users and cloud servers can extract image features directly from encrypted BTC-compressed images without the need for decompression and decryption. It can be applied to secure image retrieval applications based on BTC and is a good extension of privacy-preserving multimedia retrieval.

(2) Our scheme consists of BTC ternary permutation, quantization level diffusion, and bitplane bitshift. The three encryption steps make BTC feature histogram unchanged before and after encryption. So, the new scheme has good robustness against common content-preserving manipulations. It can be applied to searchable image encryption applications and protect the image contents while support direct feature extraction from encrypted images.

(3) The BTC feature histogram extracted by our scheme is not encrypted and can be immediately used to image processing applications such as image retrieval, image recognition, and image authentication. Secure feature extraction in our scheme can be implemented on only one server, which makes the new scheme is secure against the collusion attack of several servers.

The rest of the paper is organized as follows: In Section II, we give a general review of the state-of-the-art of privacy-preserving feature extraction. We briefly introduce the basic knowledge of BTC in Section III. Then, the proposed approach (PPBTC) is described in section IV. The experiments and analysis are provided in Section V. Finally, conclusions are made in Section VI.

II. RELATED WORKS

In recent years, researchers have extended privacy-preserving information retrieval from search over encrypted text [3]–[5], [7], [23] to secure multimedia retrieval [8], [24]–[31].

Erkin *et al.* summarized cryptographic primitives and some applications in secure signal processing [32]. However, applying cryptographic primitives to content-based multimedia retrieval is not straightforward. Lu *et al.* addressed the problem of enabling content-based retrieval over encrypted multimedia data [8]. Because the scale-invariant feature transform (SIFT) descriptor has been widely used in many image processing applications, Hsu *et al.* [15], [16] firstly addressed the problem of secure SIFT feature representation and extraction in the encrypted domain. The Paillier cryptosystem is employed to achieve a homomorphic encryption-based secure SIFT feature extraction method. This scheme uses a discrete logarithm problem (DLP) and RSA to obtain provable security, but it needs to perform modular exponentiations of large numbers and has a high computational cost. Since then, many secure SIFT feature extraction schemes are studied. To save storage space and enhance efficiency, Jiang *et al.* proposed an efficient privacy-preserving SIFT feature extraction method [17]. In this method, the authors designed an encoding method which converts a non-integer into an integer. The new encoding method and leveled homomorphic encryption are used to encrypt images. New schemes of leveled homomorphic division (LHD), comparison algorithm (LHCA), and derivative algorithm (LHDA) on encrypted images are put forward. By combining new LHD, LHDA, and LHCA, unstable key points on the edge region can be eliminated. Their method can achieve small storage overhead and low communication costs without loss of usability and confidentiality. To enhance security, Feng *et al.* designed a new outsourcing SIFT feature extraction scheme over encrypted images based on consortium blockchain with the PoW (Proof of Work) consensus mechanism [18]. This scheme was implemented with the smart contract, DAC (Distributed Autonomous Corporation), sharding technique, and D2D (Device to Device) communication. The SIFT feature is reached for consensus through the PoW mechanism and the smart contract is also introduced into the blockchain, which can strengthen the security of the scheme. Hu *et al.* [33] proposed an effective and practical privacy-preserving computation outsourcing protocol for SIFT features over massive encrypted image data. It achieves efficiency and security requirements while protecting privacy of key characteristics by randomly splitting the original image data, designing novel protocols for secure multiplication and comparison, and deliberately distributing the feature computations onto two independent cloud servers. Experimental evaluation showed its security and effectiveness. Li *et al.* [34] studied privacy-preserving face recognition based on double decryption by using SIFT. In their scheme, the client encrypts his private image data locally and outsources encrypted images to the server which performs most of the computations without the need for decryption. Theoretical analyses validated its security and efficiency. As the upgraded version of SIFT descriptors, the histogram of oriented gradients (HOG) descriptors has been widely used in objection detection. Wang *et al.* firstly focused on secure outsourcing HOG feature computation to remote cloud servers [19]. It proposed privacy-preserving HOG outsourcing protocols in which homomorphic encryption integrated with single-instruction multiple-data (SIMD) is employed to encrypt image data. A batched secure comparison protocol and the redesigning of HOG ensure the feature computation and matching in the ciphertext domain.

Moreover, some new privacy-preserving schemes for other features have been reported in the literature. Chen *et al.*

proposed a multifractal feature extraction and representation method in the encrypted domain [35]. A chaotic sequence is first used to scramble the image block-wisely, then the secure multifractal feature extraction for encrypted images is designed by exploiting the locally the randomness and special periodicity of chaotic sequence. Yang *et al.* [36] presented a privacy-preserving Hahn moments scheme in the encrypted domain by using Somewhat Homomorphic Encryption (SHE) named PPHM. A mathematical framework for the PPHM in the encrypted domain is proposed. The PPHM has low complexity and high security while obtaining good performance in both image reconstruction and image recognition. Because Hu's moments are widely employed in pattern recognition, Preethi and Cherukuri [37] addressed the problem of moment invariants computation in the encrypted domain and proposed a secure Hu's moments computation based on fully homomorphic encryption. It may be used for privacy-preserving feature extraction in a remote cloud server hosted by the third party. In recent years, researchers have studied local binary patterns (LBP) computation [38] in the encrypted domain. Sultana and Shubhangi presented a privacy-preserving LBP computation outsourcing protocol (PP-LBP) that enables extracting LBP features from encrypted images [20]. In the PP-LBP scheme, the bit plane randomization technique is used to encrypt Most Significant Bit (MSB) plane of an image, and the PP-LBP based feature extraction is performed on the last seven bit-planes of an encrypted image. The method can get the same LBP features from both encrypted and unencrypted images without any communication cost between the cloud server and the client. Xia *et al.* [21] proposed a secure LBP feature extraction algorithm. In which the images are encrypted by block permutation, pixel permutation, and image segmentation. The cloud servers can calculate directly the LBP features from encrypted images by secure multiparty computation. To eliminate the collusion between multiple servers can be a big threat to the user's privacy, Xia *et al.* extended the scheme and designed a new privacy-preserving LBP feature extraction scheme in 2019 [22]. In the updated scheme, the images are encrypted by block shuffling, intra-block shuffling, and order-preserving pixel value substitution. The secure LBP feature extraction in the encrypted domain only needs one cloud server. Cheng *et al.* proposed a novel retrieval scheme for encrypted JPEG images which exploits the histogram invariance to calculate the distances between feature vectors and achieve high precision-recall performance [39].

In this paper, we aim at expanding secure image computation outsourcing to BTC compressed domain which has been widely used in many image processing applications.

III. BLOCK TRUNCATION CODING

In this section, BTC image coding will be briefly described. BTC is first proposed by Delp and Mitchell in 1979 for data compression [40]. As a simple and fast lossy compression technique, BTC has been widely employed in many applications such as image compression, image watermarking, image retrieval, and so on [41]–[44]. It performs standard moments preserving quantization during image coding so that it can achieve both acceptable image quality and low storage space. Absolute moment block truncation coding (AMBTC) is an improved version of BTC presented by Lema and Mitchell [45] which preserves absolute moments. The AMBTC has low computational cost since square root and multiplication operations are omitted during the encoding. So, to enhance the time efficiency of feature computation outsourcing to the cloud, we consider secure feature computation in encrypted images for AMBTC compressed images.

In AMBTC, each image is first divided into non-overlapping image blocks of $s \times s$ pixels. Each $s \times s$ image block is viewed as an image vector of num dimensions where $num = s \times s$. These image vectors are sequentially processed in raster scanning. The mean value $\bar{x}$ is first calculated according to the following equation for each block.

$$\bar{x} = \frac{1}{num} \sum_{i=1}^{num} x_i \tag{1}$$

The pixels in each image block are classified into two groups according to their block mean value. The pixels with intensity less than or equal to $\bar{x}$ are classified into the first group (G_0). Otherwise, the second group (G_1). The lower mean x_l of group G_0 and the higher mean x_h of group G_1 are then calculated according to the following equation, respectively.

$$\begin{cases} x_h = \dfrac{1}{k} \displaystyle\sum_{x_i \in G1} x_i \\ x_l = \dfrac{1}{num - k} \displaystyle\sum_{x_i \in G0} x_i. \end{cases} \tag{2}$$

where q denotes the number of pixels with intensity greater than or equal to the block mean $\bar{x}$.

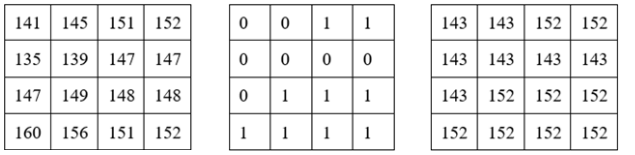
The bitplane $B = \{b_i \,|b_i \in \{0, 1\}, i = 1, 2, \cdots, num\}$ for each image block can be constructed by comparing pixel values within each block and its block mean value $\bar{x}$ as follows.

$$b_i = \begin{cases} 0 & if\ x_i < \bar{x} \\ 1 & else \end{cases} \tag{3}$$

Finally, each image block produces a ternary of compressed code (x_h, x_l, B) by applying AMBTC encoding.

In the AMBTC decoding procedure, each image block is reconstructed by replacing the '1's with higher mean x_h and the '0's by lower mean x_l.

Taking an image block of 4×4 pixels as an example, Fig. 1 illustrates AMBTC encoding and decoding procedures, where $num = 16$. The original image block has a mean of $\bar{x} = 148$. The mean $\bar{x}$ is then taken as a threshold to generate a bitplane B, as shown in Fig. 1(b), where $k = 9$. Two rounded quantization levels $x_h = 152$ and $x_l = 143$ can be computed using Eq. (2). Finally, the original image block generates a ternary of compressed code $(152, 143, 0011000001111111_2)$. In the decoding procedure, the higher mean x_h and lower



(a) original image (b) bitplane (c) reconstructed one

FIGURE 1. An example of AMBTC encoding and decoding.

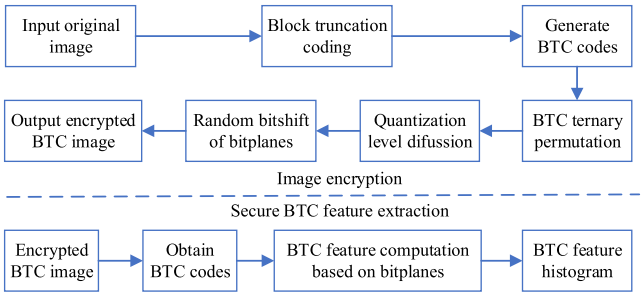


FIGURE 2. The framework for image encryption and secure BTC feature extraction.

mean x_l are used to reconstruct the image block as shown in Fig. 1(c).

IV. THE PROPOSED ALGORITHM (PPBTC)

By combining the AMBTC algorithm, a privacy-preserving image feature extraction scheme for BTC-compressed images (PPBTC) is developed. The proposed PPBTC scheme consists of image encryption and secure BTC feature extraction as shown in Fig. 2. Image encryption procedure includes 3 main subroutines: BTC ternary permutation, quantization level diffusion, and random bitshift of bitplanes. The BTC feature histogram extraction procedure completes the BTC feature computation based on the bitplanes. Data owner possesses a large number of image data to need to be outsourced for storage burden saving, computation cost-reducing, and efficient management. To protect privacy in images, the images need to be encrypted before being uploaded to the cloud server. In our scheme, the user only needs a copy of the query image and then sends a request to the cloud server. The cloud server provides BTC feature computation and returns extracted BTC features or other BCT feature-based operational results. In a word, the cloud server can complete the query tasks while not sacrificing data privacy.

A. IMAGE ENCRYPTION

During image encryption, a ternary permutation followed quantization level diffusion is first performed in the BTC compressed domain. To achieve good data security, random bitshift operations are then applied to each bitplane.

Here, a piecewise linear chaotic map (PWLCM) is employed to generate the desirable binary chaotic sequences for image encryption since PWLCM has many excellent properties such as ergodicity, pseudo-random behavior, an auto-correlation function. The PWLCM is

Algorithm 1 BTC Ternary Permutation

Input: Original image I with the size of $m \times n$, secret key k_1
Output: Scrambled BTC image $I'_{BTC} = \left(x'_h, x'_l, B'\right)_0^{tn-1}$
Initialization: image sub-block size $s = 4$.

1. Calculate the number of BTC ternaries *in* using Eq. (6).
2. Perform BTC encoding on original image I and produce BTC compressed image $I_{BTC} = (x_h, x_l, B)_0^{tn-1}$.
3. Use secret key k_1 as the initial value to generate the pseudo-random permutation Pt by Eq. (5).
4. for $\forall i \in (0, \cdots, tn-1)$ do
5. $\quad T'_i = \left(x'_h, x'_l, B'\right)_i \leftarrow T_{pt(i)} = (x_h, x_l, B)_i$
6. end for
7. Return Scrambled BTC image I'_{BTC}

defined as follows.

$$x_{i+1} = F(x_i, q) = \begin{cases} \dfrac{x_i}{q}, & x_i \in [0,q) \\ \dfrac{x_i - q}{0.5 - q}, & x_i \in [q, 0.5) \\ F(1 - x_i, q), & x_i \in [0.5, 1) \end{cases} \quad (4)$$

where $x_i \in [0, 1]$ q is a control parameter and this random sequence generated by Eq. (4) is in a chaotic state when parameter $q \in (0, 0.5)$.

1) BTC TERNARY PERMUTATION

In the process of BTC ternary permutation, we first use Eq. (4) to construct a pseudo-random permutation generator (*PRPG*) for generating random permutations. The secret key k_1 is used as an initial value of the PWLCM. The random permutation Pt is defined as follows.

$$Pt \leftarrow PRPG(k_1, tn) \quad (5)$$

where Pt is the produced random permutation within the range of $[0, 1, \cdots, tn-1]$, tn is the number of BTC ternaries in an image with a size of $m \times n$.

$$tn = \frac{m \times n}{s \times s} \quad (6)$$

Then the positions of all BTC ternaries are shuffled by the generated permutation. The permutation process is described in detail as algorithm 1.

2) QUANTIZATION LEVEL DIFFUSION AND RANDOM BITSHIFT OF BITPLANES

To enhance the image security, a pixel diffusion and a random bitshift operation are performed on the BTC ternaries subsequently. Secret key k_2 is used as the input of the permutation generator *PRPG* for generating a random permutation in the integer interval [0, 255].

$$Pg \leftarrow PRPG(k_2, 256) \quad (7)$$

Firstly, we can diffuse two quantization levels using the random permutation Pg for each BTC ternary.

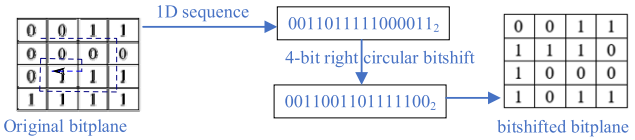


FIGURE 3. 4-bit right circular bitshift operation on a bitplane.

Algorithm 2 Quantization Level Diffusion and Random Bitshift of Bitplanes

Input: Scrambled BTC image I'_{BTC} with the size of $m \times n$, secret keys k_2 and k_3
Output: Encrypted BTC image I''_{BTC}
Initialization: image sub-block size $s = 4$.

1. Read the scrambled BTC image I'_{BTC} and obtain scrambled BTC ternaries $\left(x'_h, x'_l, B'\right)_0^{tn-1}$.
2. Compute the number tn of BTC ternaries by using Eq. (6).
3. Generate pseudo-random permutation Pg within the integer interval [0, 255], i.e. $Pg \leftarrow PRPG\,(k_2, 256)$.
4. for $\forall i \in (0, \cdots, tn-1)$ do
5. Diffuse two quantization levels (the higher mean and the lower mean) of the BTC ternary $\left(x'_h, x'_l, B'\right)_i$,
   $x''_h \leftarrow Pg\left(x'_h\right)$
   $x''_l \leftarrow Pg\left(x'_l\right)$
6. Given secret key k_3 and then generate a random number Pb within the integer interval $[0, s \times s - 1]$ for random bitshift operation of bitplanes by using Eq. (8).
7. Apply Pb-bit random right circular bitshift operation on the bitplane B' in spiral order as shown in Fig. 3.
8. end for
9. Return encrypted BTC image I''_{BTC}

At the same time, a pseudo-random number generator (*PRNG*) is constructed by using PWLCM. As an input, secret key k_3 is used to produce a pseudo-random number Pn by employing *PRNG*. The following random number generator *PRNG* produces a random number in the integer interval $[0, s \times s - 1]$.

$$Pn \leftarrow PRNG\,(k_3, i, s \times s) \tag{8}$$

where I denote the iteration number.

Finally, a Pn-bit pseudo-random right circular bitshift operation as shown in Fig. 3 is applied to each bitplane for further image scrambling.

Algorithm 2 shows the main steps of the quantization level diffusion and the circular bitshift process.

B. SECURE BTC FEATURE HISTOGRAM COMPUTATION

In our scheme, the cloud servers can extract directly BTC feature histogram from encrypted images for image search once users send a query request to the remote server.

First, we define a BTC feature based on the bitplane for each BTC ternary.

$$F_{BTC} = \sum_{i=0}^{num-1} b_i \times 2^i \tag{9}$$

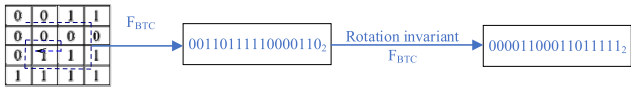


FIGURE 4. An example of BTC feature computation.

Algorithm 3 Secure BTC Feature Histogram Computation

Input: Encrypted BTC image I''_{BTC} with the size of $m \times n$.
Output: BTC histogram H
Initialization: image sub-block size $s = 4$.

1. Read the encrypted BTC image I''_{BTC} and obtain encrypted BTC ternaries $\left(x'_h, x'_l, B'\right)_0^{tn-1}$.
2. Get the number tn of encrypted BTC ternaries according to Eq.(6).
3. for $\forall i \in (0, \cdots, tn-1)$ do
4. Compute the rotation invariant BTC feature F^i_{riBTC} of the i^{th} BTC ternary using Eq.(10).
5. end for
6. All rotation invariant BTC features form the feature vector $\boldsymbol{F} = \left(F^0_{riBTC}, \ldots, F^i_{riBTC}, \ldots, F^{tn-1}_{riBTC}\right)$.
7. Produce the BTC feature histogram $\boldsymbol{H}$ of the BTC feature vector $\boldsymbol{F}$.
8. Return the BTC feature histogram $\boldsymbol{H}$.

To remove the effect of rotation, a rotation-invariant BTC feature F^{ri}_{BTC} is defined by assign a unique value to the rotation versions of the bitplane for each BTC ternary.

$$F_{riBTC} = min\left\{ROR\,(F_{BTC}, i)_{i=0}^{num-1}\right\} \tag{10}$$

where $ROR\,(x, i)$ performs a circular bit-wise right shift on the num-bit number xi times, and function $min\,(x)$ returns the minimum value of the vector x.

Fig. 4 shows an example of BTC feature computation of a bitplane with a size of 4×4, i.e. $num = 16$.

The proposed privacy-preserving feature extraction is performed according to the bitplane of each BTC ternary. The quantization level diffusion process only modifies the quantization levels, so, it doesn't impact feature extraction. And the circular bitshift operation doesn't change the feature value since the feature F_{riBTC} is rotation-invariant. At last, BTC ternary permutation only changes the pixel position, so the same BTC feature histogram can be obtained before and after encryption. The details of BTC feature histogram computation is illustrated in algorithm 3.

V. EXPERIMENTS AND ANALYSIS

The proposed secure BTC feature extraction scheme is implemented on Matlab2016 to test its performance. Some images from the USC-SIPI image database [46] are chosen to validate our secure BTC feature extraction algorithm. they contain different types of gray and color images. Their BTC compressed versions with the size of 512×512 are shown in Fig. 5.

A. ENCRYPTION EFFECT

The images are protected by performing BTC ternary permutation, quantization level diffusion, and random bitshift in

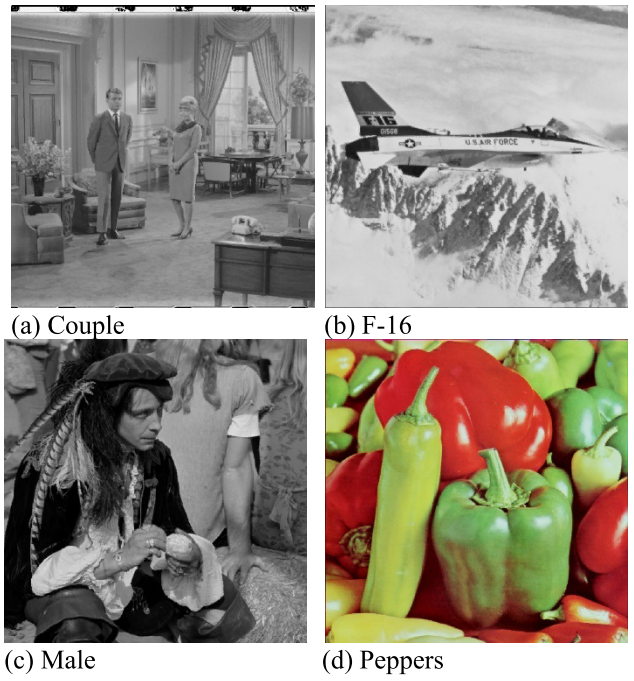


FIGURE 5. Some BTC-compressed test images.

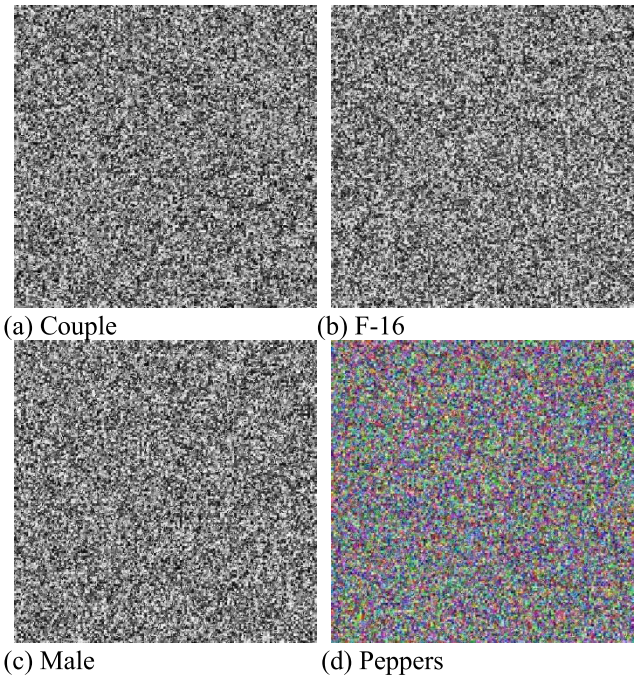


FIGURE 6. The encrypted images.

the proposed scheme. the corresponding encrypted images of BTC-compressed images shown in Fig.5 are illustrated in Fig. 6. For color images, the encryption algorithm is performed on the R, G, B components, respectively. From Fig.6, one can notice that it is extremely difficult to guess the image content from these encrypted images.

Moreover, BTC features will reveal the contour data of images once they are not effectively protected. Fig.7 (a), (c), (e), and (g) show the visual effect of BTC features extracted

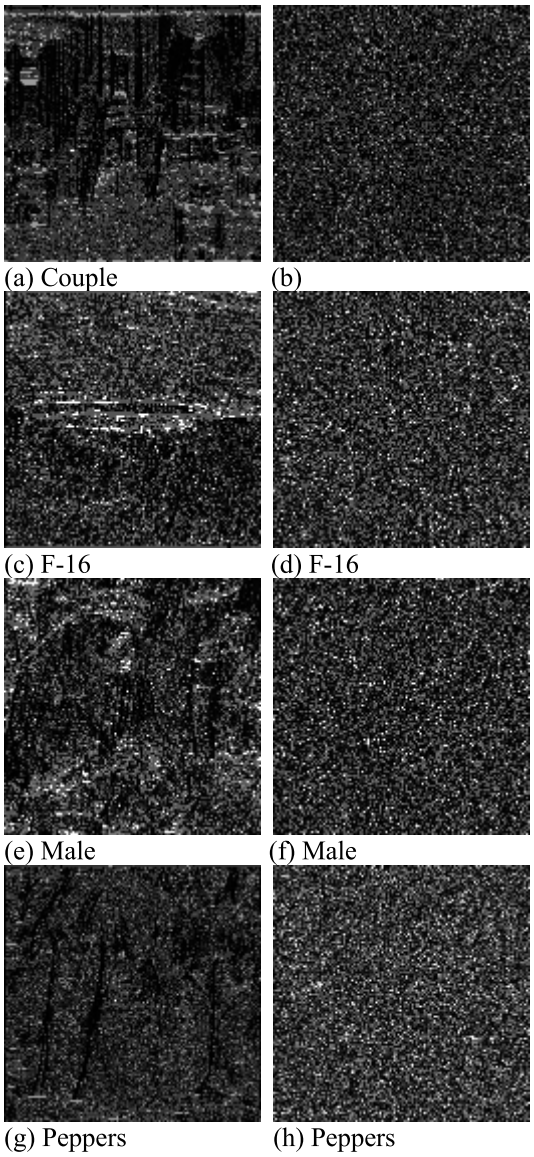


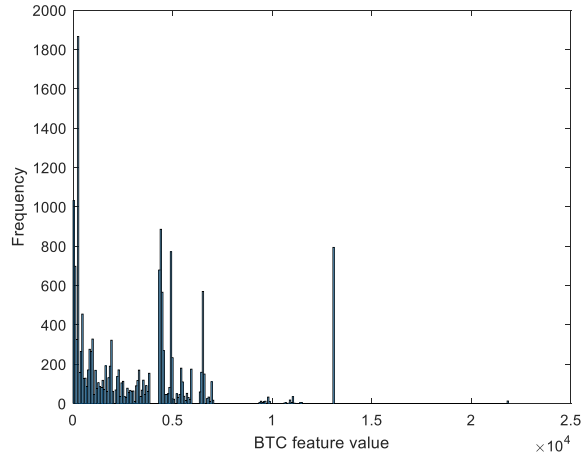
FIGURE 7. The visual effect of BTC features. (a), (c), (e), and (g): the BTC features of original BTC-compressed images. (b), (d), (f), and (h): the PPBTC features. For color images in (g) and (h), the Y component is taken as an example to show the visual effect.

from original BTC-compressed images. It can be seen that the BTC feature vector calculated from original images leak a lot of texture information about images. Correspondingly, BTC features computed from encrypted images are shown in Fig.7(b), (d), (f), and (h). From these Figures, we cannot notice any information about image content. The PPBTC features will not expose the image texture information since it is very hard to obtain the image content from the BTC features computed in the encrypted domain.

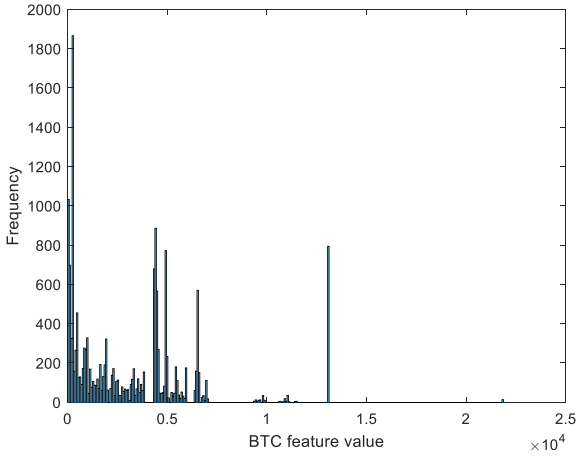
B. HOMOMORPHISM ANALYSIS

Our PPBTC scheme undergoes BTC ternary permutation, quantization level diffusion, and bitplane bitshift. According to the encryption mechanism, we have

Theorem 1: The BTC feature histogram stays the same before and after encryption.



(a) Before encryption



(b) After encryption

FIGURE 8. Extracted BTC feature histograms before and after image encryption.

Proof: Let $\varepsilon(\cdot)$ be the encryption function. Denote the encryption by BTC ternary permutation, quantization level diffusion and bitplane bitshift as $\varepsilon_1(\cdot)$, $\varepsilon_2(\cdot)$ and $\varepsilon_3(\cdot)$. Given the original image I, the overall encryption function $\varepsilon(I)$ is now $\varepsilon_3(\cdot)$ followed by $\varepsilon_2(\cdot)$ and $\varepsilon_1(\cdot)$. So, it can be written as $\varepsilon(I) = \varepsilon_1(\varepsilon_2(\varepsilon_3(I)))$. If given the feature extraction function $\mathcal{F}(\cdot)$, $\mathcal{F}(\varepsilon(I))$ represents extracting features directly from the encrypted image. According to the encryption mechanism of the functions $\varepsilon_1(\cdot)$ and $\varepsilon_2(\cdot)$, they don't change the BTC features defined by Eq. (10). So, we have

$$\mathcal{F}(\varepsilon(I)) = \mathcal{F}(\varepsilon_1(\varepsilon_2(\varepsilon_3(I)))) = \mathcal{F}(\varepsilon_1(I)) \tag{11}$$

Because the operator $\varepsilon_1(I)$ only changes the order of the BTC ternaries, $\mathcal{F}(\varepsilon(I))$ is only a permutation of the feature vector $\mathcal{F}(I)$ extracted from the original BTC-compressed image. Let $\mathcal{H}(\cdot)$ be the histogram operator. The following property can be derived,

$$\mathcal{H}(\mathcal{F}(\varepsilon(I))) = \mathcal{H}(\mathcal{F}(I)) \tag{12}$$

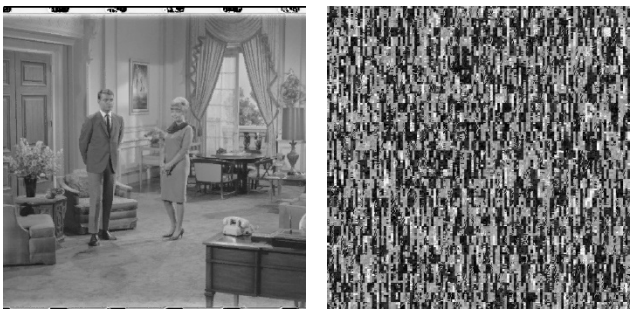
That is to say, our PPBTC scheme can guarantee that the histograms of BTC features are the same before and after encryption. ■

TABLE 1. Similarities among different original BTC-compressed images.

	Fig.5(a)	Fig.5(b)	Fig.5(c)	Fig.5(d)
Fig.5(a)	1	-0.0395	-0.0359	-0.0611
Fig.5(b)	-0.0395	1	-0.0405	-0.0564
Fig.5(c)	-0.0359	-0.0405	1	-0.0474
Fig.5(d)	-0.0611	-0.0564	-0.0474	1

TABLE 2. Similarities among different encrypted BTC-compressed images.

	Fig.6(a)	Fig.6(b)	Fig.6(c)	Fig.6(d)
Fig.6(a)	1	-0.0395	-0.0359	-0.0611
Fig.6(b)	-0.0395	1	-0.0405	-0.0564
Fig.6(c)	-0.0359	-0.0405	1	-0.0474
Fig.6(d)	-0.0611	-0.0564	-0.0474	1



(a) (b)

FIGURE 9. Secret key sensitivity.

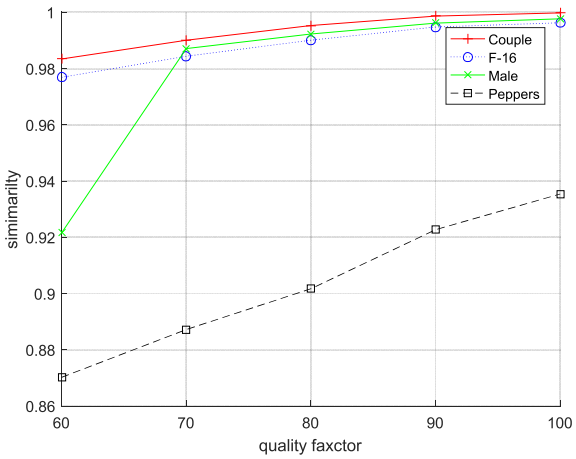
TABLE 3. Content-preserving manipulations for robustness evaluation.

manipulations	parameters
JPEG compression	Quality factor $\in$[60, 100]
Gaussian filtering	Window size =3, standard deviation $\in$[0.3, 1.0]
Median filtering	Window size $\in$[3, 11]
Gaussian noise	Mean m=0, standard deviation $\in$[0.0001, 0.0009]
Scaling	Scaling ratio $\in$[0.7, 1.3]
Rotation	Rotation angle$\in$[3, 11]

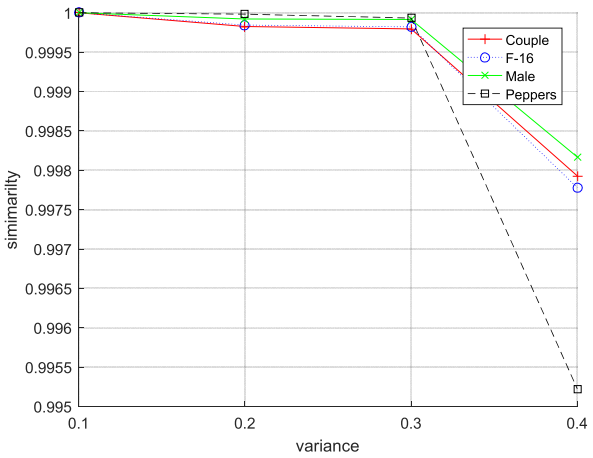
Taking the Couple image shown in Fig.5(a) as an example, the extracted histograms of BTC features from the original image and encrypted image are illustrated in Fig. 8. From Fig.8, It is easy to notice that the proposed PPBTC image encryption scheme can extract the same BTC feature histogram. This validates Theorem 1.

Given two encrypted images I_1 and I_2, their histograms are H_1 and H_2, respectively. The Pearson correlation coefficient between their histograms is used to evaluate the feature similarity, where $E(X)$ is the mathematical expectation of the distribution.

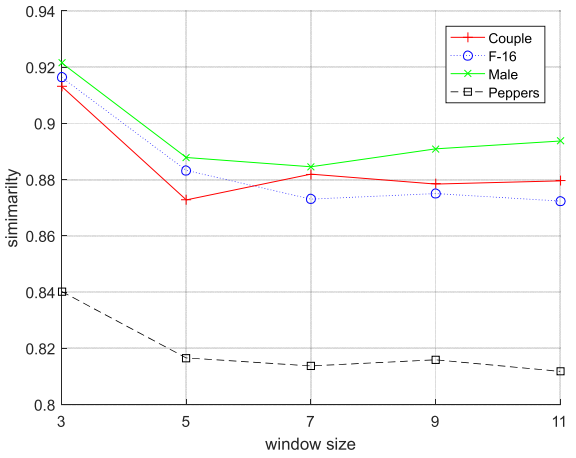
$$sim = \frac{E(H_1H_2) - E(H_1)E(H_2)}{\sqrt{E(H_1^2) - E^2(H_1)}\sqrt{E(H_2^2) - E^2(H_2)}} \tag{13}$$



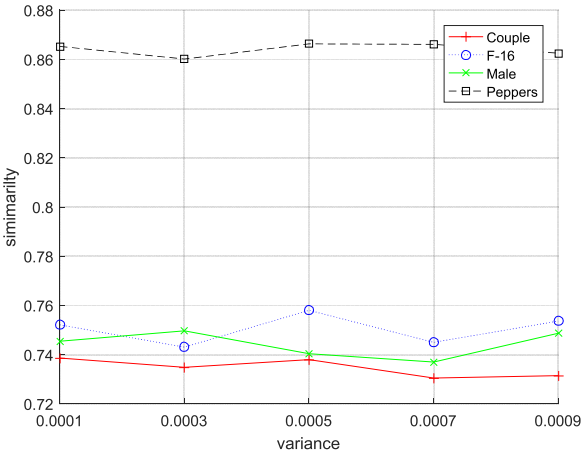
(a) JPEG compression



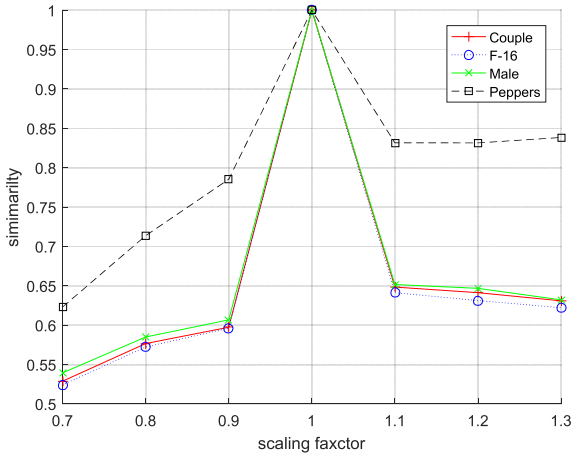
(b) Gaussian filtering



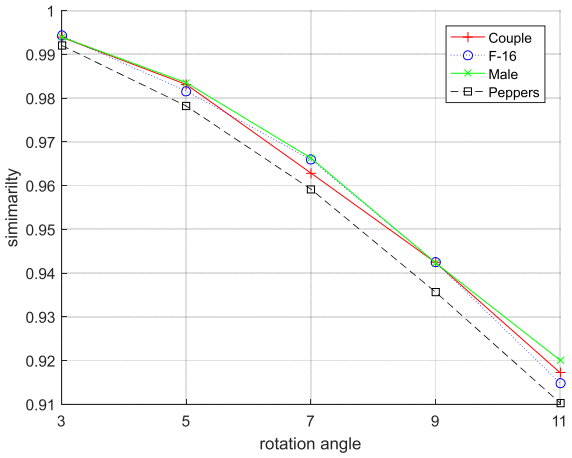
(c) median filtering



(d) Gaussian noise



(e) scaling



(f) rotation

FIGURE 10. **Robustness against different image processing attacks.**

This encryption scheme preserves the distance before and after encryption since it has the histogram invariance property.

We further calculate the similarities between different images to test the homomorphism. Table 1 and Table 2 show the similarities among various original images and encrypted images, respectively.

From Table 1 and Table 2, it can be found that the similarities of different images before encryption are the same as those of these images after encryption.

Thus, we can conclude that the new PPBTC scheme has the homomorphism property.

C. SECURITY

We have selected one gray level image 'Couple' shown in Fig.5(a) with 512×512 pixels as the sample plain image to test the security of the presented PPBTC scheme. The decrypted images are shown in Fig 9. Fig.9 (a) and Fig.9 (b) illustrate the decrypted images with the right secret key combination (k_1, k_2, k_3) and a wrong secret key combination which is only different in the secret key k_3, respectively. From Fig.9, one cannot figure out the information about image content from the wrongly decrypted image. So, it is difficult to decrypt an encrypted image without knowing the secret key.

Moreover, an ideal encryption scheme should have a large enough key space. For our PPBTC image encryption scheme, we have the following theorem.

Theorem 2: The PPBTC encryption scheme is secure against brute-force attacks.

Proof: In the proposed PPBTC scheme, three different secret keys (k_1, k_2, k_3) are used as keys. The key combination has 10^{45} possible values since each secret key has a precision of 15 decimal digits. The key space is approximate $O\left(10^{45}\right) \approx O\left(2^{150}\right)$. So, the PPBTC scheme has a larger key space than the AES-128 algorithm.

Since the AES-128 algorithm is considered safe against any brute force attacks, it can be concluded that the PPBTC scheme can defeat brute-force attacks.■

D. ROBUSTNESS

To examine the robustness of our secure PPBTC feature extraction scheme against various content-preserving manipulations. BTC-compressed images are produced based on the USC-SIPI image database and their corresponding encrypted images are utilized to test the robustness performance. These content-preserving manipulations include JPEG compression, Gaussian filtering, median filtering, noise addition, scaling, and rotation as shown in Table 3.

Taking four different types of images (shown in Fig.5) as examples, Fig. 10 (a–f) illustrates the similarity between the feature vectors of the encrypted images and the corresponding attacked versions by six image processing manipulations described by Table. 3, respectively. It can be found that the proposed PPBTC scheme has slightly lower similarity only for scaling. Generally speaking, it can obtain satisfactory similarity after suffering six kinds of common content-preserving manipulations. This indicates that the proposed privacy-preserving BTC feature computation scheme is robust against common content-preserving manipulations.

E. APPLICATION IN IMAGE RETRIEVAL

In this section, we will evaluate the effectiveness of secure feature extraction over encrypted images by applying the proposed PPBTC scheme to image retrieval. We perform a comparative analysis of the proposed privacy-preserving feature extraction algorithm with the scheme of Xia *et al.* [22]. and our PPBTC scheme using the USC-SIPI image database [46]. Two metrics, i.e., precision and recall are utilized for image matching evaluation in the encrypted domain. Precision (P) is the number of true positives (TP) over the number of true positives plus the number of false positives (FP), and recall (R) is defined as the number of true positives over the number of true positives plus the number of false negatives (FN).

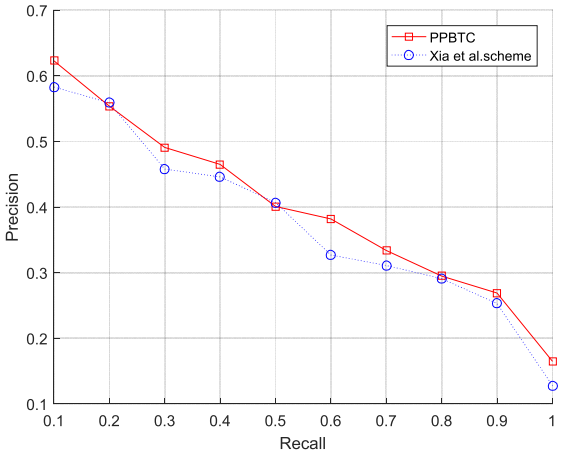


FIGURE 11. Retrieval performance comparison.

$$P = \frac{TP}{TP + FP} \tag{14}$$

$$R = \frac{TP}{TP + FN} \tag{15}$$

In Fig. 11, we illustrate precision versus recall curves for image retrieval under different privacy-preserving feature extraction. From Fig.10, our PPBTC scheme has higher precision than the schemes of Xia *et al.* [22] under the same recall. Therefore, one can say that our scheme can achieve better performance of image retrieval in terms of precision versus recall than Xia *et al.*'s scheme.

VI. CONCLUSIONS

In this paper, a privacy-preserving BTC feature extraction for encrypted images is proposed. The images are encrypted by BTC ternary permutation, quantization level diffusion, and random bitshift of bitplanes. The encryption mechanism supports direct BTC feature computation in the encryption domain and can be conducted on a single cloud server without the need for communication between multiple cloud servers. This means that it can effectively resist collusion attacks among cloud servers. Experimental results and analysis validate that our PPBTC scheme has better robustness against common content-preserving manipulations, and can be applied to secure image communication in cloud computing.

REFERENCES

[1] D. Xiaoding Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proc. IEEE Symp. Secur. Privacy (S P)*, Berkeley, CA, USA, May 2000, pp. 44–55.

[2] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Proc. Adv. Cryptol. (EUROCRYPT)*. Berlin, Germany: Springer, 2004, pp. 506–522.

[3] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized private keyword search over encrypted data in cloud computing," in *Proc. 31st Int. Conf. Distrib. Comput. Syst.*, Minneapolis, MN, USA, Jun. 2011, pp. 383–392.
[4] S. Kamara, "Encrypted search," *XRDS, Crossroads, ACM Mag. Students*, vol. 21, no. 3, pp. 30–34, 2015.
[5] Y. Wang, J. Wang, and X. Chen, "Secure searchable encryption: A survey," *J. Commun. Inf. Netw.*, vol. 1, no. 4, pp. 52–65, Dec. 2016.
[6] R. Handa, R. K. Challa, and N. Aggarwal, "Searchable encryption: A survey on privacy-preserving search schemes on encrypted outsourced data," *Concurrency Comput., Pract. Exper.*, vol. 31, no. 17, pp. 1–49, 2019.
[7] H. Pham, J. Woodworth, and M. A. Salehi, "Survey on secure search over encrypted data on the cloud," *Concurrency Comput., Pract. Exper.*, vol. 31, no. 17, pp. 1–15, 2019.
[8] W. Lu, A. Swaminathan, A. L. Varna, and W. Min, "Enabling search over encrypted multimedia databases," in *Proc. SPIE-IS T Electron. Imag.*, San Jose, CA, USA, 2009, Art. no. 725418.
[9] J. G. Jeong, B. R. Cha, and J. Kim, "Feasibility study of searchable image encryption system of streaming service based on cloud computing environment," in *Proc. Int. Conf. Data Mining Comput. Eng. (ICDMCE)*, Bangkok, Thailand, 2012, pp. 180–184.
[10] M. Jiang and G. Sun, "A chaotic searchable image encryption scheme integrating with block truncation coding," in *Proc. Int. Conf. Cloud Comput. Secur.*, Hainan, China. Cham, Switzerland: Springer, vol. 11065, 2018, pp. 349–358.
[11] J. Ye, Z. Xu, and Y. Ding, "Image search scheme over encrypted database," *Future Gener. Comput. Syst.*, vol. 87, pp. 251–258, Oct. 2018.
[12] 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.
[13] P. Golle, J. Staddon, and B. Waters, "Secure conjunctive keyword search over encrypted data," in *Appl. Cryptography Netw. Secur.* Berlin, Germany: Springer, 2004, pp. 31–45.
[14] Q. Wang, S. Hu, J. Wang, and K. Ren, "Secure surfing: Privacy-preserving speeded-up robust feature extractor," in *Proc. IEEE 36th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2016, pp. 700–710.
[15] C.-Y. Hsu, C.-S. Lu, and S.-C. Pei, "Homomorphic encryption-based secure SIFT for privacy-preserving feature extraction," in *Proc. 3rd Media Watermarking, Secur., Forensics*, San Francisco, CA, USA, 2010, vol. 7880, no. 2, Art. no. 788005.
[16] C.-Y. Hsu, C.-S. Lu, and S.-C. Pei, "Image feature extraction in encrypted domain with privacy-preserving SIFT," *IEEE Trans. Image Process.*, vol. 21, no. 11, pp. 4593–4607, Nov. 2012.
[17] L. Jiang, C. Xu, X. Wang, L. Bo, and H. Wang, "Secure outsourcing SIFT: Efficient and privacy-preserving image feature extraction in the encrypted domain," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 1, pp. 179–193, Jan./Feb. 2020.
[18] X. Feng, J. Ma, T. Feng, Y. Miao, and X. Liu, "Consortium blockchain-based SIFT: Outsourcing encrypted feature extraction in the D2D network," *IEEE Access*, vol. 6, pp. 52248–52260, 2018.
[19] Q. Wang, J. Wang, S. Hu, Q. Zou, and K. Ren, "SecHOG: Privacy-preserving outsourcing computation of histogram of oriented gradients in the cloud," in *Proc. 11th ACM Asia Conf. Comput. Commun. Secur. (ASIA CCS)*, Xi'an, China, 2016, pp. 257–268.
[20] S. F. Sultana and D. C. Shubhangi, "Privacy preserving LBP based feature extraction on encrypted images," in *Proc. Int. Conf. Comput. Commun. Informat. (ICCCI)*, Jan. 2017, pp. 1–4.
[21] Z. Xia, X. Ma, Z. Shen, X. Sun, N. N. Xiong, and B. Jeon, "Secure image LBP feature extraction in cloud-based smart campus," *IEEE Access*, vol. 6, pp. 30392–30401, 2018.
[22] Z. Xia, L. Jiang, X. Ma, W. Yang, P. Ji, and N. Xiong, "A privacy-preserving outsourcing scheme for image local binary pattern in secure industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 629–638, Jan. 2020.
[23] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 1, pp. 222–233, Jan. 2014.
[24] W. Lu, A. L. Varna, and M. Wu, "Confidentiality-preserving image search: A comparative study between homomorphic encryption and distance-preserving randomization," *IEEE Access*, vol. 2, pp. 125–141, 2014.
[25] Z. Qin, J. Yan, K. Ren, C. W. Chen, and C. Wang, "Towards efficient privacy-preserving image feature extraction in cloud computing," in *Proc. ACM Int. Conf. Multimedia (MM)*, Orlando, FL, USA, 2014, pp. 497–506.
[26] J. Yuan, S. Yu, and L. Guo, "SEISA: Secure and efficient encrypted image search with access control," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Hong Kong, Apr. 2015, pp. 2083–2091.
[27] Q. Zou, J. Wang, and X. Chen, "Secure encrypted image search in mobile cloud computing," in *Proc. 10th Int. Conf. Broadband Wireless Comput., Commun. Appl. (BWCCA)*, Krakow, Poland, Nov. 2015, pp. 572–575.
[28] Q. Zou, J. Wang, J. Ye, J. Shen, and X. Chen, "Efficient and secure encrypted image search in mobile cloud computing," *Soft Comput.*, vol. 21, no. 11, pp. 2959–2969, Jun. 2017.
[29] Q. Wang, S. Hu, K. Ren, J. Wang, Z. Wang, and M. Du, "Catch me in the dark: Effective privacy-preserving outsourcing of feature extractions over image data," in *Proc. IEEE 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.
[30] D. Li, X. Dong, and Z. Cao, "Privacy-preserving outsourced image feature extraction," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops*, Aug. 2019, pp. 59–64.
[31] A. A. Aminuddin Mohd Kamal, K. Iwamura, and H. Kang, "Searchable encryption of image based on secret sharing scheme," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Conf. (APSIPA ASC)*, Kuala Lumpurr, Malaysia, Dec. 2017, pp. 1495–1503.
[32] Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni, "Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing," *EURASIP J. Inf. Secur.*, vol. 7, no. 2, pp. 1–20, 2007.
[33] S. Hu, Q. Wang, J. Wang, Z. Qin, and K. Ren, "Securing SIFT: Privacy-preserving outsourcing computation of feature extractions over encrypted image data," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3411–3425, Jul. 2016.
[34] P. Li, T. Li, Z.-A. Yao, C.-M. Tang, and J. Li, "Privacy-preserving outsourcing of image feature extraction in cloud computing," *Soft Comput.*, vol. 21, no. 15, pp. 4349–4359, Aug. 2017.
[35] G. Chen, Q. Chen, X. Zhu, and Y. Chen, "Encrypted image feature extraction by privacy-preserving MFS," in *Proc. 7th Int. Conf. Digit. Home (ICDH)*, Guilin, China, Nov. 2018, pp. 42–45.
[36] T. Yang, J. Ma, Q. Wang, Y. Miao, X. Wang, and Q. Meng, "Image feature extraction in encrypted domain with privacy-preserving Hahn moments," *IEEE Access*, vol. 6, pp. 47521–47534, 2018.
[37] G. Preethi and A. K. Cherukuri, "Privacy preserving Hu's moments in encrypted domain," in *Intelligent Systems Design and Applications*. Cham, Switzerland: Springer, 2018, pp. 326–336.
[38] H. Yang, J. Yin, and Y. Yang, "Robust image hashing scheme based on low-rank decomposition and path integral LBP," *IEEE Access*, vol. 7, pp. 51656–51664, 2019.
[39] H. Cheng, X. Zhang, and J. Yu, "AC-coefficient histogram-based retrieval for encrypted JPEG images," *Multimedia Tools Appl.*, vol. 75, no. 21, pp. 13791–13803, Nov. 2016.
[40] E. Delp and O. Mitchell, "Image compression using block truncation coding," *IEEE Trans. Commun.*, vol. COM-27, no. 9, pp. 1335–1342, Sep. 1979.
[41] B. Balakrishnan, S. H. Darsana, J. Mathews, and M. S. Nair, "Satellite/Aerial image compression using adaptive block truncation coding technique," *J. Indian Soc. Remote Sens.*, vol. 46, no. 11, pp. 1761–1771, Nov. 2018.
[42] H. Yang and J. Yin, "A secure removable visible watermarking for BTC compressed images," *Multimedia Tools Appl.*, vol. 74, no. 6, pp. 1725–1739, Mar. 2015.
[43] N. Mohammad, X. Sun, H. Yang, J. Yin, G. Yang, and M. Jiang, "Lossless visible watermarking based on adaptive circular shift operation for BTC-compressed images," *Multimedia Tools Appl.*, vol. 76, no. 11, pp. 1–13, 2017.
[44] J. Thangarasu and P. Geetha, "Content based image retrieval using quad tree block truncation coding with color co-occurrence feature for the big data platform," *J. Comput. Theor. Nanosci.*, vol. 14, no. 8, pp. 3874–3886, Aug. 2017.
[45] M. Lema and O. Mitchell, "Absolute moment block truncation coding and its application to color images," *IEEE Trans. Commun.*, vol. COM-32, no. 10, pp. 1148–1157, Oct. 1984.
[46] USC-SIPI. *The USC-SIPI Image Database*. [Online]. Available: http://sipi.usc.edu/database/