Image Feature Extraction in Encrypted Domain With Privacy-Preserving SIFT

Chao-Yung Hsu, Chun-Shien Lu, and Soo-Chang Pei, *Fellow, IEEE*

***Abstract*— Privacy has received considerable attention but is still largely ignored in the multimedia community. Consider a cloud computing scenario where the server is resource-abundant, and is capable of finishing the designated tasks. It is envisioned that secure media applications with privacy preservation will be treated seriously. In view of the fact that scale-invariant feature transform (SIFT) has been widely adopted in various fields, this paper is the first to target the importance of privacy-preserving SIFT (PPSIFT) and to address the problem of secure SIFT feature extraction and representation in the encrypted domain. As all of the operations in SIFT must be moved to the encrypted domain, we propose a privacy-preserving realization of the SIFT method based on homomorphic encryption. We show through the security analysis based on the discrete logarithm problem and RSA that PPSIFT is secure against ciphertext only attack and known plaintext attack. Experimental results obtained from different case studies demonstrate that the proposed homomorphic encryption-based privacy-preserving SIFT performs comparably to the original SIFT and that our method is useful in SIFT-based privacy-preserving applications.**

***Index Terms*— Feature extraction, homomorphic encryption, privacy preserving, security, scale-invariant feature transform (SIFT).**

I. Introduction

In this section, we introduce privacy-preserving query in Sec. I-A, describe the importance of privacy-preserving SIFT in Sec. I-B, and give the outline of this paper in Sec. I-C.

A. Privacy-Preserving Query

Recently, people have gotten used to accessing and querying multimedia data on a server due to the increase of bandwidth capacity over the Internet. In addition, if the remote server has strong computation/storage capability with abundant resources, the users can store their data on the server side and exploit the computational power provided by the server to execute their intended tasks. In this circumstance, the Web not only provides a passive search service but also is equipped with a highly interactive mechanism. This scenario is analogous to cloud computing and is of practical use for multimedia data that demand immense computation and communication. Under this kind of framework, the transmission of personal data and permission of the server in accessing the stored data pose the issue of privacy-preserving that is usually ignored in the multimedia community.

Although encryption is a prevalent method of securing transmitted data, the data in the encrypted form (*i.e.*, ciphertext) will impede operations that are usually conducted on the plaintexts. In order to further process ciphertexts and obtain the corresponding results in the plaintext domain, some studies have been devoted to several aspects of encrypted domain operations.

Only recently, secure text document search in the encrypted domain [1], [2] has been extended to secure multimedia data search [3], [4]. While the aforementioned studies have been done on content-based multimedia retrieval over either encrypted query, or both encrypted query and database, the prevailing scale-invariant feature transform (SIFT) [5] conducted in the encrypted domain still has not been addressed. In what follows, we will target the importance of privacy-preserving SIFT (PPSIFT) and explore its broad applications.

B. Importance of Privacy-Preserving SIFT and Our Contributions

SIFT [5] is an algorithm for detecting and describing local features in images, and it has been widely used in the community of computer vision and pattern recognition due to its powerful attack-resilient feature point detection mechanism. In addition to maintaining the "robustness" of SIFT, in this paper, we will explore a homomorphic encryption-based secure SIFT methodology, called privacy-preserving SIFT (PPSIFT). In PPSIFT, privacy-preserving feature extraction and representation addresses the issue of extracting and representing media features in the encrypted domain while allowing exhibition of inherent properties in the plaintext/un-encrypted domain. Particularly, both the query and database are permitted to be encrypted to guarantee privacy-preserving as a whole. This core technology, enabling SIFT to simultaneously possess both robustness and security, will find many applications, including media retrieval [3], [6], [7], media authentication [8], face identification [9], face recognition [10]–[13], fingerprint verification [14], and video-based mobile location search [15], for the purpose of preserving privacy. More specifically, we have the following observations from the aforementioned existing

Manuscript received November 23, 2011; revised May 24, 2012; accepted May 30, 2012. Date of publication June 12, 2012; date of current version October 12, 2012. This work was supported by the National Science Council, Taiwan, under Grant NSC 98-2631-H-001-013 and Grant NSC 99-2631-H-001-020. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Hitoshi Kiya.

C.-Y. Hsu is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan, and also with the Graduate Institute of Communication Engineering, National Taiwan University, Taipei 106, Taiwan.

C.-S. Lu is with the Institute of Information Science, Academia Sinica, Taipei 115, Taiwan (e-mail: lcs@iis.sinica.edu.tw).

S.-C. Pei is with the Graduate Institute of Communication Engineering, National Taiwan University, Taipei 106, Taiwan.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2012.2204272

works: 1) privacy-preserving media retrieval [3], [16], and privacy-preserving face recognition and identification [9], [10], [12] have been studied; 2) the robustness of Roy and Sun's image authentication scheme [8] can be enhanced via a secure SIFT strategy [17] in order not to be defeated; and 3) several SIFT-based methods and applications [11], [13]–[15] potentially need privacy protection if privacy leakage is a concern.

The contributions of this paper in realizing privacy-preserving SIFT are summarized as follows. To the best of our knowledge, this work is among the first endeavors on the SIFT algorithm in the encrypted domain.

1) The Difference-of-Gaussian (DoG) transform must be executed in the encrypted domain. We investigate how DoG transform can be performed within the Paillier cryptosystem [18], which is associated with an error probability analysis. Our implementation of DoG in the encrypted domain is analog to implementations of DCT [19] and DWT [20] in the encrypted domain.
2) We present a homomorphic comparison strategy that can be conducted in the encrypted domain so that local extrema can be securely detected for SIFT feature point extraction.
3) PPSIFT is able to achieve local extrema extraction, descriptor calculation, and descriptor matching, all in the encrypted domain, without multiple rounds of communication between the user and server. On the contrary, only one-round of pre-communication is necessary for synchronization of data.
4) PPSIFT has been evaluated to find its superiority in attaining both privacy and robustness under benchmark attacks and datasets, when compared with the original SIFT.
5) Security analysis via the discrete logarithm problem and RSA is studied to show that PPSIFT is indeed secure against ciphertext only attack and known plaintext attack.
6) Our method can adapt to other feature detectors if the underlying operations for feature extraction are restricted to those considered in this paper. In fact, the operations of homomorphic addition, plaintext multiplication, and homomorphic comparison in the encrypted domain discussed here have already been covered in known feature detectors, like Speeded Up Robust Features (SURF) [21], Harris detector, and Hessian detector.

C. Organization of This Paper

The remainder of this paper is organized as follows. In Sec. II, we define the problem we would like to solve. In Sec. III, the operations on the encrypted domain are introduced, along with presenting a cryptosystem that is appropriate for the design of secure SIFT in a privacy-preserving manner. In Sec. IV, the proposed privacy-preserving SIFT method is described. In Sec. V, we provide formal security analysis for the proposed method. Experiments, including case studies, are presented to verify the usefulness of our method in Sec. VI. Finally, conclusions and future work are given in Sec. VII.

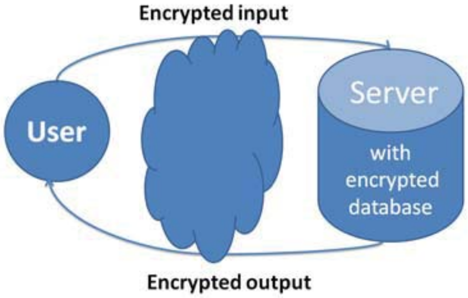


Fig. 1. Nonsymmetric query-response model operating in an cloudlike environment.

II. Problem Definition

For a multimedia query system with preservation of the user's privacy, a non-symmetric[1] query scenario in a cloud, an example of which is shown in Fig. 1, is considered in this paper. In Fig. 1, the user with scarce resources can send encrypted data as a query to the server, which possesses abundant resources and powerful computational capability, and can use the received encrypted data to finish the intended tasks (*e.g.*, feature extraction and matching). Since the user will rely on the remote but capable server, (s)he simply encrypts the data for the purpose of privacy and sends the encrypted data to the server for storage in advance. For the scenario considered in this paper, the server is assumed to be honest but curious. Thus, the server must learn nothing about either the query sent by the user or the results derived from the query in order to satisfy the purpose of privacy preserving. In other words, the server is powerful in finishing the requested tasks and sends the encrypted outputs back into the user but cannot infer what has been obtained from the information that can be available. When the user receives the encrypted outputs, they can be decrypted back to the plaintext domain. Although a query model is employed to explain the problem to be solved in this paper, our method can be broadly applied to other problems, as later described in Sec. VI.

Motivated by earlier works [3], [9], [10], [12] where privacy-preserving applications of media data have received attention recently, in this paper, we shall take digital images as the instance to describe the proposed homomorphic encryption-based secure SIFT method conducted in a privacy-preserving manner.

In our model, the user only prepares a homomorphically encrypted copy of the query image as the encrypted input, which then is sent to the server for subsequent processing. The problem here is that the server is responsible for generating the SIFT features via the framework of homomorphic encryption without knowing/learning anything to breach the user's privacy. More specifically, in order to finish SIFT in the encrypted domain, in addition to common homomorphic addition and plaintext multiplication, we observe that *homomorphic comparison* is also required. Nevertheless, the extremely challenging issue, which is a major concern of

[1] Non-symmetric here means that the user is only responsible for feeding the query and some parameters to the server but the server is capable of finishing all of the tasks for the query.

this paper, is how to accomplish comparative homomorphism securely.

It should be noted that it is not suitable to employ the framework of secure multiparty computation (SMC) [22] to solve this problem since SMC may need several rounds of interaction between the user and the server. In addition, the multiple parties in SMC can be said to possess equivalent capability. On the contrary, for the scenario (Fig. 1) considered here, the user heavily relies on the capable and powerful server to finish almost all tasks. It will be clear later that the proposed method only needs one-round of pre-communication for necessary synchronization of data when the query is initiated. Based on the received query image in the form of ciphertexts, the server carries out DoG transform, SIFT feature point extraction, feature descriptor extraction, and descriptor matching to accomplish the designated tasks in the encrypted domain and sends the encrypted outputs to the user, who will finally get the results in the plaintext domain via decryption.

III. Operations in the Encrypted Domain

In this section, we will first briefly review our previous work [17] that proposes a method of detecting SIFT features from encrypted images. Then, we will introduce the Paillier cryptosystem[2] [18], which enables one to directly operate in the ciphertext domain but can obtain the equivalent results in the plaintext domain. The goal of this section is to provide some preliminaries that motivate the study of this paper and to make this paper self-contained.

A. SIFT in an Encrypted Domain

In [17], we present two anti-SIFT attacks that can efficiently remove the feature points retrieved by conventional SIFT [5]. The idea comes from the observation that a pixel is decided as a SIFT keypoint if and only if it is a local extremum in the scale space defined by Difference-of-Gaussian (DoG) functions. As a result, an original keypoint will not be detected by SIFT if another extremum is maliciously generated nearby. In other words, there can be at least two equal extrema in a detection region such that the duplicate extremum is forced to be at one of the eight neighbors of the true one in the scale space to evade keypoint detection.

In order to tackle this problem, we present a secret key-based transformation process, which is performed on images before SIFT feature detection, such that the dominant features become recessive. This implies that the detection of SIFT features will be conducted in the transformed (or encrypted) domain instead of the original spatial domain, and the goal of secure SIFT can be achieved. Such a secret key-based transformation can be linear or non-linear. The proposed strategy is simple and composed of two steps: bit reversing and local encryption. Basically, the bit reversing step is to make SIFT detection fail and become erroneous while local encryption aims to secure SIFT detection.

Nevertheless, as we mentioned in [17], a more sophisticated design regarding secure SIFT is possible. In this paper, we shall address this issue so that the performance of the proposed method can be validated in a cryptographically secure manner.

B. Paillier Cryptosystem

In order to execute SIFT in a ciphertext domain and still obtain results equivalent to those generated in the corresponding plaintext domain, the prerequisite is to seek a cryptosystem that can provide the required operations, such as addition and multiplication. In the original SIFT, in addition to common additive and multiplicative operations, the comparison operation is a must for finishing feature point detection. Nevertheless, the design of a cryptosystem that can possess homomorphic comparison is still a challenging issue. Therefore, our goals are to seek a cryptosystem that can provide homomorphic addition and multiplication of plaintexts and to develop a new approach to achieve homomorphic comparison[3].

To achieve operations in the ciphertext domain and obtain results equivalent to those in the plaintext domain, homomorphic encryption [23] (and the references therein) has been widely investigated. We chose the Paillier cryptosystem [18] as the platform for designing our secure SIFT method because it provides additive homomorphism and plaintext multiplication, achieves provable security based on modular arithmetic, and is computationally comparable to RSA.

The operations of the Paillier cryptosystem are briefly described as follows. First, a pair of private and public keys are set. Let p and q be two large primes, and let $N = pq$. Let $Z_{N^2} = \{0, 1, \ldots, N^2 - 1\}$ and $Z^*_{N^2} \subset Z_{N^2}$ denote the set of non-negative integers that have multiplicative inverses modulo N^2. We also select $g \in Z^*_{N^2}$ to satisfy $gcd(L(g^{\lambda}\ mod\ N^2), N) = 1$, where λ defined as $\lambda = lcm(p-1, q-1)$ is the private key. The pair of N and g defines the public keys.

Second, the encryption phase is operated as follows. Let the message to be encrypted be denoted as $m \in Z_N$, which satisfies $m < N$. The ciphertext (corresponding to m) $\in Z_{N^2}$ is derived as:

$$c = E(m, r) = g^m r^N\ mod\ N^2 \tag{1}$$

where $r \in Z^*_N \subset \{0, 1, \ldots, N - 1\}$ denotes the uniformly chosen key and integer numbers modulo is employed. Since r is not fixed, the Paillier cryptosystem satisfies so-called "semantic security," which states that, for the same plaintext, different ciphertexts can be generated if r is changeable.

Third, for decrypting the ciphertext c, we use the private key λ and obtain the plaintext m as:

$$m = D(c, \lambda) = \frac{L(c^{\lambda}\ mod\ N^2)}{L(g^{\lambda}\ mod\ N^2)}\ mod\ N, \tag{2}$$

where $L(u) = \frac{u-1}{N}$.

[2] Nevertheless, it should be noted that our method does not tie to only one cryptosystem since any cryptosystems that possess at least additive homomorphism can meet our need.

[3] It should be noted that homomorphic comparison for SIFT feature detection needs to be accomplished alone on one party (*e.g.*, the server side of Fig. 1). Therefore, secure multiparty computation (SMC) [22] does not meet the goal of our paper.

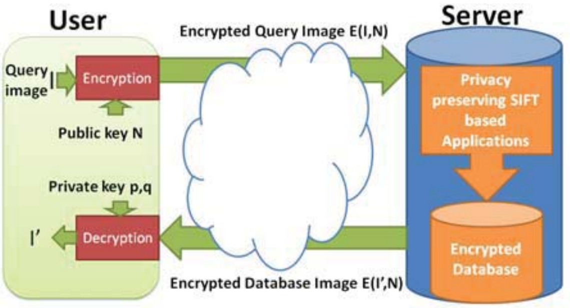

User
Server
Encrypted Query Image E(I,N)
Query image I
Encryption
Public key N
Private key p,q
I'
Decryption
Encrypted Database Image E(I',N)
Privacy preserving SIFT based Applications
Encrypted Database

Fig. 2. Framework of our privacy-preserving SIFT-based method and applications.

The Paillier cryptosystem is said to be homomorphically additive because:

$$\begin{aligned} c_1 \times c_2 &= E(m_1, r_1) \times E(m_2, r_2) \\ &= g^{(m_1+m_2)}(r_1 r_2)^N \; mod \; N^2. \end{aligned} \tag{3}$$

After decrypting the above result by $D(E(m_1, r_1) \times E(m_2, r_2), \lambda)$, we can get the plaintext $m_1 + m_2$, which is generated by executing multiplication in the ciphertext domain, as indicated in Eq. (3). Another form equivalent to Eq. (3) is expressed as:

$$\begin{aligned} c_1 \times g^{m_2} &= E(m_1, r_1) \times g^{m_2} \\ &= g^{(m_1+m_2)}(r_1)^N \; mod \; N^2 \end{aligned} \tag{4}$$

which can also be decrypted to get $m_1 + m_2$.

The Paillier cryptosystem provides plaintext multiplication based on additive homomorphism because:

$$D([E(m_1, r_1)]^{m_2} \; mod \; N^2) = (m_1 \times m_2) \; mod \; N. \tag{5}$$

The plaintext $m_1 \times m_2$ is equivalent to being generated by executing an exponentiation operation in the ciphertext domain.

The proposed PPSIFT is based on the Paillier cryptosystem, which needs to perform modular exponentiations of large numbers, and incurs a higher computational complexity (quantified by the number of modular multiplications). Strategies of speeding up modular exponentiation operation can be found in [24], [25], but they are not germane to the scope of the paper.

IV. PPSIFT: Secure SIFT in Homomorphic Encrypted Domain

In this section, we describe the proposed privacy-preserving SIFT (PPSIFT) method that is conducted in the Paillier cryptosystem. It should be noted that it is not straightforward at all to simply incorporate both SIFT and the Paillier cryptosystem directly. On the contrary, certain modifications, which are described in this section, must be introduced. Fig. 2 illustrates a framework of proposed privacy-preserving SIFT-based method and applications.

A. Difference of Gaussian in the Encrypted Domain

The first step of the SIFT framework for extracting the feature points is to execute Difference-of-Gaussian transforms. For this, the image is convolved with Gaussian filters, which are assigned different variances ρ_i's (corresponding to scales), and the differences between two neighboring Gaussian-blurred images are taken. Feature points are then chosen as local extrema of the DoG images, which occur at multiple scales. Specifically, a DoG image $DoGImg$ generated at two neighboring scales ρ_i and ρ_j is defined as:

$$DoGImg(x, y, \rho_{ij}) = Conv_G(x, y, \rho_i) - Conv_G(x, y, \rho_j) \tag{6}$$

where X and Y are, respectively, the horizontal and vertical sizes of the original image, I, $1 \leq x \leq X$ and $1 \leq y \leq Y$, and $Conv_G$ denotes the convolution of the image with the Gaussian kernel G at the i-th scale, *i.e.*,

$$\begin{aligned} Conv_G(x, y, \rho_i) &= G(u, v, \rho_i) * I(x, y) \\ &= \Sigma_{u,v} G(u, v, \rho_i) I(x-u, y-v) \quad \forall \, x \text{ and } y \end{aligned} \tag{7}$$

where $*$ denotes a convolution operation.

To preserve the users' privacy, the image I is encrypted using homomorphic encryption, as described in the previous section. The resultant encrypted data are expressed as:

$$I_e(x, y) = E(I(x, y), r) = g^{I(x,y)} r^N \; mod \; N^2 \quad \forall \, x \text{ and } y, \tag{8}$$

where $E()$ denotes the Paillier cryptosystem, as indicated in Eq. (1), and r is the uniformly chosen key. For practical implementation, the original Gaussian filter coefficients are adjusted as integers because the Paillier cryptosystem can only operate in the integer domain. For this, the integer DoG filter, $G_{Diff}(u, v, \rho_{ij})$, is derived as:

$$G_{Diff}(u, v, \rho_{ij}) = \lceil s(G(u, v, \rho_i) - G(u, v, \rho_j)) \rfloor \quad \forall \, u \text{ and } v, \tag{9}$$

where $\lceil \cdot \rfloor$ is a rounding function and s is a scaling factor used to enlarge Gaussian filter coefficients, $G()$'s, which are usually smaller than 1. It is worth noting that the proposed PPSIFT in this paper only introduces errors due to the rounding operation in Eq. (9). For the sake of notation simplification, we will simply use ρ in place of ρ_{ij} in the following if there is no confusion. Furthermore, when Gaussian kernel $G()$ is involved in the following discussion, its support will also be omitted.

By convolving the image to be encrypted with the DoG filters in the encrypted domain, the resultant encrypted image in the DoG domain can be derived for $1 \leq x \leq X$ and $1 \leq y \leq Y$ as:

$$\begin{aligned} DoGImg_e(x, y, \rho) &= E(G_{Diff}(x, y, \rho) * I(x, y), r), \\ &= E\left(\sum_{u,v} G_{Diff}(u, v, \rho) I(x-u, y-v), r\right) \\ &= \prod_{u,v} E(I(x-u, y-v), r)^{G_{Diff}(u,v,\rho)} \; mod \; N^2, \end{aligned} \tag{10}$$

where the last line is derived according to homomorphic addition and plaintext multiplication of the Paillier cryptosystem, respectively, shown in Eq. (3) and Eq. (5). Note that $DoGImg_e(x, y, \rho)$ is also interpreted as the encrypted difference between two Gaussian-blurred images at two neighboring scales. Furthermore, as mentioned in Eq. (9), a constant s is employed to enlarge the Gaussian filter coefficients and obtain $G_{Diff}(u, v, \rho)$. Now, it is clear that s cannot be too large since this will make the computational complexity of Eq. (10) intractable. On the other hand, if s is small enough, then $G_{Diff}(u, v, \rho)$ may still be truncated to zero, leading to severe performance degradation. In order to properly achieve the tradeoff between performance and computational complexity, an error probability model is theoretically derived in Appendix, experimentally verified in Sec. VI-A, and is used as the guideline to determine a proper s. Basically, our results show that if the error probability of deleting an original feature point or extra generating a new feature point would like to be as low as 10^{-6}, then s should be set to 2^{24}.

B. PPSIFT Feature Point Detection: Local Extrema Extraction via Encrypted Data Comparison

The most challenging task of PPSIFT is the local extrema extraction operating in the encrypted domain. As we introduced in Sec. III-B, the Paillier cryptosystem only provides additive homomorphism. Nevertheless, SIFT feature detection still needs homomorphic comparison. In this section, we investigate a homomorphic comparison strategy in the Paillier cryptosystem for encrypted data comparison.

In the Paillier cryptosystem, the uniformly chosen key r in Eq. (1) must be variable to satisfy semantic security. Under this circumstance, given the plaintext m_i, the resultant ciphertexts c_i's will be different according to the used user keys r_i's, leading to one-to-many mapping.

In the two-dimensional case, like the images considered here, the uniformly chosen key $r_{x,y}$, dependent on the location of a pixel, is used. Hence, a DoG image in the encrypted domain using different $r_{x,y}$'s can be derived as:

$$\begin{aligned} DoGImg_e(x, y, \rho) &= E(G_{Diff}(x, y, \rho) * I(x, y), r_{x,y}), \\ &= E\left(\sum_{u,v} G_{Diff}(x, y, \rho) I(x-u, y-v), r_{x-u,y-v}\right) \\ &= \prod_{u,v} E(I(x-u, y-v), r_{x-u,y-v})^{G_{Diff}(u,v,\rho)} \ mod\ N^2. \end{aligned} \tag{11}$$

It can be observed that Eq. (11) is generated using homomorphic addition and plaintext multiplication. Substituting Eq. (1) into Eq. (11), we have:

$$\begin{aligned} &DoGImg_e(x, y, \rho) \\ &= \prod_{u,v} E(I(x-u, y-v), r_{x-u,y-v})^{G_{Diff}(u,v,\rho)} \ mod\ N^2 \\ &= \prod_{u,v} g^{I(x-u,y-v)G_{Diff}(u,v,\rho)} r_{x-u,y-v}^{NG_{Diff}(u,v,\rho)} \ mod\ N^2 \\ &= g^{\sum_{u,v} I(x-u,y-v)G_{Diff}(u,v,\rho)} \left(\prod_{u,v} r_{x-u,y-v}^{G_{Diff}(u,v,\rho)}\right)^N \ mod\ N^2 \\ &= E\left(\sum_{u,v} I(x-u, y-v) G_{Diff}(u, v, \rho), \prod_{u,v} r_{x-u,y-v}^{G_{Diff}(u,v,\rho)}\right) \\ &= E(DoGImg(x, y, \rho), R_\rho), \end{aligned} \tag{12}$$

where a pixel $DoGImg(x, y, \rho)$ is encrypted using a combined uniformly chosen key R_ρ, which is expressed as

$$R_\rho = \prod_{u,v} r_{x-u,y-v}^{G_{Diff}(u,v,\rho)} \tag{13}$$

which is a function of the uniformly chosen key $r_{x,y}$ that is dependent on a pixel's location (x, y). Since the Gaussian kernel $G()$ is involved in the calculation of R_ρ, we know that R_ρ depends on the support of $G()$ instead of the image size, as we have mentioned in Sec. IV-A.

Comparing Eq. (11) and Eq. (12), we know that the result obtained from the scenario that the user provides encrypted data $E(I(x, y), r_{x,y})$ to the server for executing DoG in the ciphertext domain is equivalent to that obtained from directly encrypting DoG image using R_ρ at the scale ρ. Similarly, a unique characteristic is that the server does not have access to $r_{x,y}$'s and their combined one R_ρ.

1) Homomorphic Comparison: Due to the use of different r's, the resultant ciphertexts, as indicated in Eq. (12), will fall into the range between 0 and N^2-1. We propose quantization-like homomorphic comparison of ciphertexts to equivalently achieve local extrema extraction in the plaintext domain. In our method, a series of thresholds in the ciphertext domain, which will divide the ciphertext domain located between 0 and $N^2 - 1$ into several (non-uniform) quantization intervals, are designed.

For this, the user will generate a series of thresholds T_i's in the plaintext domain, where $T_i \in Z_N$, and these thresholds will be encrypted and sent to the remote server for the purpose of homomorphic comparison. Since comparison will be conducted in the encrypted domain, these thresholds are encrypted via Paillier encryption using R_ρ at scale ρ as:

$$T_{i,\rho}^e = E(T_i, R_\rho) = g^{T_i} R_\rho^N \ mod\ N^2, \tag{14}$$

where $T_{i,\rho}^e \in Z_{N^2}$. Note that R_ρ is employed by users to encrypt T_i's because the ciphertexts used for homomorphic comparisons are also encrypted using R_ρ, as indicated in Eq. (12).

In the proposed method, in addition to the encrypted query data, the additional data needed to be sent to server for subsequent privacy-preserving processing are the secure thresholds $T_{i,\rho}^e$'s and their order. Note that the calculation of R_ρ needs $G_{Diff}(u, v, \rho)$, which will be sent from the server to the user. Such pre-computation will only be executed once during the course of the query system when a user initiates his/her query task. Fig. 3 illustrates the transmission of parameters between the user and server. Nevertheless, we also note that some homomorphic comparison algorithms, like [26], employ the framework of secure multiparty computation and need a few rounds of communication.

Now, the strategy for comparison between two elements in $DoGImg_e$ in the encrypted domain will be described as follows. Basically, the principle is to compare two encrypted data according to their locations in the intervals separated by the thresholds $T_{i,\rho}^e$'s. The homomorphic addition property, as indicated in Eq. (4), is exploited. Given two ciphertexts, $E(DoGImg(x_1, y_1, \rho_1), R_{\rho_1})$ and $E(DoGImg(x_2, y_2, \rho_2), R_{\rho_2})$, the goal is to compare them

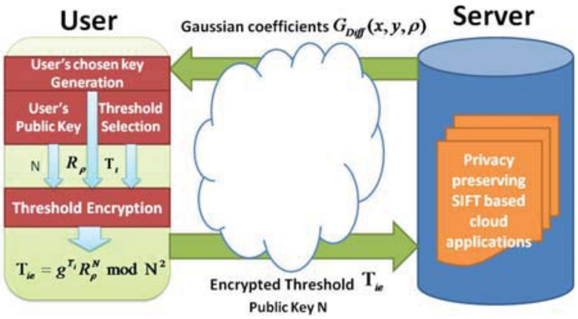

User
Server
Gaussian coefficients $G_{Diff}(x, y, \rho)$
User's chosen key Generation
User's Public Key
Threshold Selection
N
R_ρ
T_i
Threshold Encryption
$T_{ie} = g^{T_i} R_\rho^N \bmod N^2$
Privacy preserving SIFT based cloud applications
Encrypted Threshold T_{ie}
Public Key N

Fig. 3. Parameter communication between the user and server.

in the encrypted domain and finally find their magnitude relationship in the plaintext. This goal can be accomplished by identifying which quantization intervals the two ciphertexts fall into via Paillier homomorphic addition as:

$$a_k = \underbrace{argmin_{Inc}}_{\forall i}(E(DoGImg(x_k, y_k, \rho), R_\rho)g^{Inc} - E(T_i, R_\rho)) \tag{15}$$

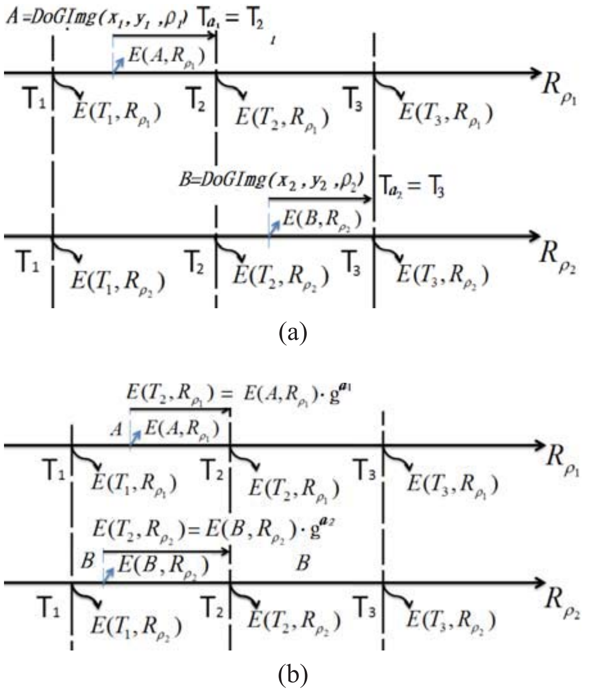
where $k = 1, 2$. In Eq. (15), g^{Inc} appears for additive homomorphism but, in fact, should be $E(Inc, 1) = g^{Inc}r^N \ mod \ N^2$ with $r = 1$ in order not to change the combined uniformly chosen key, as indicated in Eqs. (3) and (4).

Mathematically, Eq. (15) implies that the plaintext $DoGImg(x_k, y_k, \rho)$ is incrementally increased by Inc in the plaintext domain until it is finally increased to be equal to the nearest threshold T_i, which corresponds to $E(T_i, R_\rho)$ in the ciphertext domain for certain i. Here, the increment Inc is set to 1. By doing so, once two different thresholds $E(T_{a_k}, R_{\rho_k})$'s are found, in this case for $k = 1, 2$, the server can easily determine the magnitude relationship between the two ciphertexts, $E(DoGImg(x_1, y_1, \rho_1), R_{\rho_1})$ and $E(DoGImg(x_2, y_2, \rho_2), R_{\rho_2})$, since it receives the order of encrypted thresholds sent from the user. This case is illustrated in Fig. 4(a).

On the other hand, if $T_{a_1} = T_{a_2}$ is found, then the magnitude relationship between the two ciphertexts, $E(DoGImg(x_1, y_1, \rho_1), R_{\rho_1})$ and $E(DoGImg(x_2, y_2, \rho_2), R_{\rho_2})$, can still be determined by checking the magnitude relationship between a_1 and a_2. For example, as illustrated in Fig 4(b), if $a_1 > a_2$, then $E(DoGImg(x_1, y_1, \rho_1), R_{\rho_1}) < E(DoGImg(x_2, y_2, \rho_2), R_{\rho_2})$ since $E(DoGImg(x_1, y_1, \rho_1), R_{\rho_1})$ is more distant from $T^e_{i,\rho}$ for certain i.

Thus, according to this proposed homomorphic comparison strategy, the SIFT feature detection conventionally done in the plaintext domain can now be finished in the ciphertext domain without revealing the original image data. Nothing can be learned from the proposed homomorphic comparison strategy since all the operations are performed under the homomorphic addition of the Paillier cryptosystem. The security of our proposed homomorphic comparison scheme will be formally analyzed in Sec. V.

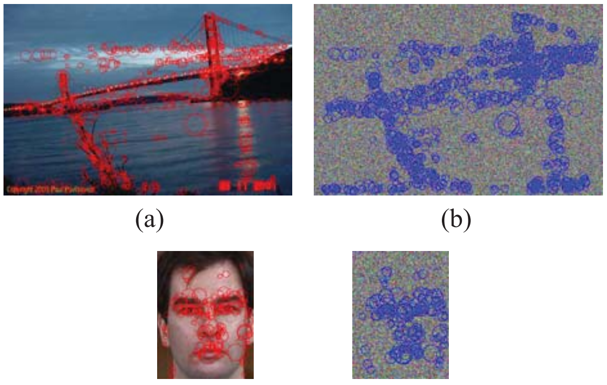
Fig. 5 illustrates a result of SIFT feature point extraction in the plaintext and ciphertext domains, respectively. In the remainder of this paper, all encrypted images are illustrated with the encrypted pixel value being normalized within [0, 255]. Each circle represents the region where a feature point resides. Large/small circles correspond to the SIFT feature points detected at the coarse/fine scales. To show the consistency of feature point detection between the original SIFT and our proposed PPSIFT, the regions of detected features are overlapped, as shown in Fig. 6. Visually, the detected locations (labeled in blue and red circles) of feature points look similar. Nevertheless, there are dissimilarities sometimes due to the effect of rounding errors introduced in Eq. (9). More advanced evaluations will be elaborated in Sec. VI.



(a)

(b)

Fig. 4. Illustration of the proposed homomorphic comparison. (a) Two ciphertexts located at different intervals. (b) Two ciphertexts located at the same interval.



(a) (b)

(c) (d)

Fig. 5. (a) and (c) Detection of SIFT features in the plaintext domain. (b) and (d) Ciphertext domain. (Best viewed on a color display).

Please also note that, in the proposed scheme, the locations of pixels are not encrypted, so the locations of SIFT features are known to the public. Such a characteristic is significantly different from our previous work [17], which aims to hide the locations of SIFT features in order to escape from being

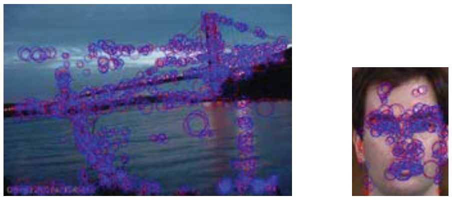


Fig. 6. Difference of detected features between original SIFT and proposed PPSIFT. Red circles represent the features of original SIFT, and blue circles represent the features of PPSIFT. (Best viewed on a color display).

maliciously tampered with. Nevertheless, for the scenario of privacy-preserving considered here, the locations of feature points will not breach the privacy because their corresponding feature descriptors (to be described later) are still in the encrypted form[4] without revealing any private data. One may argue that if the shape formed from the locations of SIFT points is directly used as the query, then conventional shape-based image retrieval can be conducted to find a set of retrieval results. However, even some private data may be revealed in this set roughly, they are not known precisely.

2) Impact of the Number of Thresholds T_i's and Their Pairwise Distances: As described in the previous subsection, the quantization-like homomorphic comparison strategy needs a series of thresholds T_i's. It is interesting to investigate the impact of the number of thresholds and their pairwise distances on the accuracy and security of homomorphic comparison.

First, we note that the different pairwise distances between a pair of thresholds will not affect the accuracy of comparison since both the magnitude relationship between $E(T_1, R_\rho)$ and $E(T_2, R_\rho)$ and between a_1 and a_2 can cooperatively finish homomorphic comparison. In the following, uniform quantization is used.

Second, we examine the impact of the number of thresholds upon the speed of computation and security. If the number of thresholds is large, meaning that the quantization interval is small, then the computation of homomorphic comparison indicated in Eq. (15) can be quicker since a_k can be found fast, but at the expense of spending large communication cost in transmitting these thresholds from the user to the server. Thus, there exists a tradeoff between communication cost and computation overhead in homomorphic comparison. If the server is considered to be resource-abundant, it is, however, preferable to use a limited number of thresholds.

3) Communication Overhead: In addition to the encrypted query data, the extra data needed to be sent to server for subsequent privacy-preserving processing are the secure thresholds $T^e_{i,\rho}$'s (Eq. (14)) and their order. Note that the calculation of R_ρ (Eq. (13)) needs $G_{Diff}(x, y, \rho)$, which will be sent from the server to the user, and what is more, such a pre-computation will only be executed once when a user initiates his/her query task. It should be noted that the aforementioned transmitted data are all integers since these integer data are used in the Paillier cryptosystem despite the difference in integer length. One finds that the overhead for transmitting the data is dominated by $log_2 N^2$ bits due to the use of Paillier encryption. Thus, the communication overhead is $O(log_2 N)$.

C. PPSIFT Feature Point Descriptor in Encrypted Domain

In this section, we describe how to derive SIFT feature descriptors in the plaintext domain, which is then extended to the ciphertext domain. First, as done in [5], an orientation assignment is executed for each detected feature point. Then, a normalized 16×16 region expanded from the region covering the derived orientation is built from which feature descriptors are obtained as follows.

An SIFT feature descriptor is established for the 16×16 region, which is further divided into sixteen 4×4 blocks, around a feature point. In addition, the calculation of the feature descriptor is accomplished at the scale where the feature is detected. Let the gradient magnitudes be denoted as $Diff_X = Conv_G(x+1, y, \rho) - Conv_G(x-1, y, \rho)$ and $Diff_Y = Conv_G(x, y+1, \rho) - Conv_G(x, y-1, \rho)$ along different directions. For each 4×4 block, the gradient magnitude and orientation are, respectively, computed for each position (x, y) within the 4×4 block as:

$$m(x, y) = \sqrt{(Diff_X)^2 + (Diff_Y)^2}, \tag{16}$$

$$\theta(x, y) = tan^{-1}\frac{Diff_X}{Diff_Y}. \tag{17}$$

Then, the histogram of weighted magnitudes defined on a number of restrictive directions is derived based on Eqs. (16) and (17).

For feature descriptor extraction conducted in the encrypted domain, the weighted magnitudes located at the four axes (*i.e.*, positive and negative x-axes and positive and negative y-axes) are calculated in this paper, which will constitute a 4-dimensional vector. Since there are a total of sixteen 4×4 blocks in a 16×16 region, a 64-dimensional feature descriptor is established. It should be noted that no more than four restrictive directions are employed in this paper because the operation of the secure inner product[5] is required to derive the included angle with the two sides not both coinciding with the $x-$ and $y-$ axes.

In this paper, the feature descriptor calculated in the encrypted domain for each 4×4 block is conducted as follows. Let $V(k)$, $0 \leq k \leq 3$, denote the 4-dimensional feature descriptor of a 4×4 block, and let $E(V(k))$ denote the ciphertext corresponding to the plaintext $V(k)$. $E(V(k))$'s are all initialized to be 1. The encrypted feature descriptors are derived according to the above conceptions based on homomorphic addition and plaintext multiplication as:

$$\begin{aligned} E(V(0)) = {} & E(V(0))E(Conv_G(x+1, y, \rho)) \\ & \times E(Conv_G(x-1, y, \rho))^{-1} mod\ N^2, \\ & \text{if } Conv_G(x+1, y, \rho) \geq Conv_G(x-1, y, \rho) \end{aligned}$$

[4] We have an interesting observation that, if the SIFT feature descriptors are not encrypted, the adversary can use them to query other databases so the originally encrypted content may be approximately guessed from the search outputs, leading to a privacy breach.

[5] In the field of secure computation, secure inner product computation without needing interaction between the user and server is another challenging issue that needs to be further studied.

$$\begin{aligned}E(V(1)) &= E(V(1))E(Conv_G(x, y+1, \rho))\\ &\quad\times E(Conv_G(x, y-1, \rho))^{-1} mod\ N^2,\\ &\quad\text{if } Conv_G(x, y+1, \rho) \geq Conv_G(x, y-1, \rho)\\ E(V(2)) &= E(V(2))E(Conv_G(x-1, y, \rho))\\ &\quad\times E(Conv_G(x+1, y, \rho))^{-1} mod\ N^2,\\ &\quad\text{if } Conv_G(x-1, y, \rho) \geq Conv_G(x+1, y, \rho)\\ E(V(3)) &= E(V(3))E(Conv_G(x, y-1, \rho))\\ &\quad\times E(Conv_G(x, y+1, \rho))^{-1} mod\ N^2,\\ &\quad\text{if } Conv_G(x, y-1, \rho) \geq Conv_G(x, y+1, \rho).\end{aligned}$$

It should be noted that the comparisons in the above equations also need to be executed in the encrypted domain via the proposed homomorphic comparison strategy.

D. PPSIFT Feature Descriptor Matching in the Encrypted Domain

The descriptor matching stage aims to compare a query descriptor with a reference descriptor for similarity evaluation via a similarity metric. Let the similarity between two descriptors, V^i and V^j, be denoted as $Sim(V^i, V^j)$. The inner product between the two descriptors is commonly used as a similarity metric, which is expressed in the plaintext domain as:

$$Sim^p_{IP}(V^i, V^j) = \sum_{k=0}^{63} V^i(k)V^j(k). \tag{18}$$

Actually, the calculation of Eq. (18) in the plaintext domain should involve the normalization factors, *i.e.*, the norms of V^i and V^j. However, they are omitted here for simplifying notations. Another reason is that the use of inner product in the encrypted domain suffers a difficulty, as discussed below. To meet the concern of privacy protection, the above similarity measure must be computed in the ciphertext domain while obtaining the same result as in the plaintext domain. Thus, Eq. (18) can be rewritten in the homomorphic encryption domain as:

$$Sim^c_{IP}(E(V^i), E(V^j)) = \prod_{k=0}^{63} E(V^i(k))^{V^j(k)}\ mod\ N^2 \tag{19}$$

to achieve the desired goal. It is not hard to derive from Eq. (19) that the same similarity measure as in Eq. (18) can be obtained by means of homomorphic addition (Eq. (3)) and plaintext multiplication (Eq. (5)).

Unfortunately, it is not possible for the server to access the plaintexts $V^j(k)$'s used in Eq. (19) due to the user's privacy protection. To conquer this problem, we adopt the ℓ_1 distance metric instead, since the calculation of ℓ_1 distance can be conducted in a secure way. More specifically, the ℓ_1 distance between two descriptors in the plaintext domain is defined as:

$$Sim^p_{\ell_1}(V^i, V^j) = |V^i - V^j|_1 = \sum_{k=0}^{63} |V^i(k) - V^j(k)|. \tag{20}$$

For ℓ_1 distance between two descriptors in the ciphertext domain, we can derive via homomorphic encryption as:

$$\begin{aligned}&Sim^c_{\ell_1}(E(V^i), E(V^j)) = E(|V^i - V^j|_1)\\ &= \prod_{k \in \{t | V^i(t) > V^j(t);\ 0 \leq t \leq 63\}} E(V^i(k))E(V^j(k))^{-1}\\ &\quad\times \prod_{k \in \{t | V^i(t) \leq V^j(t);\ 0 \leq t \leq 63\}} E(V^i(k))^{-1}E(V^j(k))\ mod\ N^2.\end{aligned} \tag{21}$$

Eq. (21) is derived based on our proposed homomorphic comparison strategy, described in Sec. IV-B.1, and the additive homomorphism of the Paillier cryptosystem. Specifically, the larger of the two ciphertexts, $E(V^i(k))$ and $E(V^j(k))$, is selected as the minuend and the small one is subtrahend via the proposed homomorphic comparison. Therefore, the absolute value of the difference between two ciphertexts can be directly calculated via homomorphic addition to be positive.

V. Security Analysis

In this section, we will provide security analysis for the proposed method. Since the Paillier cryptosystem, whose security has been investigated in [18], is employed in this paper, we will only discuss the possible security threat raised by proposed method. On the other hand, since there may exist several means to attack our method, an efficient way to analyze the security of our method is to consider the attack models instead of several single attacks. Here, we will mainly take the models of ciphertext only attack (COA) and known plaintext attack (KPA) into account.

In order to achieve the goal of data comparison in the Paillier encrypted domain, the randomness of Paillier's encryption oracle has been reduced by means of limiting the random factor "r" of encryption oracle as a pseudo-random factor R_ρ in our method. One may argue that our cryptosystem with the pseudo random factor R_ρ is not as secure as the original Paillier cryptosystem. On the other hand, the thresholds for encrypted data comparison we provide in Sec. IV-B may reveal some information to adversary. We will address these two issues in this subsection.

As mentioned earlier, we focus our security analysis on two attack models, namely, ciphertext only attack (COA) and known plaintext attack (KPA). In addition, two different KPA models are proposed with respect to the disclosure of T_i's and the removal of R_ρ. In each attack model, we assume that the adversary is an untrustworthy service provider or administrator of a server, *i.e.*, the adversary will follow the execution requirement of the protocol but may use what they see during the period of execution to infer additional information, such as the threshold table for homomorphic comparison.

A. Ciphertext Only Attack

In the ciphertext only attack model, the adversary can access to the ciphertext, which is the encrypted data available at the server, or access the threshold table used for encrypted data comparison. We first discuss the potential information leakage and then prove that the threshold table revealed to the adversary is as hard as the discrete logarithm problem (DLP) under the COA model.

The proposed encrypted data comparison strategy should be able to prevent the adversary from learning: (1) the plaintext versions of the database images; (2) the secret keys p and q used for decryption; and (3) the threshold table we provide for encrypted data comparison. For the first two concerns, the Paillier cryptosystem has been proven to be secure in [18]. Hence, we merely focus on the third problem here.

As we have described in Sec. IV-B, the encrypted threshold table is used for encrypted data comparison and the threshold values T_i's are private to protect the encrypted data. If the plaintext of the encrypted threshold table can be extracted, the threshold table can be used to decrypt the encrypted data. We will study if this is possible. To defeat our encrypted data comparison strategy, the adversary will input an encrypted DoG pixel $DoGImg_e(x, y, \rho)$ (Eq. (11)) to find the nearest encrypted threshold of the input DoG pixel via encrypted data comparison. Since the encrypted DoG pixel and its corresponding encrypted threshold have the same pseudo random factor R_ρ, the adversary may try to remove this randomness from the encrypted data via a procedure given as follows.

Given encrypted data $DoGImg_e(x, y, \rho)$, our encrypted data comparison strategy can find the nearest encrypted threshold $T^e_{i,\rho}$ (in this subsection, we need to keep in mind that an encrypted threshold $T^e_{i,\rho}$ is related to $DoGImg_e(x, y, \rho)$ at the position (x, y); different $DoGImg_e(x, y, \rho)$'s, of course, may correspond to the same or different $T^e_{i,\rho}$'s.). Then, the adversary can calculate the inverse value, $(T^e_{i,\rho})^{-1}$, of each obtained encrypted threshold, $T^e_{i,\rho}$, by means of multiplicative modular inverse in the modular domain. Ideally, $(T^e_{i,\rho})^{-1}$ can be derived to be $g^{-T_i} R_\rho^{-N} \ mod \ N^2$. Under this circumstance, the randomness term R_ρ^N of encrypted data can be removed via:

$$\begin{aligned} &DoGImg_e(x, y, \rho)(T^e_{i,\rho})^{-1} \ mod \ N^2 \\ &\quad = g^{DoGImg(x,y,\rho)} R_\rho^N g^{-T_i} R_\rho^{-N} \ mod \ N^2 \\ &\quad = g^{DoGImg(x,y,\rho)-T_i} \ mod \ N^2. \end{aligned} \tag{22}$$

Subsequently, the adversary will try to obtain the value $DT(x, y, \rho) = DoGImg(x, y, \rho) - T_i$ from Eq. (22) by solving a discrete logarithm problem (DLP).

If the DLP is solvable, $DoGImg(x, y, \rho) - T_i$ can be finally used to calculate the plaintext threshold T_i by means of solving a set of simultaneous equations as:

$$DT(x, y, \rho) = DoGImg(x, y, \rho) - T_i \tag{23}$$

where $1 \leq x \leq X$, $1 \leq y \leq Y$, $1 \leq \rho \leq Z - 1$, X and Y are, respectively, the horizontal and vertical sizes of an image, and Z denotes the number of scales used in DoG. No matter whether the set of equations in Eq. (23) forms an underdeterministic problem or not, the first crux is whether the discrete logarithm problem can be solved in a computationally efficient manner. We will formulate the security of our method as a discrete logarithm problem (DLP) according to [27] in the following.

With the variables defined in Sec. III-B, let $DoGImg(x, y, \rho) - T_i$ and $DoGImg_e(x, y, \rho)(T^e_{i,\rho})^{-1} \ mod \ N^2$ in Eq. (22) be, respectively, denoted by m and h. We have the following formulation.

Definition 1: (Security of PPSIFT as DLP) Let $g, h \in Z^*_{N^2}$. We write $\log_g(h) = m$ if $m \in Z_{N^2}$ satisfies $g^m = h$. The problem of finding such an integer m for a given $g, h \in Z^*_{N^2}$ (with $g \neq 1$) belongs to a Discrete Log Problem. No polynomial time algorithm exists to solve the Discrete Log Problem.

To sum up, the DLP is a critical problem in number theory [28]–[30] and is similar in many ways to the integer factorization problem. Under the COA model, where the adversary is assumed to have the information of the ciphertext and encrypted thresholds, the adversary will not be able to infer more information about plaintext, thus, privacy is preserved under such a COA model in our method.

B. Known Plaintext Attack

Different from the COA model, an adversary is assumed to know a number of pairs of plaintext images and their corresponding encrypted versions in a KPA model. The proposed homomorphic comparison scheme should be able to prevent the adversary from learning two kinds of major information, T_i's and R_ρ, under the KPA model. In view of this, we propose two types of the KPA model, KPA2T and KPA2R, in this subsection.

1) Disclosure of the Threshold T_i's Under the KPA Model (KPA2T): One immediate consequence of the information available to the adversary under the KPA2T model is that the adversary may be able to derive the plaintexts of encrypted thresholds from the relationship between $E(I(x, y), r)$ and the encrypted thresholds, as indicated in Eq. (22) and Eq. (23). Different from the COA model, the adversary under the KPA model can have more information available to easily solve Eq. (23). Nevertheless, similar to the COA model, the crux remains because all of the adversaries have to solve DLP first. As a consequence, the breach of our homomorphic comparison strategy under the KPA2T model is still as hard as breaking the DLP.

2) Removal of the Pseudo Random Factor R_ρ Under the KPA Model (KPA2R): The pseudo random factor R_ρ used in our homomorphic comparison strategy may be estimated and adopted to breach the security of our method. Specifically, R_ρ can be found by the adversary to construct a fake encrypted threshold table, which can be used to decrypt any encrypted data. In KPA2R, the security of our encrypted data comparison scheme is as difficult as extracting R_ρ.

The procedures of KPA2R is described as follows. First, the adversary can access some pairs of plaintexts and corresponding ciphertexts, $[DoGImg(x, y, \rho), E(DoGImg(x, y, \rho), R_\rho)]$, and also know the public key g and N. Then, the adversary will try to recover the random factor R_ρ via Paillier encryption defined in Eq. (1). One may try to figure out R_ρ by multiplying the encrypted data with

TABLE I

THEORETICAL AND PRACTICAL ERROR PROBABILITIES UNDER VARIOUS SCALING FACTORS. EACH ENTRY REPRESENTS THE EXPONENT OF AN ERROR PROBABILITY UNDER THE BASE NUMBER 10

Scaling factor (s)	$s = 2^{12}$	$s = 2^{15}$	$s = 2^{18}$	$s = 2^{21}$	$s = 2^{24}$	$s = 2^{27}$	$s = 2^{30}$
Theoretical result (31)	−2.3664	−2.8793	−3.7810	−4.7351	−6.0931	−6.7305	−7.3806
mirflickr [33]	−2.4195	−3.0250	−3.8935	−4.6567	−6.1438	−6.8428	−7.4147
ImageNet [32]	−2.5012	−3.1613	−3.9231	−4.8707	−5.9921	−6.9123	−7.3912

TABLE II

ROBUSTNESS OF OUR SCHEME VERSUS STIRMARK 3.1. ATTACKS ARE DENOTED AS SPA: THE SIGNAL PROCESSING ATTACK, INCLUDING MEDIAN FILTERING, GAUSSIAN FILTERING, SHARPENING, AND FREQUENCY MODE LAPLACIAN REMOVAL (FMLR); JPEG: COMPRESSION WITH QUALITY FACTORS RANGING FROM 0.9 TO 0.1; GLGT: GENERAL LINEAR GEOMETRIC TRANSFORM; CAR: CHANGE OF THE ASPECT RATIO; LR: LINE REMOVAL; RC: ROTATION+ CROPPING; SCALING: SCALED WITH FACTORS RANGING FROM 0.5 TO 2.0; RRS: ROTATION+RESCALING; RB: RANDOM BENDING

Stirmark 3.1	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}
SPA(6)	6	6	6	6	6	6	6	6	6	6
JPEG(12)	12	12	12	12	12	12	12	12	12	12
GLGT(3)	3	3	3	3	3	3	3	3	3	3
CAR(8)	8	8	8	8	8	8	8	8	8	8
LR(5)	5	5	5	5	5	5	5	5	5	5
Cropping(9)	8	8	8	8	8	9	8	8	8	8
RC(16)	16	16	16	16	16	16	16	16	16	16
Scaling(6)	6	6	6	6	6	6	6	6	6	6
RRS(16)	16	16	16	16	16	16	16	16	16	16
Shearing(6)	6	6	6	6	6	6	6	6	6	6
RB(1)	1	1	1	1	1	1	1	1	1	1
Flipping(1)	0	0	0	0	0	0	0	0	0	0

$g^{-E(DoGImg(x,y,\rho)}$, that is:

$$\begin{aligned} E(DoGImg(x,y,\rho))g^{-DoGImg(x,y,\rho)} &\mod N^2 \\ = g^{DoGImg(x,y,\rho)}R_\rho^N g^{-m} &\mod N^2 \\ = R_\rho^N \mod N^2 \equiv c. & \qquad (24) \end{aligned}$$

One may observe that solving Eq. (24) is equivalent to solving the RSA problem [31]. Specifically, we have the following formulation.

Definition 2: (Estimating R_ρ as an RSA problem) Let p and q be primes and $N = pq$, as previously described in the paper. We have the formulation, $c \equiv m^e \ mod \ N^2$, in RSA, where e and N are the public keys. Comparing with our problem defined in Eq. (24), we have $m = R_\rho$ and $e = N$. Thus, it can be concluded that finding an integer m for given $e, \ N \in Z_N^*$ (with $e \neq 1$) is an RSA problem.

Based on Definition 2, the random factor R_ρ cannot be extracted to fake a threshold table. Thus, our homomorphic comparison strategy is confirmed to resist KPA2R.

VI. EXPERIMENTAL RESULTS

In this section, we first verify the relationship between the scaling factor s and the error probability, which is defined in Eq. (31), to find an appropriate scaling factor in Sec. VI-A. After the scaling factor was decided, three kinds of experiments were conducted to evaluate the performance of the proposed method. In Sec. VI-B, the robustness of our method against benchmark attacks will be demonstrated. The goal is to examine whether certain robustness is lost due to secure computation of SIFT features. In Sec. VI-C, we describe a case study on privacy-preserving image recognition, which relies on feature extraction. In Sec. VI-D, a case study on privacy-preserving face recognition is examined. The aim of both case studies is to verify whether comparable performance between original-SIFT and PPSIFT can still be obtained when all of the operations of PPSIFT are conducted in the encrypted domain.

In the experimental setup, 7 octaves [5] with 6 filters of different Gaussian variances between two neighboring octaves were used for implementing DoG. The two large prime numbers, p and q, could be selected to be 1000-bit as in [20] but they could be larger for enhancing security. $g \in Z_{N^2}^*$ can be arbitrarily selected and was set to be 20 here. For feature point detection via homomorphic comparison, ten thresholds T_i's ($\in Z_N$) were arbitrarily selected for reducing the communication cost on the user side but increasing the computation overhead on the server side.

For the experiments conducted here, we do not take any advanced feature representation and indexing/search strategy into consideration since they are not the focus of this paper.

TABLE III

ROBUSTNESS OF OUR SCHEME VERSUS STIRMARK 4.0. ATTACKS ARE DENOTED AS AFFINE: AFFINE TRANSFORMATION; CONVF: CONVOLUTION FILTERING; CROPPING: CROPPED TO (3/4), (1/2), (1/4), AND (1/5) THE ORIGINAL SIZE; JPEG: COMPRESSION WITH QUALITY FACTORS RANGING FROM 0.9 TO 0.1; MF: MEDIAN FILTERING; NOISE: NOISE ADDITION; SS: SELF-SIMILARITIES; SCALING: SCALED WITH FACTORS RANGING FROM 0.5 TO 2.0; RML: REMOVING LINES; PSNR: ALL PIXEL VALUES INCREASED BY THE SAME QUANTITY; ROTATION: PURE ROTATION; RRS: ROTATION+RESCALING; AND RC: ROTATION+CROPPING

Stirmark 4.0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}
AffineT(8)	8	8	8	8	8	8	8	8	8	8
ConvF(2)	1	1	1	1	1	1	1	1	1	1
Cropping(9)	4	4	2	4	3	4	2	4	3	2
JPEG(12)	12	12	12	12	12	12	12	12	12	12
MF(4)	4	4	4	4	4	4	4	4	4	4
Noise(6)	1	1	1	1	2	1	1	1	1	2
SS(3)	3	3	3	3	3	3	3	3	3	3
Scaling(6)	6	6	6	6	6	6	6	6	6	6
RML(10)	10	10	10	10	10	10	10	10	10	10
PSNR(10)	10	10	10	10	10	10	10	10	10	10
Rotation(16)	16	16	16	16	16	16	16	16	16	16
RRS(10)	10	10	10	10	10	10	10	10	10	10
RC(10)	10	10	10	10	10	10	10	10	10	10

TABLE IV

RECOGNITION RESULTS FOR "ORIGINAL-SIFT" ON CALTECH101 AND CALTECH256 WITH 24 CATEGORIES. THE CATEGORIES LABELED WITH * COME FROM CALTECH101

Category	ak47	american-flag	backpack	calculator	car-tire	golden-gate-bridge	grand-piano	head-phones	bear	motorbikes*	revolver*	euphonium*	bonsai*	lotus*	bowling-ball	cereal-box	bulldozer	computer-mouse	dolphin	eyeglasses	duck	elk	frog	goat
top-1	10	7	2	12	0	18	11	14	1	22	13	1	0	2	0	1	0	0	3	4	3	0	0	0
top-2	13	10	7	17	1	18	18	18	1	22	17	3	0	4	0	2	0	1	6	13	7	0	0	1
top-3	16	12	11	21	1	19	22	25	1	23	23	3	0	6	0	2	1	5	11	18	9	0	1	1
top-5	21	18	17	24	1	19	26	27	1	23	25	6	2	12	1	6	2	8	15	21	14	0	1	3
top-10	28	23	26	28	4	23	28	29	2	25	30	11	8	19	7	15	10	8	22	24	25	3	3	8

Instead, we aim to verify the performance of feature extraction conducted in the ciphertext domain.

A. Scaling Factor and Feature Point Error Probability

We illustrate the relationship between the scaling factors and the error probabilities defined in Eq. (31) of Appendix. In addition to the theoretical result derived in Eq. (31), we also adopted two known image databases for obtaining practical experimental results. In the experiments, two databases, ImageNet [32] and mirflickr [33], were adopted for SIFT feature extraction. Each one of the two image databases contains 10^6 images. As described in Appendix, a feature point error happens if a feature point is theoretically/practically found but is practically/theoretically not found. Table I depicts the theoretical error probabilities and the experimental error probabilities for comparison. Note the each entry in Table I represents the exponent of an error probability under the base number that is 10. For example, when the scaling factor s is 2^{24}, the theoretical error probability is $10^{-6.0931}$. We can observe from Table I that both the theoretical and practical results are very close. This means that our analysis of feature point detection errors due to the introduction of the scaling factor s in the proposed privacy-preserving SIFT scheme is reasonable. In addition, according to Table I we can select a proper scaling factor according to the desired error probability. In the following experiments, $s = 2^{24}$ was adopted.

B. Robustness Evaluation

Ten commonly used color images with different contents (I_1: Lenna; I_2: F-16; I_3: Baboon; I_4: Peppers; I_5: Bridge; I_6: Goldhill; I_7: Sailboat; I_8: Clock; I_9: Tank; I_{10}: Splash) were adopted to verify the robustness of our secure SIFT scheme against miscellaneous attacks. The standard benchmarks, Stirmark 3.1 and 4.0, were quite suitable for simulating various manipulations of the digital images. The reader may refer to [34] for more detailed parameters of Stirmark. Basically, this experiment is analogous to image copy detection.

TABLE V

RECOGNITION RESULTS FOR "PPSIFT" ON CALTECH101 AND CALTECH256 WITH 24 CATEGORIES. THE CATEGORIES LABELED WITH * COME FROM CALTECH101

Category	ak47	american-flag	backpack	calculator	car-tire	golden-gate-bridge	grand-piano	head-phones	bear	motorbikes*	revolver*	euphonium*	bonsai*	lotus*	bowling-ball	cereal-box	bulldozer	computer-mouse	dolphin	eyeglasses	duck	elk	frog	goat
top-1	13	3	2	8	0	13	10	12	1	21	9	4	0	5	0	1	2	1	3	3	5	0	2	1
top-2	19	6	8	8	2	17	12	15	1	25	12	4	1	5	0	3	2	1	7	10	6	0	2	1
top-3	22	8	9	13	2	20	16	21	1	25	15	5	1	6	0	6	5	1	10	21	9	0	2	2
top-5	25	18	18	20	5	24	21	22	2	25	24	10	3	7	0	10	11	5	16	22	16	1	3	3
top-10	30	26	22	30	13	26	26	27	6	26	27	15	8	16	5	18	21	10	24	25	27	3	6	7

In this test, the encrypted original image was used as a query and sent to the server to find out how many modified versions could be successfully detected by comparing the detected SIFT feature vectors in the ciphertext domain. The results for robustness verification are summarized in Table II and Table III. In these two tables, each attack's name is followed by a digit, which indicates the number of times that the attack was performed with different parameters. According to our results, among 1940 modified images (there are in total 194 attacked images for each original image), 1814 of them could be correctly identified, which indicates that the correct recognition rate was 93.51%. Note that these results were obtained by controlling the false positive rate to be zero. The cases for miss recognition all occur in the attacks, including adding severe noise, cropping with (extremely) small parts remaining, and flipping, which are also the failed examples for SIFT in the plaintext domain. Our results indicate that homomorphic encryption-based secure SIFT can preserve robustness while achieving privacy.

C. Case Study on Privacy-Preserving Image Retrieval

Content-based image retrieval has been recently considered in a cloud computing environment [7], but the privacy issue is ignored. To demonstrate the usefulness of the proposed homomorphic encryption-based secure SIFT approach in achieving privacy-preserving image recognition, the Caltech101 [35] and Caltech256 datasets [36], consisting of object categories with high shape variability, were adopted. We randomly select 24 commonly used categories, each of which contains 60 images, for the experiment. Among them, 30 images per category were used as the query and the remainder were stored in the database for search purposes. Basically, this experiment is analogous to image near-duplicate detection.

It should be noted that we mainly compare the performance of original-SIFT and PPSIFT without adopting advanced feature representation and classifiers. The focus is put on the impact of homomorphic encryption on SIFT feature detection and descriptor.

Each query image is homomorphically encrypted, followed by DoG, feature point detection, and feature descriptor extraction. Then, each query is used to find the closet category via secure descriptor comparisons among the images in the database. A query image is classified into a certain category if it matches the images belonging to that category according to SIFT feature descriptor matching often. Of course, the categories a query image belongs to can also be ranked according to the number of matches in each category. Therefore, results regarding the top-k query are examined here. Table IV and Table V, respectively, show the results when top-k query, where $k = 1, 2, 3, 5, 10$, were adopted. The digit indicates the number of correct recognition according to top-k query. It can be observed from these tables that the recognition performance between original-SIFT and PPSIFT seems to be comparable.

In this experiment, we solely compare the SIFT descriptors, generated from original-SIFT and PPSIFT, in image recognition. It can be expected that the recognition rate can be remarkably improved and comparable with the state-of-the-art while still providing privacy-preserving simultaneously if sophisticatedly designed advanced feature representation and well-designed classifiers are further employed.

D. Case Study on Privacy-Preserving Face Recognition

Face recognition has been an important topic in biometrics and surveillance, and what is more, privacy-preserving face recognition has received considerable attention [9], [10], [12]. Here, a case study of privacy-preserving face recognition was conducted to verify the broad usefulness of our proposed privacy-preserving SIFT. The GT face cropped database[6], consisting of 750 color face images of size 140 × 210 from 50 subjects with 15 different face images per subject, was employed. These images were considered by allowing for strong variation in size, illumination, facial expression, and rotation both in the image plane and perpendicular to the image plane.

For creating the feature database stored on the server side, 5 images of each subject were selected and used for feature extraction. The remaining 10 face images were used as the queries to test the accuracy of PPSIFT-based face recognition. Again, the original SIFT was used to compare with PPSIFT regarding face recognition. The result is shown in Fig. 7,

[6] Georgia Tech. Face Database. Available at http://www.anefian.com/research/GTdb_crop.zip.

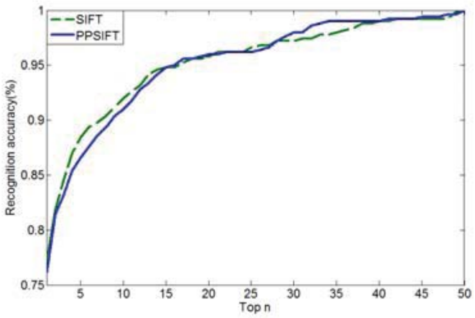


Fig. 7. Comparison of face recognition between PPSIFT and original SIFT.

where the x-axis denotes the top-n results and the y-axis is the recognition rate. We can find that both results are comparable, which implies that the detection of SIFT features in the encrypted domain does not degrade the performance of face recognition.

VII. Conclusion

We have proposed a homomorphic encryption-based privacy-preserving SIFT (PPSIFT) approach to deal with the privacy-preserving problem encountered in a cloud computing environment, where the server can finish the tasks of SIFT-based applications without learning anything to breach the user's privacy. In PPSIFT, the most challenging problem, *i.e.*, homomorphic comparison, has been solved in this paper. We show that the proposed Paillier cryptosystem-based PPSIFT scheme achieves provable security based on DLP and RSA, but the computational complexity needs to be further reduced, even if the current method is designed to be executed on the server side that owns powerful resources. We believe that the presented work is an important step toward privacy-preserving multimedia applications in an environment where privacy is a major concern.

Appendix
Error Probability of Feature Extraction Due to the Scaling Factor s

Recall the integer DoG filter defined in Eq. (9). For the sake of simplifying notations, let l denote the position of a two-dimensional pixel at (u, v) and the scales σ's are ignored. Therefore, let F_l denote the original DoG filter coefficient $G(l, \sigma_i) - G(l, \sigma_j)$ at (u, v) in Eq. (9) and let $\overline{F_l} = \lceil sF_l \rfloor$ denote the integer DoG filter, based on introducing a large integer s, where $\lceil \cdot \rfloor$ denotes the rounding function. Actually, we do not know what s should be and it will be derived to relate with the error probability here.

Next, as indicated in Eq. (11), the term $G_{Diff}(x, y, \rho)I(x, y)$ can be represented as $\overline{X_l} = \overline{F_l} \cdot I_l$, $1 \leq l \leq L$, where I_l denotes the original image pixel value and L denotes the number of pixels involved in the convolution with the DoG filter. Similarly, let $X_l = F_l \cdot I_l$. Since we would like to know the impact of the introduced factor s on SIFT feature point extraction and determine a proper value of s, the error probability $f_E(s)$ of feature point extraction due to the uses of the integer DoG filter and floating DoG filter coefficients should be minimized.

Let $F_l^d = F_l - \frac{\overline{F_l}}{s}$ denote the difference between the original floating DoG filter and the quantized integer DoG filter coefficients (note that the resultant integer DoG filter is enlarged and then quantized (or normalized) using the interval size s so that $|F_l^d|$ can range between 0 to $\frac{1}{2s}$.). Therefore, we have the difference between two DoG filtered images as:

$$Y_l = X_l - \frac{\overline{X_l}}{s} = F_l^d \cdot I_l, \quad 1 \leq l \leq L. \tag{25}$$

Some information regarding Y_l, later useful to derive the error probability $f_E(s)$, is derived in the following. The cumulative distribution function (CDF) of Y_l is first derived as:

$$\begin{aligned} P(Y_l \leq y_l) &= P(F_l^d \cdot I_l \leq y_l) \\ &= \int_{f_l^d} P(f_l^d \cdot I_l \leq y_l, F_l^d = f_l^d) df_l^d, \end{aligned} \tag{26}$$

where y_l and f_l^d are, respectively, the realization of Y_l and F_l^d. In addition, we have $-\frac{1}{2s} < f_l^d < \frac{1}{2s}$ due to $|F_l^d|$ being between 0 and $\frac{1}{2s}$, as described above, and we have $-\frac{128}{s} < y_l < \frac{128}{s}$ according to Eq. (25) and the maximum value of a pixel is 255 (for simplifying notation, 256 is used instead).

Then, the probability density function of Y_l can be calculated by differentiating the CDF in Eq. (26) as:

$$\begin{aligned} P(Y_l = y_l) &= f_{Y_l}(y_l) = \frac{\partial}{\partial y_l} P(Y_l \leq y_l) \\ &= \frac{\partial}{\partial y_l} \int_{-\frac{1}{2s}}^{\frac{1}{2s}} P(f_l^d \cdot I_l \leq y_l, F_l^d = f_l^d) df_l^d, \\ &\qquad \text{due to } -\frac{1}{2s} < f_l^d < \frac{1}{2s} \\ &= \frac{\partial}{\partial y_l} \begin{cases} \int_{\frac{-1}{2s}}^{0} P(f_l^d \cdot I_l \leq y_l, F_l^d = f_l^d) df_l^d, \\ \int_0^{\frac{1}{2s}} P(f_l^d \cdot I_l \leq y_l, F_l^d = f_l^d) df_l^d, \end{cases} \\ &= \frac{\partial}{\partial y_l} \begin{cases} \int_{\frac{-1}{2s}}^{\frac{y_l}{256}} P(f_l^d \cdot I_l \leq y_l) P(F_l^d = f_l^d) df_l^d, \\ \qquad \text{due to } -\frac{128}{s} < y_l < 0 \\ \int_{\frac{y_l}{256}}^{\frac{1}{2s}} P(f_l^d \cdot I_l \leq y_l) P(F_l^d = f_l^d) df_l^d, \\ \qquad \text{due to } 0 \leq y_l < \frac{128}{s} \end{cases} \\ &= \frac{\partial}{\partial y_l} \begin{cases} \int_{\frac{-1}{2s}}^{\frac{y_l}{256}} P(I_l \leq \frac{y_l}{f_l^d}) P(F_l^d = f_l^d) df_l^d, \\ \int_{\frac{y_l}{256}}^{\frac{1}{2s}} P(I_l \leq \frac{y_l}{f_l^d}) P(F_l^d = f_l^d) df_l^d, \end{cases} \\ &= \begin{cases} \int_{\frac{-1}{2s}}^{\frac{y_l}{256}} \frac{-1}{f_l^d} P(I_l = \frac{y_l}{f_l^d}) P(F_l^d = f_l^d) df_l^d, \\ \int_{\frac{y_l}{256}}^{\frac{1}{2s}} \frac{1}{f_l^d} P(I_l = \frac{y_l}{f_l^d}) P(F_l^d = f_l^d) df_l^d, \end{cases} \\ &= \begin{cases} \int_{\frac{-1}{2s}}^{\frac{y_l}{256}} \frac{-1}{f_l^d} \frac{1}{256} \frac{1}{s} df_l^d, \\ \int_{\frac{y_l}{256}}^{\frac{1}{2s}} \frac{1}{f_l^d} \frac{1}{256} \frac{1}{s} df_l^d, \end{cases} \quad (\because P(I_l = \frac{y_l}{f_l^d}) = \frac{1}{256}, \text{ and} \\ &\qquad P(F_l^d = f_l^d) = \frac{1}{s}) \end{aligned}$$

$$\begin{aligned}
&= \begin{cases} \frac{1}{256s}\int_{\frac{-128}{s}}^{y_l} \frac{-256}{f_l^d} d\frac{f_l^d}{256}, \\ \frac{1}{256s}\int_{y_l}^{\frac{128}{s}} \frac{256}{f_l^d} d\frac{f_l^d}{256}, \end{cases} \\
&= \begin{cases} \frac{1}{256s}(\ln|\frac{-128}{s}| - \ln|y_l|), \\ \frac{1}{256s}(\ln|\frac{128}{s}| - \ln|y_l|), \end{cases} \\
&= \frac{1}{256s}(ln\left(\frac{128}{s}\right) - ln|y_l|). \qquad (27)
\end{aligned}$$

From Eq. (27), we can find that the mean of Y_l is exactly 0 since the probability distribution of Y_l is symmetric and the variance of Y_l can be derived as:

$$\sigma_{Y_l}^2 = \int_{-\frac{128}{s}}^{\frac{128}{s}} y_l^2 f_{Y_l}(y_l)dy_l = \left(\frac{c_Y}{s}\right)^2, \qquad (28)$$

where c_Y is a constant and derived to be 42.66.

Based on the definition of the DoG pixel, $DoGImg(x, y, \rho_{ij})$, in Eq. (6) of Sec. IV-A, the DoG pixel can be represented as $X_t = \sum_{l=1}^{L} X_l$, where t denotes the position of a two-dimensional pixel at (x, y). On the other hand, we also have $\overline{X}_t = \sum_{l=1}^{L} \overline{X}_l$. Due to the central limit theorem, the quantization error $Y_t = X_t - \overline{X}_t$ is a normal distribution with zero mean and variance $\sigma_{Y_t}^2$ derived from Eq. (28) as:

$$\sigma_{Y_t} = \sqrt{\sum_{l=1}^{L}\sigma_{Y_l}^2} = \sqrt{\sum_{l=1}^{L}(\frac{c}{s})^2} = \frac{c\sqrt{L}}{s}. \qquad (29)$$

To decide a DoG local extrema as a feature point candidate, it must satisfy $DoGImg(x, y, \rho_{ij}) > \tau$, where τ is the threshold used in SIFT to filter out the unstable local extrema. Therefore, the error probability of feature point extraction due to the uses of the integer DoG filter and floating DoG filter coefficients can be defined as $P(X_t > \tau, \overline{X}_t \leq \tau) + P(X_t \leq \tau, \overline{X}_t > \tau)$, which can be further derived as:

$$\begin{aligned}
&P(X_t > \tau, \overline{X}_t \leq \tau) + P(X_t \leq \tau, \overline{X}_t > \tau) \\
&\quad = P(\overline{X}_t + Y_t > \tau, \overline{X}_t \leq \tau) + P(\overline{X}_t + Y_t \leq \tau, \overline{X}_t > \tau) \\
&\quad = \sum_{\overline{x}_t=-\infty}^{\tau} P(\overline{x}_t + Y_t > \tau, \overline{X}_t = \overline{x}_t) \\
&\qquad + \sum_{\overline{x}_t=\tau+\Delta}^{\infty} P(\overline{x}_t + Y_t \leq \tau, \overline{X}_t = \overline{x}_t) \\
&\quad = \sum_{\overline{x}_t=-\infty}^{\tau} P(Y_t > \tau - \overline{x}_t, \overline{X}_t = \overline{x}_t) \\
&\qquad + \sum_{\overline{x}_t=\tau+\Delta}^{\infty} P(Y_t \leq \tau - \overline{x}_t, \overline{X}_t = \overline{x}_t), \qquad (30)
\end{aligned}$$

where $\Delta = \frac{1}{s}$. Since $\overline{X}_t$ and Y_t are independent, Eq. (30) can be rewritten as:

$$\begin{aligned}
&P(X_t > \tau, \overline{X}_t \leq \tau) + P(X_t \leq \tau, \overline{X}_t > \tau) \\
&\quad = \sum_{\overline{x}_t=-\infty}^{\tau} P(Y_t > \tau - \overline{x}_t)P(\overline{X}_t = \overline{x}_t) \\
&\qquad + \sum_{\overline{x}_t=\tau+\Delta}^{\infty} P(Y_t \leq \tau - \overline{x}_t)P(\overline{X}_t = \overline{x}_t) \\
&\quad = \sum_{\overline{x}_t=-\infty}^{\tau} \int_{\tau-\overline{x}_t}^{\infty} P(Y_t = y_t)P(\overline{X}_t = \overline{x}_t)dy_t \\
&\qquad + \sum_{\overline{x}_t=\tau+\Delta}^{\infty} \int_{-\infty}^{\tau-\overline{x}_t} P(Y_t = y_t)P(\overline{X}_t = \overline{x}_t)dy_t \\
&\quad = \sum_{\overline{x}_t=-\infty}^{\tau} [erfc(\frac{\tau - \overline{x}_t}{\sigma_{Y_t}})]P(\overline{X}_t = \overline{x}_t) \\
&\qquad + \sum_{\overline{x}_t=\tau+\Delta}^{\infty} [1 - erfc(\frac{\tau - \overline{x}_t}{\sigma_{Y_t}})]P(\overline{X}_t = \overline{x}_t) \\
&\quad = \sum_{\overline{x}_t=-\infty}^{\tau} [erfc(\frac{s(\tau - \overline{x}_t)}{c\sqrt{L}})]P(\overline{X}_t = \overline{x}_t) \\
&\qquad + \sum_{\overline{x}_t=\tau+\Delta}^{\infty} [1 - erfc(\frac{s(\tau - \overline{x}_t)}{c\sqrt{L}})]P(\overline{X}_t = \overline{x}_t) \\
&\quad = f_E(s), \qquad (31)
\end{aligned}$$

where $erfc(\cdot)$ denotes the complementary error function with $erfc(z) = \frac{2}{\sqrt{\pi}}\int_z^{\infty}\exp(-\xi^2)d\xi$ and $f_E(s)$ denotes the error probability density function with the scaling factor s as the argument.

As we have described in Sec. IV-A, s cannot be too large since this will make the computational complexity of Eq. (10) intractable, whereas if s is too small, $G_{Diff}(x, y, \rho)$ may be truncated to zero. As a result, a proper selection of s plays an important role in achieving the tradeoff between performance and computational complexity. In order to decide a reasonable scaling factor for the proposed PPSIFT, the theoretical error probabilities obtained using various scaling factors are calculated by means of Eq. (31) and depicted in Table I.

References

[1] D. Song, D. Wagner, and A. Perrig, "Practical techniques for searches in encrypted data," in *Proc. IEEE Int. Symp. Res. Security Privacy*, May 2000, pp. 44–55.

[2] A. Swaminathan, Y. Mao, G.-M. Su, H. Gou, A. L. Varna, S. He, M. Wu, and D. W. Oard, "Confidentiality preserving rank-ordered search," in *Proc. ACM Workshop Storage, Security, Survivabil.*, 2007, pp. 7–12.

[3] W. Lu, A. Swaminathan, A. L. Varna, and M. Wu, "Enabling search over encrypted multimedia databases," *Proc. SPIE*, vol. 7254, pp. 1–11, Jan. 2009.

[4] J. Shashank, P. Kowshik, K. Srinathan, and C. Jawahar, "Private content based image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[5] D. Lowe, "Distinctive image features from scale invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[6] C. Y. Hsu, C. S. Lu, and S. C. Pei, "Homomorphic encryption-based secure sift for privacy-preserving feature extraction," *Proc. IS&T/SPIE Media Watermark., Forensics, Security*, vol. 7880, pp. 788005-1–788005-17, Jan. 2011.

[7] Z. Yang, S. Kamata, and A. Ahrary, "NIR: Content based image retrieval on cloud computing," in *Proc. IEEE Int. Conf. Intell. Comput. Intell. Syst.*, vol. 3. Nov. 2009, pp. 556–559.

[8] S. Roy and Q. Sun, "Robust hash for detecting and localizing image tampering," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2007, pp. 117–120.

[9] B. Moskovich and M. Osadchy, "Illumination invariant representation for privacy preserving face identification," in *Proc. IEEE Comput. Vis. Pattern Recognit. Workshop*, Jun. 2010, pp. 154–161.

[10] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," in *Proc. 9th Int. Symp. Privacy Enhancing Technol.*, 2009, pp. 235–253.

[11] J. Krizaj, V. Struc, and N. Pavesic, “Adaptation of sift features for robust face recognition,” in *Proc. Int. Conf. Image Anal. Recognit.*, 2010, pp. 394–404.

[12] A. Sadeghi, T. Schneider, and I. Wehrenberg, “Efficient privacy-preserving face recognition,” in *Proc. 12th Annu. Int. Conf. Inf. Security Cryptol.*, 2009, pp. 229–244.

[13] C. Velardo and J. L. Dugelay, “Face recognition with daisy descriptors,” in *Proc. ACM Multimedia Security Workshop*, 2010, pp. 95–100.

[14] U. Park, S. Pankanti, and A. K. Jain, “Fingerprint verification using sift features,” *Proc. SPIE*, vol. 6944, pp. 69440K-1–69440K-9, Mar. 2008.

[15] Z. Ye, X. Chen, and Z. Li, “Video based mobile location search with large set of sift points in cloud,” in *Proc. MCMC Workshop ACM Multimedia Conf.*, 2010, pp. 25–30.

[16] 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. Security*, vol. 7, no. 2, pp. 1–20, 2007.

[17] C. Y. Hsu, C. S. Lu, and S. C. Pei, “Secure and robust sift,” in *Proc. ACM Multimedia*, 2009, pp. 637–640.

[18] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Proc. Eurocrypt*, 1999, pp. 223–238.

[19] T. Bianchi, A. Piva, and M. Barni, “On the implementation of the discrete Fourier transform in the encrypted domain,” *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 1, pp. 86–97, Mar. 2009.

[20] P. Zheng and J. Huang, “Implementation of the discrete wavelet transform and multiresolution analysis in the encrypted DOM,” in *Proc. ACM Multimedia Conf.*, 2011.

[21] H. Bay, T. Tuytelaars, and L. V. Gool, “Surf: Speeded up robust features,” in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[22] A. Yao, “Protocols for secure computations,” in *Proc. IEEE Symp. Found. Comput. Sci.*, Nov. 1982, pp. 160–164.

[23] C. Gentry, “Fully homomorphic encryption using ideal lattices,” in *Proc. 41st Annu. ACM Symp. Theory Comput.*, 2009, pp. 169–178.

[24] A. Rezai and P. Keshavarzi, “High-performance modular exponentiation algorithm by using a new modified modular multiplication algorithm and common-multiplicand-multiplication method,” in *Proc. World Congr. Internet Security*, 2011, pp. 192–197.

[25] C. D. Walter, “Montgomery exponentiation needs no final subtractions,” *Electron. Lett.*, vol. 35, no. 21, pp. 1831–1832, 1999.

[26] I. Damgard, M. Geisler, and M. Kroigard, “Homomorphic encryption and secure comparison,” *Int. J. Appl. Cryptography*, vol. 1, no. 1, pp. 22–31, 2008.

[27] J. Seberry and J. Pieprzyk, *Cryptography: An Introduction to Computer Security*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[28] C. P. Schnorr, “Factoring integers and computing discrete logarithms via diophantine,” in *Proc. 10th Annu. Int. Conf. Theory Appl. Cryptographic Tech.*, 1991, pp. 281–293.

[29] V. Shoup, “Lower bounds for discrete logarithms and related problems,” *Proc. 16th Annu. Int. Conf. Theory Appl. Cryptographic Tech.*, 1997, pp. 256–266.

[30] E. Teske, “Speeding up Pollard’s rho method for computing discrete logarithms,” in *Proc. ANTS III*, 1998, pp. 541–553.

[31] A. S. R. Rivest and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Commun. ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[32] *ImageNet* [Online]. Available: http://www.image-net.org/

[33] *Mirflickr*. LIACS-MediaLab, Leiden Univ., Leiden, The Netherlands [Online]. Available: http://medialab.liacs.nl/mirflickr/

[34] F. Petitcolas, R. J. Anderson, and M. G. Kuhn, “Attacks on copyright marking systems,” in *Proc. Int. Workshop Inf. Hiding*, 1998, pp. 218–238.

[35] F.-F. Li, R. Fergus, and P. Perona, “Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories,” in *Proc. CVPR Workshop Generat.-Model Based Vis.*, 2004, p. 178.

[36] G. Griffin, A. Holub, and P. Perona, “Caltech-256 object category dataset,” California Institute Technology, Pasadena, Tech. Rep. CNS-TR-2007-001, 2007.

Chao-Yung Hsu received the Ph.D. degree from the Graduate Institute of Communication Engineering, National Taiwan University, Taipei, Taiwan, in 2012.

He was a Research Assistant with the Institute of Information Science, Academia Sinica, Taipei, Taiwan, from 2003 to 2012. His current research interests include multimedia signal processing.

Chun-Shien Lu received the Ph.D. degree in electrical engineering from National Cheng-Kung University, Tainan, Taiwan, in 1998.

He has been with the Institute of Information Science, Academia Sinica, Taipei, Taiwan, since August 2002, where he is currently an Associate Research Fellow. His current research interests include compressed sensing and sparse representation, multimedia signal processing, and security and privacy-preserving in multimedia and sensor networks.

Dr. Lu was a recipient of the Ta-You Wu Memorial Award of National Science Council in 2007. He was a co-recipient of the National Invention and Creation Award in 2004. He was an Area Chair in the International Conference on Acoustics, Speech, and Signal Processing in 2012 and is currently an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING.

Soo-Chang Pei (SM’89–F’00) received the Ph.D. degree from the University of California, Santa Barbara, in 1975.

He has been a Professor with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, since 1984. His current research interests include digital signal processing, image processing, optical information processing, and laser holography.

Dr. Pei became an IEEE Fellow in 2000 for his contributions to the development of digital eigenfilter designs, color image coding, and signal compressions, and to the electrical engineering education in Taiwan. He was a recipient of the Distinguished Research Award from the National Science Council from 1990 to 1998, the Academic Achievement Award in Engineering from the Ministry of Education in 1998, the Pan Wen-Yuan Distinguished Research Award in 2002, and the National Chair Professor Awards from the Ministry of Education in 2002 and 2008.