中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-10-006

Homomorphic Encryption-based Secure SIFT for Privacy-Preserving Feature Extraction

Chao-Yung Hsu, Chun-Shien Lu, Soo-Chang Pei

July, 12 2010 || Technical Report No. TR-IIS-10-006

http://www.iis.sinica.edu.tw/page/library/TechReport/tr2010/tr10.html

Homomorphic Encryption-based Secure SIFT for Privacy-Preserving Feature Extraction

Chao-Yung Hsu
IIS, Academia Sinica and
Grad. Inst. Comm. Eng., Nat'l Taiwan Univ.
Taipei, Taiwan, ROC
cyhsu@iis.sinica.edu.tw

Chun-Shien Lu
IIS, Academia Sinica
Taipei, Taiwan, ROC
lcs@iis.sinica.edu.tw

Soo-Chang Pei
Grad. Inst. Comm. Eng., Nat'l Taiwan Univ.
Taipei, Taiwan, ROC
pei@cc.ee.ntu.edu.tw

***Abstract*—Privacy has received much 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 retrieval and search with privacy-preserving will be seriously treated. In view of the fact that scale-invariant feature transform (SIFT) has been widely adopted in various fields, this paper is the first to address the problem of secure SIFT feature extraction and representation in the encrypted domain. Since all the operations in SIFT must be moved to the encrypted domain, we propose a homomorphic encryption-based secure SIFT method for privacy-preserving feature extraction and representation based on Paillier cryptosystem. In particular, homomorphic comparison is a must for SIFT feature detection but is still a challenging issue for homomorphic encryption methods. To conquer this problem, we investigate a quantization-like secure comparison strategy in this paper. Experimental results demonstrate that the proposed homomorphic encryption-based SIFT performs comparably to original SIFT on image benchmarks, while preserving privacy additionally. We believe that this work is an important step toward privacy-preserving multimedia retrieval in an environment, where privacy is a major concern.**

***Keywords*-feature extraction; homomorphic encryption; privacy preserving; security; SIFT**

I. Introduction

Recently, people are getting 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 computation power provided by the server to execute their intended tasks. Under this circumstance, Web not only provides passive search service but also is equipped with high 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, however, create the privacy issue that is usually ignored in the multimedia community.

Although encryption is a prevalent way in securing the transmitted data, the data in the encryption form (*i.e.*, ciphertext) will impede the 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 devoted to encrypted domain operations on several aspects.

Text document instead of multimedia information retrieval in the encrypted domain has received much attention in the literature for privacy protection. Song *et al.* [25] identified if a query term is in an encrypted text document or not by using Boolean search. Aiming at returning the documents in the order of their relevance to the query, Swaminathan *et al.* [26] presented a framework for rank-ordered search over encrypted text documents. In addition, similar privacy protection has also been applied on data mining [1]. Nevertheless, it is not straightforward to achieve multimedia retrieval and object recognition over encrypted data. This is mainly due to the reason that encrypted data fail to preserve the distance between feature vectors if the employed cryptographic primitives are not designed especially for intended goals.

Only recently, secure text document search has been extended to secure multimedia data search. Shashank *et al.* [24] claimed to first address the problem of protecting the privacy of the query image when searching over a public non-encrypted database. The limitation is that encrypted database is not permitted. Lu *et al.* [16], [17] studied content-based multimedia retrieval over encrypted databases, where both the query and database data are encrypted and their privacy is protected. The major concern of their methods is that the user needs to provide encrypted image indices to the server. In particular, if the approach for encrypting image indices, largely depending on applications, is required to be changed, then the user's overhead will be increased as well. 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) [15] conducted in

the encrypted domain is still lacking.

SIFT is an algorithm of detecting and describing local features in images and has been widely used [3], [7], [9], [13] due to its powerful attack-resilient keypoint detection mechanism. In this paper, we focus on presenting a homomorphic encryption-based secure SIFT method for privacy-preserving feature extraction and representation. This core technology will find many applications, including media retrieval, (near-) duplicate detection, and so on. Particularly, both the query and database are permitted to be encrypted to guarantee privacy-preserving.

The contributions of the proposed approach are summarized as follows.

1) To achieve secure SIFT, the Difference-of-Gaussian (DoG) transform is executed in the encrypted domain. We investigate how DoG transform can be performed in Paillier cryptosystem [20].
2) Usually, the existing homomorphic cryptosystems only provide additive and multiplicative homomorphism. We study and present a secure comparison method that can be conducted in the encrypted domain so that local extrema can be securely detected for SIFT feature point extraction.
3) Our method is able to achieve local extrema decision, descriptor calculation, and descriptor matching, all in the encrypted domain, without multiple rounds of communications between the user and server. On the contrary, only one-round pre-communication is necessary for synchronization of data. To the best of our knowledge, this work is among the first endeavors on the SIFT algorithm in the encrypted domain and has promising privacy-preserving multimedia applications.
4) The proposed privacy-preserving secure SIFT method has been evaluated to find its superiority in attaining both privacy and robustness under benchmark attacks and datasets, when compared with the original SIFT.

The remainder of this paper is organized as follows. We define in Sec. II the problem we would like to solve. In Sec. III, the operations on the encrypted domain are introduced, which contain our preliminary result of secure SIFT that motivates the research of this paper and a cryptosystem that is appropriate for the design of secure SIFT in a privacy-preserving manner. In Sec. IV, the proposed homomorphic encryption-based secure SIFT method with preservation of privacy is described. Our method is mainly composed of four steps, including Difference-of-Gaussian transform, feature point extraction, feature descriptor extraction, and descriptor matching, which are all accomplished in the encrypted domain for privacy concern. Experimental results are presented to verify the proposed method in Sec. V. Finally, conclusions and future work are given in Sec. VI.

II. Problem Definition

For a multimedia query system with preservation of user's privacy, as an example shown in Fig. 1, the user sends the encrypted data as a query to the server, who possesses abundant resources and powerful computation capability, can use the received encrypted data to finish the intended tasks (*e.g.*, feature extraction and media retrieval). Since the users will rely on the remote but capable server, his/her data will be encrypted for the purpose of privacy and sent to the server for storage in advance. The server must learn nothing about either the query itself sent by the user or the results derived from the query. In other words, the server is powerful in finishing the requested tasks and sends the encrypted outputs back to the user but does not know what have been obtained. When the user receives the encrypted outputs, they can be decrypted back to the plaintext domain. In this paper, we shall take digital images as the case study to describe the proposed homomorphic encryption-based secure SIFT method conducted in a privacy-preserving manner.

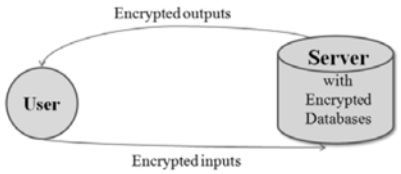


Figure 1. A query-response model operated in the encrypted domain.

In our model, the user only prepares a homomorphically encrypted copy of the query image as the encrypted inputs, which are then sent to the server for subsequent processing, while the server is responsible for generating the SIFT features via the framework of homomorphic encryption. More specifically, in order to finish SIFT in the encrypted domain, in addition to common homomorphic addition and multiplication we observe that *homomorphic comparison* is also required. Nevertheless, the extremely challenging problem, which is a major concern of this paper, is how to accomplish comparative homomorphism securely. Fig. 2 shows an example of how to perform SIFT from the encrypted image in Fig. 2(b), which is an encrypted version of Fig. 2(a) using Paillier cryptosystem [20].

It should be noted that it is not suitable to employ the framework of secure multiparty computation (SMC) [27] to achieve this goal 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 considered here, the user heavily relies on the capable and powerful server to finish almost all tasks. It will be clear later that our proposed method only needs one-round pre-communication for necessary synchronization of data when the query is initiated. Based on the received query image in the form

(a) (b)

Figure 2. Plaintext image (a) and its corresponding ciphertext image (b) obtained using Paillier cryptosystem [20].

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 [11] that proposes to detect SIFT features from encrypted images. Then, we will introduce the Paillier cryptosystem [20], which enables 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 make this paper self-contained.

A. SIFT in an Encrypted Domain

In our previous work [11], we present two anti-SIFT attacks that can efficiently remove the feature points retrieved by conventional SIFT [15]. 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 enforced to be at one of the eight neighbors around 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 either 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 erroneous while local encryption aims to secure SIFT detection.

The performance has been evaluated by examining the security against anti-SIFT attacks, authentication capability in locating maliciously tampered regions, and robustness against the benchmark, Stirmark. It has also been incorporated with sparse representation for secure image copy detection and recognition [12].

Nevertheless, as we have already mentioned in [11], a more sophisticated design regarding secure SIFT is possible. In this paper, we shall address this issue so that the performance of 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, multiplication, and so on. 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 goal is to seek a cryptosystem that can provide additive and multiplicative homomorphism, and develop a new approach to achieve homomorphic comparison [1].

To achieve operations in the ciphertext domain and obtain results equivalent to those in the plaintext domain, homomorphic encryption [8], [23] has been widely investigated. We choose the Paillier cryptosystem [20] as the platform for designing our secure SIFT method because Paillier cryptosystem provides additive and multiplicative homomorphism, achieves provable security based on modular arithmetic, and is computationally comparable to RSA. In fact, Paillier cryptosystem has been widely adopted in various applications [5]. Some recent promising privacy-preserving applications include secure transform [2], face recognition [6], secure watermark detection [18], sensor network surveillance [19], and secure distortion computation [22].

The operations of 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 $Z_{N^2} = \{0, 1, ..., N^2 - 1\}$ and $Z^*_{N^2} \subset Z_{N^2}$ denotes 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

[1] It should be noted that secure 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) [27] does not meet the goal of our paper.

satisfies $m < N$. The ciphertext of $m \in Z_{N^2}$ is derived as:

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

where $r \in Z_N^*$ denotes the user chosen key and integer numbers modulo is employed.

Third, for decrypting the ciphertext c in the decryption phase, 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 $L(x) = \frac{x-1}{N}$.

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 is also homomorphically multiplicative 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 exponentiation operation in the ciphertext domain.

IV. Secure SIFT in Homomorphic Encrypted Domain

In this section, we describe the proposed secure SIFT method that is conducted in the homomorphic encryption domain. The traditional SIFT consists of four major parts: Difference-of-Gaussian (DoG) transform, feature point (keypoint) detection, feature description, and descriptor matching. We will give a particular account of these four parts, which are all operated on encrypted data.

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 then 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 $DoGImg(x, y, \rho_{ij})$ generated at two neighboring scales ρ_i and ρ_j is defined as:

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

where $Con_G(x, y, \rho_i)$ denotes the convolution of the original image $I(x, y)$ with the Gaussian kernel $G(x, y, \rho_i)$ at the i-th scale, *i.e.*,

$$Con_G(x, y, \rho_i) = G(x, y, \rho_i) * I(x, y). \tag{7}$$

To preserve the users' privacy, the image $I(x, y)$ is encrypted by using homomorphic encryption, as described in the previous section. The resultant encrypted data is expressed as:

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

where $E()$ denotes Paillier cryptosystem, as indicated in Eq. (1) and r is the user chosen key. For practical implementation, we present to scale the original Gaussian filter coefficients to be integers with a constant s in view of the fact that Paillier cryptosystem can only operate in the integer domain. For this, the integer DoG filter, $G_{Diff}(x, y, \rho_{ij})$, is derived as:

$$G_{Diff}(x, y, \rho_{ij}) = round(s[G(x, y, \rho_i) - G(x, y, \rho_j)]), \tag{9}$$

where $round()$ is a rounding function and s is a constant used to enlarge Gaussian filter coefficients, $G()$'s, which are usually smaller than 1. It is worth noting that the secure SIFT proposed 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. In addition, when Gaussian kernel $G()$ is involved in the following discussions, its support will also be omitted.

By convolving the image to be encrypted with the DoG filter in the encrypted domain for SIFT, the resultant encrypted image in the DoG domain can be derived as:

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

where the last equation is derived according to homomorphic addition and multiplication of 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.

It is worth mentioning here that the above computation of DoG can maintain privacy without significant information loss except for the rounding errors that are caused due to

only integer operations are permitted within the framework of Paillier cryptosystem.

B. SIFT Feature Point Detection: Local Extrema Extraction via Integer Comparison in Encrypted Domain

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

1) Direct Comparison on Single Encrypted Data (One-to-One Mapping): To achieve the comparison operation in the ciphertext domain, an intuitive way is to directly compare the received encrypted data. Under this circumstance, the result yielded after comparison can be decrypted to obtain the corresponding plaintexts. In other words, the one-to-one mapping between pairs of (plaintext, ciphertext) can be revealed, leading to breach the privacy of plaintexts. This problem cannot be ignored if the encrypted data are sent to the server for subsequent processing and the server is malicious in that he/she would like to recover the plaintexts from the received ciphertexts.

More specifically, suppose we design a transformation function[2], $F()$, that aims to transform the ciphertexts into another domain to accomplish comparison. Such a transformation is required since the encrypted data, basically a random string, cannot be directly used for processing. Let K pairs of plaintexts and ciphertexts be denoted as (p_i, c_i), where $1 \leq i \leq K$. Consider the result obtained after comparing $F(c_i)$'s as $F(c_1) < F(c_2) < ... < F(c_K)$. After decryption, the relationship among the plaintexts can be revealed. Even worse, the information about plaintexts can still be revealed via comparing ciphertexts (the sorting list of $F(c_i)$'s) to know the order of plaintexts without needing decryption.

Recall from the Paillier encryption procedure shown in Eq. (1) that if the user chosen key r is fixed, then the plaintext and its corresponding ciphertext will form a one-to-one mapping, and the number of possible ciphertexts is exactly equal to that of plaintexts instead of N^2 (note that N is the integer number modulo). It is easy to reveal the plaintexts if the sorting list regarding the ciphertexts can be obtained, violating the need of privacy during operations. This is possible because given the public keys, N and g, and the ciphertext c, the adversary can exhaustive choose $r \in Z_N^*$ to solve the plaintext m according to Eq. (1). Note that the adversary will not choose to derive m from Eq. (2) since the secret key λ related to the two primes p and q is unknown. Fig. 3 illustrates an example of breaking an encrypted image if a fixed user chosen key in Paillier cryptosystem is used. We can observe that the visual quality of the recovered image in Fig. 3(b) looks acceptable, despite the quantization artifacts.

[2]In the next subsection, Eq. (15) shows an example.

(a) (b)

Figure 3. Breaking an encrypted image generated using a fixed user chosen key in Paillier cryptosystem: (a) original image; (b) image recovered from a Paillier encrypted image.

Let's also consider this security breach problem from another viewpoint. If for some reasons and applications it is required to directly compare the encrypted data to obtain the magnitude relations among the plaintexts, the privacy breach problem described above is also suitable for use here. Therefore, it is concluded that (1) if Paillier cryptosystem is used, the user chosen key r must be chosen at random or content-dependent and (2) homomorphic comparison on single encrypted data is insecure.

2) Direct Comparison on Linear Combination of Encrypted Data (One-to-Many Mapping): According the above observations, the user chosen key r in Eq. (1) must be variable. 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. This states the property of semantic security in the Paillier cryptosystem.

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

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

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

$$\begin{aligned} & DoGImg_e(x,y,\rho) \\ = \; & \prod_{x,y} E(I(x,y), r_{x,y})^{G_{Diff}(x,y,\rho)} \; mod \; N^2 \end{aligned}$$

$$= \prod_{x,y} g^{I(x,y)G_{Diff}(x,y,\rho)} r_{x,y}^{NG_{Diff}(x,y,\rho)} \ mod\ N^2$$
$$= g^{\sum_{x,y} I(x,y)G_{Diff}(x,y,\rho)} (\prod_{x,y} r_{x,y}^{G_{Diff}(x,y,\rho)})^N \ mod\ N^2$$
$$= E(\sum_{x,y} I(x,y)G_{Diff}(x,y,\rho), \prod_{x,y} r_{x,y}^{G_{Diff}(x,y,\rho)})$$
$$= E(DoGImg(x,y,\rho), R_\rho), \quad (12)$$

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

$$R_\rho = \prod_{x,y} r_{x,y}^{G_{Diff}(x,y,\rho)}, \quad (13)$$

which is a function of the user 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 access to $r_{x,y}$'s and their combined ones R_ρ's.

Homomorphic Comparison: Due to one-to-many mapping, the resultant ciphertexts, as indicated in Eq. (12), will fall into the range between 0 and N^2-1, and the whole range may be completely occupied. In addition, the ciphertexts are accompanied with R_ρ. Therefore, we propose quantization-like secure comparison of ciphertexts to equivalently achieve local extrema extraction in the plaintext domain. In our method, a series of thresholds, which will divide the ciphertext domain located between 0 and N^2-1 into several (non-uniform) quantization intervals, in the ciphertext domain 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 secure comparison. Since comparison will be conducted in the encrypted domain, these thresholds are encrypted via Paillier encryption using R_ρ at scale ρ as:

$$T_{ie} = E(T_i, R_\rho) = g^{T_i} R_\rho^N \ mod\ N^2, \quad (14)$$

where $T_{ie} \in Z_{N^2}$. Note that R_ρ is employed by users to encrypt T_i's because the ciphertexts used for secure 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 secure processing are the secure thresholds T_{ie}'s and their order. Note that the calculation of R_ρ needs $G_{Diff}(x,y,\rho)$, which will be sent from the server to the user. Such a pre-computation will only be executed once during the course of the query system when a user initiates his/her query task. Nevertheless, we also note that some secure comparison algorithms like [4] employ the framework of secure multiparty computation and need a few rounds of communications.

Now, the strategy for comparison between two elements in $DoGImg_e(x,y,\rho)$ in the encrypted domain will be described as follows. Basically, our idea is to compare two encrypted data according to their locations in the intervals separated by the thresholds T_{ie}'s. Given two ciphertexts, $E(DoGImg(x_1,y_1,\rho), R_\rho)$ and $E(DoGImg(x_2,y_2,\rho), R_\rho)$, at scale ρ and at two different locations, (x_1,y_1) and (x_2,y_2), respectively, the goal is to compare them 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:

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

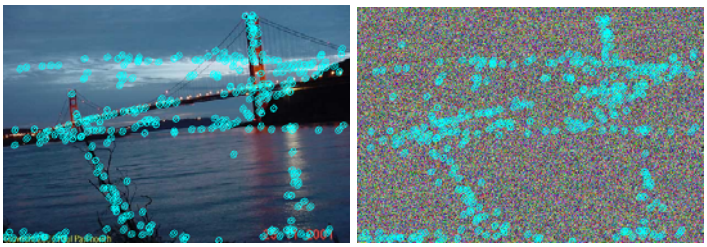
where $k = 1,2$. In Eq. (15), g^{Inc} appears for additive homomorphism but, in fact, it should be $E(Inc,1) = g^{Inc} r^N \ mod\ N^2$ with $r = 1$ in order not to change the combined r value, as indicated in Eqs. (3) and (4).

Mathematically, Eq. (15) implies that the plaintext $DoGImg$ (x_k,y_k,ρ) 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 $E(T_k,R_\rho)$'s are found in this case for $k = 1,2$, the server can determine the magnitude relationship between the two ciphertexts, $E(DoGImg(x_1,y_1,\rho),R_\rho)$ and $E(DoGImg(x_2,y_2,\rho),R_\rho)$, since it receives the order of encrypted thresholds sent from the user.

On the other hand, if $E(T_1,R_\rho) = E(T_2,R_\rho)$ is found, then the magnitude relationship between the two ciphertexts, $E(DoGImg(x_1,y_1,\rho),R_\rho)$ and $E(DoGImg(x_2,y_2,\rho),R_\rho)$, can still be determined by checking the magnitude relationship between a_1 and a_2. For example, if $a_1 > a_2$, then $E(DoGImg(x_1,y_1,\rho),R_\rho) < E(DoGImg(x_2,y_2,\rho),R_\rho)$ since $E(DoGImg(x_1,y_1,\rho),R_\rho)$ is more distant from T_{ie} for certain i. Thus, according to this proposed secure 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.

Fig. 4 illustrates a result of SIFT feature point extraction with respect to Fig. 2 in the plaintext and ciphertext domains, respectively. Visually, the detected locations (labeled in

'blue' circle) of feature points look similar. More advanced evaluations will be elaborated in Sec. V.



(a) (b)

Figure 4. Detection of SIFT features in the plaintext domain (a) and ciphertext domain (b), which correspond to the pictures in Fig. 2, respectively. (best viewed on a color display)

Please also note that in the proposed scheme the locations of pixels are not encrypted, so that the locations of SIFT features are known to the public and the server. Such a characteristic is significantly different from our previous work [11], which aims to hide the locations of SIFT features in order to escape from being maliciously tampered with. Nevertheless, for the scenario which is 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[3].

The Impact of Number of Thresholds T_i's and Their Pairwise Distances: As described in the above subsection, the quantization-like secure comparison strategy needs a series of thresholds T_i's. It is interesting to investigate the impact of 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 a_1 and a_2 can cooperatively finish secure comparison. Hence, in the following analyses, we assume the use of uniform quantization for simplicity.

Second, we examine the impact of number of thresholds upon the speed of computation and security. If the number of thresholds is large, meaning that the interval size is small, then the computation of homomorphic comparison indicated in Eq. (15) can be speeded-up since a_k can be found fast. Nevertheless, if large number of thresholds is adopted, then the communication cost spent in transmitting these thresholds from the user to the server is large. 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.

Finally, we discuss the security of secure comparison accomplished using Eq. (15). Given the encrypted data shown in Eq. (12) and the public key (N, g), the plaintexts cannot be exactly recovered since each combined user chosen key R_ρ, as shown in Eq. (13), is a combination of location-based user chosen keys $r_{x,y}$'s and is unknown publicly.

C. SIFT Feature Point Descriptor in Encrypted Domain

In this section, we first describe how to derive SIFT feature descriptors in the plaintext domain, which is then extended to the ciphertext domain. First, as done in [15], orientation assignment is executed for each detected feature point. Then, a normalized region of size 16×16 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 feature descriptor is accomplished at the scale, where the feature is detected. Let the gradient magnitudes be, respectively, denoted as $Diff_X = Con_G(x+1, y, \rho) - Con_G(x-1, y, \rho)$ and $Diff_Y = Con_G(x, y+1, \rho) - Con_G(x, y-1, \rho)$ 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 in total sixteen 4×4 blocks in a 16×16 region, a 64-dimensional feature descriptor is established. It should be noted that no more than the four restrictive directions are employed in this paper because the operation of secure inner product[4] is required to derive the included angle with the two sides not both coinciding with the $x-$ and $y-$ axes. Our empirical results, however, reveal that the SIFT descriptors with 64 dimensionalities are sufficient to maintain robustness nearly without any loss.

In this paper, the feature descriptor calculated in the encrypted domain for each 4×4 block is conducted as follows.

[3]We have an interesting observation that if the SIFT feature descriptors are not encrypted, the adversary can use them to query another databases so that the originally encrypted content may be approximately guessed from the search outputs, leading to privacy breach. This issue, currently not the scope of this paper, is worth further studying if we try to break the Paillier cryptosystem-based applications.

[4]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.

Let $V(k)$, $0 \leq k \leq 3$, denote the 4-dimensional feature descriptor of a 4×4 block. They are derived according to the above conceptions based on homomorphic addition and multiplication as:

$$V(0) = V(0)E(Con_G(x+1,y,\rho))E(Con_G(x-1,y,\rho))^{-1},$$
$$\text{if } Con_G(x+1,y,\rho) \geq Con_G(x-1,y,\rho)$$
$$V(1) = V(1)E(Con_G(x,y+1,\rho))E(Con_G(x,y-1,\rho))^{-1},$$
$$\text{if } Con_G(x,y+1,\rho) \geq Con_G(x,y-1,\rho)$$
$$V(2) = V(2)E(Con_G(x-1,y,\rho))E(Con_G(x+1,y,\rho))^{-1},$$
$$\text{if } Con_G(x-1,y,\rho) \geq Con_G(x+1,y,\rho)$$
$$V(3) = V(3)E(Con_G(x,y-1,\rho))E(Con_G(x,y+1,\rho))^{-1},$$
$$\text{if } Con_G(x,y-1,\rho) \geq Con_G(x,y+1,\rho).$$

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

D. SIFT Feature Descriptor Matching in Encrypted Domain

Once the encrypted descriptors have been calculated, the stage of descriptor matching can be widely used in many applications such as object recognition and media retrieval. The descriptor matching stage aims to compares a query descriptor with each candidate descriptor in the database for similarity evaluation via a similarity metric. Let the similarity between two descriptors, V^i and V^j, be denoted as $Sim(V^i, V^j)$. Since the inner product between two descriptors is commonly used as a similarity metric, which is expressed in the plaintext domain as:

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

For 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 homomorphic multiplication (Eq. (5)).

Unfortunately, it is, however, not possible for the server to access the plaintexts $V^j(k)$'s used in Eq. (19) due to 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:

$$Sim^p_{\ell_1}(E(V^i), E(V^j)) = E(|V^i - V^j|_1)$$
$$= \prod_{k=0}^{63} E(V^i(k))E(V^j(k))^{-1} \ mod\ N^2. \tag{21}$$

Finally, V^i and V^j are considered to be similar if the similarity value defined in Eq. (21) is smaller than a threshold, σ. The threshold σ can be found by minimizing the probability of false positive as:

$$\sigma = argmin_\sigma P\{Sim^p_{\ell_1}(E(V^i), E(V^j)) \leq \sigma \mid V^i \text{ and } V^j \text{ are dissimilar}\}. \tag{22}$$

E. Comparison with Original SIFT

For the sake of clarity, the differences between the original SIFT (denoted as "original-SIFT") and the proposed homomorphic encryption-based SIFT (abbreviated as "HE-SIFT") are summarized as follows.

1) The major difference is that HE-SIFT is entirely conducted in the encrypted domain, wherein DoG conducted in the encrypted domain is presented and homomorphic comparison is proposed for privacy-preserving feature extraction.
2) The descriptor vector length of HE-SIFT is 64 for the reason of avoiding executing inner product without allowing interaction between parties in the encrypted domain, as mentioned in Sec. IV-C, while the descriptor length of original-SIFT is 128.
3) The metric used for privacy-preserving matching between two descriptors in HE-SIFT is ℓ_1 distance while it is inner product in original-SIFT.

The performance difference for both original-SIFT and HE-SIFT will be evaluated in the next section.

V. Experimental Results

Two kinds of experiments were conducted to evaluate the performance of proposed method. In Sec. V-A, 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. V-B, we describe a case study on image recognition, which is a popular application based on feature extraction. The aim is to verify whether comparable performance between original-SIFT and HE-SIFT can still be obtained even all operations of HE-SIFT are conducted in the encrypted domain.

For experimental setup, the two large prime numbers, p and q, were usually selected to be 100-bit as done in [22], but they could be larger, and $g \in Z^*_{N^2}$ can be arbitrarily selected and was set to be 20 here. For implementation of DoG, 7 octaves and 6 voices between two neighboring octaves were selected. For feature point detection via secure comparison, ten thresholds $T_i \in Z_N$ were arbitrarily selected for reducing the communication cost on the user side but increasing the computation overhead on the server side.

A. Robustness Evaluation

Six 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) 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 [21] for more detailed parameters of Stirmark. Basically, this experiment is analogous to image copy detection.

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 I and Table II. 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 1224 modified images (there are in total 204 attacked images for each original image), 1151 of them could be correctly identified, which indicates that the correct recognition rate was 94.04%. 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 severe noise adding, cropping with (extremely) small parts remaining, and flipping, which are also the failed examples for SIFT in the plaintext domain. Obviously, our results indicate that homomorphic encryption-based secure SIFT can preserve robustness while achieving privacy.

Table I
ROBUSTNESS OF OUR SCHEME VS. 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
SPA(6)	6	6	6	6	6	6
JPEG(12)	12	12	12	12	12	12
GLGT(3)	3	3	3	3	3	3
CAR(8)	8	8	8	8	8	8
LR(5)	5	5	5	5	5	5
Cropping(9)	8	8	8	8	8	9
RC(16)	16	16	16	16	16	16
Scaling(6)	6	6	6	6	6	6
RRS(16)	16	16	16	16	16	16
Shearing(6)	6	6	6	6	6	6
RB(1)	1	1	1	1	1	1
Flipping(1)	0	0	0	0	0	0

Table II
ROBUSTNESS OF OUR SCHEME VS. STIRMARK 4.0: ATTACKS ARE DENOTED AS AFFINET: AFFINE TRANSFORMATION; CONVF: CONVOLUTION FILTERING; CROPPING: CROPPED TO $\frac{3}{4}$, $\frac{1}{2}$, $\frac{1}{4}$, AND $\frac{1}{5}$ 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
AffineT(8)	8	8	8	8	8	8
ConvF(2)	1	1	1	1	1	1
Cropping(9)	4	4	2	4	3	4
JPEG(12)	12	12	12	12	12	12
MF(4)	4	4	4	4	4	4
Noise(6)	1	1	1	1	2	1
SS(3)	3	3	3	3	3	3
Scaling(6)	6	6	6	6	6	6
RML(10)	10	10	10	10	10	10
PSNR(10)	10	10	10	10	10	10
Rotation(16)	16	16	16	16	16	16
RRS(10)	10	10	10	10	10	10
RC(10)	10	10	10	10	10	10

B. Case Study on Image Recognition

To demonstrate the usefulness of the proposed homomorphic encryption-based secure SIFT approach in achieving privacy-preserving image recognition, the Caltech101 [14] and Caltech256 datasets [10] 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 purpose. Basically, this experiment is analogous to image near-duplicate detection. Fig. 5 shows some examples of Caltech images, where the first row shows four images of the Caltech256 category "golden-gate-bridge," and the second row shows four images of the Caltech101 category "bowling-ball."

It should be noted that we mainly compare the performance of original-SIFT and HE-SIFT without adopting advanced feature representation and classifiers. The focus

is put on the impact of homomorphic encryption on SIFT feature detection and descriptor.

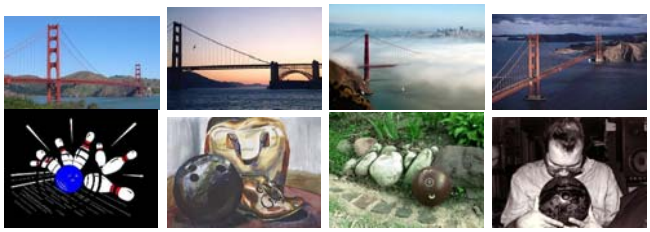


Figure 5. Top row: four examples of the Caltech256 category "golden-gate-bridge"; Bottom row: four examples of the Caltech101 category "bowling-ball".

For each query image, it 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 to the images belonging to that category according to SIFT feature descriptor matching most 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 III and Table IV, 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. Some pictures excerpted from two of the 23 categories are shown in Fig. 5. It can be observed from these tables that the recognition performance between original-SIFT and HE-SIFT seems to be comparable. To be specific, we have the following observations.

- If original-SIFT (slightly) outperforms HE-SIFT, we conjecture that this is mainly due to the reason that both secure SIFT feature detection and descriptor extraction are affected by privacy-preserving operations in the encrypted domain.
- If HE-SIFT (slightly) outperforms original-SIFT, it is presumed that HE-SIFT captures less sophisticated features than original-SIFT since the feature length of the former is only half the latter, as we have described in Sec. IV-C and Sec. IV-E. Under this circumstance, rough feature representation more fits the experiment of near-duplicate object classification conducted here.

In this experiment, we solely compare the SIFT descriptors, generated from original-SIFT and HE-SIFT, 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 sophisticated designed advanced feature representation and well-designed classifiers are further employed.

VI. Conclusions and Future Work

We have proposed a homomorphic encryption-based secure SIFT approach to deal with the privacy-preserving problem encountered in a cloud computing environment, where the server can finish the tasks of SIFT-based image retrieval and management without learning anything to breach the user's privacy. Notably, we address the privacy issue that is relatively ignored in the media retrieval literature. We present a new method to enable SIFT to be done in the framework of Paillier cryptosystem, where the most challenging problem; *i.e.*, secure comparison, has been solved in this paper. We believe that the presented work is an important step toward privacy-preserving multimedia retrieval in an environment, where privacy is a major concern.

For future work, we will analyze the errors introduced due to integral operations in encrypted domain. At present, we mainly verify the performance of the proposed homomorphic encryption-based SIFT at the feature level; *i.e.*, SIFT descriptors built upon the SIFT feature points are used for experiments. However, advanced features (such as visual words, descriptive visual words/phrases, query expansions, and etc) built upon the encrypted SIFT features should be further investigated to improve the retrieval performance. Moreover, privacy-preserving feature extraction finds broad applications and is also a key to secure video surveillance. Our another work indeed shows promising results in object detection of encrypted surveillance videos.

VII. Acknowledgments

This work was supported by National Science Council, Taiwan, ROC, under Grants NSC 98-2631-H-001-013 and NSC 99-2631-H-001-020.

References

[1] C. C. Aggarwal and P. S. Yu. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer-Verlag, 2008.

[2] T. Bianchi, A. Piva, and M. Barni. On the implementation of the discrete fourier transform in the encrypted domain. *IEEE Trans. on Information Forensics and Security*, 4(1):86–97, 2009.

[3] O. Chum, M. Perdoch, and J. Matas. Geometric min-hashing: Finding a (thick) needle in a haystack. In *IEEE CVPR*, pages 00–01, 2009.

[4] I. Damgard, M. Geisler, and M. Kroigard. Homomorphic encryption and secure comparison. *Int. Journal of Applied Cryptography*, 1(1):22–31, 2008.

[5] I. Damgard and M. Jurik. A generalization, a simplification and some applications of paillier probabilistic public-key system. In *Public Key Cryptography*, pages 119–136, 2001.

[6] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft. Privacy-preserving face recognition. In *PETS, LNCS 5672*, pages 235–253, 2009.

Table III
RECOGNITION RESULTS FOR "ORIGINAL-SIFT" ON CALTECH101 AND CALTECH256 WITH 23 CATEGORIES. THE CATEGORIES LABELED WITH* COME FROM CALTECH101.

top k query \ 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
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
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
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
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
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

Table IV
RECOGNITION RESULTS FOR "HE-SIFT" ON CALTECH101 AND CALTECH256 WITH 23 CATEGORIES. THE CATEGORIES LABELED WITH* COME FROM CALTECH101.

top k query \ 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
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
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
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
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
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] J. J. Foo, J. Zobel, R. Sinha, and S. Tahaghoghi. Detection of near-duplicate images for web search. In *CIVR*, pages 00–11, 2007.

[8] C. Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, pages 169–178, 2009.

[9] K. Grauman and T. Darrell. Efficient image matching with distributions of local invariant features. In *CVPR*, pages 627–634, 2005.

[10] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. In *Technical Report. California Institute of Technology*, 2007.

[11] C. Y. Hsu, C. S. Lu, and S. C. Pei. Secure and robust sift. In *ACM MM*, 2009.

[12] L. W. Kang, C. Y. Hsu, H. W. Chen, and C. S. Lu. Secure sift-based sparse representation for image copy detection and recognition. In *IEEE Int. Conf. on Multimedia and Expo*, 2010.

[13] Y. Ke, R. Sukthankar, and L. Huston. Efficient near-duplicate and sub-image retrieval. In *ACM MM*, pages 00–01, 2004.

[14] 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 *IEEE CVPR Workshop on Generative-Model Based Vision*, pages 178–178, 2004.

[15] D. Lowe. Distinctive image features from scale invariant keypoints. *Int. Journal of Computer Vision*, 60(2):91–110, 2004.

[16] W. Lu, A. Swaminathan, A. L. Varna, and M. Wu. Enabling search over encrypted multimedia databases. In *SPIE Conference on Media Forensics and Security*, pages 1–11, 2009.

[17] W. Lu, A. L. Varna, A. Swaminathan, and M. Wu. Secure image retrieval through feature protection. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1533–1536, 2009.

[18] M. Malkin and T. Kalker. A cryptographic method for secure watermark detection. In *Information Hiding Workshop, LNCS 4437*, pages 26–41, 2007.

[19] V. A. Oleshchuk. Privacy preserving monitoring and surveillance in sensor networks. In *IPSA, LNCS 4743*, pages 485–492, 2007.

[20] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Eurocrypt, LNCS 1592*, pages 223–238, 1999.

[21] F. Petitcolas, R. J. Anderson, and M. G. Kuhn. Attacks on copyright marking systems. In *Int. Workshop on Information Hiding, LNCS 1575*, pages 218–238, 1998.

[22] S. D. Rane, W. Sun, and A. Vetro. Secure distortion computation between untrusting parties using homomorphic encryption. In *IEEE Int. Conf. on Image Processing*, pages 1485–1488, 2009.

[23] R. Rivest, L. Adelman, and M. Dertouzos. *Foundations of Secure Computation*. Academic, London, U.K., 1978.

[24] J. Shashank, P. Kowshik, K. Srinathan, and C. Jawahar. Private content based image retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[25] D. Song, D. Wagner, and A. Perrig. Practical techniques for searches in encrypted data. In *IEEE Int. Symposium on Research in Security and Privacy*, pages 44–55, 2000.

[26] A. Swaminathan, Y. Mao, G.-M. Su, H. Gou, A. L. Varna, S. He, M. Wu, and D. W. Oard. Condentiality preserving rank-ordered search. In *ACM Workshop on Storage, Security, and Survivability*, pages 7–12, 2007.

[27] A. Yao. Protocols for secure computations. In *IEEE Symp. Foundations Computer Science*, pages 160–164, 1982.