Privacy-preserving Outsourcing of Image Global Feature Detection

Zhan Qin*, Jingbo Yan⋆, Kui Ren*, Chang Wen Chen*, Cong Wang†, Xinwen Fu§
*Department of Computer Science and Engineering, State University of New York at Buffalo
⋆State Key Laboratory of Integrated Services Networks, Xidian University
†Computer Science Department, City University of Hong Kong
§Department of Computer Science, University of Massachusetts Lowell
*{zhanqin, kuiren, chencw}@buffalo.edu, ⋆jbyan@xidian.edu.cn, †congwang@cityu.edu.hk, §xinwenfu@cs.uml.edu

***Abstract*—The amount and availability of user-contributed image data have been dramatically increased during the past ten years. Popular multimedia social networks, e.g. Flicker, commonly utilize user image data to construct user behavior models, social preferences, etc., for the purpose of effective advertisement, better user retention and attraction, and many others. Existing practices of data utilization, however, seriously deteriorate users' personal privacy and have led to increasing criticisms and legislation pressures. In this paper, we aim to construct a privacy-preserving feature detection scheme over encrypted image data. The proposed system enables an interested party to perform a variety of image feature detection tasks, including visual descriptors in MPEG-7 standard, while protecting user privacy relating to image contents. We implement a prototype system based on somewhat homomorphic encryption scheme and the benchmark Caltech256 database. The experimental results show that our system can guarantee effective image feature detection without sacrificing user privacy.**

I. Introduction

Nowadays, the ever-increasing data mining algorithms are implemented by various social network service (SNS) providers to discover the users' behavior preferences, social links, etc. As over six million images are being uploaded to SNS providers everyday [1], the user-contributed image data becomes a key enabler to these mining applications. More and more SNS providers cooperate with interested parties like advertisers or retailers in mining these massive data for tremendous business interests.

However, the participation of interested parties in the user-contributed data analysis compromises users' privacy at different levels. The analyzed data may reveal private information, including personal identity, home address, or even financial profiles. A recently revealed privacy invasion to user-contributed data is from PRISM [2]. In this program, the security authority inspects and analyzes the private data of the millions of users. As a typical example of cooperative interested party in current scheme, the security authority has little-to-none controlled access to user-contributed data. Public concerns and criticisms on existing data mining schemes of interested party increase unprecedentedly [3].

There is an emerging research field on discovering solutions to enable image processing over encrypted image data [4], [5], [16], [17]. In secure multimedia data search [4], [16], the authors propose designs to protect the privacy of the query image when searching over a public database by utilizing oblivious retrieval techniques. Other works focus on extracting/detecting image feature in the ciphertext domain. In [5], [6], a scheme is proposed to utilize Paillier encryption scheme to enable a secure local feature detection (SIFT) in the ciphertext domain (Detials refer to Sec. V).

In this paper, we focus on providing a solution to enable an interested party to detect global feature of image data without compromising users' privacy. In the proposed system, the interested party performs image feature detection algorithms to generate the image features from the encrypted image data shared by SNS provider. After that, as an example application in target advertising, it can utilize the similarities (Euclidean distances) between the extracted features of the users' images and the benchmark images to identify target users. The challenge lies in how to preserve the privacy of users' image against the interested party, while enabling the functionalities of the image global feature detection algorithms over encrypted data. To address the challenge, we propose a privacy-preserving image feature detection system. It utilizes the homomorphic properties of Somewhat Homomorphic Encryption (*SHE*) scheme to decompose the existing image feature detection algorithms into circuit-level operations that can be performed in the ciphertext domain.

Compared to existing works, the contributions of this paper can be summarized as follows: To the best of our knowledge, the proposed system is the first to enable the privacy-preserving image feature detection with the involvement of interested parties. Specifically, it reconstructs the image global feature detection algorithms from MPEG-7 standard in the ciphertext domain by utilizing circuit-level homomorphic operations supported by *SHE* scheme. It analyzes the security and gives the proof to justify the privacy-preserving guarantee of the proposed system. It describes a complete implementation and reports the results across hundreds of images from Caltech256 image database. The implementation evaluation shows the correctness and effectiveness of the proposed design. The remaining of this paper is organized as follows: The problem formulation is described in Sec. II. The constructions of privacy-preserving image feature detection system are proposed in Sec. III, followed by the performance evaluation in Sec. IV. Finally, the related work and conclusion are presented in Sec. V and VI.

II. Problem Formulation

A. System Model

As shown in Fig. 1, the proposed system consists of two main entities: SNS Provider (SP) and Interested Party (IP):

978-1-4799-3512-3/14/$31.00 ©2014 IEEE

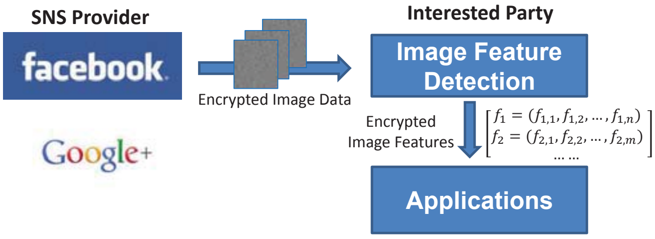


Figure 1. System Model

SP holds massive user-contributed image data and intends to sell the privilege of mining these data to IP for business interest. In addition, it is also under an obligation to protect the privacy of these user-contributed data. IP may represents an individual or an enterprise, which is interested in mining the features of the users' image data collected by SP and intends to utilize them in applications for different purposes, e.g. targeted advertising and security surveillance. Due to SP's obligation on protecting the privacy of image data, IP should not be allowed to access the content or the features of user-contributed image data in plaintext domain. The proposed system consists of two phases. One is *Data Preprocessing*: During the data preprocessing phase, for an image I, SP prepares $C \leftarrow \mathsf{Encode}(I)$ and sends C to IP, where $\mathsf{Encode}(\cdot)$ is an encoding algorithm that enables feature detection over the encrypted image C. Such encoding algorithm should be lightweight and support as many feature detection algorithms as possible. Hence, SP only needs to encode its image data once, and the major computation work over the encoded image data is taken by IP. The other is *Image Feature Detection*: IP performs feature detection algorithms over the encoded image data C to detect the encrypted features. The proposed system focuses on implementing popular *visual descriptor* algorithms: *color descriptor* in the ciphertext domain. Afterward, these detected image features can be processed to compute the similarities (Euclidean distance) for image matching or other applications. Meanwhile, the privacy of the image data is protected against IP.

In our design, IP gets the maximum flexibility and scalability to control the image feature detection procedures. An essential procedure in image matching-the computation of similarities among feature vectors is performed by IP. In fact, for image matching, SP only needs to perform decryption for the matching results of each application, since SP only needs to encrypt the image and upload its ciphertext once. If SP undertakes the feature computation by itself and uploads the encrypted features instead of the image, the flexibility of the design would be poor. SP has to perform the feature computation over the whole image database, encrypt and upload them again, once the parameters of feature computation are changed by IP. Note that the benchmark image may contain private information. Thus, it is required to be protected against SP in certain applications, e.g. face recognition and medical images.

B. Design Goals

Security: The proposed system should enable the outsourcing of image feature detection algorithms without compromising the privacy of the image data against IP. In other words, IP can only get the size and format of each image. In our design, both IP and SP are assumed to be curious-but-honest. Since the feature detection phase is non-interactive, we define the privacy as indistinguishability of two encoded images with the same size and format for all probabilistic polynomial-time adversaries $\mathcal{A}$:

$$Pr[\mathcal{A}(C_b, I_0, I_1) = b)] = \frac{1}{2} + negl(\kappa)$$

in which b is randomly chosen from $\{0, 1\}$, plaintexts of I_0 and I_1 is given to $\mathcal{A}$, κ is the security parameter, and $negl()$ is a negligible function of κ.

Effectiveness: The effectiveness of the image features generated by the proposed system should be approximate to the output of the original image feature detection algorithms. This means that if all entities honestly run the system, the image features detected by IP in ciphertext domain should be approximate to the features detected in plaintext domain. And the effectiveness of these features on further applications also should similar to the features from the original algorithms.

III. System Construction

A. Data Preprocessing

In order to encrypt the image data, we utilize the most recent RLWE based Somewhat Homomorphic Encryption (*SHE*) scheme from [7] to realize such encryption process in SP. We define *SHE* scheme to be a tuple of algorithms: $\mathcal{SHE} =$ ($\mathbf{SH.Gen}$,$\mathbf{SH.Enc}$,$\mathbf{SH.Add}$,$\mathbf{SH.Mult}$,$\mathbf{SH.Dec}$). Among these algorithms, $\mathbf{SH.Gen}$ defines the secret key sk from sampling a ring element. It randomly generates ring elements a_1, e, and computes the public key $\mathsf{pk} = (a_0 = -a_1 s - te, a_1)$. The $\mathbf{SH.Enc}$ and $\mathbf{SH.Dec}$ are shown below:

$$\mathbf{SH.Enc}(\mathsf{pk}, m) = \mathsf{ct} = (c_0, c_1) \tag{1}$$

where $c_0 = a_0 u + tg + m, c_1 = a_1 u + tf$, and

$$\mathbf{SH.Dec}(\mathsf{sk}, \mathsf{ct}) = m = \sum_{i=0}^{\delta} c_i s^i (mod\ t); \tag{2}$$

The homomorphic properties on addition and multiplication of *SHE* scheme are briefly described as follows:

$$\mathbf{SH.Add}(\mathsf{ct}, \mathsf{ct}') = (c_0 + c_0', \ldots, c_{max} + c_{max}'); \tag{3}$$

$$\mathbf{SH.Mult}(\mathsf{ct}, \mathsf{ct}') = (\sum_{i=0}^{\delta} c_i v^i) \times (\sum_{i=0}^{\delta} c_i' v^i); \tag{4}$$

The above properties are valid within the finite homomorphic additions and multiplications. Assume that a pixel of image I at position (x, y) is represented as

$$I(x, y) = (R_{x,y}, G_{x,y}, B_{x,y}) = (r_j^{(x,y)}, \ldots, g_j^{(x,y)}, \ldots, b_j^{(x,y)})$$

where $j = \{0, \ldots, 7\}$, in a RGB color format.

We define the encoding algorithm as shown in Alg. 1. The notation κ represents the security parameter, in which, the dimension of the polynomial n, the modulus q, the message space t, the error parameter σ and the bound of the multiplication D are configured as the security parameter κ. And $C(x, y)$ stands for a bitwise ciphertext of a pixel value at position (x, y) represented in a RGB color model: $C(x, y) = (ct_{r_7}, \ldots, ct_{r_0}, ct_{g_7}, \ldots, ct_{g_0}, ct_{b_7}, \ldots, ct_{b_0})$, where

Algorithm 1 SHE base Image Encoding

```
Input:
    The matrix of pixels for current processed image $I$;
Output:
    The encrypted image matrix $C$
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathbf{SH.Gen}(1^\kappa)$;
for All $(x, y) \in I$ do
    Encrypting the Image $I$:
    $\forall (x, y)\ in\ I : C(x, y) \leftarrow \mathbf{SH.Enc}(\mathsf{pk}, I(x, y))$;
end for
return The ciphertext matrix $C$.
```

$ct_* = (c_{1,*}, c_{2,*})$. Note that the homomorphic comparison between bitwise ciphertexts is much more efficient than in integer-wise ciphertexts. Meanwhile, the bitwise ciphertexts are able to be packed into a single ciphertext as the integer-wise ciphertext, which is more efficient in homomorphic multiplications and additions. Here, we briefly show the packing technique from [7]. It packs n bitwise ciphertexts of $\{b_{n-1}, ..., b_0\}$ into a single integer-wise ciphertext of a polynomial $b(x) = b_{n-1}x^{n-1} + ... + b_0$: Given n ciphertexts $(c_{0,i}, c_{1,i})$ from the encryption of the bits $\{b_i\}$, the ciphertext c_{int} of the binary integer $b_{n-1}...b_0$ is transformed from:

$$c_{int} = (\textstyle\sum_{i=0}^{n-1} c_{0,i}x^i, \sum_{i=0}^{n-1} c_{1,i}x^i) \qquad (5)$$

This is easy to see that the packing of bitwise encrypted ciphertext utilizes the homomorphic evaluation properties of the scheme. Thus, it potentially requires a higher dimension to contain all the bits together, as a longer ciphertext to be operated in the corresponding operations in the *SHE* scheme.

B. *Image Feature Detection*

Color is one of the most expressive features in image recognition. This paper focuses on *color descriptors* from *visual descriptors* in MPEG-7 standard [8]. In this standard, the most fundamental quality of visual content is color descriptor. There are four descriptors that are defined to describe color: three of them represent in the form of color histogram, and the rest one captures the spatial distribution of color in an image by discrete cosine transform with quantization as shown in the following parts. More detailed information regarding the *color descriptors* can be found in the references and other related MPEG-7 documents.

1) Color Descriptor: Histogram Descriptor: In color feature detection, one of the most fundamental and popular feature descriptors is histogram descriptor. Three prevalent color descriptors in MPEG-7: *Dominant Color Descriptor (DCD)*, *Scalable Color Descriptor (SCD)*, and *Color Structure Descriptor (CSD)* are all generated from a color histogram vector [8]. With respect to *DCD*, *SCD*, and *CSD*, the color histogram is an indispensable component [9]. Note that some color descriptors require color model conversion, e.g. color model transforms from RGB to YCbCr. It is simple to map one color space to another. For example, the RGB model can map into the YCbCr model by multiplying a 3 × 3 constant matrix. The details of color model conversions and scaling procedures in transforming a color histogram to an individual *color descriptor* are various according to different color model and color descriptors. Thus, the descriptions of these transformation are omitted in this paper.

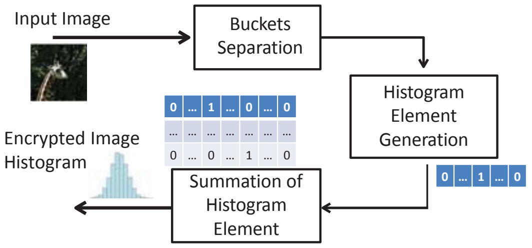

Input Image
Buckets Separation
Histogram Element Generation
0 ... 1 ... 0
0 ... 1 ... 0 ... 0
0 ... 0 ... 1 ... 0
Encrypted Image Histogram
Summation of Histogram Element

Figure 2. The Detection Process of Image Histogram

In order to compute the histogram vector in the ciphertext domain, the system needs to enable the comparisons of ciphertexts. The basic idea in enabling these homomorphic comparisons is to utilize a private information retrieval alike technique. Fig. 2 shows the workflow of privacy-preserving image color histogram detection algorithm. Note that for a simple expression, instead of $\mathbf{SH.Add}$ and $\mathbf{SH.Mult}$, we use $\oplus$, $\ominus$, and $\otimes$ to represent homomorphic addition, subtraction (very similar to addition), and multiplication. The privacy-preserving binary-based RGB image histogram extraction consists of three stages. The detailed procedures are described as follows:

Buckets Separation: We separate the space of the colors into buckets. In our instance, the RGB space is separated into 64 buckets by dividing each color space into 4 buckets. Note that the size of the buckets is tunable, which could be 512, 4096, etc. Here we use 64 buckets as an illustrative example. Recall that $\bar{x} = 1 - x$ when $x \in \{0, 1\}$. For an image ciphertext C, IP invokes $Bucketize(C)$ to generate bucket labels for each pixel. The bucketization algorithm is shown in Alg. 2. The output bucket labels $R_{bin1}, ..., B_{bin4}$ are encrypted binary values. For example, in each color domain, there are four binary labels and only one multiplication is the ciphertext of 1. This procedure enables the privacy-preserving comparison of different values to compute an encrypted histogram vector without leaking any additional information.

Histogram Element Generation: The histogram element $C_{h(x,y)}$ is computed through multiplications of each bucket labels: $C_{h(x,y)} = (R_{bin_1} \otimes G_{bin_1} \otimes B_{bin_1}, ..., R_{bin_4} \otimes G_{bin_4} \otimes B_{bin_4})$ A histogram element contains 64 multiplications of buckets' labels, in which only one of them is the ciphertext of integer 1, others are ciphertexts of 0.

Summation of Histogram Element: The summation of the histogram element $C_{h(x,y)}$ is the ciphertext of the color histogram vector $C_{H_I} = \sum_{x,y} C_{h(x,y)}$. Here, the length of C_{H_I} is 64, which could be 512 and etc. This procedure completes the computation process of histogram descriptor.

2) Color Descriptor: Color Layout Descriptor: The *Color Layout Descriptor (CLD)* captures the spatial layout of the representative colors on a grid superimposed over a region, which is based on coefficients of the DCT. This is a very compact descriptor being highly efficient in fast browsing and search applications. As shown in Fig. 3, the detection process of *CLD* consists of four stages:

Image Partitioning: In the image partitioning stage, the encrypted image is divided into 64 blocks to guarantee the invariance to resolution or scale. As an encrypted image is represented by a $[M \times N]$ matrix, the partitioned form contains 64 matrices, each of which has the size $[M/8 \times N/8]$.

Representative Color Selection: After the partitioning, a

Algorithm 2 Bucketization

```
Input:
    The encrypted image matrix: $C$;
Output:
    The bucket labels: $R_{bin1}, ..., B_{bin4}$;
for All $(x, y) \in I$ do
    $C(x, y)$, compute $c_{\bar{r_j}} = c_1 \ominus c_{r_j}$, similar to $c_{\bar{g_j}}$ and $c_{\bar{b_j}}$, where $j = \{1, ...7\}$, in which $c_1 = \mathbf{SH}.\mathbf{Enc}(pk, 1)$;
    Compute bucket labels: $R_{bin_1} = c_{\bar{r}_7} \otimes c_{\bar{r}_6}$; $R_{bin_2} = c_{r_7} \otimes c_{\bar{r}_6}$; $R_{bin_3} = c_{\bar{r}_7} \otimes c_{r_6}$; $R_{bin_4} = c_{r_7} \otimes c_{r_6}$; $G_{bin_1} = c_{\bar{g}_7} \otimes c_{\bar{g}_6}$; $G_{bin_2} = c_{g_7} \otimes c_{\bar{g}_6}$; $G_{bin_3} = c_{\bar{g}_7} \otimes c_{g_6}$; $G_{bin_4} = c_{g_7} \otimes c_{g_6}$; $B_{bin_1} = c_{\bar{b}_7} \otimes c_{\bar{b}_6}$; $B_{bin_2} = c_{b_7} \otimes c_{\bar{b}_6}$; $B_{bin_3} = c_{\bar{b}_7} \otimes c_{b_6}$; $B_{bin_4} = c_{b_7} \otimes c_{b_6}$;
end for
return The 12 bucket labels: $R_{bin1}, ..., B_{bin4}$;.
```

single representative color is selected from each block. The MPEG-7 standard recommends the use of the average of the pixel colors in a block as the corresponding representative color, which is simpler and achieves a sufficient description accuracy. The selection results in a tiny image icon of size 8×8 is $Avg = \frac{M \times N}{64} \sum_{x,y \ in \ block} I_i(x, y)$. The Average value represented in the ciphertext domain is $C_{Avg} = Q_{64} \sum_{x,y \ in \ block}^{\oplus} C(x, y)$. Note that Q_{64} represents a scaling factor for the average value. Once the tiny image icon is obtained, the color space conversion between RGB and YCbCr is applied.

DCT Transformation: In this stage, the luminance (Y), the blue and red chrominance (Cb and Cr) of a block are transformed into an 8×8 DCT matrix, consisting of three sets of 64 DCT coefficients. Different from the Paillier based scheme in [10], we propose a *SHE* based scheme to calculate the DCT in a $2D$ array. The formulas over the plaintext domain are shown as follows:

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} cos \frac{\pi(2m+1)p}{2M} cos \frac{\pi(2n+1)q}{2N} \quad (6)$$

where, $0 \leq p \leq M-1$, $0 \leq q \leq N-1$, A_{mn} is a scaling factor for DCT algorithm, and with

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}}, & p = 0 \\ \sqrt{\frac{2}{M}}, & 1 \leq p \leq M-1 \end{cases} \quad (7)$$

$$\alpha_q = \begin{cases} \frac{1}{\sqrt{N}}, & q = 0 \\ \sqrt{\frac{2}{N}}, & 1 \leq q \leq N-1 \end{cases} \quad (8)$$

The corresponding integer DCT is defined as

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} C(m, p) C(n, q) \quad (9)$$

where

$$C(m, p) = [Q_m cos \frac{\pi(2m+1)p}{2M}] \quad (10)$$

$$C(n, q) = [Q_n cos \frac{\pi(2n+1)q}{2N}] \quad (11)$$

Q_m and Q_n are suitable scaling factors for the cosine values.

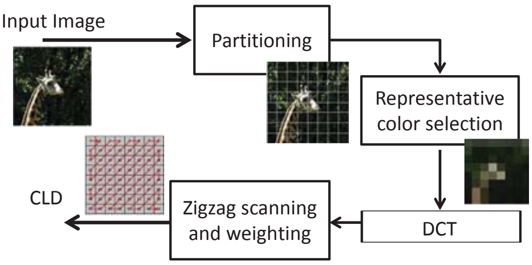


Figure 3. The Detection Process of Color Layout Descriptor

Now, IP needs to generate $C(m, p)$ and $C(n, q)$ for different partitioning parameters in DCT transformation. Since the matrices $C(m, p)$ and $C(n, q)$ are public parameters (independent from the content of the image), IP is able to generate corresponding ciphertexts of these matrices before performing the detection algorithm.

$$C_M = \{c_M | c_M(m, p) = \mathbf{SH}.\mathbf{Enc}(\mathsf{pk}, C(m, p))\} \quad (12)$$

$$C_N = \{c_N | c_N(n, q) = \mathbf{SH}.\mathbf{Enc}(\mathsf{pk}, C(n, q))\} \quad (13)$$

Here we use C_M and C_N to represent two encrypted cosine factor matrix. As shown in Alg. 3, IP generates the DCT coefficients $C_{B_{pq}}$ from the representative color, which is the average color C_{Avg} represented in the ciphertext domain.

Zigzag Scanning: A zigzag scanning is performed with the three sets of 64 DCT coefficients (the YCbCr color space), following the scheme presented in Fig 3. The purpose of the zigzag scanning is to group the low frequency coefficients of the 8x8 matrix. Finally, the system outputs the three sets of matrices $C_{B_{pq}}$ corresponding to the *CLD* of the encrypted image as the image feature. This completes the detection process of *CLD*.

C. Security Analysis

We evaluate the security of the proposed system by showing that the encrypted image data and the detected feature vectors in the ciphertext domain do not reveal any information to IP except the size and format of the image.

Theorem 3.1: If the underlying *SHE* scheme is IND-CPA secure, the proposed system protects private information of the image in the sense that two encrypted images are computationally indistinguishable.

Proof Sketch. In the proposed system, all the uploaded private images are encrypted under *SHE* scheme. The encrypted images are computationally indistinguishable, if the underlying *SHE* scheme is IND-CPA secure. Due to the space limitation, the reduction procedures are briefly described as follows: A probabilistic polynomial-time simulator is constructed to take the encrypted detection results from IP as input. It simulates IP's view during the execution of the corresponding image feature detection procedures. Depending on the input detection results, the simulator generates the corresponding ciphertexts that IP processes in Sec. III. Then, these ciphertexts are utilized to determine the values of the image data over the plaintext domain. The above reduction completes the proof of Theorem 3.1: The computational indistinguishablility of encrypted image follows the IND-CPA security of the RLWE based *SHE* scheme performed by SP.

Algorithm 3 DCT Transformation

```
Input:
    The representative color $C_{Avg}$;
Output:
    The DCT coefficients: $C_{B_{pq}}$;
Generate encrypted computation parameters:
    $C_M = \mathbf{SH.Enc}(pk, Q_m cos\frac{\pi(2m+1)p}{2M})$,
    $C_N = \mathbf{SH.Enc}(pk, Q_n cos\frac{\pi(2n+1)q}{2N})$,
    $C_{\alpha_p} = \mathbf{SH.Enc}(pk, \alpha_p)$, and
    $C_{\alpha_q} = \mathbf{SH.Enc}(pk, \alpha_p)$;
for All $(x, y) \in I$ do
    Compute $C_{B_{pq}} = C_{\alpha_p} \otimes C_{\alpha_q} \sum_{\oplus,m=0}^{M-1} \sum_{\oplus,n=0}^{N-1} C_{Avg}(m,n) \otimes C_M(m,p) \otimes C_N(n,q)$;
end for
return The DCT coefficients: $C_{B_{pq}}$;
```

Moreover, in the image feature detection phase, the cloud server gains no access to td or any other additional information about the image data. Thus, the security of the scheme relies on the polynomial LWE assumption. As in the work from [11], we only need to consider the security against the distinguishing attack of [12]. Specifically, for given security parameters of the *SHE* scheme, we can deduce the optimal value of Hermite factor δ:

$$lg(\delta) = \frac{1}{nlg(q)}(\frac{lg^2(\frac{cq}{\sigma})}{4}) \tag{14}$$

In order to determine specific parameters, the value of n and prime q are fixed. This allows us to utilize the heuristic runtime estimate of the distinguishing attack [12] to solve the Eq. 14 from [7]:

$$lgt_{Adv} = \frac{1.9}{lg\delta} - 110 \tag{15}$$

In fact, the best accuracy an adversary can achieve in this game is at the probability of $(1/2)^{log[q]}$, in which q belongs to the security parameter κ.

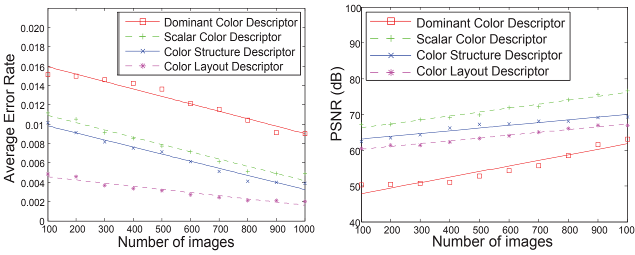
IV. Performance Evaluation

We implement the proposed system on the Java 6 Standard runtime environment update 24, on an Intel Core 2 Duo processor at 2.1GHz with 4 GB of RAM running Linux. The experiments are conducted on a real database from Caltech256 [13]. It contains 30,252 images, and 102 categories. The largest category contains 802 images. Without loss of generality, The key parameters of *SHE* scheme are the dimension of the polynomial n, the modulus q, the message space t and the bound of the multiplication D are configured as the security parameter κ. The instantiation of *SHE* scheme provides the security level of 128 bits.

A. Correctness Evaluation

The correctness of the proposed system is analyzed from the view of mathematics. We conduct the experiments regarding comparison between feature vectors detected in the ciphertext domain and the plaintext domain. Denote β to be a detected feature vector from the detection algorithm in the plaintext domain, and β' to be a feature vector from the proposed system after decoding. We define the error of the proposed system as:

$$Err_{\beta'} = \|\frac{\beta - \beta'}{\beta'}\|$$



(a) The average error rate.

(b) The PSNR of color features.

Fig. IV and Fig. IV illustrate the average error rate and PSNR of various *color descriptors*, respectively. In Fig. IV, the maximum error rate of descriptors in the proposed system is lower than 1.6%. The error is introduced by the fixed point numbers in numerical system implementation and the ciphertext comparison procedures of algorithms. Among various descriptors, the *DCD* has the highest error rate and lowest PSNR. This is caused by the high accuracy requirement on ciphertext comparison for detecting the dominant color.

B. Effectiveness Evaluation

For assessing the effectiveness of the proposed system in real applications, the experiments focus on the precision of the similarity comparison results. Basically, the experiments are analogous to the application of privacy-preserving image similarity comparison described in Sec. II. To generate a testing dataset, we randomly select 10 commonly used categories, each of which contains 100 images. Among these categories, 10 images per category are selected as benchmark images owned by IP. The rest of images are considered as users' images from SP. Each of the users' images and the benchmark images is encoded and the corresponding feature vector (*CLD*) is detected in the ciphertext domain by the proposed system. Then these encrypted feature vectors of users' images and benchmark images are compared by computing the Euclidean distance between them in the ciphertext domain. The results regarding the top-k similar images are examined here.

In the experiments, we solely compare the results of similarity comparison in the ciphertext domain with the results of comparison over plaintext domain. Based on criterion in [14], we define the precision criterion of the similarity comparison results as:

$$precision = \frac{\#\{relevant\ image \cap\ retrieved\ image\}}{\#\{\ retrieved\ image\}}$$

The *relevant image* is defined as the ground truth from the comparison over the plaintext domain. Table 4 shows the *precision* of the proposed system of top-k similar images to the benchmark images, where $k = 1, 2, 3, 5, 10$. When $k \leq 2$, the average precision is above 95%. The comparison results over different domains are almost the same. The proposed system does not compromise the precision of image similarity comparison. When $k \geq 3$, the precision of the system reduces, but remains above 80% in average. It is expected that the

Category	k=1	k=2	k=3	k=5	k=10
ant	98%	96%	95%	95%	90%
airplane	97%	95%	90%	82%	70%
anchor	97%	93%	90%	88%	84%
chair	94%	91%	85%	83%	63%
camera	98%	90%	86%	82%	76%
motorbikes	95%	91%	84%	75%	69%

Figure 4. The average precision of *CLD* on Caltech256 with 10 categories

precision can be improved, if advanced feature detection algorithm or classifier are further implemented.

C. Efficiency Evaluation

To enhance the efficiency, we implement batching techniques from [7] and [15]. Given n images that being selected, the number of pixels that can be batched into one ciphertext of 1024 bits is $1024/(b + log_2 n)$, where b is the total number of bits for representing numbers. In the experiments, we manage to batch 15 elements to adapt the accuracy concern. The parameters are configured to fit the proposed system.

In order to evaluate the efficiency of the proposed system, the experiments are conducted regarding the time cost of data preprocessing phase and image feature detection phase. Fig. 5 plots the time spent in the two phases of detection algorithms with different types for an image with size of 438×256. The plot shows that the *CLD* algorithm has the highest time complexity, since it requires the highest degree of multiplication in transformation stage and color space conversion procedure. The time cost of the proposed system remains to be fully practical, i.e. about the order of minute. However, compared with the basic computation over *SHE* scheme in [7], it has been improved with an acceptable efficiency.

V. Related Work

The global image feature detection system in the ciphertext domain is proposed to explore the prospects of private image computation outsourcing, inspired by works in [4], [5], [16]. The secure image retrieval allows content-based search over encrypted multimedia databases. Therefore, it offers flexible approaches to manage private multimedia collections online. In [4], the features extracted from image are encrypted in a distance-preserving scheme to enable the direct comparisons for similarity evaluation. In [16], the current search indexes for multimedia data are encrypted while achieving an efficient searching functionality. However, as to the common privacy-preserving computation scenarios, it requires to protect both input and output privacy of the image while outsourcing the computation procedures. The most similar work to this paper is Hsu's work [5], [6]. They propose a scheme based on the homomorphic properties of Paillier cryptosystem to enable a secure SIFT computation over the ciphertext. However, their scheme has a serious drawback in their homomorphic comparison algorithm, which leads to an impractical complutational complexity on cloud side (e.g. Assuming a 512-bits modulus, 2^{256} multiplications are required to perform one comparison).

VI. Conclusion and Future Work

In this paper, we proposed a privacy-preserving global feature detection system over encrypted image data. The proposed system transforms the color descriptor detection algorithms designed for plaintext into circuit-level homomorphic operations for ciphertext. We constructed a prototype system for detecting various color descriptors. The security of the proposed system was analyzed and proved. The experiments based on Caltech256 image database show that the proposed system is correct and effective for different color descriptors and categories of image. As our ongoing work, we will continue to research on privacy-preserving image detection algorithms for the effective utilization over encrypted data. Interesting problems include designing more complicated privacy-preserving local feature detection algorithms like SIFT, and analyzing the influence of privacy-preserving mechanisms to the image retrieval algorithms.

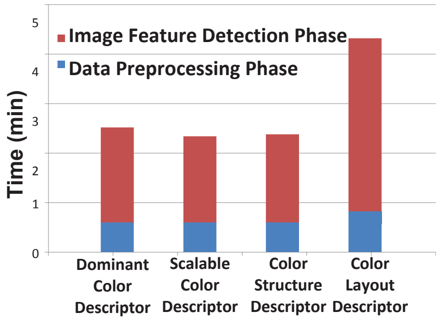


Figure 5. Time Complexity of Encryption Phase and Feature Detection Phase

References

[1] S. Lawson, "SSDs boost instagram's speed on amazon EC2," *Computer World*, Web. 21 Sep. 2012.

[2] G. Greenwald, and E. MacAskill, "NSA prism program taps in to user data of apple, google and others," *The Guardian*, 7.6(2013): 1–43.

[3] C. Wang, et al. "Toward secure and dependable storage services in cloud computing," *IEEE Trans. Services Computing*, 5.2(2012): 220–232.

[4] W. Lu, et al., "Secure image retrieval through feature protection," in *Proc. of ICASSP' 09*, 2009.

[5] C.-Y. Hsu, et al. "Image feature extraction in encrypted domain with privacy-preserving sift," *IEEE Trans. Image Processing*, 21.11(2012): 4593–4607.

[6] C.-Y. Hsu, et al. "Homomorphic encryption-based secure SIFT for privacy-preserving feature extraction," in *Proc. of SPIE'11*, 2011

[7] M. Naehrig, et al. "Can homomorphic encryption be practical?" in *Proc. of CCSW'11*, 2011.

[8] T. Sikora, "The mpeg-7 visual standard for content description-an overview," *IEEE Trans. CSVT*, 11.6(2001): 696–702.

[9] K. Ivanova, et al. "Features for art painting classification based on vector quantization of mpeg-7 descriptors," *Data Engineering and Management*, Springer, pp. 146–153, 2012.

[10] T. Bianchi, et al. "Encrypted domain dct based on homomorphic cryptosystems," *EURASIP Journal on Information Security*, 2009(2009): 1–1.

[11] R. Lindner, and C. Peikert, "Better key sizes (and attacks) for lwe-based encryption," *Topics in Cryptology–CT-RSA 2011*, Springer, pp. 319–339, 2011.

[12] D. Micciancio, and O. Regev, "Worst-case to average-case reductions based on gaussian measures," *SIAM Journal on Computing*, 37.1(2007): 267–302.

[13] G. Griffin, et al. "Caltech-256 object category dataset," *Technical Report. California Institute of Technology*, 2007.

[14] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," in *Proc. of CVPR'04*, 2004.

[15] M. Yasuda, et al. "Packed homomorphic encryption based on ideal lattices and its application to biometrics," in *Security Engineering and Intelligence Informatics*, Springer, pp. 55–74, 2013.

[16] W. Lu, et al. "Enabling search over encrypted multimedia databases," in *Proc. of SPIE'09*, 2009.

[17] J. Bringer, et al. "Privacy-preserving biometric identification using secure multiparty computation: An overview and recent trends," *Signal Processing Magazine*, 30.2(2013): 42–52.