Towards Privacy-preserving Content-based Image Retrieval in Cloud Computing

Zhihua Xia, *Member, IEEE,* Yi Zhu, Xingming Sun, *Senior Member, IEEE,*
Zhan Qin, *Member, IEEE* and Kui Ren, *Senior Member, IEEE*

Abstract—Content-based image retrieval (CBIR) applications have been rapidly developed along with the increase in the quantity, availability and importance of images in our daily life. However, the wide deployment of CBIR scheme has been limited by its the severe computation and storage requirement. In this paper, we propose a privacy-preserving content-based image retrieval scheme, which allows the data owner to outsource the image database and CBIR service to the cloud, without revealing the actual content of the database to the cloud server. Local features are utilized to represent the images, and earth mover's distance (EMD) is employed to evaluate the similarity of images. The EMD computation is essentially a linear programming (LP) problem. The proposed scheme transforms the EMD problem in such a way that the cloud server can solve it without learning the sensitive information. In addition, local sensitive hash (LSH) is utilized to improve the search efficiency. The security analysis and experiments show the security and efficiency of the proposed scheme.

Index Terms—Cloud computing, searchable encryption, image retrieval, local feature, earth mover's distance

1 Introduction

THANKS to low-cost storage and easy web hosting, the world has witnessed a tremendous growth in the quantity, availability and importance of images in our daily life. Images start to play a crucial role in diverse fields like medicine, journalism, advertising, design, education and entertainment, etc. The need for efficient storage and retrieval of images is reinforced by the increase of large-scale image databases among all kinds of areas. Meanwhile, as an emerging technology, Content-based Image Retrieval (CBIR) shows enough promise and maturity to be helpful in many real-world image retrieval/matching applications. For example, clinicians may use CBIR to retrieve the similar cases of the patients to facilitate the clinical decision-making process [1]. As another example, law enforcement agencies usually compare the evidence from the crime scene with the records in their archives [2]. However, such kind of CBIR service is intensive in both computation and storage intensive. A large image database usually consists of millions of images. Sometimes, one digital image might contain more than 20 million dimensions and its size could be above 40 megabytes, such as mammography images [3]. Moreover, CBIR typically has high computational complexity due to the high dimensionality of image data.

Cloud computing offers a great opportunity to provide on-demand access to ample computation and storage resource, which makes it a primary choice for image storage and CBIR outsourcing. By deploying such image retrieval outsourcing, the data owner is no longer needed to maintain the image database locally. An authorized data user can query the cloud for CBIR service without interacting with the data owner. Despite the tremendous benefits, privacy becomes the biggest concern about CBIR outsourcing. For example, the patients will not want to disclose their medical images. In fact, the Health Insurance Portability and Accountability Act (HIPAA) sets legal requirements to protect patients' privacy.

Contribution. In this paper, we study the privacy-preserving CBIR outsourcing problem and present a practical solution. We exploit techniques from security, image processing and information retrieval domains to achieve secure and efficient searching over encrypted images. The proposed scheme supports local-feature based CBIR with the earth mover's distance (EMD) as similarity metric. In particular, a secure transformation is designed so that the cloud server can solve the EMD problem with the privacy preserved. Local sensitive hash in employed to achieve constant search efficiency.

The rest of this paper is organized as follows. Section 2 summaries the related works. Section 3 introduces the system architecture and preliminaries. The scheme design is presented in Section 4. The security of the proposed scheme is analyzed in Section 5. In Section 6, we implement proposed scheme and study its efficiency. Finally, a conclusion is given in Section 7.

2 Related works

Searchable symmetric encryption (SSE) on text domain has been widely studied in the literature. Song *et al.* [4] proposed the first SSE scheme, and the search time of their scheme is linear to the size of the data collection. Goh [5] proposed formal security definitions for SSE and designed a scheme based on Bloom filter. The search time of Goh's scheme is $O(n)$, where n is the cardinality of the document

- *Zhihua Xia, Yi Zhu, Xingming Sun are with the Jiangsu Engineering Center of Network Monitoring, School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, China. E-mail: xia_zhihua@163.com; zhuyi1344@gmail.com; sunnudt@163.com*
- *Zhan Qin and Kui Ren are with the Department of Computer Science and Engineering, State University of New York at Buffalo. E-mail: zhanqin@buffalo.edu; kuiren@buffalo.edu*

Manuscript received Feb. 15, 2015; revised Aug. 17, 2015.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

collection. Curtmola *et al.* [6] proposed two schemes (SSE-1 and SSE-2) which achieve the optimal search time. Their SSE-1 scheme is secure against chosen-keyword attacks (CKA1) and SSE-2 is secure against adaptive chosen-keyword attacks (CKA2). The aforementioned works are some early ones which only support Boolean search to identify whether or not a query term is present in an encrypted document. Afterward, abundant works are proposed under different threat models to achieve various search functionality, such as similar search [7], [8], [9], multi-keyword ranked search [10], [11], [12], dynamic search [12], [13], [14], etc. However, few of these schemes are straightforwardly appropriate to an image retrieval task.

Shashank *et al*. [15] proposed a private content-based image retrieval (PCBIR) scheme which protected the privacy of the query images, but directly exposed the unencrypted image database to the server. Some researchers devoted to outsourcing the computation of image feature extraction to the could server in a privacy-preserving manner [16], [17], [18], [19], which can be the key technique to the privacy-preserving CBIR. Nevertheless, the index construction and similar search based on the encrypted features are needed to be further addressed.

To the best of our knowledge, Lu *et al*. [20] constructed the first image retrieval scheme over encrypted images. The authors extracted the visual words to represent the images and calculated the Jaccard similarity between two sets of visual words to evaluate the similarity of the two corresponding images. Order Preserving Encryption and Min-Hash Algorithm were employed to protect the information of visual words. In another work, Lu *et al.* [21] investigated three image feature protection techniques, i.e. bitplane randomization, random projection and randomized unary encoding. The features encrypted with bitplane randomization and randomized unary encoding can be used to calculate Hamming Distance in encryption domain. The features encrypted with random projection can be used to calculate L1 distance in encryption domain. Similar with [21], Cheng *et al.* [22] designed a CBIR system by utilizing the bitplane randomization and random projection. Ferreira *et al.* [23] proposed an image encryption method which is suitable for privacy-preserving image retrieval. In Ferreira *et al.*'s scheme, the color information is encrypted by deterministic algorithm to support content-based image retrieval, and the texture information is encrypted by probabilistic algorithm for better security. The encrypted color information can be utilized to construct the searchable index.

The aforementioned privacy-preserving CBIR schemes mainly focused on the global-feature based methods. Compared with the global feature, local-feature based CBIR usually retrieves more accurate results, but need to be combined with more complicated distance metric, such as earth mover's distance (EMD) [24], [25]. In this paper, we propose a practical framework for privacy-preserving CBIR outsourcing. The proposed scheme supports local-feature based CBIR with EMD as similarity metric .

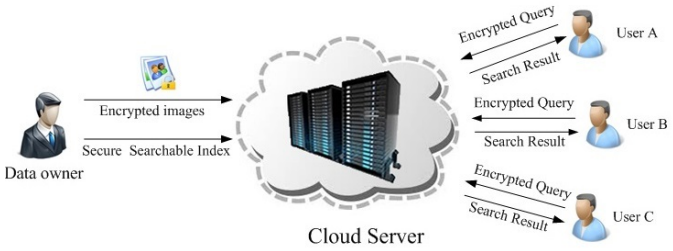


Fig. 1: Architecture of proposed scheme

3 System Overview and Preliminaries

3.1 System architecture

As shown in Fig. 1, the proposed scheme involves three types of entities: data owner, data users and cloud server. The data owner holds a large-scale image database $\mathcal{M}$=$\{m_1,...,m_n\}$ to be outsourced, where n is the image number of the database. The data owner generates a searchable index for the image database $\mathcal{M}$. For privacy-preserving, the data owner needs to encrypt the image database and the search index, and then outsources the encrypted image database and index to the cloud. Since it is expected that the cloud can provide the CBIR service without interacting with the data owner once the database is outsourced, we need to construct a special searchable encryption scheme that supports CBIR over encrypted data. During the CBIR query phase, the authorized user submits an encrypted query trapdoor to the cloud server. Then, the cloud server compares the similarities between the query image and the images in the database, and returns the encrypted similar images to the data user. Finally, the authorized user decrypts the received images.

3.2 Security model

In this paper, we consider the semi-honest (also known as honest-but-curious) cloud server, who will correctly follow the designated protocol specification, but keep and analyse the communication history, trying to derive sensitive information. The data owner and authorized users are always trusted. The scheme is designed to prevent the cloud server from knowing the content of image database and users' queries. Similar to searchable encryption scheme, we don't consider the information leakage due to access pattern. For instance, if images m_i and m_j are returned as the search results of the same query, it is easy to deduce that images m_i and m_j are similar to each other. In fact, this issue can be effectively solved by applying an existing ORAM scheme, e.g. [26].

3.3 Notations

- $\mathcal{M}$, $\mathcal{S}$, Ξ: image database, signature database, centroid database.
- $K_{\mathcal{M}}$, $K_{\mathcal{S}}$, K_{Ξ}: secret key for image, signature, centroid database encryption.
- K_j, g_j, $j = 1,\ldots,\lambda$: secret keys, LSH functions for index construction.
- ϕ: one way hash function.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

- s_q: signature of query image m_q.
- $\mathcal{I}$, $\mathcal{T}_q$: secure searchable index, trapdoor for the query image m_q.
- $\mathcal{S}_q$, $\mathcal{M}_q$: set of retrieved signatures and set of retrieved images to query image m_q.
- Ω: transformed EMD optimization problem set.
- $ID(\cdot)$: the identity of image in the database.

3.4 Bag-of-words model

CBIR usually involves extraction of visual features and search in the visual feature space for similar images. So it has two intrinsic challenges. The first challenge is how to mathematically describe an image, which is referred to as the feature extraction step. A feature is defined to capture a certain visual property of an image, either globally for the entire image or locally for a small group of pixels. The commonly used global features contain color histograms [27], texture features [28], shape features [29], etc. The advantage of global feature is its high speed for both extracting features and computing similarity. However, as a trade-off between accuracy and computation, the global feature based CBIR is often too rigid to represent an image. Local features based on local invariants such as corner points or interest points, are typically more robust for spatial transformation and usually retrieve more accurate results. The most popular local features contain Scale-invariant feature transform (SIFT) features [30] and image patches [31].

For local feature based CBIR, a set of local features are extracted from local regions of an image. One popular CBIR approach using local features is called bag-of-words model. In this model, local features are extracted from all images in the database and then jointly clustered. The cluster centers are used as 'words' to form the vocabulary. Note that each local feature is a feature vector consisting of multiple elements. After being clustered, the vectors of cluster centers will be kept. For each of other local features, only the identifier of the most close cluster center is kept. Then, the image $m_t \in \mathcal{M}$ can be denoted as a bag of words:

$$s_t = \{(c_1^{(t)}, w_1^{(t)}), \ldots, (c_i^{(t)}, w_i^{(t)}), \ldots, (c_{k_t}^{(t)}, w_{k_t}^{(t)})\}, \quad (1)$$

where c_i denotes a cluster center, w_i is the number of local features that are clustered to the class centered by c_i in m_t, and k_t is the total number of relevant cluster centers to image m_t. In this paper, s_t is regarded as the signature of the image m_t. Then, the similarity between a query image m_q and an image $m_t \in \mathcal{M}$ can be calculated as the earth mover's distance between their signatures s_q and s_t. Please note that c_i is a feature vector whose dimensionality depends on the specific feature extraction method.

3.5 Earth mover's distance

The earth mover's distance can be applied to evaluate the similarity between the distributions [24], [25]. Given two distributions, the distribution with smaller sum of weights can be viewed as a mass of earth which rightly spread in space, and the distribution with larger sum of weights can be viewed as an array of holes in the same space. The EMD measures the minimal cost of moving all the earth into the holes. An unit of work will be counted when a unit of earth is transported for a unit of distance. The EMD transforms the matching problem to the transportation problem. Two distributions have the least transportation cost can be viewed as the most similar ones. Given two image signatures $s_t = \{(c_1^{(t)}, w_1^{(t)}), \ldots, (c_i^{(t)}, w_i^{(t)}), \ldots, (c_{k_t}^{(t)}, w_{k_t}^{(t)})\}$, for $t = 1, 2$, the EMD is defined in terms of an optimal flow $\mathcal{F}$=$\{f_{i,j}\}$, which minimizes the work required to move earth from one signature to another, denoted as $W(s_1, s_2, \mathcal{F}) = \sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j}d_{i,j}$, where $d_{i,j} = d(c_i^{(1)}, c_j^{(2)})$ is the distance between $c_i^{(1)}$ and $c_j^{(2)}$, e.g. the Euclidean distance in $\mathcal{R}^d$. Meanwhile, the flow $f_{i,j}$ must satisfy the following constraints:

- $f_{i,j} \geq 0, 1 \leq i \leq k_1, 1 \leq j \leq k_2$;
- $\sum_{i=1}^{k_1} f_{i,j} \leq w_j^{(2)}, 1 \leq j \leq k_2$;
- $\sum_{j=1}^{k_2} f_{i,j} \leq w_i^{(1)}, 1 \leq i \leq k_1$;
- $\sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j} = \min(\sum_{i=1}^{k_1} w_i^{(1)}, \sum_{j=1}^{k_2} w_j^{(2)})$.

Once the optimal flow $f_{i,j}^*$ is found, the EMD between s_1 and s_2 is defined as

$$EMD(s_1, s_2) = \frac{\sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j}^* d_{i,j}}{\sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j}^*}, \quad (2)$$

where numerator is the minimal transportation cost and the denominator is the total movement. The signatures can always be converted to proper probability distributions by normalizing the weights to add up to 1 [32]. Hence, we can simplify the constraints to:

- $f_{i,j} \geq 0, 1 \leq i \leq k_1, 1 \leq j \leq k_2$;
- $\sum_{i=1}^{k_1} f_{i,j} = w_j^{(2)}, 1 \leq j \leq k_2$;
- $\sum_{j=1}^{k_2} f_{i,j} = w_i^{(1)}, 1 \leq i \leq k_1$;
- $\sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j} = \sum_{i=1}^{k_1} w_i^{(1)} = \sum_{j=1}^{k_2} w_j^{(2)} = 1$.

The EMD distance can be converted to an LP optimization problem as follows:

$$\begin{aligned} minimize \quad & \sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j}d_{i,j}, \\ subject\ to \quad & f_{i,j} \geq 0, 1 \leq i \leq k_1, 1 \leq j \leq k_2, \\ & \sum_{i=1}^{k_1} f_{i,j} = w_j^{(2)}, 1 \leq j \leq k_2, \\ & \sum_{j=1}^{k_2} f_{i,j} = w_i^{(1)}, 1 \leq i \leq k_1, \\ & \sum_{i=1}^{k_1}\sum_{j=1}^{k_2} f_{i,j} = 1. \end{aligned} \quad (3)$$

4 THE PROPOSED SCHEME DESIGN

In this section, we describe the design of our privacy-preserving CBIR scheme. Firstly, we introduce the framework of the proposed scheme. Next, we introduce two technologies which will be employed in index construction. Finally, we present the details of the whole scheme.

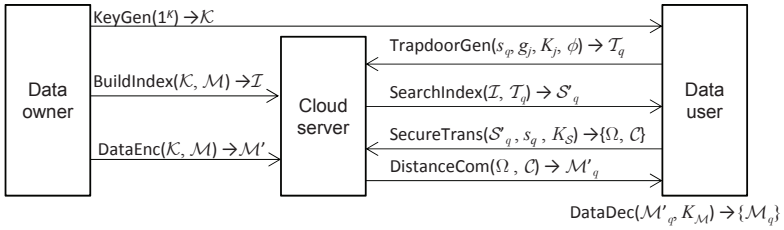


Fig. 2: Flow chart of the proposed scheme

4.1 The framework of the scheme

The proposed scheme consists of a tuple of probabilistic polynomial time algorithms Γ={KeyGen, BuildIndex, DataEnc, TrapdoorGen, SearchIndex, SecureTrans, DistanceCom, DataDec}. The flow chart of the proposed scheme is illustrated in Fig. 2.

In the initiation phase, given an image database $\mathcal{M}$, the data owner runs KeyGen(1^κ), BuildIndex$(\mathcal{K}, \mathcal{M})$ and DataEnc$(\mathcal{K}, \mathcal{M})$ to generate $\mathcal{K}$, $\mathcal{I}$ and $\mathcal{M}'$. Then the data owner outsources $\mathcal{I}$ and $\mathcal{M}'$ to the cloud server and sends the $\mathcal{K}$ to the authorized data users.

In the image retrieval phase, an authenticated data user runs TrapdoorGen(s_q, g_j, K_j, ϕ) to generate trapdoor $\mathcal{T}_q$, and then submits $\mathcal{T}_q$ to the cloud server. The cloud server runs SearchIndex$(\mathcal{I}, \mathcal{T}_q)$ to obtain a set of encrypted signature $\mathcal{S}'_q$. This means that the corresponding set of images $\mathcal{M}_q \subseteq \mathcal{M}$ is a subset of images that may be similar to the query image m_q. The cloud server then sends $\mathcal{S}'_q$ to the data user. Upon receiving $\mathcal{S}'_q$, the data user runs SecureTrans$(\mathcal{S}'_q, s_q, K_\mathcal{S})$ to construct the set of encrypted EMD problem Ω, and then sends Ω to cloud server. Upon receiving the set of encrypted EMD problem Ω, the cloud server runs DistanceCom(Ω) to find the most similar images to the query image, and then sends the set of encrypted image set $\mathcal{M}'_q$ to the data user. Finally, the data user runs DataDec$(\mathcal{M}'_q, \mathcal{K}_\mathcal{M})$ to get the set of similar images $\mathcal{M}_q$.

4.2 Local sensitive hash on signature centroid

The calculation of EMD problem between the query image and the images in database will cause a time complexity linear to the cardinality of image set. It will be unusable in a real world application with the huge number of images. Thus, we need a method to filter out the dissimilar images quickly, and then only calculate the EMD problems with the remaining images. In this paper, the local sensitive hash calculated with signature centroid is employed to filter out the dissimilar images quickly.

Centroid of the signature. A lower bound of EMD between two signatures is the Euclidean distance between their centroids [24]. The centroid of the signature s_t is defined as

$$\xi_t = \sum_{i=1}^{k_t} c_i^{(t)} w_i^{(t)}. \tag{4}$$

Note that the centroid of the signature is a point in multiple dimension space. Then, the lower bound of the EMD between the signature of s_t and s_q can be defined as

$$EMD(s_t, s_q) = \sum_{i=1}^{k_t}\sum_{j=1}^{k_q} f_{i,j}^* d_{i,j} \geq \| \sum_{i=1}^{k_t} c_i^{(t)} w_i^{(t)} - \sum_{j=1}^{k_q} c_j^{(q)} w_j^{(q)} \|_2 .$$

TABLE 1: An example of j-th hash table

$\phi(K_j, B_{j,1})$	$ID(m_7), ID(m_9), ID(m_{12}), ID(m_{17})$
$\phi(K_j, B_{j,2})$	$ID(m_{13}), ID(m_{24}), ID(m_{35}), ID(m_{67})$
$\phi(K_j, B_{j,3})$	$ID(m_{27}), ID(m_{49}), ID(m_{62}), ID(m_{73})$
$\cdots$	$\cdots$
$\phi(K_j, B_{j,N_j})$	$ID(m_{47}), ID(m_{59}), ID(m_{92}), ID(m_{117})$

Local sensitive hash. Locality Sensitive Hash (LSH) has the property that close items will collide with a higher probability than distant ones, which can be successfully applied to approximate queries of vector space [33]. A hash function family $\mathcal{H} = \{h : \mathcal{U} \to \mathcal{R}\}$ is called $(\rho, \alpha\rho, p_1, p_2)$-sensitive for any $x, y \in \mathcal{U}$ if

$$\begin{cases} Pr\{h(x) = h(y)\} \geq p_1 & for \quad d(x,y) \leq \rho \\ Pr\{h(x) = h(y)\} \leq p_2 & for \quad d(x,y) \geq \alpha\rho, \end{cases}$$

where the constant $\alpha > 1$ and probabilities $p_1 > p_2$.

To enlarge the gap between p_1 and p_2, multiple hash functions can be jointly used to construct another function family $\mathcal{G} = \{g : \mathcal{U} \to \mathcal{R}^\lambda\}$ where $g(v) = (h_1(v), h_2(v), ...h_\lambda(v))$ is the concatenation of λ LSH functions $h_i \in \mathcal{H}$.

Specially, the LSH families based on p-stable distribution [33] allow each hash function $h_{a,b} : R^l \to Z$ to map a l-dimensional vector v into a integer. The p-stable LSH can be defined as $h_{a,b}(v) = \lfloor \frac{(a \cdot v + b)}{\theta} \rfloor$, where a is a l-dimensional random vector with chosen entries following a p-stable distribution, b is a real number chosen uniformly from the range $[0, \theta)$ and θ is a constant. In our scheme, LSH based on p-stable distribution is utilized to construct the pre-filter tables to improve the search efficiency.

Local sensitive hash on signature centroid. For each image m_t, we calculate its signature s_t and signature centroid ξ_t. We define the $\Xi = \{\xi_t\}_{t=1}^n$ as the database of signature centroids. Then, we map the centroids with LSH. According to the property of LSH, the similar centroids will be mapped into the same bucket, i.e. obtaining the same hash value.

In practice, to provide more possible results, the hash process will be repeated L times with different $g_i \in \mathcal{G}, i = 1, ..., L$, generating L pre-filter tables (i.e. hash tables). The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

Algorithm 1. BuildIndex

Input: the centroid database Ξ, the set of hash functions $\{g_i\}_{i=q}^{L}$, the set of keys for hash value $\{K_i\}_{i=q}^{L}$, the one way hash function ϕ.
Output: secure index $\mathcal{I}$.
1. For each $j = 1, \cdots, L$, data owner builds the j-th hash table by applying function g_j over all the elements in centroid database Ξ. One example is shown in Tab. 1.
2. For each $j = 1, \cdots, L$, data owner picks a random key K_j and replaces each LSH hash digest $B_{j,i}, i = 1, \cdots, N_j$ in the j-th hash table with $\phi(K_j, B_{j,i})$.
3. For each $j = 1, \cdots, L$, data owner further fills the j-th hash tables with identifiers of corresponding images $ID(m_i)$.

Return: Index $\mathcal{I}$ consists of L secure hash tables.

set of search results is the union from the multiple hash tables. Let $\{B_{j,b}\}_{j\in[1,L],b\in[1,N_j]}$ denote the derived bucket (i.e. hash value) set of LSH hash functions, where N_j refers to the total number of buckets in the j-th pre-filter table. An example of the j-th pre-filter table is illustrated in Table 1. The parameters L and λ can be adjusted so that it performs favourably, as suggested by the methodology from LSH community [34].

Please note that, LSH does not necessarily have the one-way property. Therefore, we cannot directly outsource the pre-filter tables to the cloud server as it may leak information about the centroid database. To enhance the security, a one-way hash function ϕ is employed to protect the buckets, as shown in Table 1.

4.3 LP transformation on EMD problem

The image feature vectors in plaintext may reveal information about image content. For example, a color histogram with large blue component would indicate the likely presence of sky or ocean, and the shape descriptors may disclose the information about the likely object in the image [20]. The data user wants to leverage the computation power of cloud server to compute EMD. However, the unencrypted signatures may reveal content of the images. In this subsection, we explain how to make the cloud server securely compare the EMDs of different images to the query image without revealing the sensitive information. Given the query image m_q and an image $m_t \in \mathcal{M}$, we use the matrix expression of Formula (3) for concise explanation:

$$\begin{aligned} minimize \quad & \mathbf{c}^T\mathbf{x}, \\ subject\ to \quad & \mathbf{U}\mathbf{x} = \tau, \\ & \mathbf{V}\mathbf{x} = \mathbf{E} \end{aligned} \tag{5}$$

where $\mathbf{c}$ is an $k_t k_q \times 1$ distance vector whose elements are $d_{i,j}$ in Formula (3), $\mathbf{x}$ is an $k_t k_q \times 1$ flow vector whose elements are $f_{i,j}$ in Formula (3). Here, the symbol k_t is the total number of cluster centers associated to the signature s_t, and k_q is the total number of cluster centers associated to the signature s_q. $\mathbf{U}$ is an $1 \times k_t k_q$ known matrix to transform $\sum_{i=1}^{k_t}\sum_{j=1}^{k_q} f_{i,j} = 1$, and all the elements of $\mathbf{U}$ are equal to 1. $\mathbf{V}$ is an $(k_t + k_q) \times k_t k_q$ known matrix to transform $\sum_{i=1}^{k_t} f_{i,j} = w_j^{(q)}$, $1 \le j \le k_q$ and $\sum_{j=1}^{k_q} f_{i,j} = w_i^{(t)}$, $1 \le i \le k_t$. An example of $\mathbf{V}$ with $k_t = 3, k_q = 4$ is illustrated in Fig. 3. τ is the minimal total weight, and in the case of normalized weights, τ is equal to 1. $\mathbf{E} = ((w_i^{(t)})_{i=1}^{k_t}, (w_j^{(q)})_{j=1}^{k_q})$ is the weight vector. We denote the problem in Formula (5) by a tuple $\Psi = (\mathbf{c}, \mathbf{U}, \tau, \mathbf{V}, \mathbf{E})$.

$$\begin{pmatrix} 1,1,1,0,0,0,0,0,0,0,0,0 \\ 0,0,0,1,1,1,0,0,0,0,0,0 \\ 0,0,0,0,0,0,1,1,1,0,0,0 \\ 0,0,0,0,0,0,0,0,0,1,1,1 \\ 1,0,0,1,0,0,1,0,0,1,0,0 \\ 0,1,0,0,1,0,0,1,0,0,1,0 \\ 0,0,1,0,0,1,0,0,1,0,0,1 \end{pmatrix}$$

Fig. 3: An example of $\mathbf{V}$ with $k_t = 3, k_q = 4$

Our design goal is to find a secure and efficient transformation, using a secret key K_T, to make cloud compute the EMD optimization problem while protecting the input and output privacy. It means that the cloud solves the randomly transformed EMD optimization problem without knowing the input distance vector $\mathbf{c}$, weight vector $\mathbf{E}$ and the output flow vector $\mathbf{x}$. Our secure transformation contains two main steps. First of all, to protect the distance vector $\mathbf{c}$ and output flow vector $\mathbf{x}$, we perform the transformation $\mathbf{x}$=$\mathbf{\Lambda y}$-$\mathbf{r}$, where $\mathbf{\Lambda}$ is an $k_t k_q \times k_t k_q$ non-singular matrix and $\mathbf{r} = (r_1, r_2, ..., r_{k_t k_q})^T$ is an $k_t k_q \times 1$ vector. The original problem is then transformed to:

$$\begin{aligned} minimize \quad & \mathbf{c}^T\mathbf{\Lambda y} - \mathbf{c}^T\mathbf{r}, \\ subject\ to \quad & \mathbf{U\Lambda y} = \tau + \mathbf{Ur}, \\ & \mathbf{V\Lambda y} = \mathbf{E} + \mathbf{Vr} \end{aligned} \tag{6}$$

Next, we multiply the $(k_t + k_q) \times (k_t + k_q)$ generalized permutation $\mathbf{G}$ to protect $\mathbf{E}$ and multiply a real positive value γ to protect optimal value. Then, the original problem is transformed to:

$$\begin{aligned} minimize \quad & \gamma\mathbf{c}^T\mathbf{\Lambda y} - \gamma\mathbf{c}^T\mathbf{r}, \\ subject\ to \quad & \mathbf{U\Lambda y} = \tau + \mathbf{Ur}, \\ & \mathbf{GV\Lambda y} = \mathbf{G}(\mathbf{E} + \mathbf{Vr}) \end{aligned} \tag{7}$$

As the constant term $\gamma\mathbf{c}^T\mathbf{r}$ does not affect the optimal solution, the final transformed problem can be formed as:

$$\begin{aligned} minimize \quad & \mathbf{c'}^T\mathbf{y}, \\ subject\ to \quad & \mathbf{U'y} = \tau', \\ & \mathbf{V'y} = \mathbf{E'} \end{aligned} \tag{8}$$

where $\mathbf{c'}^T = \gamma\mathbf{c}^T\mathbf{\Lambda}$, $\mathbf{U'} = \mathbf{U\Lambda}$, $\tau' = \tau + \mathbf{Ur}$, $\mathbf{V'} = \mathbf{GV\Lambda}$, $\mathbf{E'} = \mathbf{G}(\mathbf{E} + \mathbf{Vr})$. This problem has similar structure to the original problem in Formula (5). We use $\Omega = (\mathbf{c'}, \mathbf{U'}, \tau', \mathbf{V'}, \mathbf{E'})$ to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

denote the secure transformed problem, where the secret transformation key $K_T = (\mathbf{G}, \mathbf{\Lambda}, \mathbf{r}, \gamma)$. The whole transformation process is illustrated in Algorithm 2.

After solving the EMD optimization problem, we also want cloud to compute EMD in Formula (2) so as to sort the results. The numerator of EMD equation is the optimal values of the original problem in Formula (3) and the denominator is the sum of the elements of optimal solution. However, the cloud server solves the transformed problem in Formula (8) instead of the original one, thereby the privacy is preserved. The difference of optimal value between original LP problem and transformed LP problem is a constant term $\gamma \mathbf{c}^T \mathbf{r}$, which can be computed by data users before outsourcing the EMD problems. We also want the sum of elements of optimal solutions between original EMD problem and transformed EMD problem differs in a constant term after the transformation $\mathbf{x} = \mathbf{\Lambda}\mathbf{y} - \mathbf{r}$. So we apply one additional constraint when constructing $\mathbf{\Lambda}$ so that the sum of each column's elements in $\mathbf{\Lambda}^{-1}$ equals to 1. Then we can derive that $\sum_{i=1}^{k_t}\sum_{j=1}^{k_q} x_{ij} = \sum_{i=1}^{k_t}\sum_{j=1}^{k_q} y_{ij} - \sum_{i=1}^{k_t k_q} r_i$. When outsourcing the secure transformed problems to cloud server, the data users also send two offset constants $\gamma \mathbf{c}^T \mathbf{r}$ and $\sum_{i=1}^{k_t k_q} r_i$ to cloud server. Then, the cloud server can solve the transformed EMD problems and use these constants to compute the EMDs in an order preserving way. The offset constants themselves will reveal little information. After that, the cloud server sorts the results and returns the top-k most similar encrypted images to the data user.

4.4 Construction details

In this subsection, we will describe the whole service flow of the privacy-preserving CBIR scheme.

- $\mathcal{K} \leftarrow \mathsf{KeyGen}(1^\kappa)$. The data owner generates secret key set $\mathcal{K} = \{K_\mathcal{M}, K_\mathcal{S}, K_\Xi, \{K_j\}_{j=1}^L\}$.
- $\mathcal{I} \leftarrow \mathsf{BuildIndex}(\Xi, \{g_j\}_{j=1}^L, \{K_j\}_{j=1}^L, \phi)$. Firstly, the data owner extracts signature s_t from each image $m_t \in \mathcal{M}$, generating the signature set $\mathcal{S} = \{s_t\}_{t=1}^n$. Then, the data owner calculates the centroid ξ_t of the signature $s_t \in \mathcal{S}$, generating the centroid database $\Xi = \{\xi_t\}_{t=1}^n$. Then, the data owner constructs the secure LSH tables for the centroid database Ξ, using LSH functions g_j, keys K_j and one-way hash function ϕ as mentioned in Subsection 4.2. The index $\mathcal{I}$ consists of the generated secure LSH tables.
- $\{\mathcal{M}', \mathcal{S}'\} \leftarrow \mathsf{DataEnc}(\mathcal{M}, \mathcal{S}, K_\mathcal{M}, K_\mathcal{S})$. The data owner encrypts the image database $\mathcal{M}$ using key $K_\mathcal{M}$ to form the encrypted database $\mathcal{M}'$, and encrypts the signature database $\mathcal{S}$ using key $K_\mathcal{S}$ to form the encrypted signature database $\mathcal{S}'$. The data owner sends the encrypted image database $\mathcal{M}'$, encrypted signature database $\mathcal{S}'$ and the secure LSH tables to cloud server. The authorized data users can retrieve the encrypted outsourced images later from the cloud server. To achieve this functionality, the data owner needs to share the following information with data users: $K_\mathcal{M}, K_\mathcal{S}, \{K_j, g_j\}_{j=1}^L$ and ϕ. The data user receives these shared information to preprocess the image query.
- $\mathcal{T}_q \leftarrow \mathsf{TrapdoorGen}(s_q, \{K_j, g_j\}_{j=1}^L, \phi)$. For a given query image, the data user extracts its signature s_q using the feature extraction and clustering algorithm, and then computes the centroid ξ_q. Next, the data user applies LSH functions g_j, secret keys K_j, $j = 1, \cdots, L$ and one way hash function ϕ to construct trapdoor $\mathcal{T}_q = \{\phi(K_i, g_i(\xi_q))\}_{i=1}^L$. Then the data user sends the trapdoor $\mathcal{T}_q$ to the cloud server.
- $\mathcal{S}'_q \leftarrow \mathsf{SearchIndex}(\mathcal{I}, \mathcal{T}_q)$. After receiving the trapdoor $\mathcal{T}_q$, the cloud server uses $\mathcal{T}_{q,j} = \phi(K_j, g_j(\xi_q))$ to find in j-th hash table the bucket that has the same value with $\mathcal{T}_{q,j}$. The image identities in all of the found buckets are collected in an image identity set $\mathcal{ID}_q$. Then, a corresponding encrypted signature set $\mathcal{S}'_q$ is formed according to $\mathcal{ID}_q$. Next, the signature set $\mathcal{S}'_q$ is sent to the data user.
- $\{\Omega, \mathcal{C}\} \leftarrow \mathsf{SecureTrans}(\mathcal{S}'_q, s_q, K_\mathcal{S})$. After receiving the encrypted signature set $\mathcal{S}'_q$, the data user decrypts them to get the unencrypted signature set $\mathcal{S}_q$, using secret key $K_\mathcal{S}$. Then, the data user computes the distance vector $\mathbf{c}_{t,q}^T$ for each signature pair $(s_q, s_t), \forall s_t \in \mathcal{S}_q$. The data user also uses the weights information to form the EMD optimization problem $\Psi_{q,t} = (\mathbf{c}_{q,t}, \mathbf{U}_{q,t}, \tau_{q,t}, \mathbf{V}_{q,t}, \mathbf{E}_{q,t})$ for each pair (s_q, s_t). After that, the data user generates secure transformation key $K_{T_{q,t}} = (\mathbf{G}_{q,t}, \mathbf{\Lambda}_{q,t}, \mathbf{r}_{q,t}, \gamma_q)$ and transforms the original EMD problem in Formula (5) to Formula (8) and computes two offset constants $\gamma_q \mathbf{c}_{q,t}^T \mathbf{r}_{q,t}$ and $\sum_{i=1}^{k_q k_t} r_{q,t,i}$. Next, the data user outsources the set of all secure transformed EMD problems $\Omega = \{\Omega_{q,t} | s_t \in \mathcal{S}_q\}$ and corresponding offset constants $\mathcal{C} = \{\gamma_q \mathbf{c}_{q,t}^T \mathbf{r}_{q,t}, \sum_{i=1}^{k_q k_t} r_{q,t,i} | s_t \in \mathcal{S}_q\}$ to cloud server. Note that for each signature pair (s_q, s_t), we adopt different transformed key $K_{T_{q,t}}$, which can protect the sensitive information in a one-time pad manner.
- $\mathcal{M}'_q \leftarrow \mathsf{DistanceCom}(\Omega, \mathcal{C})$. After receiving the secure transformed EMD optimization problems and the corresponding offset constants, the cloud server solves the transformed EMD problems using the existing solving algorithm without learning the sensitive information. After that, it uses the offset constants to compute the EMDs in an order preserving way. Next, the cloud server sorts the EMDs and returns the set of ranked encrypted images $\mathcal{M}'_q$ to data user.
- $\mathcal{M}_q \leftarrow \mathsf{DataDec}(\mathcal{M}'_q, K_\mathcal{M})$. After receiving the ranked encrypted images, the data user decrypts the images using the secret key $K_\mathcal{M}$ and gets the unencrypted similar images to the search request.

5 Security Analysis

The proposed scheme utilizes pre-filter tables to group similar images together so as to improve search efficiency. Thus, the cloud server knows those images in the same bucket are similar to each other. In addition, the server knows those images retrieved by the same query are similar to each other as well. The above information leakage is a compromise for efficiency.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

Algorithm 2. Secure Transformation

Input: Original problem $\Psi = (\mathbf{c}, \mathbf{U}, \tau, \mathbf{V}, \mathbf{E})$ and secure key $K_T = (\mathbf{G}, \mathbf{\Lambda}, \mathbf{r}, \gamma)$
Output: Transformed problem $\Omega = (\mathbf{c}', \mathbf{U}', \tau', \mathbf{V}', \mathbf{E}')$
1: Pick a non-singular $k_t k_q \times k_t k_q$ matrix $\mathbf{\Lambda}$ and $k_t k_q \times 1$ vector $\mathbf{r}$ to perform the transformation as Formula(6);
2: Pick an $(k_t + k_q) \times (k_t + k_q)$ generalized permutation matrix $\mathbf{G}$ and multiple it to the constraints as Formula (7);
Return: The transformed problem Ω as shown in Formula (8).

$\mathcal{K} \leftarrow \mathsf{KeyGen}(1^\kappa)$
1. Generate secret keys $K_\mathcal{M}, K_\mathcal{S}, K_\Xi, \{K_j\}_{j=1}^L$;
2. Output the secret key $\mathcal{K} = \{K_\mathcal{M}, K_\mathcal{S}, K_\Xi, \{K_j\}_{j=1}^L\}$.

$\mathcal{I} \leftarrow \mathsf{BuildIndex}(\Xi, \{g_j, K_j\}_{j=1}^L, \phi)$
See **Algorithm 1**.

$\{\mathcal{M}', \mathcal{S}'\} \leftarrow \mathsf{DataEnc}(\mathcal{M}, \mathcal{S}, K_\mathcal{M}, K_\mathcal{S})$
1. Encrypt all of image $m_t \in \mathcal{M}$ with secure key $K_\mathcal{M}$, generating encrypted image set $\mathcal{M}'$;
2. Encrypt all of image $s_t \in \mathcal{S}$ with secure key $K_\mathcal{S}$, generating encrypted image set $\mathcal{S}'$;
3. Output $\{\mathcal{M}', \mathcal{S}'\}$

$\mathcal{T}_q \leftarrow \mathsf{TrapdoorGen}(s_q, \{g_j, K_j\}_{j=1}^L, \phi)$
1. For query signature s_q, compute its centroid ξ_q and initialize the trapdoor $\mathcal{T}_q = \emptyset$;
2. For each LSH function g_j and key $K_j, j = 1, ..., L$,
   1) Generate subtrapdoor $\mathcal{T}_{q,j} = \phi(K_j, g_j(\xi_q))$,
   2) $\mathcal{T}_q \leftarrow \mathcal{T}_q \cup \mathcal{T}_{q,j}$;
3. Output $\mathcal{T}_q$.

$\mathcal{S}'_q \leftarrow \mathsf{SearchIndex}(\mathcal{I}, \mathcal{T}_q)$
1. Initialize the candidate encrypted signature set $\mathcal{S}'_q = \emptyset$, image identity set of candidate image $\mathcal{ID}_q = \emptyset$;
2. For each $\mathcal{T}_{q,j}, j = 1, ..., L$:
   1) In the j-th hash table of index $\mathcal{I}$, retrieve the set of $\mathcal{ID}_{\mathcal{T}_{q,j}}$ from bucket $B_{\mathcal{T}_{q,j}}$ where the encrypted bucket value equals to $\mathcal{T}_{q,j}$;
   2) $\mathcal{ID}_q \leftarrow \mathcal{ID}_q \cup \mathcal{ID}_{\mathcal{T}_{q,j}}$;
3. Retrieve the corresponding encrypted signatures $\mathcal{S}'_q = \emptyset$ according to identity set $\mathcal{ID}_q$;
4. Output retrieval encrypted signature set $\mathcal{S}'_q$.

$\{\Omega, \mathcal{C}\} \leftarrow \mathsf{SecureTrans}(\mathcal{S}'_q, s_q, K_\mathcal{S})$
1. For each encrypted $s'_t \in \mathcal{S}'_q$;
   1) Calculate $s_t = Dec_{K_\mathcal{S}}(s'_t)$;
   2) Compute the distance vector $\mathbf{c}_{q,t}^T$ for signature s_t and query signature s_q;
   3) Formulate the LP problem $\Psi_{q,t} = (\mathbf{c}_{q,t}, \mathbf{U}_{q,t}, \tau_{q,t}, \mathbf{V}_{q,t}, \mathbf{E}_{q,t})$ by using the weights information in s_t;
   4) Generate secure transformation key $K_{T_{q,t}} = (\mathbf{G}_{q,t}, \mathbf{\Lambda}_{q,t}, \mathbf{r}_{q,t}, \gamma_q)$;
   5) Compute the transformed problem $\Omega_{q,t} = (\mathbf{c}'_{q,t}, \mathbf{U}'_{q,t}, \tau'_{q,t}, \mathbf{V}'_{q,t}, \mathbf{E}'_{q,t})$ using $K_{T_{q,t}}$ according to **Algorithm 2**.
   6) Calculate the offset values $\mathcal{C}_{q,t} = \{\gamma_q \mathbf{c}_{q,t}^T \mathbf{r}_{q,t}, \sum_{i=1}^{k_q k_t} r_{q,t,i}\}$;
2. Output the transformed problem set $\Omega = \{\Omega_{q,t} | s_t \in \mathcal{S}_q\}$ and the corresponding offset set $\mathcal{C} = \{\mathcal{C}_{q,t} | s_t \in \mathcal{S}_q\}$.

$\mathcal{M}'_q \leftarrow \mathsf{DistanceCom}(\Omega, \mathcal{C})$
1. Solve the transformed problems in Ω, and use the corresponding offset values to compute the EMDs in an order preserving way;
2. Sort the results according to EMD;
3. Output the set of top-k ranked encrypted image $\mathcal{M}'_q$.

$\mathcal{M}_q \leftarrow \mathsf{DataDec}(\mathcal{M}'_q, K_\mathcal{M})$
1. Decrypt all of the encrypted image $m'_t \in \mathcal{M}'_q$, generating the unencrypted image set $\mathcal{M}_q$;
2. Output $\mathcal{M}_q$.

Fig. 4: The details of Privacy-preserving CBIR Framework

5.1 Security of encryption

In this subsection, we argue that the encrypted database, secure searchable index and encrypted query will not reveal extra information to the cloud. First of all, it is easy to see that the image database is well protected if the encryption scheme is CPA-secure. The keywords in hash tables are encrypted by one-way hash function. For each $j = 1, ..., L$, data owner picks a random key K_j and encrypts each keyword $B_{j,i}, i = 1, ..., N_j$ in the j-th hash table as $\phi(k_j, B_{j,i})$. Since the keywords in the same hash table are unique

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

and every hash table is encrypted with different key, the same keywords are never encrypted with the same key twice. Thus, the encrypted keywords are indistinguishable from random. Similar to the secure searchable index, the encrypted query is also well protected.

5.2 Security of LP transformation

In order to protect the input and output privacy, we first perform the transformation that $\mathbf{x} = \mathbf{\Lambda y} - \mathbf{r}$, where $\mathbf{\Lambda}$ is an $k_t k_q \times k_t k_q$ non-singular matrix and $\mathbf{r}$ is an $k_t k_q \times 1$ vector. Next, we multiply the $(k_t + k_q) \times (k_t + k_q)$ matrix $\mathbf{G}$ to the constraints to protect $\mathbf{E}$. In this subsection, we prove that the two transformed EMD problems are indistinguishable from each other. In this case, the transformed EMD problem will reveal no extra information of the original one. We are going to show that over the random choice $K \leftarrow KeyGen(1^\kappa)$:

$$\forall \Psi_0, \Psi_1 : \mathsf{SD}(\mathsf{SecureTrans}(K, \Psi_0), \mathsf{SecureTrans}(K, \Psi_1)) \leq \mu(\kappa),$$

where $\mu(\cdot)$ is a negligible function, and $\mathsf{SD}(\cdot, \cdot)$ stands for the statistical distance. We assume that our system uses finite precision floating numbers, and each entry x_i of the original solution $\mathbf{x}$ should be in range $(-\delta, \delta)$, where $\delta = poly(\kappa)$ and κ is our security parameter.

Firstly, we argue that the transformed optimal solution $\mathbf{y}$ does not reveal $\mathbf{x}$. Recall that $\mathbf{x} = \mathbf{\Lambda y - r}$ and $\mathbf{y} = \mathbf{\Lambda^{-1}}(\mathbf{x} + \mathbf{r})$, where $\mathbf{\Lambda}$ is a randomly chosen invertible matrix. We don't have stringent requirement on the random choice of $\mathbf{\Lambda}$, as long as it is invertible and satisfies $\mathbf{1^T \Lambda} = \mathbf{1^T}$ to keep the problem structure consistent with the original one. For a random $\mathbf{r}$, we uniformly pick each entry r_i of $\mathbf{r}$ from a relatively big interval $[-2^\kappa, 2^\kappa]$ with fixed precision. we denote uniform distribution from $[-2^\kappa, 2^\kappa]$ with fixed precision as $\mathcal{U}(-2^\kappa, 2^\kappa)$. If we pick a random vector $\mathbf{r}^*$, where each entry in $\mathbf{r}^*$ is sampled from the uniform distribution $\mathcal{U}(-2^\kappa, 2^\kappa)$, we show that the distribution of $\mathbf{x}$ + $\mathbf{r}$ is statistically close to $\mathbf{r}^*$.

Theorem 1. The statistical distance $\mathsf{SD}(\mathbf{x} + \mathbf{r}, \mathbf{r}^*) \leq \mu(\kappa)$, where $\mu(\kappa)$ is a negligible function.

Proof. We first show that for each individual entry, $\mathsf{SD}(x_i + r_i, r_i^*)$, $i \in \{1, \ldots, k_t k_q\}$, is negligible. We have two hypothesises $\mathcal{H}_0$ and $\mathcal{H}_1$, whose ranges are $[-2^\kappa - \delta, 2^\kappa + \delta]$ and $[-2^\kappa, 2^\kappa]$, respectively. The distinguisher will output $\mathcal{H}_0$ if the input is from $[-2^\kappa - \delta, -2^\kappa)$ or $(2^\kappa, 2^\kappa + \delta]$, otherwise, it will output a random guess $\epsilon \leftarrow \{0, 1\}$. Thus, the distinguisher's success probability is

$$\begin{aligned} p &= \frac{1}{2} + Pr[x_i + r_i \in [-2^\kappa - \delta, -2^\kappa)] \\ &\quad + Pr[x_i + r_i \in (2^\kappa, 2^\kappa + \delta]] \\ &= \frac{1}{2} + \frac{2\delta}{2 \times 2^\kappa + 2\delta} \\ &= \frac{1}{2} + \frac{\delta}{2^\kappa + \delta} \\ &\leq \frac{1}{2} + \frac{\delta}{2^\kappa} \\ &= \frac{1}{2} + \mu'(\kappa) \end{aligned}$$

where $\mu'(\kappa) = \frac{\delta}{2^\kappa}$ is a negligible function. Then, by applying union bound, we have $\mathsf{SD}(\mathbf{x+r}, \mathbf{r}^*) \leq \mu(\kappa)$ where $\mu(\kappa) = k_t k_q * \mu'(\kappa)$ as claimed. Therefore, from the cloud's view, $\mathbf{x+r}$ and $\mathbf{r}^*$ are statistically indistinguishable.

End of Proof.

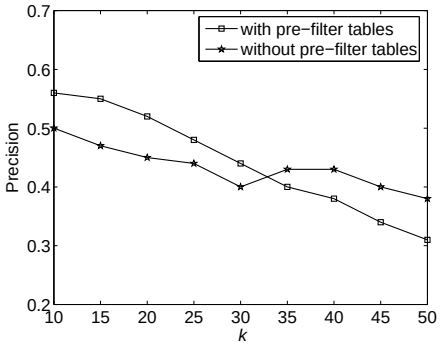


Fig. 5: Search precision

As the $\mathbf{\Lambda}$ is a random chosen matirx, $\mathbf{y} = \mathbf{\Lambda}^{-1}(\mathbf{x} + \mathbf{r})$ and $\mathbf{y}^* = \mathbf{\Lambda}^{-1}\mathbf{r}^*$ are statistically indistinguishable from the cloud's view according to **Theorem 1**. That is to say, $\mathbf{y}$ will not reveal $\mathbf{x}$.

Similarly, we show that $\mathbf{E}' = \mathbf{G}(\mathbf{E} + \mathbf{Vr})$ statistically hides $\mathbf{E}$. Since $\mathbf{E} = \mathbf{Vx}$, we have $\mathbf{E'}$=$\mathbf{GV(x+r)}$. Again, we assume that each entry of $\mathbf{r}$ is sampled from the uniform distribution $\mathcal{U}(-2^\kappa, 2^\kappa)$. According to **Theorem 1**, $\mathbf{E}' = \mathbf{GV}(\mathbf{x} + \mathbf{r})$ and $\mathbf{E}^* = \mathbf{GVr}^*$ are statistically indistinguishable from the cloud's view, i.e. $\mathsf{SD}(\mathbf{E}', \mathbf{E}^*) \leq \mu(\kappa)$. Therefore, $\mathbf{E}'$ will not reveal $\mathbf{E}$ since matrix $\mathbf{G}$ is also randomly generated.

Altogether, based on that $\mathsf{SD}(\mathbf{E}', \mathbf{E}^*) \leq \mu(\kappa)$, $\mathsf{SD}(\mathbf{x} + \mathbf{r}, \mathbf{r}^*) \leq \mu(\kappa)$, and all parameters $\mathbf{G}, \mathbf{r}, \mathbf{\Lambda}$ are randomly selected for each EMD problem, we have

$$\mathsf{SD}(\mathsf{SecureTrans}(K, \Psi_0), \mathsf{SecureTrans}(K, \Psi_1)) \leq \mu(\kappa).$$

6 Implementation and Performance

In this section, we present an experimental evaluation of the proposed CBIR scheme on a real-world image database: Corel Image Set [35]. The performance of our scheme is evaluated in terms of the efficiency and search precision. The experimental scheme is implemented by C++ language on a Windows Server with Intel Core 2 Processor 2.0 GHz.

The performance of the scheme is dependent on several parameters, including the parameter in the p-stable hash function θ, the number of jointed LSH functions in the construction of pre-filter table λ, and number of pre-filter tables L. Experientially, we set $\theta = 4, \lambda = 2$, and $L = 20$ to achieve a satisfying result. Please note that the parameters used here are not claimed to be the optimum ones. Generally, a small L, a largerλ, or a smaller θ will improve the search efficiency but trade off the search precision.

6.1 Precision

In our experiments, the "precision" for a query is defined as that in [36]: $P_k = k'/k$, where k' is the number of real similar images in the retrieved k images. According to analysis in subsection 4.3, the secure transformation of the EMD problem will not influence retrieval precision. However, the pre-filter tables which are utilized to improve the search efficiency will affect the retrieval precision. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

retrieval precision of our scheme with and without the pre-filter tables are tested and compared. Fig. 5 shows that when $k \leq 30$, the scheme with pre-filter tables performs better than the direct EMD computation scheme. It is a surprise windfall.

There are many reasons contribute to the outcome, we consider two factors here. The first is that it is not ensured that the local features combined with EMD metric will achieve optimum retrieval precision. The second one is that local sensitive hash helps to filter out some dissimilar images which will be wrongly judged to be the similar ones to the query image by EMD metric. As shown in Fig. 5, the proposed scheme with the LSH tables does not degrade a lot in terms of the search precision.

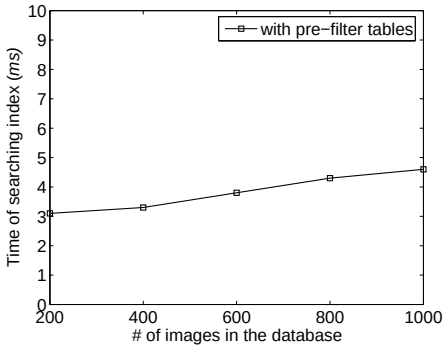


Fig. 6: Time of searching index

6.2 Efficiency

6.2.1 Index construction

In this paper, we extract SIFT features [30] to represent the images. We consider the feature extraction and clustering as two pre-processes to the index construction. Then, the index construction process mainly includes signature generation, centroid calculation, and hash calculation. The time consumption in feature extraction, clustering and index construction is listed in Table 2. Compared with the index construction, feature extraction and clustering are the two steps which consume more time. The time consumption in feature extraction, clustering and index construction is not a negligible overhead for the data owner. However, these are one-off operations before data outsourcing and are affordable for the data owner. In addition to the time consumption, we list the storage consumption of the index in Table 3, which is affordable for cloud server.

TABLE 2: Time consumption of feature extraction, clustering operation, and index construction

Size of image database	200	400	600	800	1000
Time of feature extraction (s)	377	781	1026	1411	1738
Time of clustering operation (s)	119	255	370	567	750
Time of index construction (s)	48	96	131	171	217

TABLE 3: Storage consumption of index

Size of image database	200	400	600	800	1000
Index size(KB)	356	714	1072	1431	1789

6.2.2 Trapdoor Generation

For a query request, the trapdoor generation consists of feature extraction, signature generation, centroid calculation and hash calculation. Similar to the index construction, the feature extraction consumes the most of the time. The average time consumption of trapdoor generation is 2.094 seconds in our 40 times of experiment.

6.2.3 Time of search operation

After receiving the query trapdoor, the cloud server searches on the index (i.e. the LSH tables) to obtain a identity set of candidate images. Then, the corresponding signature set is formed and sent to the query user. The query user decrypts the signatures, constructs the secure transformed EMD problems, and then sends the transformed problems back to the cloud server. The cloud server solves transformed problems to obtain the top-k ranked images. Finally, the ranked images are sent to the query user for decryption.

The time consumption of search on index, transformation of EMD problems, and computation of transformed EMD problems are separately tested and illustrated in Fig. 6, 7 and 8, respectively. In our scheme, the LSH tables are constructed to improve the search efficiency. Thus, the time consumption of search with and without the hash tables is tested and compared.

The time consumption of searching index refers to the time that is costed on the search of LSH tables. Thus, the scheme without LSH tables takes no time on it. As shown in Fig. 6, the time cost of searching index is tiny and is constant with the increase of images in database. As shown in Fig. 7 and 8, the time complexity of scheme without LSH tables is linear to the size of image database. However, the time consumption of the scheme with the LSH tables increases slightly to the expansion of database. Altogether, the proposed scheme achieves nearly constant search time while increasing the size of image database.

Among the previous works, the schemes in [20], [21], [22] mainly focus on the security of the features, not the efficiency. The search time complexities of these works are $O(n)$. In [23], a hierarchical index is built to improve the search efficiency using Bag-Of-Visual-Words and k-means algorithm. We compare our experimental results to the data published in [23].

In a query of our scheme, the time consumption on the cloud server side includes the time cost on index search and computation of EMD problems, while the time consumption on the user side includes the time cost on trapdoor generation and transformation of EMD problems. As shown in Table 4, the cloud in our scheme consumes less time than that in [23] during a query process, while the query user in our scheme consumes more time than that in [23]. The previous works, including [23], mainly devote to secure outsourcing of global-feature based CBIR, but our work solves the problems in local-feature based CBIR. Thus, the comparison of performance may not be quite fair. The local feature-based schemes generally consume more time than the global ones. Please note that, most of the time consumed on the query user side in our scheme is spent on the extraction of local features.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCC.2015.2491933, IEEE Transactions on Cloud Computing

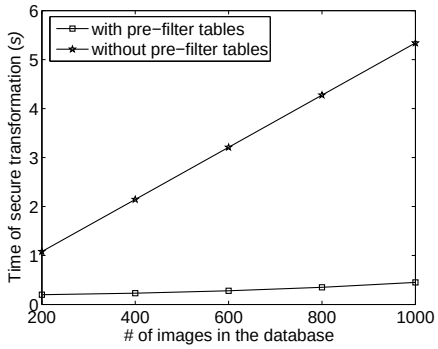


Fig. 7: Time of secure transformation

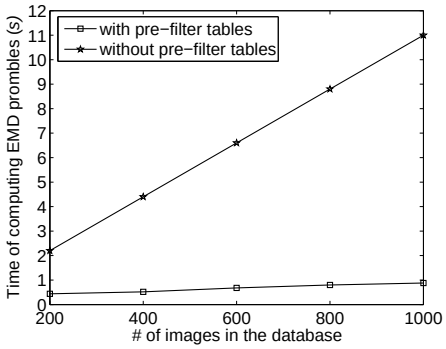


Fig. 8: Time of calculation of EMD problems

7 Conclusion and Future Work

In this paper, we propose a privacy-preserving content-based image retrieval scheme, which allows the data owner to outsource image database and the CBIR service to the cloud without revealing the actual content of the database. Local features are utilized to represent the images, and earth mover's distance (EMD) is employed to evaluate the similarity of images. We transform the EMD problem so that the cloud server can solve the problem without learning the sensitive information. In order to improve the search efficiency, we design a two-stage structure with LSH. In the first stage, dissimilar images are filtered out by pre-filter tables to shrink the search scope. In the second stage, the remaining images are compared under EMD metric one by one for refined search results. The security analysis and experiments show the security and efficiency of the proposed scheme. In the future, we will study how to outsource the feature extraction to the cloud server so as to further relieve the burden of data owner and data user.

Acknowledgments

This work is supported by the NSFC (61173141, 61232016, U1405254, 61173142, 61173136, 61373133), 201301030, 2013DFG12860, BC2013012, Fund of Jiangsu Engineering Center of Network Monitoring (KJR1308, KJR1402), Fund of MOE Internet Innovation Platform (KJRP1403), and PAPD fund.

TABLE 4: Time consumption per one query

	Time consumption of cloud (s)	Time consumption of user (s)	Number of images
Ferreira *et al.*'s scheme [23]	2.35	0.95	1491
ours	0.88	2.54	1000

References

[1] C. Pavlopoulou, A. C. Kak, and C. E. Brodley, "Content-based image retrieval for medical imagery," in *Medical Imaging 2003*. International Society for Optics and Photonics, 2003, pp. 85–96.

[2] A. K. Jain, J.-E. Lee, R. Jin, and N. Gregg, "Content-based image retrieval: An application to tattoo images," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2009, pp. 2745–2748.

[3] J. M. Lewin, R. E. Hendrick, C. J. DOrsi, P. K. Isaacs, L. J. Moss, A. Karellas, G. A. Sisney, C. C. Kuni, and G. R. Cutter, "Comparison of full-field digital mammography with screen-film mammography for cancer detection: Results of 4,945 paired examinations 1," *Radiology*, vol. 218, no. 3, pp. 873–880, 2001.

[4] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on*. IEEE, 2000, pp. 44–55.

[5] E.-J. Goh *et al.*, "Secure indexes." *IACR Cryptology ePrint Archive*, vol. 2003, p. 216, 2003.

[6] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definitions and efficient constructions," in *Proceedings of the 13th ACM conference on Computer and communications security*. ACM, 2006, pp. 79–88.

[7] M. Kuzu, M. S. Islam, and M. Kantarcioglu, "Efficient similarity search over encrypted data," in *Data Engineering (ICDE), 2012 IEEE 28th International Conference on*. IEEE, 2012, pp. 1156–1167.

[8] C. Wang, K. Ren, S. Yu, and K. M. R. Urs, "Achieving usable and privacy-assured similarity search over outsourced cloud data," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 451–459.

[9] Z. Xia, Y. Zhu, X. Sun, and L. Chen, "Secure semantic expansion based search over encrypted cloud data supporting similarity ranking," *Journal of Cloud Computing*, vol. 3, no. 1, pp. 1–11, 2014.

[10] W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," in *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*. ACM, 2013, pp. 71–82.

[11] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 25, no. 1, pp. 222–233, 2014.

[12] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data," *Parallel and Distributed Systems, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2015.

[13] S. Kamara and C. Papamanthou, "Parallel and dynamic searchable symmetric encryption," in *Financial Cryptography and Data Security*. Springer, 2013, pp. 258–274.

[14] D. Cash, J. Jaeger, S. Jarecki, C. Jutla, H. Krawczyk, M.-C. Rosu, and M. Steiner, "Dynamic searchable encryption in very large databases: Data structures and implementation," in *Proc. of NDSS*, vol. 14, 2014.

[15] J. Shashank, P. Kowshik, K. Srinathan, and C. Jawahar, "Private content based image retrieval," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[16] C.-Y. Hsu, C.-S. Lu, and S.-C. Pei, "Secure and robust SIFT," in *Proceedings of the 17th ACM international conference on Multimedia*. ACM, 2009, pp. 637–640.

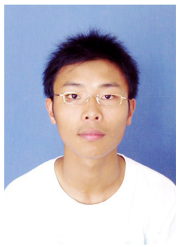
[17] ——, "Image feature extraction in encrypted domain with privacy-preserving SIFT," *Image Processing, IEEE Transactions on*, vol. 21, no. 11, pp. 4593–4607, 2012.

[18] P. Zheng and J. Huang, "An efficient image homomorphic encryption scheme with small ciphertext expansion," in *Proceedings of the 21st ACM international conference on Multimedia*. ACM, 2013, pp. 803–812.

[19] Z. Qin, J. Yan, K. Ren, C. W. Chen, and C. Wang, "Towards efficient privacy-preserving image feature extraction in cloud computing,"

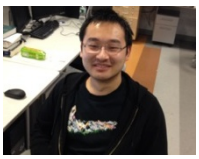
in *Proceedings of the ACM International Conference on Multimedia*. ACM, 2014, pp. 497–506.
[20] W. Lu, A. Swaminathan, A. L. Varna, and M. Wu, "Enabling search over encrypted multimedia databases," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2009, pp. 725 418–725 418.
[21] W. Lu, A. L. Varna, A. Swaminathan, and M. Wu, "Secure image retrieval through feature protection," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 1533–1536.
[22] B. Cheng, L. Zhuo, Y. Bai, Y. Peng, and J. Zhang, "Secure index construction for privacy-preserving large-scale image retrieval," in *Big Data and Cloud Computing (BdCloud), 2014 IEEE Fourth International Conference on*. IEEE, 2014, pp. 116–120.
[23] B. Ferreira, J. Rodrigues, J. Leitão, and H. Domingos, "Privacy-preserving content-based image retrieval in the cloud," *arXiv preprint arXiv:1411.4862*, 2014.
[24] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.
[25] H. Ling and K. Okada, "An efficient earth mover's distance algorithm for robust histogram comparison," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 5, pp. 840–853, 2007.
[26] B. Pinkas and T. Reinman, "Oblivious RAM revisited," in *Advances in Cryptology–CRYPTO 2010*. Springer, 2010, pp. 502–519.
[27] J. R. Smith and S.-F. Chang, "Tools and techniques for color image retrieval." in *Storage and Retrieval for Image and Video Databases (SPIE)*, vol. 2670, 1996, pp. 2–7.
[28] B. S. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 8, pp. 837–842, 1996.
[29] M. Yang, K. Kpalma, and J. Ronsin, "A survey of shape feature extraction techniques," *Pattern recognition*, pp. 43–90, 2008.
[30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.
[31] T. Deselaers, D. Keysers, and H. Ney, "Discriminative training for object recognition using image patches," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 157–162.
[32] E. Levina and P. Bickel, "The earth mover's distance is the Mallows distance: Some insights from statistics," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2. IEEE, 2001, pp. 251–256.
[33] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proceedings of the twentieth annual symposium on Computational geometry*. ACM, 2004, pp. 253–262.
[34] A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," in *Foundations of Computer Science, 2006. FOCS'06. 47th Annual IEEE Symposium on*. IEEE, 2006, pp. 459–468.
[35] J. Z. Wang, J. Li, and G. Wiederhold, "Simplicity: Semantics-sensitive integrated matching for picture libraries," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 9, pp. 947–963, 2001.
[36] H. Müller, W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun, "Performance evaluation in content-based image retrieval: Overview and proposals," *Pattern Recognition Letters*, vol. 22, no. 5, pp. 593–601, 2001.

Zhihua Xia received his BS in Hunan City University, China, in 2006, PhD in computer science and technology from Hunan University, China, in 2011. He works as a lecturer in School of Computer & Software, Nanjing University of Information Science & Technology. His research interests include digital forensic and privacy protection in cloud computing.



Yi Zhu is currently pursuing his MS in computer science and technology at the School of Computer and Software, in Nanjing University of Information Science & Technology, China. His research interests include privacy protection in cloud computing.

Xingming Sun received his BS in mathematics from Hunan Normal University, China, in 1984, MS in computing science from Dalian University of Science and Technology, China, in 1988, and PhD in computing science from Fudan University, China, in 2001. He is currently a professor in School of Computer & Software, Nanjing University of Information Science & Technology, China. His research interests include network and information security, digital watermarking, and data security in cloud.



Zhan Qin is a PhD candidate at Department of Computer Science and Engineering of State University of New York at Buffalo. He received his B.E in information engineering from Beijing Institute of Technology in 2010, his M.Sc. in electronic engineering from Columbia University in 2012. He has worked at Microsoft Research Asia in the summer of 2011. His research interests focus on security and privacy of cloud computing.

Kui Ren received the Ph.D. degree from the Worcester Polytechnic Institute, Worcester, MA, USA. He is currently an Associate Professor of Computer Science and Engineering and the Director of the UbiSeC Laboratory with University at Buffalo, Buffalo, NY, USA. His research has been supported by NSF, DoE, AFRL, MSR, and Amazon. His current research interest spans cloud and outsourcing security, wireless and wearable system security, and human-centered computing. He is a member of the Association for Computing Machinery, and a Distinguished Lecturer of the IEEE Vehicular Technology Society. He was a recipient of the NSF CAREER Award in 2011 and the Sigma Xi/IIT Research Excellence Award in 2012. He received several best paper awards, including the IEEE ICNP 2011. He serves as an Associate Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING, the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, the IEEE Wireless Communications, the IEEE INTERNET OF THINGS JOURNAL, the IEEE TRANSACTIONS ON SMART GRID, Pervasive and Mobile Computing (Elsevier), and The Computer Journal (Oxford University Press). He was a Board Member of the Internet Privacy Task Force in Illinois.