Received December 15, 2013, accepted January 15, 2014, date of publication February 20, 2014, date of current version March 4, 2014.

Digital Object Identifier 10.1109/ACCESS.2014.2307057

Confidentiality-Preserving Image Search: A Comparative Study Between Homomorphic Encryption and Distance-Preserving Randomization

WENJUN LU[1], AVINASH L. VARNA[2], (Member, IEEE), AND MIN WU[3], (Fellow, IEEE)
[1]Google, Mountain View, CA 95014, USA
[2]Intel, Pheonix, AZ 85226, USA
[3]Department of Electrical and Computer Engineering and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, USA

Corresponding author: W. Lu (luwj02@gmail.com)

ABSTRACT Recent years have seen increasing popularity of storing and managing personal multimedia data using online services. Preserving confidentiality of online personal data while offering efficient functionalities thus becomes an important and pressing research issue. In this paper, we study the problem of content-based search of image data archived online while preserving content confidentiality. The problem has different settings from those typically considered in the secure computation literature, as it deals with data in rank-ordered search, and has a different security-efficiency requirement. Secure computation techniques, such as homomorphic encryption, can potentially be used in this application, at a cost of high computational and communication complexity. Alternatively, efficient techniques based on randomizing visual feature and search indexes have been proposed recently to enable similarity comparison between encrypted images. This paper focuses on comparing these two major paradigms of techniques, namely, homomorphic encryption-based techniques and feature/index randomization-based techniques, for confidentiality-preserving image search. We develop novel and systematic metrics to quantitatively evaluate security strength in this unique type of data and applications. We compare these two paradigms of techniques in terms of their search performance, security strength, and computational efficiency. The insights obtained through this paper and comparison will help design practical algorithms appropriate for privacy-aware cloud multimedia systems.

INDEX TERMS Content based image retrieval, secure search, secure cloud computing, homomorphic encryption, visual words, min-hash, random projection, order preserving encryption.

I. INTRODUCTION

With the arrival of the cloud computing paradigm and the proliferation of online services, the Internet stores not only information for sharing, but also a large amount of personal data demanding restricted access and privacy protection. Secure management of personal data stored online is an increasingly important issue, which demands a balance between data confidentiality and availability. Technologies that can enable secure online data management are going to be critically important for cloud computing to reach its full potential.

Traditional privacy protection for online personal data focuses on access control and secure data transmission to ensure that the data can be securely transmitted to the server and unauthorized people cannot access the data. Once the data arrives at the server, the server decrypts the data and operates on plaintext in order to provide services to users, such as search and data summarization. This makes the user's private information vulnerable to untrustworthy service providers and malicious intruders. For example, personal photo albums can potentially be viewed by a system administrator if stored online in plaintext. Encryption of the data stored on the server using traditional cryptographic ciphers directly makes it difficult for the server to process the data, and for the user to retrieve information from the encrypted database. Therefore, it is both desirable and necessary to develop technologies for information retrieval over encrypted databases that can protect users' privacy without sacrificing the usability and accessibility of the information.

Due to the widespread use of digital cameras and smartphones, digital images have become a significant part of today's personal data collections. Storing and managing large volume of image data online is a desirable option for convenient data access anywhere anytime. Motivated by these important technological trends, we study in this paper the problem of content-based search of online image database, such as a personal online album, while minimizing information leak and preserving data confidentiality against unauthorized access including the service provider, with focus on comparing two major types of techniques, namely, homomorphic encryption and distance preserving randomization. Below, we first review some related work.

A. RELATED WORK

Prior work in the area of information retrieval in the encrypted domain focused on text documents. Song et al. [1], Brinkman et al. [2], and Boneh et al. [3] explored Boolean search to identify whether a query term is present in an encrypted text document. Swaminathan et al. [4] proposed a framework for rank-ordered search over encrypted text documents, so that documents can be returned in the order of their relevance to the query term. In that work, several protocols are studied to address different operational constraints such as different communication cost allowed to perform the secure search. Secure text retrieval techniques can also be applied to keyword based search of image data. However, keyword search relies on having accurate text description of the content already available, and its search scope is confined to the existing keyword set. In contrast, content-based search over an encrypted image database provides more flexibility, whereby sample images are presented as queries and documents with similar visual content in the database are identified.

An emerging area of work related to confidentiality preserving image retrieval is secure signal processing, which aims at performing signal processing tasks while keeping the signals being processed secret. Erkin et al. [5] provided a review of related cryptographic primitives and some applications of secure signal processing in data analysis and content protection. However, applying cryptographic primitives to the task of content-based image retrieval is not straightforward. Effective image retrieval typically relies on evaluating the similarity of two documents using the distance between their visual features, such as color histograms, shape descriptors, or salient points [6]. By design, traditional cryptographic primitives do not preserve the distance between feature vectors after encryption. Given the much larger data volume for image data than that of text and other generic data, efficiency and scalability are critical for image retrieval but can be difficult to achieve using cryptographic primitives alone. Another work by Shashank et al. [7] addresses the problem of protecting the privacy of the query image when searching over a public database, where the images in the database are not encrypted. By appropriately formulating the query message and response message during multiple rounds of communication between the user and the server, the server is made oblivious to the actual search path and thus unaware of the query content.

Recent work in the area of secure computation for privacy protection addressed related but different problems under various application settings [8]–[13]. Yiu et al. [8] considered privacy preserving range query over geospatial coordinates using a k-dimensional tree. Extending such technique to image retrieval is difficult because features used for content-based image retrieval are high dimensional vectors and kd-tree is known to be inefficient in high dimensional spaces. Wong et al. [9] proposed secure k-NN computation that can determine which of two encrypted database entries has a smaller distance to the query, while keeping the actual distance value secret. This work can potentially be used for rank-ordered image retrieval, but the efficiency is limited because each comparison only answers a binary question of which one among the two being larger or smaller. Erkin et al. [10], Sadeghi et al. [11] and Osadchy et al. [12] studied privacy preserving face recognition, where one party verifies the existence of a given face image in a database hosted on another party's servers. The two parties want to keep their own data secret from each other. Additive homomorphic encryption schemes are used to allow similarity computation in the encrypted domain. Recent work by Yan et al. [14] proposed more efficient protocols based on garbled circuit for general biometric matching problem. The common properties of such biometric matching work are that two parties are involved in the computation and communication cost is inevitable because there have been no efficient homomorphic encryption schemes yet that allow both addition and multiplication.

Given that cryptography based approaches are typically too heavy-weight in terms of computation and communication, Lu et al. [15], [16] studied the problem of confidentiality-preserving image search from a practical perspective and proposed techniques that are efficient to be used for practical applications. By jointly utilizing image processing, information retrieval, and cryptographic primitives, the techniques proposed in [15] and [16] randomize the visual features and search indexes with approximate distance preserving property so that the content similarity between encrypted image content can be directly computed by the server without interacting with the user. Compared to homomorphic encryption used in secure computation, these techniques do not have the highest security as required for cryptographic ciphers, but still provide a good amount of randomness and protection on the visual features and the search process. The advantage of being highly efficient makes them good candidates for practical web applications that have less stringent requirement on security but demand high efficiency and least user involvement.

B. CONTRIBUTIONS AND PAPER ORGANIZATION

The problem of confidentiality-preserving content-based search of image has both practical applications and challenging research issues. The application settings of this problem are quite different from existing secure computation work [10]–[13] and therefore brings interesting research values: (1) the search is rank-ordered where the server needs to return the documents ranked according to their similarity to the query, while existing secure computing work typically focus on a binary matching problem, such as biometric matching and keyword search, and the server may be made oblivious to the binary matching result; (2) the user owns all the data and the server merely offers storage and search functionality, while in secure multiparty computation scenario, both parties have their own private data that need to be kept secret from each other when computing a joint function; (3) we consider retrieval over large volumes of image data using high dimensional visual features, which requires efficient processing and good scalability in order to be practical; (4) The application considered in this paper is more consumer-oriented, which has less stringent requirement on security but demands highly efficient solutions and least user involvement; while the applications considered in secure computation literature typically involve very sensitive data such as biometrics, thus demanding very high security at the cost of heavier computation and communication cost.

In this paper, we review and compare two major types of techniques for the problem of confidentiality-preserving image search. The first type is based on homomorphic encryption and cryptography protocols. We discuss how existing additive homomorphic encryption and the recent advancement in fully homomorphic encryption can be potentially used for image search. The second type is the randomization techniques for visual features and search indexes [15], [16]. We compare these two types of techniques in terms of their search accuracy on an actual encrypted image database, as well as their security strength and computational efficiency. The comparative study here also applies to the problem of searching general multimedia data with multiple media modalities. We hope such a quantitative comparison between these two major techniques on secure search may reveal some insights on the practical design of secure computation techniques for real-world application in this era of cloud computing.

The paper is organized as follows: Section II formulates the problem of confidentiality preserving image search and discuss its unique application settings. Section III reviews the two major types of techniques for secure search, namely, the homomorphic encryption based techniques and the feature/index randomization based techniques. Section IV provides detailed quantitative comparison in terms of search accuracy, security strength, and computational efficiency of the two types of techniques. Conclusions are drawn in Section V.

II. PROBLEM FORMULATION

We now use image as an example to discuss the problem formulation. In order to protect data privacy, images need to be encrypted before being transferred to a remote server. Image encryption can be done using state-of-the-art ciphers such as AES or RSA by treating images as ordinary data, or using image specific encryption such as selective and format-compliant encryption [17]–[19] to enable post-processing such as transcoding. As these techniques are built upon established cryptographic encryption tools, it is computationally difficult for an adversary to decrypt the encrypted image files.

In modern image retrieval techniques, content similarity is typically evaluated using search indexes or visual features, such as color histograms and salient points, instead of comparing images pixel by pixel. Therefore, encryption of images alone is not sufficient for privacy preserving retrieval because search indexes and visual features in plaintext can reveal information about image content. For example, a color histogram with large values for the blue components would indicate the presence of sky or ocean, and salient features such as SIFT [20] can reveal information about distinctive objects in the image. In order to be able to search through the encrypted database without leaking information from the plaintext search indexes or image features, image features or indexes need to be properly protected on the user side before being transferred to the server. The computation of image similarity then needs to be carried out on those protected features and indexes. A system model for the confidentiality-preserving search scenario is shown in Fig. 1, where the left part depicts the database construction stage and the right part depicts the search stage. There are two entities in this model:

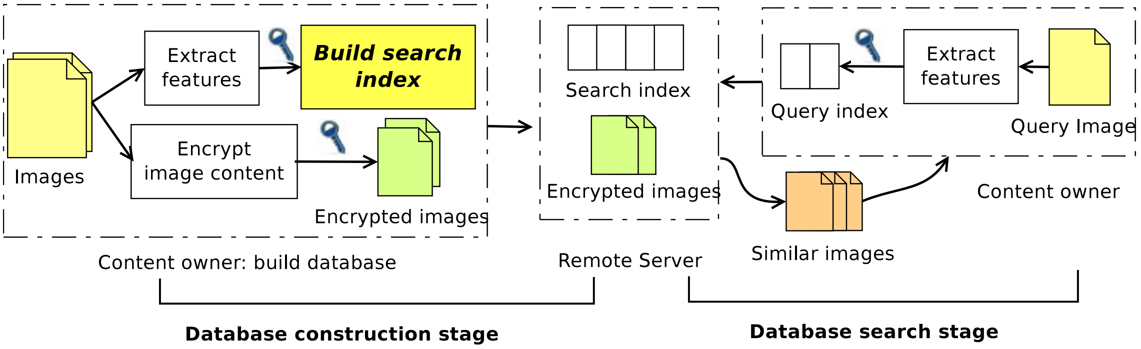


FIGURE 1. System model for confidentiality-preserving image retrieval.

a user who owns the private image collections, and a server who stores the encrypted data and performs retrieval based on a given encrypted query. During database construction, the user encrypts the images using standard ciphers and protects visual features or search indexes. After encryption, the user sends the encrypted images and protected features/indexes to the server for storage. During search, the user sends the similarly protected visual feature or search index of the query image to the server, who performs search using only the protected features or indexes. Finally, a list of encrypted images are ranked by their similarity to the query and returned to the user. The block ''Build search index'' corresponds to techniques of feature encryption or randomization, which we will discuss in Section III.

This secure search problem and formulation have several unique properties that are different from existing secure computation literature: (1) In secure two-party computation problem, each party will hold its own data in plaintext, and they want to compute a joint function without revealing their own data. Each party will perform computation on its plaintext data and the encrypted data from the other party. In our application, although there are two parties, the user and the server, only one party, i.e., the user owns the entire data. The server merely stores the encrypted data and helps perform some computation task. The server needs to do the computation entirely in the encrypted domain. (2) We are considering rank-ordered search where the server needs to return the documents ranked according to their similar to the query, while existing secure computing work typically focus on a binary matching problem, such as biometric matching and keyword search, and the server may be made oblivious to the binary matching result. The requirement of returning images to the user inevitably reveals some information about the ciphertext, i.e., the returned images will be likely to be all similar to the query. Such a property may affect the security level that can be achieved and the design of secure search techniques. (3) The data that we consider in this application is personal image collection and the visual features from those image data. There is an inherent semantic gap between low level visual features and high level concepts, and we can expect that the security requirement on such data will not be as stringent as that required for highly sensitive data such as biometrics. Therefore, a proper understanding on the security objective and security-efficiency tradeoff will be critical in the design of privacy-aware search techniques.

III. TECHNIQUES FOR CONFIDENTIALITY-PRESERVING SEARCH

In this section, we review existing techniques that can serve as candidate solutions for confidentiality-preserving image search. Two major types of techniques will be discussed here. One is based on homomorphic encryption and cryptographic protocols that are commonly used in secure computation literature. The other is based on distance preserving randomization for visual features and search indexes [15], [16]. The comparison of these techniques in terms of search performance, security strength, and computational efficiency will be discussed in Section IV.

A. HOMOMORPHIC ENCRYPTION AND CRYPTOGRAPHIC PROTOCOLS

Semantically secure homomorphic public-key encryption schemes are central cryptographic tool for many secure multiparty computation problem. Below, we briefly review the basics of simple additive homomorphic encryption and recent advance of fully homomorphic cryposystems, then discuss how such techniques can be applied for the problem of secure image search.

1) ADDITIVE HOMOMORPHIC ENCRYPTION

In an additive homomorphic cryptosystem, given encryptions $[a]$ and $[b]$, the encryption of their summation $[a+b]$ can be computed by $[a+b] = [a][b]$, where all the computations are performed in the encrypted domain, without decryption. Following the above property, the multiplication of an encrypted value $[a]$ with a known constant b in the clear can be computed as $[ab] = [a]^b$.

One of the representative additive homomorphic cryptosystem is proposed by Paillier [21], which is based on the decisional composite residuosity problem. Let $n = pq$ of size k, where p and q are large prime numbers and k is the required bit length of the security key typically from the range $1000 - 2048$. Randomly select a base g ($g = n+1$ will do). Then to encrypt a plaintext message $m \in \mathbb{Z}_n$, the user will select a random value $r \in \mathbb{Z}_n$ and computes the ciphertext $c = g^m r^n \bmod n^2$. The parameters (n, g) are the public keys and the pair (p, q) serves as the private key. Given a ciphertext c, its plaintext message m can be obtained by $m = \frac{L(c^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} \bmod n^2$, where $L(u) = \frac{u-1}{n}$. It is easy to see that the Paillier is additively homomorphic and for an encryption $[m]$, re-randomizing it can be done without knowing the private key by $[m]r^n \bmod n^2$. More details of the Paillier cryptosystem can be found in [21].

2) FULLY HOMOMORPHIC ENCRYPTION

Earlier homomorphic cryptosystems [21]–[24] support either addition or multiplication between encrypted values, but not both operations at the same time. This brings challenges to many secure computation problems because many operations such as computing the Euclidean distance between two encrypted vectors require both addition and multiplication. With only additive or multiplicative homomorphic cryptosystem, a cryptography protocol that involves communication between the two computing parties is typically required.

More recently, in a breakthrough work, Gentry [25] constructed a fully homomorphic encryption (FHE) scheme capable of evaluating an arbitrary number of additions and multiplications (thus compute any function) on encrypted data. The mathematics and construction details in [25] are quite involved, but the basic idea can be summarized as follows. An initial ''somewhat homomorphic'' scheme based on ideal lattice is constructed to allow evaluation of essentially unlimited addition and a certain amount of multiplication. This initial scheme is somewhat homomorphic because the errors in the ciphertext grows with more operations, so only a limited amount of multiplication can be supported. To achieve fully homomorphic encryption, the ciphertext has to be re-encrypted through a technique called ''bootstrapping'', so that errors in the ciphertext can be cleaned and an unlimited number of operations can be allowed.

Following this first construction of fully homomorphic encryption, there have been subsequent developments that try to improve the efficiency of FHE [26]–[30]. Although the most recent solutions of FHE have improved upon the initial construction of Gentry, with more efficient encryption and shorter ciphertexts, there is still a considerable amount of effort until FHE can be practical for real-world applications. It has been discussed in [31] the possibility of using a somewhat homomorphic encryption, which is more efficient than their FHE counterparts, for applications that require only a limited amount of multiplication.

3) USING HOMOMORPHIC ENCRYPTION FOR IMAGE SEARCH

As discussed in Section II, the application of rank-ordered image search has different settings from many secure computation work such as privacy-preserving face recognition [10], [11]. The challenge here is that the database has access only to the encrypted images and encrypted features, and rank-ordered search results rather than a binary exact matching is required. To the best of our knowledge, there is no existing work that address the problem of rank-ordered image search using homomorphic encryption. Below, we discuss possible scenarios and constructions of using homomorphic encryption for secure image search.

We first provide some notations. We assume that there are N images in the database, and each image has a visual feature $\mathbf{f}_i \in \mathbb{R}^n$. The query image is denoted as Q and its visual feature is $\mathbf{q} \in \mathbb{R}^n$. Paillier homomorphic encryption of a plaintext message m is denoted as $[m]$, and fully homomorphic encryption is denoted as $[[m]]$. The encryption of a feature vector is just the encryption of its individual components, i.e., $[\mathbf{f}] = \{[f_1], [f_2], \ldots, [f_n]\}$.

(1) Scenario-1: Additive homomorphic with encrypted query: In this base-line scenario, we use additive homomorphic encryption to encrypt the visual features of both the database images and the query image. Since the database will return a list of encrypted images similar to the query image, encrypting the feature of the query image is important to prevent the server from inferring the content of returned images using the query feature.

The computational task in this scenario is to compute distance between encrypted vectors $[\mathbf{f}]$ and $[q]$. Take the commonly used L_2 distance as example, we need to compute $\sum_{i=1}^{n}(f_i - q_i)^2$ using only encrypted values $[f_i], [q_i]$. Unfortunately, with additive homomorphic encryption alone, such computation is impossible without decryption because the computation involves both addition and multiplication. Since the database holds only the encrypted features without knowing the secret key, in order to proceed with the computation, the database needs to send back all the encrypted features $[\mathbf{f}_i]$, $i \in \{1, \ldots, N\}$ to the user. The user then decrypts all the features and compute distances on his/her end. The ranking result on the computed distances will be sent back to the database to retrieve similar images. Although the visual features typically have smaller size than the image itself, this naïve base-line scenario is still highly impractical because each query will require the database sending back the entire database of encrypted features. To be more efficient, the user might as well stores all the visual features on his/her local machine and computes similarity by his/herself. This alternative costs storage space and computational burden on the user and fails to utilize the computation power of online services.

(2) Scenario-2: Additive homomorphic with plaintext query: In order to fully utilize the computational power of the cloud, we need to minimize the computation and involvement on the user side. In this scenario, we make a relaxation such that the query feature is not encrypted but sent in plaintext to the database.

The computational task in this scenario is to compute distance between an encrypted feature $[\mathbf{f}]$ and a plaintext feature $\mathbf{q}$. This can be done directly in the database without communication with the user. We give two examples with dot product and L_2 distance, respectively. Computing dot product between a plaintext vector and an encrypted vector is directly supported by additive homomorphic. To see this, the dot product $\mathbf{f} \cdot \mathbf{q} = \sum_{i=1}^{n} f_i q_i$ can be computed in the encrypted domain as $[\mathbf{f} \cdot \mathbf{q}] = \Pi_{i=1}^{n}[f_i]^{q_i}$, where $\mathbf{q}$ is the plaintext query feature. For L_2 distance, $\|\mathbf{f} - \mathbf{q}\|_2 = \sum_{i=1}^{n}(f_i - q_i)^2 = \sum f_i^2 - 2\sum f_i q_i + \sum q_i^2$. The encrypted distance value thus can be computed as $[\sum f_i^2] \cdot (\Pi[f_i]^{q_i})^{-2} \cdot [\sum q_i^2]$. To allow the database compute the distance without interacting with the user, the user can provide the database an encrypted value $[\sum f_i^2]$ for each feature in the database.

The N encrypted distance values between the query feature and every database feature will then be sent back to the user for decryption and ranking. The security consideration of allowing the query feature in clear is that the database can infer the content of the query image and the final images returned from the search. To mitigate such a security concern, the user can add some noise to the ranking result, so that not all requested images will be similar to the query. Adding noise increases security at the cost of less accurate search.

(3) Scenario-3: Fully homomorphic with encrypted query: In this last scenario, we consider that FHE is used to encrypt features from both the query and database images. Despite that there is no efficient FHE implementations available, this scenario still helps us understand how FHE, if efficiently available in the future, can help address the problem of confidentiality-preserving image search.

With both query and database features encrypted by FHE, the computation of any distance function between $[[\mathbf{f}]]$ and $[[\mathbf{g}]]$ can be done directly in the encrypted domain without interaction with the user. However, the ranking of the encrypted distance values cannot be done alone by the database. This is because a semantically secure FHE should prevent the database from learning any information from the ciphertext, therefore, the database cannot learn ranking information from the encrypted distances without interacting with user. To obtain the final ranking, the database can either send N encrypted distance values to the user or send $N(N-1)/2$ encrypted binary values indicating the pair-wise relation of encrypted distances. The user then computes the ranking and requests similar images from the database. We can see that even FHE cannot completely eliminate the interaction with the user in order to complete the task of content-based image search.

B. RANDOMIZATION TECHNIQUES FOR VISUAL FEATURES AND SEARCH INDEXES

Given the high computational and communication complexity involved in using homomorphic encryption for the task of rank-ordered image search, Lu et al. [15], [16] proposed to address the problem from a practical perspective and ask what can be done now as efficient solutions for this kind of practical applications. Considering that such consumer-oriented applications have less stringent security requirement than that required for highly sensitive data such as biometrics, but requires high efficiency and minimum user involvement, privacy-aware search techniques that are efficient and provides a reasonable amount of protection can be interesting alternatives to homomorphic encryption based schemes. Below, we briefly review two types of such alternatives proposed in [15] and [16], namely, distance preserving randomization of visual features and search indexes, respectively.

1) DISTANCE PRESERVING RANDOMIZATION OF VISUAL FEATURES

The idea of feature randomization is to scramble the content of visual features but approximately preserve the distance between features after randomization. Three types of feature randomization schemes are proposed in [15], namely, bit-plane randomization, random projection, and the randomized unary encoding.

a: BIT-PLANE RANDOMIZATION

The motivation behind the bit-plane randomization is that feature vectors with small distances are likely to have similar patterns among their most significant bit-planes (MSB). Given a feature vector $\mathbf{f} = [f_1, \ldots, f_n] \in \mathbb{R}^n$, each component f_i is represented in its binary form as $[b_{i1}, \ldots, b_{il}]^T$, where b_{i1} is the first MSB, b_{il} is the least significant bit (LSB), and l is the total number of bit-planes. The j^{th} bit-plane of $\mathbf{f}$ is composed of the j^{th} MSB of the n feature components, denoted as $[b_{1j}, b_{2j}, \ldots, b_{nj}]$. To scramble the feature vector, each bit-plane is XORed with a binary random vector and then randomly permuted. The XOR pattern and permutation for the same j^{th} bit-plane is the same so that Hamming distance between corresponding bit-planes is exactly preserved. The randomization of the j^{th} bit-plane is illustrated in Fig. 2, where $[r_{1j}, r_{2j}, \ldots, r_{nj}]$ is the binary random vector used for XOR.

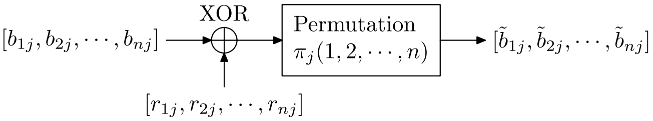

XOR
$[b_{1j}, b_{2j}, \cdots, b_{nj}]$
Permutation
$\pi_j(1, 2, \cdots, n)$
$[\tilde{b}_{1j}, \tilde{b}_{2j}, \cdots, \tilde{b}_{nj}]$
$[r_{1j}, r_{2j}, \cdots, r_{nj}]$

FIGURE 2. Randomization of the j^{th} bit-plane.

All the randomized bit-planes are reassembled to form the randomized feature vector $\mathcal{E}(\mathbf{f}) = [\tilde{f}_1, \ldots, \tilde{f}_n]$. The distance between two randomized feature vectors $\mathcal{E}(\mathbf{f})$ and $\mathcal{E}(\mathbf{q})$ is computed using a weighted sum of Hamming distances between their individual bit-planes:

$$d_{\mathcal{E}}(\mathcal{E}(\mathbf{f}), \mathcal{E}(\mathbf{q})) = \sum_{i=1}^{n} \sum_{j=1}^{l} |\tilde{b}_{ij}^{(\mathbf{f})} - \tilde{b}_{ij}^{(\mathbf{q})}| \times 2^{-j}. \tag{1}$$

This distance metric between randomized features is an upper bound on the L_1 distance between the original features:

$$d_{\mathcal{E}}(\mathcal{E}(\mathbf{f}), \mathcal{E}(\mathbf{g})) \geq \sum_{i=1}^{n} \left| \sum_{j=1}^{l} (b_{ij}^{(\mathbf{f})} - b_{ij}^{(\mathbf{g})}) \times 2^{-j} \right| = \|\mathbf{f} - \mathbf{g}\|_1.$$

b: RANDOM PROJECTION

The motivation to use random projection as a feature randomization technique is that close points in a high dimensional space will be mapped to close points in a low dimensional space with high probability. Random projection has been proposed to be used for secure image retrieval [15] and secure biometric matching [32].

Given a feature vector $\mathbf{f} \in \mathbb{R}^n$, a key-dependent Gaussian random matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$ with independent standard Gaussian components will be generated. The randomized feature will be computed as

$$\mathcal{E}(\mathbf{f}) = \mathbf{R} \cdot \mathbf{f} = [\mathbf{r}_1 \cdot \mathbf{f}, \ldots, \mathbf{r}_m \cdot \mathbf{f}] \in \mathbb{R}^m, \tag{2}$$

where $\mathbf{r}_i \cdot \mathbf{f}$ is the dot product between the i^{th} row of $\mathbf{R}$ and $\mathbf{f}$. The distance preserving property of random projection can be derived by considering the L_1 distance of randomized features:

$$\begin{aligned} &d_{\mathcal{E}}(\mathcal{E}(\mathbf{f}), \mathcal{E}(\mathbf{g})) \\ &\quad = \sum_{i=1}^{m} |\mathbf{r}_i \cdot \mathbf{f} - \mathbf{r}_i \cdot \mathbf{g}| = \sum_{i=1}^{m} \|\mathbf{r}_i\|_2 \cdot \|\mathbf{f} - \mathbf{g}\|_2 \cdot |\cos(\theta_i)| \\ &\quad = \|\mathbf{f} - \mathbf{g}\|_2 \cdot \sum_{i=1}^{m} \|\mathbf{r}_i\|_2 \cdot |\cos(\theta_i)| \approx c \cdot \|\mathbf{f} - \mathbf{g}\|_2 \end{aligned} \tag{3}$$

Here, θ_i is an independent and identically distributed random variable representing the angle between the vector $\mathbf{f}-\mathbf{q}$ and the random vector $\mathbf{r}_i$. By the law of large numbers, $\|\mathbf{r}_i\|_2 \approx$ const and $\sum_{i=1}^{m}|\cos(\theta_i)| \approx$ const. Thus, the distance $d_{\mathcal{E}}(\cdot,\cdot)$ between randomized features is proportional to the L_2 distance between the original feature vectors with high probability [33].

The security of random projection based scheme is due to the fact that without knowing the secret key and thus the projection matrix $\mathbf{R}$, it is extremely difficult to reconstruct the exact original features from the projected ones.

c: RANDOMIZED UNARY ENCODING

To further improve security, randomized unary encoding is proposed by combining the idea of bit-plane randomization and random projection. Given $\mathbf{f} = [f_1, \ldots, f_n]$, its unary encoding $\mathcal{U}(\mathbf{f})$ is:

$$\mathcal{U}(\mathbf{f}) = [\mathcal{U}(f_1), \mathcal{U}(f_2), \ldots, \mathcal{U}(f_n)],$$
$$\mathcal{U}(f_i) = \underbrace{11\cdots11}_{f_i}\underbrace{00\cdots00}_{M-f_i}.$$

Here M is the maximum possible value for any component of $\mathbf{f}$. This unary encoding $\mathcal{U}(\mathbf{f})$ is XORed with a vector of binary random variables $\mathbf{r}$ and then randomly permuted. The XOR and random permutation preserve the Hamming distance among $\mathcal{U}(\mathbf{f}), \forall\mathbf{f}$, which also equals the L_1 distance between original feature vectors. Denoting the randomization by XOR and permutation as $\mathcal{E}_1(\cdot)$, we have $\|\mathcal{E}_1(\mathcal{U}(\mathbf{f})) - \mathcal{E}_1(\mathcal{U}(\mathbf{g}))\|_2 = \|\mathbf{f}-\mathbf{g}\|_1$. Finally, random projection is applied on $\mathcal{E}_1(\mathcal{U}(\mathbf{f}))$, which reduces the randomized feature length and serves as an additional layer of randomization to improve security. Denote the random projection as $\mathcal{E}_2(\cdot)$, the overall randomization function $\mathcal{E}(\cdot)$ is $\mathcal{E}(\mathbf{f}) = \mathcal{E}_2(\mathcal{E}_1(\mathcal{U}(\mathbf{f}))) \in \mathbb{R}^m$. Considering the L_1 distance of randomized features, we have the approximate distance preserving property

$$d_{\mathcal{E}}(\mathcal{E}(\mathbf{f}), \mathcal{E}(\mathbf{q})) \approx c \cdot \|\mathcal{E}_1(\mathcal{U}(\mathbf{f})) - \mathcal{E}_1(\mathcal{U}(\mathbf{q}))\|_2 = c \cdot \|\mathbf{f}-\mathbf{q}\|_1.$$

2) DISTANCE PRESERVING RANDOMIZATION OF SEARCH INDEXES

Since image features are typically high dimensional vectors, comparing every pair of such vectors when doing a search is computationally prohibitive for a large database. Modern image retrieval techniques often achieve efficiency and scalability through well-designed search indexes. Below, we briefly review search index randomization technique proposed in [16].

a: SECURE INVERTED INDEX

Inverted index [34] is a widely used indexing structure in text document retrieval, where each keyword has an associated inverted index listing the documents that contain this keyword and the number of occurrences of this word in each of these documents. Only documents that appear in the query word's inverted index need to be considered during retrieval. By utilizing the visual words representation of images [35], inverted index can be constructed for image documents and facilitates efficient search and retrieval over large image databases.

In order to generate inverted index, a vocabulary tree is first created, where each node in the tree denotes a representative feature vector and each leaf node represents a visual word. Such a vocabulary tree can be constructed using hierarchical k-means clustering on a set of training features. Given the vocabulary tree, each visual feature of an image will be treated as a word and assigned to the closest visual word in the vocabulary tree, as shown in Fig. 3. An example of the inverted index is given in Fig. 4, which shows that image I_j contains w_j occurrences of the visual word i.

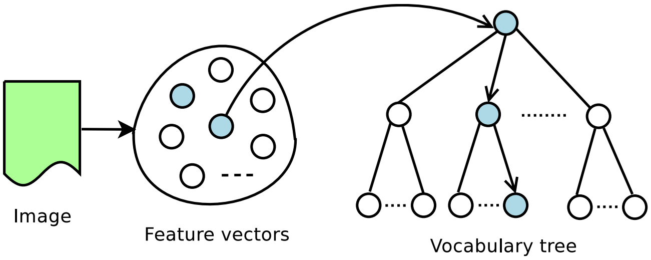


FIGURE 3. Inverted index generation by content owner.

Word ID	i			
Image ID	I_1	I_2	$\cdots$	I_{N_i}
Word frequency	w_1	w_2	$\cdots$	w_{N_i}

FIGURE 4. Data structure of inverted index.

To randomize the inverted index, a random permutation $\tau(\cdot)$ is first applied on the word IDs so that the i^{th} word will now have an ID $\tau(i)$. Following that, order preserving encryption (OPE) [36] is applied on all the word frequency values in the inverted indexes so that encrypted values have a distribution closer to uniform, thus minimizes the amount of information leaked to the server. At the same time, the preservation of the order information ensures that image similarity can still be compared in the encrypted domain.

After randomizion, the visual words representations of the query image and an image in the database are denoted by $\{\mathcal{E}(Q_1), \ldots, \mathcal{E}(Q_V)\}$ and $\{\mathcal{E}(D_1), \ldots, \mathcal{E}(D_V)\}$, respectively, where V is the total number of visual words, and Q_i and D_i represent the occurrence frequency of word i in the query and database images, respectively.

The similarity of two images is then measured by the Jaccard similarity between $\mathcal{E}(Q)$ and $\mathcal{E}(D)$:

$$\mathrm{Sim}(Q, D) \triangleq \frac{|\mathcal{E}(Q) \cap \mathcal{E}(D)|}{|\mathcal{E}(Q) \cup \mathcal{E}(D)|} \triangleq \frac{\sum_{i=1}^{V} \min(\mathcal{E}(Q_i), \mathcal{E}(D_i))}{\sum_{i=1}^{V} \max(\mathcal{E}(Q_i), \mathcal{E}(D_i))}. \quad (4)$$

As the order information used in $\min(\cdot,\cdot)$ and $\max(\cdot,\cdot)$ is preserved by the order preserving encryption, the Jaccard similarity computed from the encrypted indexes reflects the similarity of the plaintext indexes.

b: SECURE MIN-HASH

The min-Hash algorithm, first proposed by Broder et al. [37], provides another efficient way to compute the Jaccard similarity between the visual words representations of two images. The basic idea of the min-Hash algorithm is as follows: For any given set $\mathcal{A}$ such as the visual words representation, its min-Hash is defined as $m(\mathcal{A}, f) = \arg\min_{x \in \mathcal{A}} f(x)$, where f is a randomized hash function with the property that $\Pr[f(x) < f(y)] = \Pr[f(x) > f(y)] = 0.5$, $\forall x, y \in \mathcal{A}$ and $x \neq y$. The probability that two sets have the same min-Hash value is given by their Jaccard similarity defined in Equation (4).

To compare the similarity between a query image and an image in the database, their visual words representations $\{Q_1, Q_2, \ldots, Q_V\}$ and $\{D_1, D_2, \ldots, D_V\}$ will be converted to two sets:

$$\mathcal{A}(Q_{MH}) = \{X_1^1, \ldots, X_1^{Q_1}, \ldots, X_N^1, \ldots, X_N^{Q_N}\},$$
$$\mathcal{A}(D_{MH}) = \{X_1^1, \ldots, X_1^{D_1}, \ldots, X_N^1, \ldots, X_N^{D_N}\}.$$

Here, X_i^j is a unique element indexed by i and j. The min-Hash values generated from $\mathcal{A}(Q)$ and $\mathcal{A}(D)$ are essentially elements randomly selected from the two sets, and they satisfy

$$\begin{aligned}\Pr[m(\mathcal{A}(Q_{MH}), f) &= m(\mathcal{A}(D_{MH}), f)] \\ &= \mathrm{Sim}(Q_{MH}, D_{MH}) = \frac{\sum_{i=1}^{N} \min(Q_i, D_i)}{\sum_{i=1}^{N} \max(Q_i, D_i)}.\end{aligned} \tag{5}$$

In order to obtain a reliable estimate of $\mathrm{Sim}(Q, D)$, k independent hash functions $f_1, f_2, \ldots, f_k$ are used to generate k min-Hash values for $\mathcal{A}(Q)$ and $\mathcal{A}(D)$, respectively. The concatenation of the k min-Hash values for $\mathcal{A}(Q)$ forms a *sketch* of the image Q, and a sketch of the image D is formed similarly. The number of identical values in their sketches, denoted by $s(Q, D) = |\{i : 1 \leq i \leq k | m_i(Q) = m_i(D)\}|$, is then used to measure the similarity between the two images.

IV. COMPARISON METHODOLOGY AND RESULTS

In this section, we compare the two major types of confidentiality-preserving search techniques discussed above, namely, homomorphic encryption based and feature/index randomization based. Detailed experiments and quantitative analysis are provided in terms of their search accuracy, security strength, and computational efficiency.

A. COMPARISON ON SEARCH ACCURACY

For the task of content-based image search, a good search accuracy means that the top ranked images have high similarity to the query. Due to the semantic gap between visual features and high level concepts, irrelevant images may be returned as similar images and degrade the search accuracy. A good confidentiality-preserving search technique should retain as good search accuracy as possible when compared to conventional search without any protection.

1) EXPERIMENT SETUP

We perform the content-based search experiments on an image database containing 1000 color images from the Corel dataset [38]. These images are grouped by content into 10 categories, with 100 images in each category of African, Beach, Architecture, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountain, and Food. Image sizes are either 256 × 384 or 384 × 256. We choose this database for its existing categorization and it has been used as ground-truth for evaluating color image retrieval [39] and image annotation [40]. Sample images from the database are shown in Fig. 5. Another thing worth mentioning here is that the secure computation techniques discussed in this paper only operate on image features or search indexes rather than on image itself, and the computation of image features is done on the user side before sending all the data to the server. Therefore, the choice of dataset and image size is less critical here.

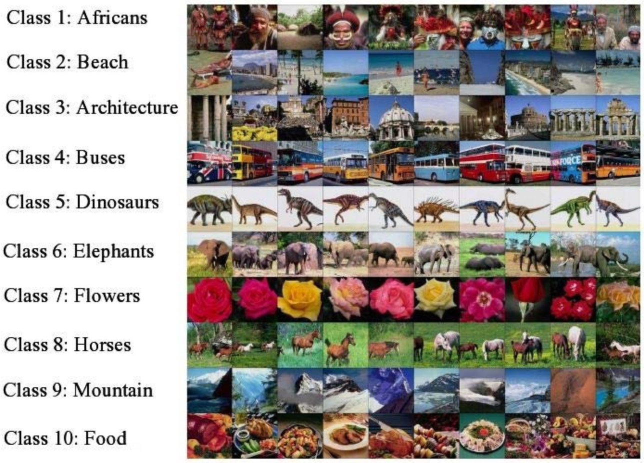


FIGURE 5. Selected content of the Corel dataset (figure from [39]).

We use global color histogram for the homomorphic encryption based techniques and feature randomization techniques, and use localized color histogram to generate indexes for the index randomization techniques. The color histograms are in the HSV (Hue, Saturation, and Value of brightness) color space [41]. For localized color histogram, we divide an image into 256 blocks and extract a 128-dimensional color histogram from each block by quantizing the three channels of hue, saturation, and intensity value into 8, 4, and 4 levels, respectively. To generate the visual words representation, we obtain a training set of 256,000 local histograms from the entire database and perform hierarchical clustering to build the vocabulary tree. Each node in the vocabulary tree except the leaf nodes has 10 children and the tree has height 3, which gives 10^3 visual words.

Search accuracy is evaluated using precision-recall curves, where precision and recall are defined as

$$precision = \frac{\text{\# of positive images among returned images}}{\text{\# of returned images}},$$

$$recall = \frac{\text{\# of positive images among returned images}}{\text{\# of positive images in the database}}.$$

A higher precision value at a given recall value indicates better retrieval performance. Our experiments use every image in the database as a query, and positive images are those images in the same category as the query.

2) COMPARISON ON SEARCH ACCURACY

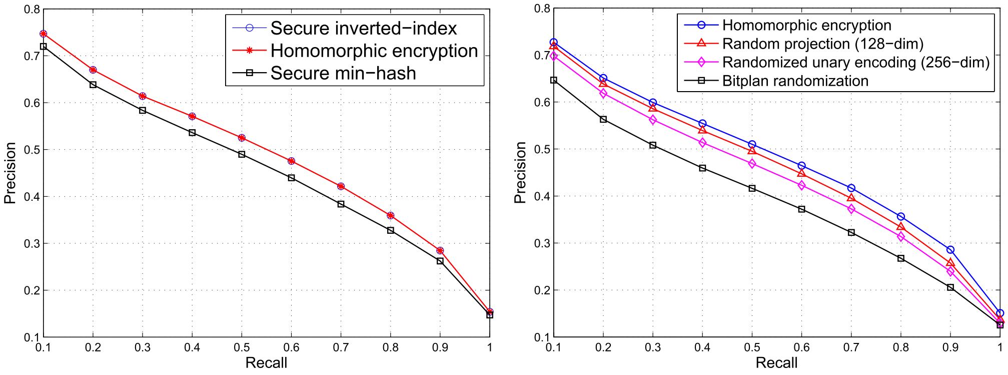
For conventional content-based image search without any protection, color histograms can be compared using L_1 distance. In the confidentiality-preserving search, the color histogram is either encrypted using homomorphic encryption or scrambled using feature/index randomization techniques.

Between the two types of confidentiality-preserving techniques, feature/index randomization technique scrambles the visual features or search indexes, and approximately preserves the distance between original features. The approximate distance preserving property ensures that the search accuracy is preserved with only slight degradation. To understand the search accuracy after involving homomorphic encryption, we note that homomorphic encryption operates on integer values. This implies that if the feature vector is in floating point, it has to be properly scaled and quantized. This brings quantization error to the distance computation, although such error can be made quite small thus the impact on the search performance is small. The color histogram used in this experiment contain only integer values, so homomorphic encryption will not cause quantization error and the distance between encrypted features will be exactly preserved. Therefore, we expect confidentiality-preserving search using homomorphic encryption to have the same performance as the conventional search. This is confirmed in our experiments.

Based on the experiment set-up discussed in the previous subsection, we obtain the search accuracy for index randomization techniques that operate on local color histograms, feature randomization techniques that operate on global color histograms, and homomorphic encryption based techniques for both local and global color histograms, respectively. The results are shown in Fig. 6. For techniques that operate on search indexes extracted from local color histograms [Fig. 6(a)], both the homomorphic encryption and secure inverted index retain the accuracy of using plaintext indexes. The secure min-hash technique has a slight performance drop at hash length 256, but its performance can be made close to the plaintext index by increasing the hash length. It should be noted that the distance metric used in secure inverted index and secure min-hash are the Jaccard similarity and the number of identical elements, respectively. Computing such distance metrics between vectors encrypted by homomorphic encryption is involved and requires heavy communication with the user.

For techniques that operate on global color histogram [Fig. 6(b)], the search accuracy using homomorphic encryption technique is the same as the search accuracy using plaintext features, and is better than the other three feature randomization techniques, namely, bit-plane randomization, random projection, and randomized unary encoding. We can see from this figure that random projection and randomized unary encoding preserve the search accuracy with a slight degradation. However, the search accuracy of using random projection and randomized unary encoding can also be made arbitrarily close to the performance of plaintext search by increasing the feature dimension. Among the three feature randomization techniques, bit-plane randomization has relatively larger degradation on the search accuracy because the distance between randomized features is only an upper bound on the original L_1 distance.

The comparison above demonstrates that homomorphic encryption can retain the exact search accuracy of a conventional scheme that operates on plaintext features, while the index and feature randomization techniques also achieve



(a) (b)

FIGURE 6. **Comparison of search accuracy between different techniques. (a) Index randomization vs. Homomorphic encryption. (b) Feature randomization vs. Homomorphic encryption.**

performance very close to that of the homomorphic encryption. The gap between index/feature randomization techniques and plaintext search can be made arbitrarily small by increasing the feature dimension for techniques such as random projection, randomized unary encoding, and secure min-hash. It should be noted that homomorphic encryption will greatly expand the encrypted feature size, which we will discuss later in Section IV-B.4.

B. COMPARISON ON SECURITY-EFFICIENCY TRADE-OFF

In this section, we discuss the security concerns for the application of confidentiality preserving image search, develop a suite of metrics, and demonstrate quantitative results on the protection level achieved by the different techniques; we will then specifically discuss the challenges in employing techniques such as homomorphic encryption and cryptography protocols in terms of their computational and communication complexity.

1) SECURITY OBJECTIVE FOR RANK-ORDERED IMAGE SEARCH

In the confidentiality preserving image search scenario considered in the paper, the server stores only the encrypted images and randomized features, and performs retrieval based on randomized query features. We model the server as a semi-honest adversary, i.e., it follows the execution requirement of the protocol but may use what it sees during the execution to infer additional information. Such a semi-honest model is applicable to such scenarios as web service providers, who would like to learn as much as possible about the users for benefits such as better targeted ads, but would not deliberately break the users' privacy. A user who uses these third-party services wants to utilize the service's computational power for reliable storage, easy access, and better organization of his/her private data set, but wants to reveal as little information as possible to the server beyond what is necessary for the server to provide the necessary services.

Given that the database images are already encrypted using highly secure ciphers, the security objective will be to minimize information revealed from the encrypted or randomized features and from the search process. Content-based image retrieval relies on comparison of different types of visual features to capture visual or semantic similarity between images. Storing raw features without any protection or randomization is never wise, because visual features can reveal important information about image content. First of all, raw features have fixed structure, from which an adversary can infer certain aspects of image content. For example, each bin in a color histogram reveals proportion of that color in the image. A large proportion of blue color might indicate sky or sea, while a large proportion of green color can suggest trees or grasses. Second, storing raw features allows an adversary to compare them with features of other known images. For example, a close match of salient features such as SIFT can give an adversary high confidence that an encrypted image may contain certain objects such as buildings and landmarks. Both the homomorphic encryption based technique and feature/index randomization techniques will hide the fixed visual feature structure and values, and make it difficult for an adversary to probe the content of encrypted images using known images.

The second source of information leakage is from the search process, where the server will compute distance between the query feature and all the features stored in the database. The result is a list of images ranked by their similarity to the query. The information revealed in this process is the similarity among database images. We will see that such an information leakage is inevitable for the application of rank-ordered search. The first major reason is that the server provides the search functionality and needs to return the similar images. Therefore, the server will know that the returned images are similar to each other. This is different from some secure multi-party computation problem such as binary matching of biometrics or text keywords where only a binary answer is returned and the server can be made oblivious of the matching result. We will show in the following subsections that the utility requirement of returning similar images has some inherent security implications that need to be taken into account when designing secure solutions. The second reason is efficiency. Allowing the server to compare distance between randomized features is necessary to achieve a practical scheme that avoids multiple rounds of communication between the server and the user, as is typically required in secure multi-party computation. This is particularly important for search over large image databases beyond a few hundred or thousand entries, because for each query, the communication bandwidth involved in sending intermediate encrypted values, such as homomorphicly encrypted distance values with respect to all database entries, back to the user for distance comparison is formidably expensive.

For homomorphic encryption based technique, the server can infer image similarity by observing the search results. For feature/index randomization techniques, the server can directly compute feature distances to infer the similarity information about the encrypted images, and learn the distance distribution between the raw features, because the randomization techniques are approximately distance-preserving. For text documents, the relative frequency of letters or words may reveal its plaintext counterpart, but image content and their signal representations are far more diverse than letters and words. Therefore, we expect that the distribution of distances among visual features encodes only a limited amount of information and cannot be easily used to infer the plaintext image content by an adversary. In the following subsections, we design several experiments to study the security implication of revealing distance distribution and validate the above arguments.

2) PROTECTION ON INDIVIDUAL FEATURES

As we mentioned earlier, the raw visual features have fixed structure, so that each element in a feature vector has physical meanings that may reveal image content information.

Simple permutation of the feature vector is not sufficient because visual feature values typically have smoothness and correlation property that can be exploited. Homomorphic encryption of each feature value essentially converts the feature vector into a random vector where each component can be considered as an i.i.d. random variable; feature/index randomization techniques scramble the feature structure and increase randomness of the resulting feature values by jointly using cryptographic primitives and signal processing techniques, while approximately preserving distance between feature vectors.

We use three different metrics to measure the level of protection achieved by the different encryption and randomization techniques. The three metrics are autocorrelation function, entropy, and conditional entropy of the feature vectors. We also generate random feature vectors whose values are drawn from i.i.d. uniform distribution to simulate the results that we can expect from homomorphic encryption of the feature vectors.

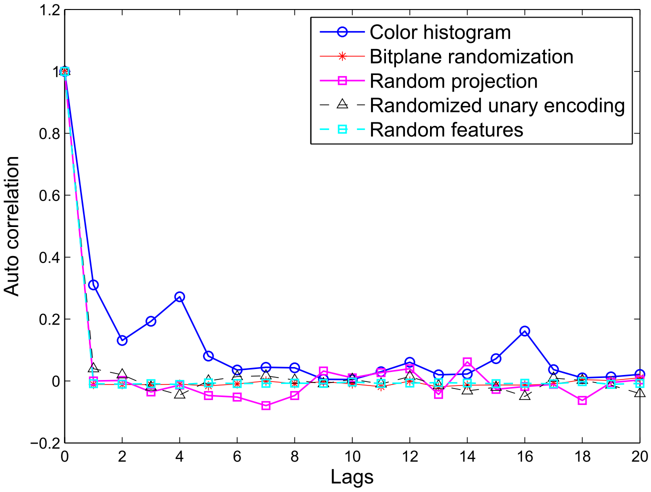


FIGURE 7. **Autocorrelation function on randomized features.**

The first metric, the autocorrelation function of a feature vector, measures how correlated the neighbouring feature elements are. The autocorrelation function for the raw color histogram, visual words representation, and randomized features/indexes using different algorithms are shown in Figs. 7 and 8. We can see that the original color histogram and visual words representation both have non-negligible correlation for lags larger than 0, which means there exists at least some correlation between nearby feature values. For both encrypted and randomized features/indexes, the correlation between neighboring feature values or index dimensions have been reduced to close to 0, similar to what we can expect from a sequence of i.i.d. random numbers.

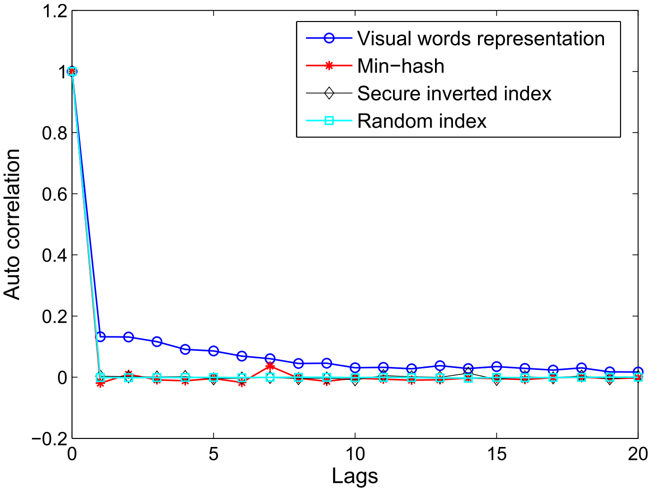


FIGURE 8. **Autocorrelation function on randomized indexes.**

The other two metrics are entropy and conditional entropy of the feature vectors. Given all the feature vectors generated from the 1000 images in the Corel image database, we quantize the entire range of feature values into 256 levels. We then consider the quantized feature value as a random variable and measure its entropy. A higher entropy indicates the feature value has a distribution closer to uniform, thus higher randomness. The conditional entropy $H(X_2|X_1)$ measures randomness of a feature value given its immediate neighbor. The conditional entropy can be approximated by $H(X_2|X_1) = -\sum_{ij} \mu_i P_{ij} \log P_{ij}$, where μ_i is the ensemble distribution of the feature values and P_{ij} is the transition probability.

The entropy and conditional entropy for randomized features/indexes from different algorithms are shown in Table 1. The results are averaged over 50 runs of randomized features generated by different secret keys. We can see that both the raw color histogram and visual words representation have relatively low entropy and conditional entropy, which implies that raw features and indexes have limited randomness and demonstrate inherent smoothness and correlation among feature values. The features encrypted by homomorphic

TABLE 1. **Entropy and conditional entropy for randomized features/indexes.**

Feature type	$H(X)$	$H(X_1\|X_2)$	Index type	$H(X)$	$H(X_1\|X_2)$
Color histogram	1.95	1.71	Visual words	2.59	2.50
Bitplane randomization	7.72	5.05	Min-hash	7.93	5.82
Random projection	7.00	6.80	Secure inverted index	7.97	7.26
Random unary encoding	6.90	6.80			
Gaussian random vectors	6.89	6.72	Uniform random vectors	8.00	7.58

encryption are expected to have i.i.d. uniform distribution, so their randomness is compared to uniform random vectors, which achieves the highest entropy and entropy rate. The feature/index randomization techniques also generate protected features with high entropy similar to that of pure random vectors. Since we used 256 levels to quantize the feature values, the maximum possible entropy is 8 bits for a uniform random variable, and lower for a Gaussian random variable. The features from random projection and randomized unary encoding follow Gaussian distribution, and we can see their randomness is close to what can be achieved by a Gaussian random vector; while the other randomized features/indexes all approximately follow an uniform distribution, and their entropies are close to 8 bits.

The above experiments indicate that the feature/index randomization techniques can generate features and indexes that have similar randomness to a pure random vector or features after the homomorphic encryption. The feature structure is scrambled, and the correlations among individual feature values are significantly reduced. The physical meaning in the feature vectors are therefore hidden from the adversaries.

3) PROTECTION ON THE SEARCH PROCESS

During the search process, the server will compute distance between randomized features and return a list of encrypted images ranked by their similarity to the query. Therefore, the server will know that the returned images are likely to be similar, and for feature/index randomization techniques, the server will also know the distance distribution among the randomized features. In this subsection, we carry out several experiments to see if revealing such information will be of significant security concern for feature/index randomization techniques.

a: CLUSTERING ON RANDOMIZED FEATURES

For homomorphic encryption schemes, the distance between feature vectors are encrypted and thus not directly obtainable by the server. From server's perspective, the encrypted features are the same as a set of i.i.d. uniform random vectors. For feature/index randomization techniques, since the server can compute the distances between randomized features, it will be able to perform clustering of all the features in the database and group encrypted images into clusters where each cluster contains images that are likely to be similar to each other. In the Corel image database that we used here, there are 10 categories each with 100 images. A perfect clustering will generate 10 categories each with the exact 100 images from that category. The better clustering that the server can obtain using the distances among features, the more information about the database is revealed from the feature distance information.

We carry out K-means clustering on the randomized features/indexes as well as on the i.i.d. uniform random vectors that we expect from homomorphic encryption, respectively. We assume that the server knows the number of clusters in the database as prior knowledge. We employ two scores to measure the randomness of the clustering result. The first one is the average entropy of image categories over the 10 clusters. We consider the image category as a random variable, taking values from 1 to 10. After clustering, each cluster will contain a list of images each with a category number. The entropy of image category can be computed for each cluster and averaged to get a value of average cluster entropy. A perfect clustering will generate an average cluster entropy of 0, and a higher entropy indicates that the clustering is more random and more different from the ground-truth. The second score is the number of unique image categories among the 10 clusters. For each cluster, we consider the dominant image category as a cluster category, and then count the total number of unique cluster categories from the clustering. A perfect clustering will generate 10 unique categories. The clustering results are shown in Table 2, which have averaged over 50 runs of K-means clustering with different initial random centroids.

Several points can be learned from the results in Table 2. First, clustering on the random feature vectors or vectors after homomorphic encryption achieve the highest entropy and fewest unique cluster categories, indicating their clustering result is most different from the ground-truth. Second, the randomized features and indexes from the randomization techniques achieve similar randomness to that of the raw color histogram. This can be expected from the approximate distance preserving property of the randomization algorithms. Third and most importantly, even the clustering results on raw color histogram are still somewhat different from the ground-truth. We observed that each cluster typically contains images from 3 to 6 different categories. This can be mainly attributed to the semantic

TABLE 2. K-mean clustering results.

Feature type	Average cluster entropy	Number of unique cluster categories
Color histogram	1.61	8.72
Random projection	1.50	8.40
Randomized Unary encoding	1.56	8.52
Secure Min-hash	1.92	7.96
Secure inverted index	2.16	7.56
Bitplane randomization	2.49	7.74
Random feature vectors	3.26	7.06

gap in image search, where low level visual features cannot capture very well high level semantic concepts. In other words, there is a gap from knowing the visual features to knowing the semantic concept of the image, which actually helps add another favorable uncertainty layer for image related security applications.

b: IMAGE CATEGORY INDISTINGUISHABILITY

From previous experiment, we know that the server is not able to obtain the ground-truth clustering from the randomized features. In this subsection, we perform experiments to demonstrate that even if the server can obtain the ground-truth clustering, these clusters of randomized or encrypted features will be highly indistinguishable from the server's point of view.

We assume that the server has the prior knowledge of the category names in the database, but does not know which name corresponds to each of the 10 clusters of encrypted images. In the Corel image database used in this paper, the 10 categories are "African", "Beach", "Architecture", "Buses", "Dinosaurs", "Elephants", "Flowers", "Horses", "Mountain", and "Food". The first experiment we carry out here is to see that given a plaintext image from one of the 10 categories, whether the server can successfully associate it with the correct cluster of encrypted images. Since the server does not know the secret key used in randomization, we will randomize the feature of the known plaintext image using a randomly chosen key and use the randomized feature as query to compare with features in the database. The retrieval performance of using every image in the database as query but randomize its feature using a randomly chosen key is shown in Figs. 9 and 10.

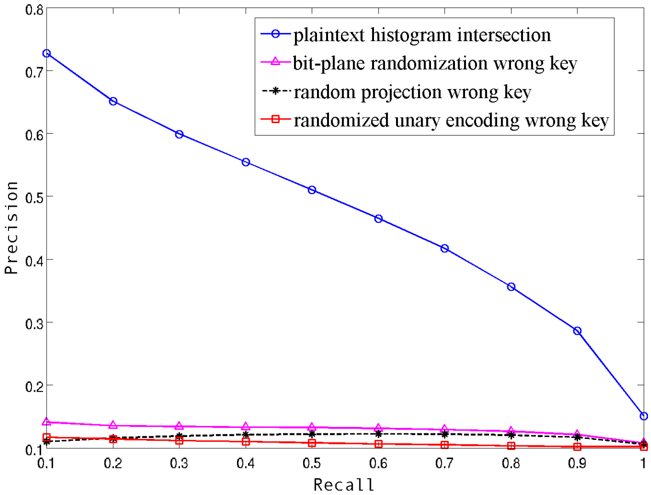


FIGURE 9. Retrieval using a wrong key for feature protection schemes.

Since the database has 100 images in each of the 10 categories, a random selection from the database would imply a precision value around 0.1 for all recall values. From this figure, we can see that the retrieval precision of feature/index randomization techniques is reduced to around 0.1 if a different secret key is used to randomize the feature or index. In other words, a query index randomized by a different key from the correct one will be equally like to be closest to any randomized feature in the database. Therefore, without knowing the correct secret key, retrieval from an encrypted database is equivalent to picking images randomly from the database. The chance of the server associating a plaintext image of known category to the correct cluster in the encrypted database is no better than random guessing.

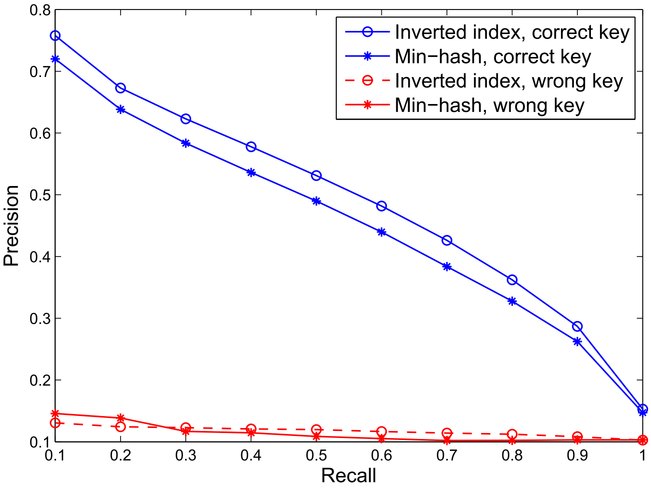


FIGURE 10. Retrieval using a wrong key for secure index schemes.

Next, we carry out an experiment to see when the server has multiple plaintext images from some image category, whether the distribution of visual features among those images of the same category can be used to differentiate clusters of encrypted images. For each of the 10 categories in the Corel database, we first divide the 100 images in that category equally into two sets $S_{i1}, S_{i2}, i = 1, 2, \ldots, 10$, each with 50 images. The distance distributions from $\{S_{i1}\}$ are used as query to search for the closest match in distance distributions from $\{S_{i2}\}$. The purpose of such an experiment is to see whether the distance distribution of visual features has sufficient discriminative power to differentiate different image categories. The less distinctive the distance distributions are among image categories, the less information about the image database is revealed. Kullback-Leibler divergence is used as a distance metric for the distributions, and the probability of correct match over 100 runs with different secret keys is shown in Table 3.

From the table, we can see that for the 10 categories in the Corel database used in this paper, the probability of correctly matching two distance distributions from the same category is 40% for both the raw color histogram and visual words representation. This relatively low match accuracy, as compared to the search accuracy using visual features, implies that distance distribution between the visual features is not a very good discriminative feature to differentiate different image categories. After randomization, the match accuracy on the randomized features and indexes are further reduced. Especially for randomized unary encoding, the match

TABLE 3. Search accuracy using distance distribution of different categories.

Feature type	Prob. of match	Feature type	Prob. of match
Color histogram	40%	Visual words	40%
Bitplane randomization	29.7%	Secure inverted index	24.8%
Random projection	25.4%	Secure Min-hash	16.6%
Randomized unary encoding	9.8%	Random feature vectors	9.7%

accuracy is close to 10%. This means that for $M = 10$ categories of images, obtaining category information after randomized unary encoding is essentially a random guess with an accuracy of $1/M$. So for a larger image database with more categories, thus larger M, we can expect the match accuracy to further decrease.

The experiments in this subsection show that due to the extremely diverse representation of image data and the semantic gap between the visual features and semantic concept, it is extremely difficult for an adversary to learn useful information about the image content from the distance distribution of visual features. Measured by the suite of scores developed in the above experiments, the feature/index randomization techniques can achieve security performance close to that of homomorphic encryption.

4) CHALLENGES IN EMPLOYING HOMOMORPHIC ENCRYPTION BASED TECHNIQUES

From the previous comparisons, we have seen that homomorphic encryption schemes achieve exactly the same search accuracy as that using plaintext features, and offer the highest amount of randomness in terms of confidentiality protection for the visual features and the search process. The feature/index randomization techniques, although not designed as encryption schemes, come very close to the performance of homomorphic encryption schemes in terms of both search accuracy and minimal information leakage from correlation, entropy, and clustering aspects. In this section, we discuss some practical challenges and considerations when employing these two types of techniques for confidentiality preserving rank-ordered image search.

a: SECURITY BENEFIT OF CRYPTOGRAPHIC APPROACHES

Feature/index randomization techniques proposed in [15] and [16] are designed with efficiency and distance preserving property in mind, but strictly speaking, they are not encryptions as those commonly used cryptographic ciphers. Secure cryptographic ciphers require semantic security, which demands randomized encryption. Due to the distance preserving requirement, the feature/index randomization schemes are deterministic. Homomorphic encryption, on the other hand, offers randomized encryption and prevents the server from computing distance between encrypted features directly. In some secure computation problems such as those involve highly sensitive text documents and biometrics, such a security benefit of randomized encryption and computation results obfuscation would be important. However, as we shall see next, for rank-ordered image search, the security benefit from homomorphic encryption may not always justify its high computational complexity.

In the comparisons presented in the previous section, we have shown that distance distribution among visual features does not leak sensitive information about the image content in the database. Furthermore, the requirement for the server to return a list of encrypted images similar to the query brings some inherent security implication that may diminish the benefits from cryptographic techniques such as homomorphic encryption. The main reason is that the utility requirement of returning similar images inevitably reveals the information that those returned images are similar to each other. Therefore, even if the encryption for the individual features are semantically secure, some information about the ciphertext will be revealed.

The less appropriateness of requiring semantic security is similar to the application of statistical databases, where the database is required to return global statistical information about the private data it holds. Such a utility requirement makes semantic security impossible for statistical database, as shown by Dwork [42]. Instead, differential privacy is used to quantify security from a different perspective for those applications where ciphertext carries utility to the adversary and semantic security is impossible. Typical technique to achieve differential privacy is to add noise to the returns from the database at the cost of noisy and less useful results. Exploring differential privacy formulation for the problem of image search can be an interesting issue for future research, but may not be trivial or feasible given the unique application settings of the problem considered in this paper. In summary, we can see that using homomorphic encryption does not bring significant security benefit over feature/index randomization techniques for the problem of confidentiality-preserving image search.

b: EFFICIENCY COST OF CRYPTOGRAPHIC APPROACHES

In addition to the limited security benefit, the huge computational and communication complexity is another major limitation of employing homomorphic encryption schemes at this moment. First of all, today's popular homomorphic encryption techniques such as Paillier's cryptography system, are computationally intensive and incurs a large amount of ciphertext expansion. to offer a practical sense of the complexity, we list in Table 4 a comparison between the Paillier homomorphic encryption and several randomization

TABLE 4. Efficiency comparison of feature randomization schemes.

	Encryption time	Ciphertext size / expansion factor
Paillier Homomorphic	1778.5s	32005KB / 241.8
Bitplane randomization	0.24s	159KB / 1.2
Random projection	0.38s	462KB / 3.5
Randomized unary encoding	9.64s	457KB / 3.5
Secure inverted index	0.32s	246KB / 2.1
Secure Min-hash	3.04s	296KB / 2.5

TABLE 5. Summary of comparison.

	Paillier homomorphic encryption	Feature/index randomization
Advantages	• High search accuracy • Randomized encryption	• High search accuracy • Computationally efficient • Minimum user involvement
Disadvantages	• High computational complexity • Frequent user involvement	• Deterministic randomization

algorithms reviewed in this paper. The encryption time and ciphertext size are for all 1000 features/indexes in the Corel database. All implementations are in C/C++ and run on a Linux desktop with 3.0GHz dual core CPU and 4GB RAM. The homomorphic encryption implementation is based on a C library from http://acsc.cs.utexas.edu/libpaillier/. The randomized features are stored in binary format and further compressed using ZIP. We can see that homomorphic encryption takes 2-4 orders of magnitude of longer time to encrypt the 1000 color histograms from the Corel database, and results in 2-3 orders of magnitude of larger expansion on the feature size. The ciphertext expansion also implies that homomorphic encryption will incur high communication cost in order to transfer the encrypted features to the server. Among the feature/index randomization techniques, randomized unary encoding and secure min-hash have relatively longer running time, because they have more randomization steps in their algorithms.

Fully homomorphic encryption is an active research area today. The current complexity is still much higher than Paillier and far from being practical. Therefore, we do not include fully homomorphic encryption in this comparison. Given the promising progress seen in recent years, we look forward to potential break-through along this line of research and believe that advances on both the homomorphic encryption and joint signal processing and randomization techniques would help expand the solution space to accommodate a broad variety of applications with different trade-off and performance requirements.

C. SUMMARY OF COMPARISON

In this section, we compared the two types of techqnieus for confidentiality-preserving image search, namely, homomorphic encryption based techniques and feature/index randomization techniques. The main advantages and disadvantages of both techniques are summarized in Table 5.

The advantages of using homomorphic encryption are that it retains the search accuracy of plaintext features and offers randomized encryption so that the server cannot obtain distance between encrypted features directly. The disadvantage is that the currently established homomorphic techniques are too computation and communication intensive to be practical, requiring frequent user involvement in order to obtain the ranking results. On-going and future advancement in cryptography techniques, such as that on efficient fully homomorphic encryption and light weight secure comparison protocols [43], [44], will be critical in making cryptography based approach more practical for the application of content-based image retrieval.

On the other hand, feature/index randomization techniques have the advantage of being highly efficient and requiring minimum user-involvement. The search accuracy and confidentiality protection offered by feature/index randomization are very close to that of homomorphic encryption schemes. The limitation of feature/index randomization is that they are deterministic methods and thus the server can learn distance distribution of randomized features. We have provided various experiments to demonstrate that the revealing distance distribution is not a significant security concern for image data, and that the utility requirement of rank-ordered search has some inherent security implications that may diminish the security benefit of using homomorphic encryption.

V. CONCLUSION

In this paper, we have studied the problem of confidentiality-preserving content-based image search. This problem has many practical applications such as secure online services that help manage personal image collections, and the problem also has several challenging research issues, such as achieving a good trade-off between security and efficiency for practical applications that demands high efficiency and least user involvement. We have reviewed two major types of techniques for this problem, namely techniques based on homomorphic encryption, and techniques based on visual feature and search index randomizations. We have provided quantitative comparison of these two types of techniques in terms of search accuracy, security strength, and computational efficiency. The homomorphic based technique is more secure but too heavy-weight in terms of computational complexity, communication load, and user involvement for practical applications, while the feature/index randomization techniques offer very high efficiency using deterministic distance-preserving randomization at the cost of revealing some information about the distance distribution among randomized features. Both techniques achieve good search accuracy as compared to conventional search that does not have privacy protection. We hope the comparison study offered in this paper can provide useful insights in designing privacy-aware techniques for the problem of confidentiality-preserving image search as well as other real-world secure online applications with various levels of security and efficiency requirements.

REFERENCES

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

[2] R. Brinkman, J. M. Doumen, and W. Jonker, "Using secret sharing for searching in encrypted data," in *Proc. Workshop Secure Data Manag. Connected World*, 2004, pp. 18–27.

[3] D. Boneh, G. Crescenzo, R. Ostrovsky, and G. Persiano, "Public-key encryption with keyword search," in *Proc. Eur.*, 2004, pp. 506–522.

[4] A. Swaminathan, Y. Mao, G.-M. Su, H. Gou, A. L. Varna, S. He, *et al.*, "Confidentiality preserving rank-ordered search," in *Proc. ACM Workshop Storage, Sec., Survivability*, 2007, pp. 7–12.

[5] Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, *et al.*, "Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing," *EURASIP J. Inf. Sec.*, vol. 7, no. 2, pp. 1–20, 2007.

[6] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surveys*, vol. 40, no. 2, pp. 1–5, 2008.

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

[8] M.-L. Yiu, G. Ghinita, C. S. Jensen, and P. Kalnis, "Outsourcing search services on private spatial data," in *Proc. IEEE 25th Int. Conf. Data Eng.*, Apr. 2009, pp. 1140–1143.

[9] W. K. Wong, D. W.-L. Cheung, B. Kao, and N. Mamoulis, "Secure kNN computation on encrypted databases," in *Proc. 35th SIGMOD Int. Conf. Manag. Data*, 2009, pp. 139–152.

[10] Z. Erkin, M. Franz, J. Guajardo, S. Katzenbeisser, I. Lagendijk, and T. Toft, "Privacy-preserving face recognition," *Privacy Preserving Technol., LNCS*, vol. 5672, pp. 235–253, Aug. 2009.

[11] A.-R. Sadeghi, T. Schneider, and I. Wehrenberg, "Efficient privacy-preserving face recognition," in *Proc. 12th Int. Conf. Inf. Sec. Cryptol.*, 2009, pp. 229–244.

[12] M. Osadchy, B. Pinkas, A. Jarrous, and B. Moskovich, "Scifi—A system for secure face identification," in *Proc. IEEE Symp. Sec. Privacy*, May 2010, pp. 239–254.

[13] W. Jiang, M. Murugesan, C. Clifton, and L. Si, "Similar document detection with limited information disclosure," in *Proc. IEEE 24th Int. Conf. Data Eng.*, Apr. 2008, pp. 735–743.

[14] D. E. Yan Huang, L. Malka, and J. Katz, "Efficient privacy-preserving biometric identification," in *Proc. 18th Network Distrib. Syst. Sec. Symp.*, 2011, pp. 1–9.

[15] W. Lu, A. L. Varna, A. Swaminathan, and M. Wu, "Secure image retrieval through feature protection," in *Proc. IEEE Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 1533–1536.

[16] W. Lu, A. Swaminathan, A. L. Varna, and M. Wu, "Enabling search over encrypted multimedia databases," *Proc. SPIE*, vol. 7254, pp. 7254–7318, Jan. 2009.

[17] Y. Mao and M. Wu, "A joint signal processing and cryptographic approach to multimedia encryption," *IEEE Trans. Image Process.*, vol. 15, no. 7, pp. 2061–2075, Jul. 2006.

[18] M. Grangetto, E. Magli, and G. Olmo, "Multimedia selective encryption by means of randomized arithmetic coding," *IEEE Trans. Multimedia*, vol. 8, no. 5, pp. 905–917, Oct. 2006.

[19] H. Kim, J. Wen, and J. D. Villasenor, "Secure arithmetic coding," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 2263–2272, May 2007.

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

[21] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. 17th Int. Conf. Theory Appl. Cryptograph. Tech.*, 1999, pp. 223–238.

[22] T. El Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Proc. CRYPTO 84th Adv. Cryptol.*, 1985, pp. 10–18.

[23] S. Goldwasser and S. Micali, "Probabilistic encryption," *J. Comput. Syst. Sci.*, vol. 28, no. 2, pp. 270–299, 1984.

[24] I. Damgård and M. Jurik, "A generalisation, a simplification and some applications of Paillier's probabilistic public-key system," in *Proc. 4th Int. Workshop Pract. Theory Public Key Cryptograph.*, 2001, pp. 119–136.

[25] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proc. 41st Annu. ACM Symp. Theory Comput. (STOC)*, 2009, pp. 169–178.

[26] M. Van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, "Fully homomorphic encryption over the integers," in *Advances in Cryptology EUROCRYPT* (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag, 2010, pp. 24–43.

[27] D. Stehlé and R. Steinfeld, "Faster fully homomorphic encryption," in *Advances in Cryptology ASIACRYPT* (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag, 2010, pp. 377–394.

[28] N. P. Smart and F. Vercauteren, "Fully homomorphic encryption with relatively small key and ciphertext sizes," in *Public Key Cryptography* (Lecture Notes in Computer Sciences). Berlin, Germany: Springer-Verlag, 2010, pp. 420–443.

[29] Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-LWE and security for key dependent messages," in *CRYPTO* (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag, 2011, pp. 505–524.

[30] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proc. FOCS*, 2011, pp. 1–43.

[31] M. Naehrig, K. Lauter, and V. Vaikuntanathan, "Can homomorphic encryption be practical?" in *Proc. 3rd ACM Workshop Cloud Comput. Security Workshop, Ser. CCSW*, New York, NY, USA, 2011, pp. 113–124.

[32] J. K. Pillai, V. M. Patel, R. Chellappa, and N. K. Ratha, "Secure and robust iris recognition using random projections and sparse representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1877–1893, Sep. 2011.

[33] W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemp. Math.*, vol. 26, no. 1, pp. 189–206, 1984.

[34] J. Zobel and A. Moffat, "Inverted files versus signature files for text indexing," *ACM Trans. Database Syst.*, vol. 23, no. 4, pp. 453–490, 1998.

[35] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 2161–2168.

[36] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order preserving encryption for numeric data," in *Proc. SIGMOD*, 2004, pp. 563–574.

[37] A. Broder, M. Charikar, A. Frieze, and M. Mitzenmacher, "Min-wise independent permutations," in *Proc. 30th ACM Symp. Theory Comput.*, 1998, pp. 327–336.

[38] [Online]. Available: http://wang.ist.psu.edu/∼jwang/test1.tar

[39] S. Jeong, C. Won, and R. Gray, "Image retrieval using color histograms generated by Gauss mixture vector quantization," *Comput. Vis. Image Understand.*, vol. 94, nos. 1–3, pp. 44–66, 2004.

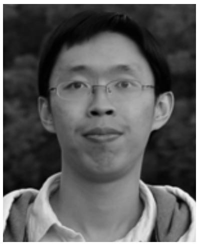
[40] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 394–410, Mar. 2007.

[41] R. C. Gonzales and R. E. Woods, *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2007.

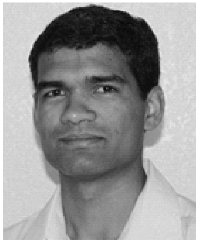
[42] C. Dwork, "Differential privacy," in *Proc. 33rd Int. Colloq. Autom., Lang., Program. (ICALP)*, 2006, pp. 1–12.

[43] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(Leveled) fully homomorphic encryption without bootstrapping," in *Proc. 3rd Innov. Theoretical Comput. Sci. Conf.*, 2012, pp. 309–325.

[44] Z. Brakerski, "Fully homomorphic encryption without modulus switching from classical GapSVP," in *Advances in Cryptology Crypto*. New York, NY, USA: Springer-Verlag, 2012, pp. 868–886.



WENJUN LU received the B.S.E. (Hons.) degree in Department of Automation from Tsinghua University, Beijing, China, and the Ph.D. degree in Department of Electrical and Computer Engineering from the University of Maryland, College Park, in 2006 and 2011, respectively. He was a Research Intern with the National Library of Medicine of the U.S. National Institute of Health in summer 2007, a Research Intern with Technicolor in summer 2009, and a Research Intern with Google in summer 2010. Since 2011, he has been with Google Inc., Mountain View. His R&D interests are in the areas of information security, multimedia forensics, and image processing.



AVINASH L. VARNA (S'06–M'12) received the B.Tech. and Ph.D. degrees in electrical engineering from IIT Madras, and the University of Maryland, College Park, in 2005 and 2011, respectively. He is currently with Intel, Chandler, AZ, USA. His current research interests include digital rights management, security of embedded systems, information forensics, and multimedia security. Dr. Varna received a Silver Medal from the International Chemistry Olympiad in 2001. He was awarded the Distinguished Dissertation Fellowship by the Department of Electrical and Computer Engineering and the Litton Fellowship for academic excellence by the University of Maryland. His co-authored paper with the 2011 ACM Multimedia Conference received a Best Student Paper Award. He was awarded two Divisional Recognition Awards in 2012 and an Operational Excellence Award in 2013 by Intel.

MIN WU (S'95–M'01–SM'06–F'11) received the B.E. degree in electrical engineering and the B.A. degree in economics from Tsinghua University, Beijing, China (both with the highest honors), in 1996, and the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, USA, in 2001. Since 2001, she has been with the University of Maryland, College Park, MD, USA, where she is currently a Professor and a University Distinguished Scholar-Teacher. She leads the Media and Security Team with the University of Maryland, with main research interests on information security and forensics, and multimedia signal processing. She has co-authored two books and holds eight U.S. patents on multimedia security and communications. She is a co-recipient of the two Best Paper Awards from the IEEE Signal Processing Society and EURASIP. She received the NSF CAREER Award in 2002, the TR100 Young Innovator Award from the MIT Technology Review Magazine in 2004, the ONR Young Investigator Award in 2005, the Computer World 40 Under 40 IT Innovator Award in 2007, the IEEE Mac Van Valkenburg Early Career Teaching Award in 2009, and the University of Maryland Invention of the Year Award in 2012. She has served as the Vice Presidentof the Finance of the IEEE Signal Processing Society from 2010 to 2012, and the Chair of the IEEE Technical Committee on Information Forensics and Security from 2012 to 2013.