Hindawi Publishing Corporation
EURASIP Journal on Information Security
Volume 2009, Article ID 859859, 16 pages
doi:10.1155/2009/859859

Research Article

An Extended Image Hashing Concept: Content-Based Fingerprinting Using FJLT

Xudong Lv and Z. Jane Wang

Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada V6T 1Z4

Correspondence should be addressed to Xudong Lv, xudongl@ece.ubc.ca

Received 27 March 2009; Revised 25 June 2009; Accepted 23 September 2009

Recommended by Patrick Bas

Dimension reduction techniques, such as singular value decomposition (SVD) and nonnegative matrix factorization (NMF), have been successfully applied in image hashing by retaining the essential features of the original image matrix. However, a concern of great importance in image hashing is that no single solution is optimal and robust against all types of attacks. The contribution of this paper is threefold. First, we introduce a recently proposed dimension reduction technique, referred as Fast Johnson-Lindenstrauss Transform (FJLT), and propose the use of FJLT for image hashing. FJLT shares the low distortion characteristics of a random projection, but requires much lower computational complexity. Secondly, we incorporate Fourier-Mellin transform into FJLT hashing to improve its performance under rotation attacks. Thirdly, we propose a new concept, namely, content-based fingerprint, as an extension of image hashing by combining different hashes. Such a combined approach is capable of tackling all types of attacks and thus can yield a better overall performance in multimedia identification. To demonstrate the superior performance of the proposed schemes, receiver operating characteristics analysis over a large image database and a large class of distortions is performed and compared with the state-of-the-art image hashing using NMF.

1. Introduction

Digital media has profoundly changed our daily life during the past decades. However, the massive proliferation and extensive use of media data arising from its easy-to-copy nature also pose new challenges to effectively manage such abundance of data (e.g., fast media searching, indexing) and protection of intellectual property of multimedia data. Among the various techniques proposed to address these challenges, image hashing has been proven to be an efficient tool because of its robustness and security.

An image hash is a compact and exclusive feature descriptor for a specific image. Robustness and security are its two desired properties [1, 2]. Different from traditional hash, image hash does not suffer from the sensitivity to minor degradations of original data because of its perceptual robustness. Such a property requires two images that are perceptually identical in human visual system (HVS) and are mapped to similar hash values. Obviously, the more robust a hash is, the less sensitive it is to large distortions upon the original images, which in turn inevitably incurs another problem that distinct images may be misclassified to the same group. Hence, tradeoff between robustness and anticollision of distinct images is of great concern. Additionally, by incorporating the pseudorandomization techniques, a hash is hardly obtained by unauthorized adversaries without the secret key. Therefore, the unpredictability encrypts the image hash and guarantees its security against illegal access.

Behaving as a secure tag for image data, image hashing facilitates significant developments in many areas such as image and video watermarking [3]. It is worth mentioning that different applications may impose different requirements in a hashing design. For the purpose of image authentication, it is required that minor unmalicious modifications which do not alter the content of the data should preserve the authenticity of the data [4, 5]. The robustness of image hash assures its capability to authenticate the content by ignoring the effect of minor unmalicious modifications on the original

data. For the management of large image databases [6], image hashing allows efficient media indexing, identification, and retrieval by avoiding exhaustively searching through all the entries, thus reducing computational complexity of similarity measurements. Moreover, specific hashing designed based on some specific features of image data, such as color, edges, and other information, obviously contributes to the content-based image retrieval (CBIR) system [7] at the semantic level. In this paper, we are particularly interested in image identification and explore the application of image hashing in this direction.

Although there exist various frameworks to design robust and secure hashes [8–10], a hashing scheme generally consists of two aspects: one is feature extraction and the other is pseudorandomization technique. Most hashing schemes combine both aspects to generate an intermediate hash as the first step and then incorporate a compression operation in postprocessing to generate the final hash [1, 10, 11]. Obviously, the robustness and security, two principal properties of hashing, lie in the first step. In order to resist routine unmalicious degradations (e.g., noising, compression) and other malicious attacks (e.g., cropping, rotation), the more invariant features are extracted, the more robust a hash scheme is. However, using features directly makes the scheme susceptible to forgery attacks. Therefore, pseudorandomization techniques should be employed in the hash schemes to assure the security.

Aiming at resisting both routine unmalicious degradations and malicious attacks, various approaches have been proposed in literatures for constructing image hashes, although there is no universallyoptimal hashing approach that is robust against all types of attacks. For example, Radon Soft Hash algorithm (RASH) [12] shows robustness against geometric transformation and some image processing attacks using Radon transform and principle component analysis (PCA). Swaminathan's hashing scheme [8] incorporates pseudorandomization into Fourier-Mellin transform to achieve better robustness to geometric operations. However, it suffers from some classical signal processing operations such as noising. It was also proposed in [9] to generate the hash by detecting invariant feature points, though the expensive searching and removal of feature points by malicious attacks such as cropping and blurring limit its performance in practice. Other content-preserving features based on statistics [1] and spectrum information [2, 13] have also contributed to the development of image hashing and enlightened some novel directions.

Recently, several image hashing schemes based on dimension reduction have been developed and reported to outperform previous techniques. For instance, using low-rank matrix approximations obtained via singular value decomposition (SVD) for hashing was explored in [14]. Its robustness against geometric attacks motivated other solutions in this direction. Monga introduced another dimension reduction technique, called nonnegative matrix factorization (NMF) [15], into their new hashing algorithm [16]. The major benefit of NMF hashing is the structure of the basis resulting from its nonnegative constraints, which lead to a parts-based representation. In contrast to the global representation obtained by SVD, the non-negativity constraints result in a basis of interesting local features [17]. Based on the results in [16], the NMF hashing possesses excellent robustness under a large class of perceptually insignificant attacks, while it significantly reduces misclassification for perceptually distinct images. Note that, for simplicity, we sometimes refer the NMF-NMF-SQ hashing scheme, which was shown to provide the best performance among NMF-based hashing schemes investigated in [16], simply as NMF hashing in this paper.

Inspired by the potential of dimension reduction techniques for image hashing, we introduced Fast Johnson-Lindenstrauss transform (FJLT), a dimension reduction technique recently proposed in [18], into our new robust and secure image hashing algorithm [19]. FJLT shares the low-distortion characteristics of a random projection process but requires a lower computational complexity. It is also more suitable for practical implementation because of its high computational efficiency and security due to the random projection. Since we mainly focus on invariant feature extraction and are interested in image identification applications, the FJLT hashing seems promising because of its robustness to a large class of minor degradations and malicious attacks. Considering the fact that NMF hashing was reported to significantly outperform other existing hashing approaches [16], we use it as the comparison base for the proposed FJLT hashing. Our preliminary experimental results in [19] showed that FJLT hashing provides competitive or even better identification performance under various attacks such as additive noise, blurring, and JPEG compression. Moreover, its lower computational cost also makes it attractive.

However, geometric attacks such as rotation could essentially tamper the original images and thus prevent the accurate identification if we apply the hashing algorithms directly on the manipulated image. Even for the FJLT hashing, it still suffers from the rotation attacks with low identification accuracy. To address this concern, motivated by the work [8, 20], we plan to apply the Fourier-Mellin transform (FMT) on the original images first to make them invariant to geometric transform. Our later experimental results show that, under rotation attacks, the FJLT hashing combined with the proposed FMT preprocessing yields a better identification performance than that of the direct FJLT hashing.

Considering that a specific feature descriptor may be more robust against certain types of attacks, it is desirable to take advantage of different features together to enhance the overall robustness of hashing. Therefore we further propose an extended concept, namely, content-based fingerprinting, to represent a combined, superior hashing approach based on different robust feature descriptors. Similar to the idea of having the unique fingerprint for each human being, we aim at combining invariant characteristics of each feature to construct an exclusive (unique) identifier for each image. Under the framework of content-based fingerprinting, the inputs to the hashing algorithms are not restricted to the original images only, but can also be extendable to include various robust features extracted from the images, such

as color, texture, and shape. An efficient joint decision scheme is important for such a combinational framework and significantly affects the identification accuracy. Our experimental results demonstrate that the content-based fingerprinting using a simple joint decision scheme can provide a better performance than the traditional one-fold hashing approach. More sophisticated joint decision-making schemes are worth further being investigated in the future.

The rest of this paper is organized as follows. We first introduce the background and theoretic details about FJLT in Section 2. We then describe the proposed hashing algorithm based on random sampling and FJLT in Section 3. In Section 4, we propose the RI-FJLT hashing by combining the Fourier-Mellin transform and FJLT hashing to achieve better geometric robustness. To combine the advantages of both FJLT and RI-FJLT hashing algorithms, a general framework and experimental results of content-based fingerprinting using FJLT hashing for multimedia identification are presented in Section 5. The analytical and experimental results are exhibited in Section 6 to demonstrate the superior performance of the proposed schemes. The conclusion and suggestions for future work are given in Section 7.

2. Theoretical Background

Based on the literature review in Section 1, the current task of image hashing is to extract more robust features to guarantee the identification accuracy under manifold manipulations (e.g., noising, blurring, compression, etc.) and incorporate the pseudorandomization techniques into the feature extraction to enhance the security of the hash generation. According to the information theory [21], if we consider the original image as a source signal, similar to a transmission channel in communication, the feature extraction process will make the loss of information inevitable. Therefore, how to efficiently extract the robust features as lossless as possible is a key issue that the hashing algorithms such as SVD [14], NMF [16], and our FJLT hashing want to tackle.

2.1. Fast Johnson-Lindenstrauss Transform. The Johnson-Lindenstrauss (JL) theorem has found numerous applications, including searching for approximate nearest neighbors (ANNs) [18] and dimension reduction in database, and so forth, by the JL lemma [22], n points in Euclidean space can be projected from the original d dimensions down to lower $k = \mathcal{O}(\varepsilon^{-2}\log n)$ dimensions while just incurring a distortion of at most $\pm\varepsilon$ in their pairwise distances, where $0 < \varepsilon < 1$. Based on the JL theorem, Alion and Chazelle [18] proposed a new low-distortion embedding of l_p^d into $l_p^k (p = 1 \text{ or } 2)$, called Fast Johnson-Lindenstrauss transform (FJLT). FJLT is based on preconditioning of a sparse projection matrix with a randomized Fourier transform. Note that we will only consider the l_2 case ($p = 2$) because our hash is measured by the l_2 norm. For the l_1 case, interested readers please refer to [18].

Briefly speaking, FJLT is a random embedding, denoted as $\Phi = \text{FJLT}(n, d, \varepsilon)$, that can be obtained as a product of three real-valued matrices:

$$\Phi = P \cdot H \cdot D, \tag{1}$$

where the matrices P and D are random and H is deterministic [18].

(i) P is a k-by-d matrix whose elements P_{ij} are drawn independently according to the following distribution, where $\mathcal{N}(0, q^{-1})$ means a Normal distribution with zero-mean and variance q^{-1},

$$\begin{aligned} P_{ij} &\sim \mathcal{N}(0, q^{-1}) \quad \text{with probability } q, \\ P_{ij} &= 0 \qquad\qquad\quad \text{with probability } (1-q), \end{aligned} \tag{2}$$

where

$$q = \min\left\{\frac{c\log^2 n}{d}, 1\right\}, \tag{3}$$

for a large enough constant c.

(ii) H is a d-by-d normalized Hadamard matrix with the elements as

$$H_{ij} = d^{-1/2}(-1)^{\langle i-1, j-1\rangle}, \tag{4}$$

where $\langle i, j\rangle$ is the dot-product of the m-bit vectors of i, j expressed in binary.

(iii) D is a d-by-d diagonal matrix, where each diagonal element D_{ii} is drawn independently from $\{-1, 1\}$ with probability 0.5.

Therefore, $\Phi = \text{FJLT}(n, d, \varepsilon)$ is a k-by-d matrix, where d is the original dimension number of the data and k is the lower dimension number, which is set to be $c'\varepsilon^{-2}\log n$. Here, n is the number of data points, ε is the distortion rate, and c' is a constant. Given any data point X from a d-dimension space, it is intuitively mapped to the data point X' at a lower k-dimension space by the FJLT and the distortion of their pairwise distances could be illustrated by Johnson-Lindenstrauss lemma [18].

2.2. The Fast Johnson-Lindenstrauss Lemma

Lemma 1. *Fix any set X of n vectors in $\mathbb{R}^d$, $0 < \varepsilon < 1$, and let $\Phi = FJLT(n, d, \varepsilon)$. With probability at least 2/3, the following two events occur.*

(1) *For all $x \in X$,*

$$(1-\varepsilon)k\|x\|_2 \le \|\Phi x\|_2 \le (1+\varepsilon)k\|x\|_2. \tag{5}$$

(2) *The mapping $\Phi : \mathbb{R}^d \to \mathbb{R}^k$ requires*

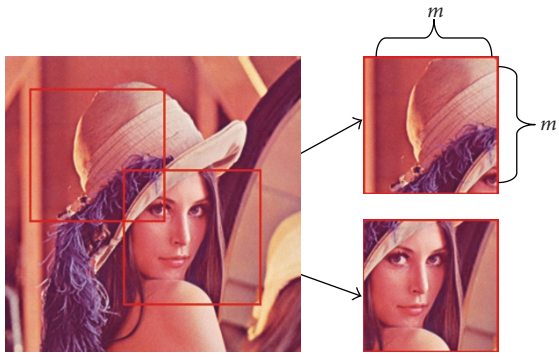


Figure 1: An example of random sampling. The subimages selected by random sampling with size $m \times m$.

$$\mathcal{O}\left(d \log d + \min\left(d\varepsilon^{-2} \log n, \varepsilon^{-2} \log^3 n\right)\right) \quad (6)$$

operations.

Proofs of the previous theorems can be found in [18]. Note that the probability of being successful (at least 2/3) arises from the random projection and could be amplified to $(1 - \delta)$ for any $\delta > 0$, if we repeat the construction $\mathcal{O}(\log(1/\delta))$ times [18]. Since the random projection is actually a pseudorandom process determined by a secret key in our case, most of the keys (at least 2/3) are satisfied with the distortion bound described in FJLT lemma and could be used in our hashing algorithm. Hence, the FJLT will make our scheme widely applicable for most of the keys and suitable to be applied in practice.

3. Image Hashing via FJLT

Motivated by the hashing approaches based on SVD [14] and NMF [16], we believe that dimension reduction is a significantly important way to capture the essential features that are invariant under many image processing attacks. For FJLT, three benefits facilitate its application in hashing. First, FJLT is a random projection, enhancing the security of the hashing scheme. Second, FJLT's low distortion guarantees its robustness to most routine degradations and malicious attacks. The last one is its low computation cost when implemented in practice. Hence, we propose to use FJLT for our new hashing algorithm. Given an image, the proposed hashing scheme consists of three steps: random sampling, dimension reduction by FJLT, and ordered random weighting. Due to our purpose, we are only interested in feature extraction and randomization. The hash generated by FJLT is just an intermediate hash. For readers who are interested in generating the final hash by compression step, as in the frameworks[8, 9], they are suggested to refer [1, 11] for details.

3.1. Random Sampling. The idea of selecting a few subimages as original feature by random sampling, as shown in Figure 1, is not novel [14, 16]. However, in our approach, we treat each subimage as a point in a high-dimensional space rather than a two-dimensional matrix as in SVD hashing [14] and NMF hashing [16]. For instance, the subimage in Figure 1, which is a m-by-m patch, is actually a point in the m^2-dimensional space in our case, where we focus on gray images.

Given an original color image, we first convert it to a gray image and pseudorandomly select N subimages depending on the secret key and get $\{R_i\}$, for $1 \le i \le N$. Each R_i is a vector with length m^2 by concatenating the columns of the corresponding subimage. Then we construct our original feature as.

$$\text{Feature} = \{R_1, R_2, \ldots, R_N\}, \quad \text{with size } m^2 \times N. \quad (7)$$

The advantage of forming such a feature is that we can capture the global information in the *Feature* matrix and local information in each component R_i. Even if we lose some portions of the original image under geometric attacks such as cropping, it will only affect one or a few components in our *Feature* matrix and have no significant influence on the global information. However, the Feature matrix with the high dimension (e.g., m^2, when $m = 64$) is too large to store and match, which motivates us to employ dimension reduction techniques.

3.2. Dimension Reduction by FJLT. Based on the theorems in Section 2, FJLT is able to capture the essential features of the original data in a lower-dimensional space with minor distortion, if the factor ε is close to 0. Recall the construction $\Phi = \text{FJLT}(n, d, \varepsilon)$, our work is to map the Feature matrix from a high-dimensional space to a lower-dimensional space with minor distortion. We first get the three real-valued matrices P, H, and D in our case, which is $\Phi = \text{FJLT}(N, m^2, \varepsilon)$, where H is deterministic but P and D are pseudorandomly dependent on the secret key. The lower dimension k is set to be $c'\varepsilon^{-2} \log N$ and c' is a constant. Then we can get our intermediate hash (IH) as

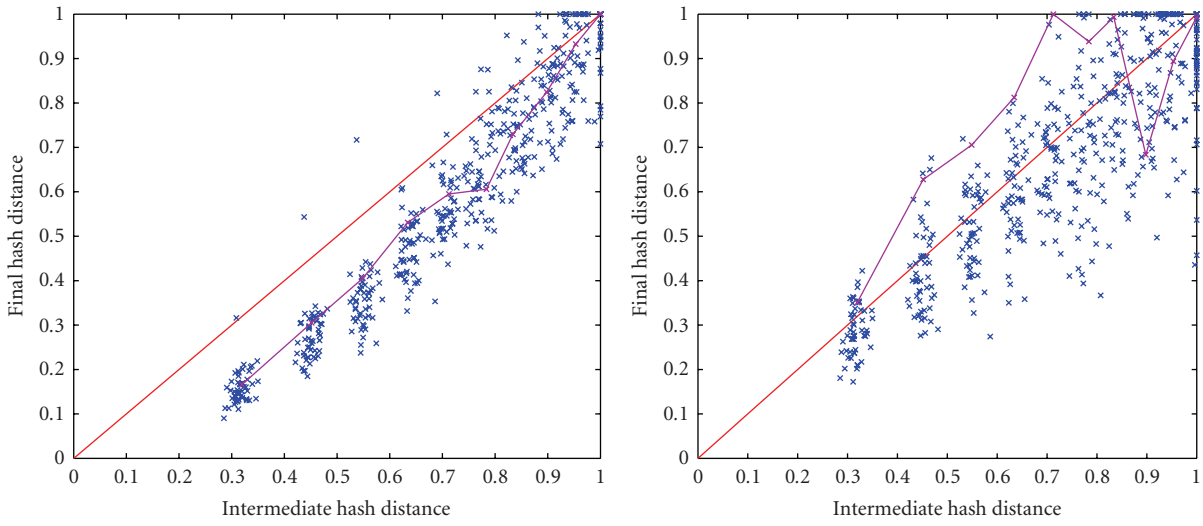
$$IH = \Phi(\text{Feature}) = P \cdot H \cdot D \cdot \text{Feature}, \quad \text{with size } k \times N. \quad (8)$$

Here, the advantage of FJLT is that we can determine the lower dimension k by adjusting the number of data points, which is the number of image blocks by random sampling in our case, and the distortion rate ε. This provides us with a good chance to get a better identification performance. However, the smaller ε is, the larger k is. Hence we need to make a tradeoff between ε and k in a real implementation.

3.3. Ordered Random Weighting. Although the original feature set has been mapped to a lower-dimensional space with a small distortion, the size of intermediate hash can still be large. For instance, if we set $N = 20, \varepsilon = 0.1$, and $c' = 2$, the size of IH will be 600-by-20. To address this issue, similar to the NMF-NMF-SQ hashing in [16], we can introduce the pseudorandom weight vectors $\{w_i\}_{i=1}^{N}$ with $w_i \in R^k$ drawn from the uniform distribution $U(x \mid 0, 1)$ by the secret key, and we can calculate the final secure hash as

$$\text{Hash} = \{\langle IH_1, w_1\rangle, \langle IH_2, w_2\rangle, \ldots, \langle IH_N, w_N\rangle\}, \quad (9)$$

where IH_i is the ith column in IH, and $\langle IH_i, w_i\rangle$ is the inner product of the vectors IH_i and w_i. Hence, the final hash is



(a) Ordered (b) Unordered

FIGURE 2: An example of the correlations between the final hash distance and the intermediate hash distance based on 50 images under Salt and Pepper noise attacks (with variance level: $0 \sim 0.1$) when employing ordered random weighting and unordered random weighting.

obtained as a vector with length N for each image, which is compact and secure. However, the weight vector w_i drawn from $U(x \mid 0, 1)$ could diminish the distance between the hash components IH_i and IH_i' from two images and degrade the identification accuracy later. Here we describe a simple example to explain this effect. Suppose we have two vectors $A = \{10, 1\}$ and $A' = \{1, 1\}$, the Euclidean distance is 9. In the first case, if we assign the weight vector $w = \{0.1, 0.9\}$ to A and A', after the inner product (9), the hash values of A and A' will be 1.9 and 1, respectively. Obviously, the distance between A and A' is significantly shortened. However, if we assign the weight $w = \{0.9, 0.1\}$ to A and A' in the second case, after the inner product (9), the hash values of A and A' will be 9.1 and 1, respectively. The distance between A and A' is still 8.1. We would like to maintain the distinction of two vectors and avoid the effect of an inappropriate weight vector as the first case.

To maintain this distance-preserving property, a possible simple solution, referred as ordered random weighting, is to sort the elements of IH_i and w_i in a descending order before the inner product (9) and make sure that a larger weight value will be assigned to a larger component. In this way, the perceptual quality of the hash vector is retained by minimizing the influence of the weights. To demonstrate the effects of ordering, we investigate the correlation between the intermediate hash distances and the final hash distances when employing the unordered random weighting and ordered random weighting. Intuitively, for both the intermediate hash and the final hash, the distance between the hash generated from the original image (without distortion) and the hash from its distorted copy should increase when the attack/distortion is more severe. One example is illustrated in Figure 2, where we investigate 50 nature images and their 10 distorted copies with Salt and Pepper noise attacks (with variance level: $0 \sim 0.1$) from our database described in Section 5.1. We observe that the normalized intermediate hash distance and the final hash distance are highly correlated when using ordered random weighting, as shown in Figure 2(a), while the distances are much less correlated under unordered random weighting, as shown in Figure 2(b). In Figure 2, one example of distance correlation based on one of the 50 nature images is indicated by the solid purple lines, where a monotonically increasing relationship between the distances is clearly noticed when using ordered random weighting. Figure 2 suggests that the ordered random weighting in the proposed hashing approach maintains the property of low distortion in pairwise distances of the FJLT dimension reduction technique.

Furthermore, we also investigate the effect of ordering on the identification performance by comparing the ordered and unordered random weighting approaches. One illustrative example is shown in Figure 3, where the distances between different hashes are reported. Among 50 original images, we randomly pick out one as the target image and use its distorted copies as the query images to be identified. To compare the normalized Euclidean distances between the final hashes of the query images and the original 50 images, the final hash distances between the target image and its distorted copies are indicated by red squares, and others are marked by blue crosses. For the Salt and Pepper noise attacks (with variance level: $0 \sim 0.1$) as shown in Figures 3(a) and 3(b), we can see that, when using both ordered random weighting and unordered random weighting, the query images could be easily identified as the true target image based on the identification process described in Section 3.4.1. It is also clear that the ordered random weighting approach should provide a better identification performance statistically since the distance groups are better separated. For the Gaussian blurring attacks (with filter size: $3 \sim 21$) as

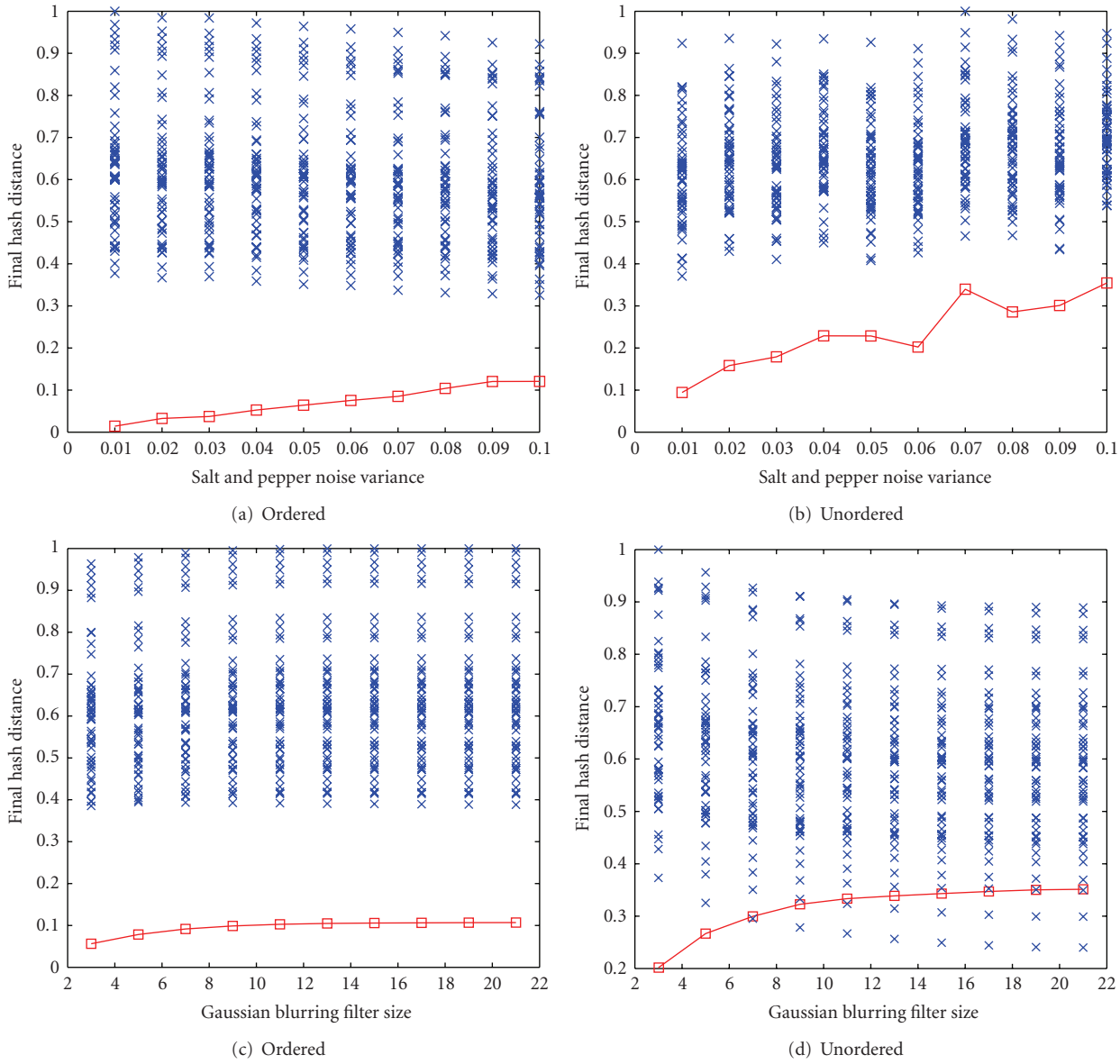


FIGURE 3: Illustrative examples to demonstrate the effect of ordering on the identification performance. The final hash distances between the query images and the original 50 images are shown for comparing the ordered random weighting and the unordered random weighting approaches. (a) and (b) The query images are under Salt and Pepper noise attacks. (c) and (d) The query images are under Gaussian blurring attacks.

shown in Figures 3(c) and 3(d), it is clear that the correct classification/identification can only be achieved by using the ordered random weighting. Based on the two examples illustrated in Figure 3 and the tests on other attacks described in Section 6.1, we notice that the identification performance under the blurring attacks is significantly improved using the ordered random weighting when compared with the unordered approach. The improvement is less significant under noise and other attacks. In summary, we observe that ordered random weighting maintains better the distance-preserving property of FJLT compared with the unordered random weighting and thus yields a better identification performance.

3.4. Identification and Evaluation

3.4.1. Identification Process. Let $S = \{s_i\}_{i=1}^{N}$ be the set of original images in the tested database and define a space $H(S) = \{H(s_i)\}_{i=1}^{N}$ as the set of corresponding hash vectors. We use Euclidean distance as the performance metric to measure the discriminating capability between two hash vectors, defined as

$$\text{Distance} = \|H(s_1) - H(s_2)\|_2 = \sqrt{\sum_{i=1}^{n} (h_i(s_1) - h_i(s_2))^2}, \tag{10}$$

where $H(s_i) = \{h_1(s_i), h_2(s_i), \ldots, h_n(s_i)\}$ means the corresponding hash vector with length n of the image s_i. Given a tested image D, we first calculate its hash $H(D)$ and then obtain its distances to each original image in the hash space $H(S)$. Intuitively, the query image D is identified as the $\hat{i}$th original images which yields the minimum corresponding distance, expressed as

$$\hat{i} = \arg\min_i \{\|H(D) - H(s_i)\|_2\}, \quad i = 1, \ldots, N. \tag{11}$$

The simple identification process described above can be considered as a special case of the K-nearest-neighbor classification approach with $K = 1$. Here K is set as 1 since we only have one copy of each original image in the current database. For a more general case, if we have K multiple copies of each original image with no distortion or with only slight distortions, we could adopt the K-nearest neighbor (KNN) algorithm for image identification in our problem.

3.4.2. Receiver Operating Characteristics Analysis. Except investigating identification accuracy, we also study the receiver operating characteristics (ROC) curve [23] to visualize the performance of different hashing approaches, including NMF-NMF-SQ hashing, FJLT hashing, and Content-based fingerprinting proposed later. The ROC curve depicts the relative tradeoffs between benefits and cost of the identification and is an effective way to compare the performances of different hashing approaches.

To obtain ROC curves to analyze the hashing algorithms, we may define the probability of true identification $P_T(\xi)$ and probability of false alarm $P_F(\xi)$ as

$$\begin{aligned} P_T(\xi) &= Pr(\|H(I) - H(I_M)\|_2 < \xi), \\ P_F(\xi) &= Pr(\|H(I) - H(I'_M)\|_2 < \xi), \end{aligned} \tag{12}$$

where ξ is the identification threshold. The images I and I' are two distinct original images and the images I_M and I'_M are manipulated versions of the image I and I', respectively. Ideally, we hope that the hashes of the original image I and its manipulated version I_M should be similar and thus be identified accurately, while the distinct images I and I'_M should have different hashes. In other words, given a certain threshold ξ, an efficient hashing should provide a higher $P_T(\xi)$ with a lower $P_F(\xi)$ simultaneously. Consequently, when we obtain all the distances between manipulated images and original images, we could generate a ROC curve by sweeping the threshold ξ from the minimum value to the maximum value, and further compare the performances of different hashing approaches.

4. Rotation Invariant FJLT Hashing

Although the Fast Johnson-Lindenstrauss transform has been shown to be successful in the hashing in our previous preliminary work [19], the FJLT hashing can still be vulnerable to rotation attacks. Based on the hashing scheme described in Section 3, random sampling can be an effective approach to reduce the distortion introduced by cropping, and scaling attack can be efficiently tackled by upsampling and downsampling in the preprocessing. However, to successfully handle the rotation attacks, we need to introduce other geometrically invariant transform to improve the performance of the original FJLT hashing.

4.1. Fourier-Mellin Transform. The Fourier-Mellin transform (FMT) is a useful mathematical tool for image recognition and registration, because its resulting spectrum is invariant to rotation, translation, and scaling [8, 20]. Let f denote a gray-level image defined over a compact set of $\mathbb{R}^2$, the standard FMT of f in polar coordinates (log-polar coordinates) is given by

$$M_f(k, v) = \frac{1}{2\pi} \int_0^{2\pi} \int_0^{\infty} f(r, \theta) r^{-iv} e^{-ik\theta} \mathrm{d}\theta \frac{\mathrm{d}r}{r}. \tag{13}$$

If we make $r = e^{\gamma}, \mathrm{d}r = e^{\gamma}\mathrm{d}\gamma$, (13) is clearly a Fourier transform like

$$M_f(k, v) = \frac{1}{2\pi} \int_0^{2\pi} \int_{-\infty}^{\infty} f(e^{\gamma}, \theta) e^{-iv\gamma} e^{-ik\theta} \mathrm{d}\gamma \mathrm{d}\theta. \tag{14}$$

Therefore, the FMT could be divided into three steps, which result in the invariance to geometric attacks.

(i) *Fourier Transform*. It converts the translation of original image in spatial domain into the offset of angle in spectrum domain. The magnitude is translation invariant.

(ii) *Cartesian to Log-Polar Coordinates*. It converts the scaling and rotation in Cartesian coordinates into the vertical and horizontal offsets in Log-Polar Coordinates.

(iii) *Mellin Transform*. It is another Fourier transform in Log-Polar coordinates and converts the vertical and horizontal offsets into the offsets of angles in spectrum domain. The final magnitude is invariant to translation, rotation, and scaling.

However, the inherent drawback of the Fourier transform makes FMT only robust to geometric transform, but vulnerable to many other classical signal processing distortions such as cropping and noising. As we know, when converting an image into the spectrum domain by 2D Fourier transform, each coefficient is contributed by all the pixels of the image. It means that the Fourier coefficients are dependent on the global information of the image in the spatial domain. Therefore, the features extracted by Fourier-Mellin transform are sensitive to certain attacks such as noising and cropping, because the global information is no longer maintained. To overcome this problem, we have modified the FMT implementation in our proposed rotation-invariant FJLT (RI-FJLT) hashing.

4.2. RI-FJLT Hashing. The invariance of FMT to geometric attacks such as rotation and scaling has been widely applied in image hashing [3, 8] and watermarking [20, 24]. It also motivates us to address the deficiency of FJLT hashing by

incorporating FMT. Here, we propose the rotation-invariant FJLT hashing by introducing FMT into the FJLT hashing. Specially, the proposed rotation-invariant FJLT hashing (RI-FJLT) consists of three steps.

Step 1. Converting the image into the Log-Polar coordinates

$$I(x, y) \longrightarrow G(\log\rho, \theta), \tag{15}$$

where x and y are Cartesian coordinates and ρ and θ are Log-Polar coordinates. Any rotation and scaling will be considered as vertical and horizontal offsets in Log-Polar coordinates. An example is given in Figure 4.

Step 2. Applying Mellin transform (Fourier transform under Log-Polar coordinates) to the converted image and return the magnitude feature image.

Step 3. Applying FJLT hashing in Section 3 to the magnitude feature image derived in Step 2.

For the conversion in Step 1, since the pixels in Cartesian coordinates are not able to be one-to-one mapped to pixels in the Log-Polar coordinates space, some value interpolation approaches are needed. We have investigated three different interpolation approaches for the proposed RI-FJLT hashing, including nearest neighbor, bilinear and bicubic interpolations, and found that the bilinear is superior to others. Therefore we only report the results under bilinear interpolation here. Note that we abandon the first step of FMT in RI-FJLT hashing, because we only focus on rotation attacks (other translations are considered as cropping) and it is helpful to reduce the influence of noising attacks by removing the Fourier transform step. The performance will be illustrated in Section 6. However, since Step 2 can inevitably be affected by attacks such as noising, some preprocessing such as median filtering can help improve the final identification performance.

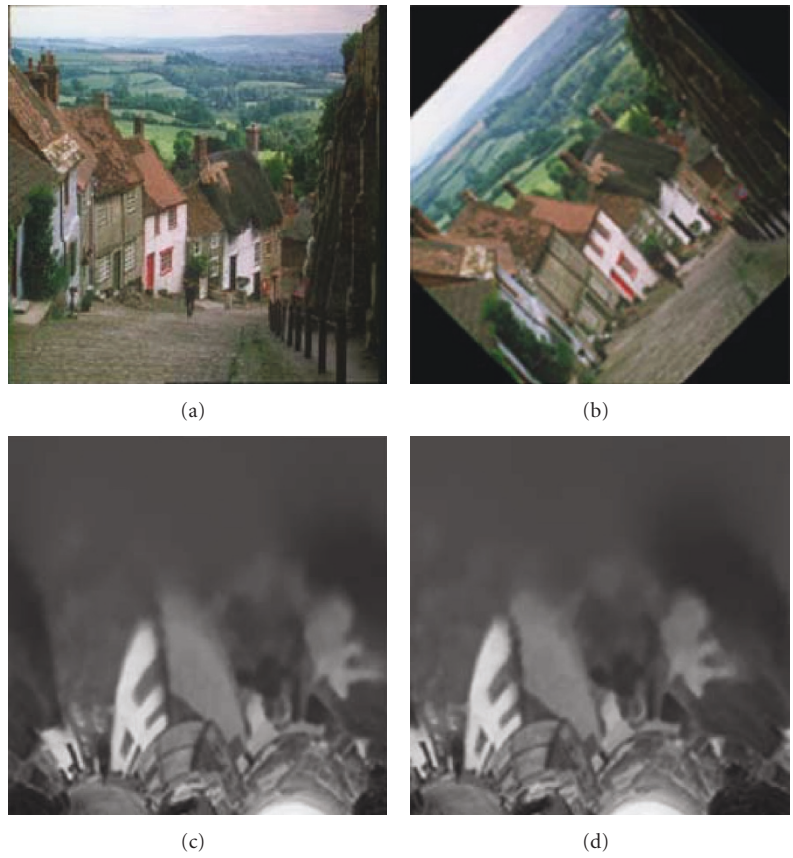
5. Content-Based Fingerprinting

5.1. Concept and Framework. Considering that certain features can be more robust against certain attacks, to take advantage of different features, we plan to propose a new content-based fingerprinting concept. This concept combines benefits of conventional content-based indexing (used to extract discriminative content features) and multimedia hashing. Here we define content-based image fingerprinting as a combination of multiple robust feature descriptors and secure hashing algorithms. Similar to the concept of image hash, it is a digital signature based on the significant content of image itself and represents a compact and discriminative description for the corresponding image. Therefore, it has a wide range of applications in practice such as integrity verification, watermarking, content-based indexing, identification, and retrieval. The framework is illustrated in Figure 5.

Specially, each vertical arrow in Figure 5 represents an independent hashing generation procedure, which consists of robust feature extraction and intermediate hash generation proposed by [8, 10]. Because it is the combination of various hash descriptors, the content-based fingerprinting can be considered as an extension and evolution of image hashing and thus offers much more freedom to accommodate different robust features (color, shape, texture, salient points, etc., [7]) and design efficient hashing algorithms to successfully against different types of attacks and distortions. Similar to the idea of finding one-to-one relationships between the fingerprints and an individual human being, the goal of content-based fingerprinting is to generate an exclusive digital signature, which is able to uniquely identify the corresponding media data no matter which content-preserving manipulation or attack is taken on.

Compared with the traditional image hashing concept, the superiority of content-based fingerprint concept lies in its potential high discriminating capability, better robustness, and multilayer security arising from the combination of various robust feature descriptors and a joint decision-making process. Same as in any information fusion processes, theoretically the discrimination capability of the content-based fingerprinting with effective joint decision-making scheme should outperform a single image hashing. Since the content-based fingerprint consists of several hash vectors, which are generated based on various robust features and different secret keys, it is argued that the framework of content-based fingerprinting results in a better robustness and multilayer security when an efficient joint decision-making is available. However, combining multiple image hashes approaches requires additional computation cost for the generation of content-based fingerprinting. The tradeoff between computation cost and performance is a concern with great importance in practice.

5.2. A Simple Content-Based Fingerprinting Approach. From the experimental results in Section 6, we note that FJLT hashing is robust to most types of the tested distortions and attacks except for rotation attacks and that RI-FJLT hashing provides a significantly better performance for rotation attacks at the cost of the degraded performances under other types of attacks. Recall an important fact that it is relatively easy to find a robust feature to resist one specific type of distortion; however it is very difficult, if not impossible, to find a feature which is uniformly robust to against all types of distortions and attacks. Any desire to generate an exclusive signature for the image by a single image hashing approach is infeasible. Here we plan to demonstrate the advantages of the concept of content-based fingerprinting by combining the proposed FJLT hashing and RI-FJLT hashing. The major components of the content-based fingerprinting framework include hash generations and the joint decision-making process which should take advantage of the combinations of the hashes to achieve a superior identification decision-making. Regarding the joint decision-making, there are many approaches in machine learning [25] that can be useful. Here we only present a simple decision-making



(a) (b) (c) (d)

Figure 4: An example of conversion from Cartesian coordinates to Log-Polar coordinates. (a) Original Goldhill. (b) Goldhill rotated by 45°. (c) Original Goldhill in Log-Polar coordinates. (d) Rotated Goldhill in Log-Polar coordinates.

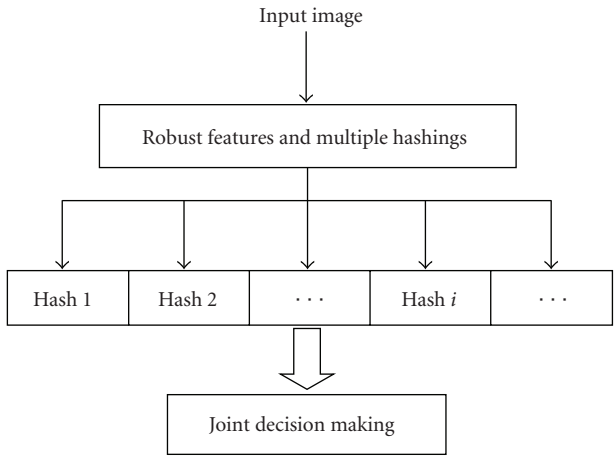


Figure 5: The conceptual framework of the content-based fingerprinting.

process in rank level [26] to demonstrate the superiority of content-based fingerprinting.

Given an image d with certain distortion, we, respectively, generate the hash vectors H_f^d and H_r^d by FJLT and RI-FJLT hashing. Suppose that the hash values of original images s are H_f^s and H_r^s generated by FJLT and RI-FJLT hashing, respectively. We denote $P_f(s \mid d)$ as the confidence measure that we identify image d as image s when applying the FJLT hashing. Similarly, $P_r(s \mid d)$ is denoted for that of the RI-FJLT hashing. Here, we simply define

$$P_f(s \mid d) = W_f\left(1 - \frac{\mathrm{Norm}\left(H_f^d - H_f^s\right)}{\mathrm{Norm}\left(H_f^s\right)}\right),$$
$$P_r(s \mid d) = W_r\left(1 - \frac{\mathrm{Norm}\left(H_r^d - H_r^s\right)}{\mathrm{Norm}\left(H_r^s\right)}\right), \tag{16}$$

where W_f and W_r are preselected weights in the case of FJLT and RI-FJLT hashing, respectively, and Norm means the Euclidean norm. Considering the poor performances of RI-FJLT hashing under many other types of attacks except for rotation ones, we intuitively introduce a weight W, where $0 \le W \le 1$, to the original confidence measures of FJLT and RI-FJLT hashing to decrease the possible negative influence of RI-FJLT hashing and maintain the advantages of both FJLT and RI-FJLT hashing in the proposed content-based fingerprinting under different attacks.

Regarding the identification decision making, given a tested image d, we calculate all the confidence measures $P_f(s_i \mid d)_{i=1}^N$ and $P_r(s_i \mid d)_{i=1}^N$ over the image database of $S = \{s_i\}_{i=1}^N$ by using FJLT and RI-FJLT hashing, and make

Table 1: Content-preserving manipulations and parameter settings.

Manipulation	Parameters Setting	Number
Additive noise		
Gaussian noise	Sigma: 0 ∼ 0.1	10
Salt and Pepper noise	Sigma: 0 ∼ 0.1	10
Speckle noise	Sigma: 0 ∼ 0.1	10
Blurring		
Gaussian blurring	Filter size: 3 ∼ 21, Sigma = 5	10
Circular blurring	Radius: 1 ∼ 10	10
Motion blurring	Len: 5 ∼ 15, θ : 0° ∼ 90°	9
Geometric attacks		
Rotation	Degree = 5° ∼ 45°	9
Cropping	5%, 10%, 20%, 25%, 30%, 35%	6
Scaling	25%, 50%, 75%, 150%, 200%	5
JPEG compression	Quality factor = (5 ∼ 50)	10
Gamma correction	γ = (0.75 ∼ 1.25)	10

the identification decision correspondingly by selecting the highest one among $P_f(s_i \mid d)_{i=1}^N$ and $P_r(s_i \mid d)_{i=1}^N$. Note that if a confidence measure $P(s \mid d)$ is negative, it means that the image d is outside the confidence interval of the image s and the confidence measure is assigned to be zero.

6. Analytical and Experimental Results

6.1. Database and Content-Preserving Manipulations. In order to evaluate the performance of the proposed new hashing algorithms, we test FJLT hashing and RI-FJLT hashing on a database of 100 000 images. In this database, there are 1000 original color nature images, which are mainly selected from the ten sets of categories in the content-based image retrieval database of the University of Washington (http://www.cs.washington.edu/research/imagedatabase/) as well as our own database. Therefore, some of the original images can be similar in content if they come from the same category, and some are distinct if they come from the different categories. For each original color image with size 256 × 384, we generate 99 similar but distorted versions by manipulating the original image according to eleven classes of content-preserving operations, including additive noise, filtering operations, and geometric attacks, as listed in Table 1. All the operations are implemented using Matlab. Here we give some brief explanations of some ambiguous manipulations. For image rotation, a black frame around the image will be added by Matlab but some parts of image will be cut if we want to keep its size the same as the original image. An example is given in Figure 4(b). Here our cropping attacks refer to the removal of the outer parts (i.e., let the values of the pixels on each boundary be equal to null and keep the significant content in the middle).

6.2. Identification Results and ROC Analysis. Our preliminary study [19] on a small database showed that FJLT hashing provides nearly perfect identification accuracy for the standard test images such as Baboon, Lena, and Peppers. Here we will measure the FJLT hashing and the new proposed RI-FJLT hashing on the new database, which consists of 1000 nature images from ten categories. Ideally, to be robust to all routine degradations and malicious attacks, no matter what content-preserving manipulation is done, the image with any distortion should still be correctly classified into the corresponding original image.

It is worth mentioning that all the pseudorandomizations of NMF-NMF-SQ hashing, FJLT hashing, and content-based fingerprinting are dependent on the same secret key in our experiment. As discussed in [16], the secret keys, more precisely the key-based randomizations, play important roles on both increasing the security (i.e., making the hash unpredictable) and enhancing scalability (i.e., keeping the collision ability from distinct images low and thus yielding a better identification performance) of the hashing algorithm. Therefore, the identification accuracy of a hashing algorithm is determined simultaneously by both the dimension reduction techniques (e.g., FJLT and NMF) and the secret keys. As shown in NMF hashing in [16], if we generate hashes of different images with varied secret keys, the identification performance can be further improved significantly because the secret key boosts up the cardinality of the probability space and brings down the probability of false alarm. In this paper, because we mainly focus on examining the identification capacity of hashing schemes themselves rather than the effects of secret keys, to minimize the effects of the factor of the secret keys, we use the same key in generating hash vectors for different images.

6.2.1. Results of FJLT Hashing. Following the algorithms designed in Section 3, we test the FJLT hashing with the parameters chosen as $m = 64$, $N = 40$, $\varepsilon = 0.1$, key = 5, as summarized in Table 3. Note that most of the keys could be used in FJLT hashing because of its robustness to secret keys, which has been illustrated in [19]. Since the NMF-NMF-SQ hashing has been shown to outperform the SVD-SVD and PR-SQ hashing algorithms having the best known robustness properties in the existing literature, we compare the performance of our proposed FJLT hashing algorithm with NMF-NMF-SQ hashing when testing on the new database. For the NMF approach, the parameters are set as $m = 64$, $p = 10$, $r_1 = 2$, $r_2 = 1$, and $M = 40$ according to [16]. It is worth mentioning that, to be consistent with the FJLT approach, we chose the same size of subimages and length of hash vector in NMF hashing (denoted as m and M), which facilitate a fair comparison between them later. We also tried the setting $p = 40$ (with p represents the number of subimages in the NMF approach), but it was found that the choice of $p = 10$ yields a better performance. Consequently, NMF hash vector has the same length 40 as the FJLT hash vector. We first examine the identification accuracy of both hashing algorithms under different attacks, and the identification results are shown in Table 2. It is clearly noted that the proposed FJLT hashing consistently yields a higher identification accuracy than that of NMF hashing under different types of tested manipulations and attacks.

Table 2: Identification accuracy for manipulated images by NMF-NMF-SQ (NMF) hashing, FJLT hashing, and content-Based fingerprinting (CBF) based on FJLT and RI-FJLT hashing.

Manipulations	NMF	FJLT	CBF
Additive noise			
Gaussian noise*	59.38%	69.5%	62.36%
Salt and Pepper noise	81.87%	96.87%	97.71%
Speckle noise	78.27%	99.83%	99.77%
Blurring			
Gaussian blurring	98.31%	99.49%	99.04%
Circular blurring	98.36%	99.51%	99.09%
Motion blurring	98.88%	99.81%	99.66%
Geometric attacks			
Rotation	16.43%	36.86%	86.54%
Cropping	16.75%	96.6%	96.14%
Scaling	98.47%	100%	100%
JPEG compression	99.7%	100%	100%
Gamma correction	5.22%	86.62%	74.26%

*With the help of median filter in preprocessing, the identification accuracy of NMF hashing under Gaussian noise could be improved to 90.61% and 99.5% for FJLT hashing.

Table 3: Parameter setting in the FJLT hashing algorithm.

Parameter	Value
Size of the subimage	$m = 64$
Length of the hash vector	$N = 40$
Parameters of FJLT	$\varepsilon = 0.1$, $c = 250$, $c' = 1$.
Secret key	key = 5

We then present a statistical comparison of the proposed FJLT and NMF hashing algorithms by studying the corresponding ROC curves. We first generate the overall ROC curves for all types of tested manipulations when applying different hashing schemes, and the resulting ROC curves are shown in Figure 6. From Figure 6, one major observation is that the proposed FJLT hashing outperforms NMF-NMF-SQ hashing. To test the robustness to each type of attacks, a ROC curve is also generated for a particular attack and hash algorithm. Since we note from Table 2 that the proposed FJLT hashing significantly outperforms NMF-NMF-SQ for additive noise, cropping and gamma correction attacks, we show the ROC curves corresponding to the six attacks (i.e., Gaussian noise, Salt and Pepper noise, Speckle noise, Rotation attacks, Cropping and Gamma correction) in Figure 7. Once again, the ROC curves in Figure 7 reinforce the observation that FJLT hashing significantly outperform the state-of-art NMF hashing. However, both of them are still a little sensitive to Gaussian noise as shown in Figure 7(a). The underlying reason is that we did not incorporate any preprocessing such as median filter into FJLT hashing or NMF hashing, because we would investigate the robustness of FJLT and NMF hashing themselves to additive noise. In practice, the preprocessing such as image denoising before applying image hashing could further improve the robustness to additive noise (referring to the annotation below Table 2), since both FJLT hashing and NMF hashing

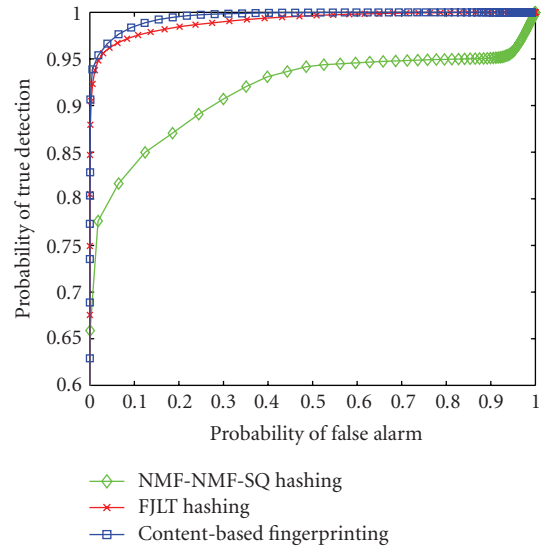


Figure 6: The overall ROC curves of NMF-NMF-SQ hashing, FJLT hashing, and content-based fingerprinting under all types of tested manipulations.

are strongly robust to blurring. As for the attacks such as JPEG compression and Blurring, since we observe perfect identification performances and no false alarms in our own experiments, we do not report the ROC curves further, which are similar to the ROC results via NMF hashing shown in [16].

Here we try to give some intuitive explanations regarding the observed performances of the two hashing algorithms. In NMF hashing, the dimension reduction technique is based on the approximative nonnegative matrix factorization, which factorizes the image matrix into two lower rank matrices. However, the problem of choosing a low rank r

(e.g., r_1, r_2 in the NMF hashing) is of great importance, though it is observed to be sensitive to the data. While for FJLT hashing, the mapping is obtained by a coefficients matrix and a subimage is treated as a point in a high-dimensional space (in our case, the dimension is $64 \times 64 = 4096$). One advantage of FJLT hashing is that minor modifications in the content will not affect the integrity of the global information, which results in a better performance. However, as illustrated in Table 2 and the ROC curve in Figure 7(d), both FJLT hashing and NMF hashing provide poor performances under rotation attacks, and we shall investigate this problem further.

6.2.2. Results of RI-FJLT Hashing. In Table 2, we note that one drawback of FJLT hashing is its vulnerability to rotation attacks. Especially, as shown by an example in Figure 4, for a large rotation degree of 45, FJLT hashing failed to identify the image content. Here we apply the RI-FJLT hashing approach presented in Section 6 to overcome this drawback.

We generated 36 rotated versions for each test image in the database and the rotation degrees are varied from 5 to 180 with an interval of 5 degrees. Though not investigated further here, it is worth mentioning that, before the conversion from Cartesian coordinates to Log-Polar coordinates, some preprocessing operations such as median filtering can be helpful to enhance the identification performance [8], especially under additive noise distortions. We have employed median filter as preprocessing in RI-FJLT hashing. The identification results under rotation attacks are shown in Table 4. We can see from the table that FJLT hashing is obviously sensitive to rotation attacks and thus its identification accuracy greatly degrades with the increase of rotation degree. It is also noted that RI-FJLT hashing still consistently achieves almost perfect identification accuracy under rotation attacks even with large rotation degrees.

Although the invariance of Fourier-Mellin transform benefits the FJLT hashing with the robustness to rotation attacks, such robustness to rotation comes at the cost of degraded identification accuracy for other types of manipulations and attacks. We have intuitively discussed the reasons for this observation in Section 4. We argue that it may not be feasible to be robustly against various attacks by only depending on single feature descriptor. This observation motivates us to look for an alternative solution that is the content-based fingerprinting we proposed in Section 5 to tackle this problem.

6.2.3. Results of Content-Based Fingerprinting. Since FJLT hashing is demonstrated to be robust against a large class of distortions except for rotation attacks and RI-FJLT hashing achieves superior performance under rotation attacks at the cost of sensitivity to other manipulations, it accounts for the fact that it is very difficult to design a globally optimal hashing approach that could handle all of the distortions and manipulations. Hence, we combine FJLT hashing and RI-FJLT hashing following the framework of content-based fingerprinting proposed in Section 5 and test its performance on the database described in Section 6.1. Considering the poor performance of RI-FJLT hashing on other manipulations, we need to introduce an elaborate weight shown in Section 5.2 to the confidence measure of RI-FJLT hashing to get rid of its negative influence and try to maintain the advantages of both FJLT and RI-FJLT hashing in the proposed content-based fingerprinting. Based on our preliminary study, we set $W_f = 1$ to keep the advantages of FJLT hashing and find that a good weight W_r could be drawn from the interval range $\{0.85 \sim 0.9\}$. We set $W_r = 0.895$ in our implementation and exhibit the results in Table 2.

To have a fair comparison between different approaches, though we combine the FJLT hashing and the RI-FJLT hashing in the content-based fingerprinting, the length of the overall fingerprint vector is still chosen as 40 (with 20 components from the FJLT hashing and the left 20 from the RI-FJLT hashing), which is the same as that of the FJLT hashing and the NMF hashing. It is clear that the simple joint decisionmaking complements the drawback of FJLT hashing under rotation attacks by incorporating the RI-FJLT hashing into the proposed content-based fingerprinting. The ROC curves for FJLT hashing, NMF hashing, and the proposed content-based fingerprinting under rotation attacks are shown in Figure 7(d). Obviously, among the three approaches, the content-based fingerprinting yields the highest true positive rates when the same false positive rates are considered. The ROC curves of the content-based fingerprinting approach under other types of attacks are also illustrated in Figure 7. We note that the robustness of content-based fingerprinting to additive noise, cropping, and Gamma correction slightly degrades, as shown in Figure 7. One possible explanation could be that the current simple decision-making process is not the theoretically optimal one that could eliminate the negative effect of RI-FJLT hashing under these attacks. However, the overall performance of content-based fingerprinting as illustrated by the ROC curve in Figure 6 demonstrates that it is superior and more flexible than a single hashing approach, because the selection of features and secure hashes can be adapted to address different practical application concerns. Therefore, the proposed content-based fingerprinting can be a promising extension and evolution of traditional image hashing.

6.3. Unpredictability Analysis. Except for the robustness against different types of attacks, the security in terms of unpredictability that arises from the key-dependent randomization is another important property of hashing and the proposed content-based fingerprinting. Here we mainly focus on the unpredictability analysis of FJLT hashing, because the unpredictability of the RI-FJLT hashing and the content-based fingerprinting proposed arise from the FJLT hashing. Higher amount of the randomness in the hash values makes it harder for the adversary to estimate and forge the hash without knowing the secret keys. Since it is believed that a high differential entropy is a necessary property of secure image hashes, we evaluate the security in terms of unpredictability of FJLT hashing by quantifying

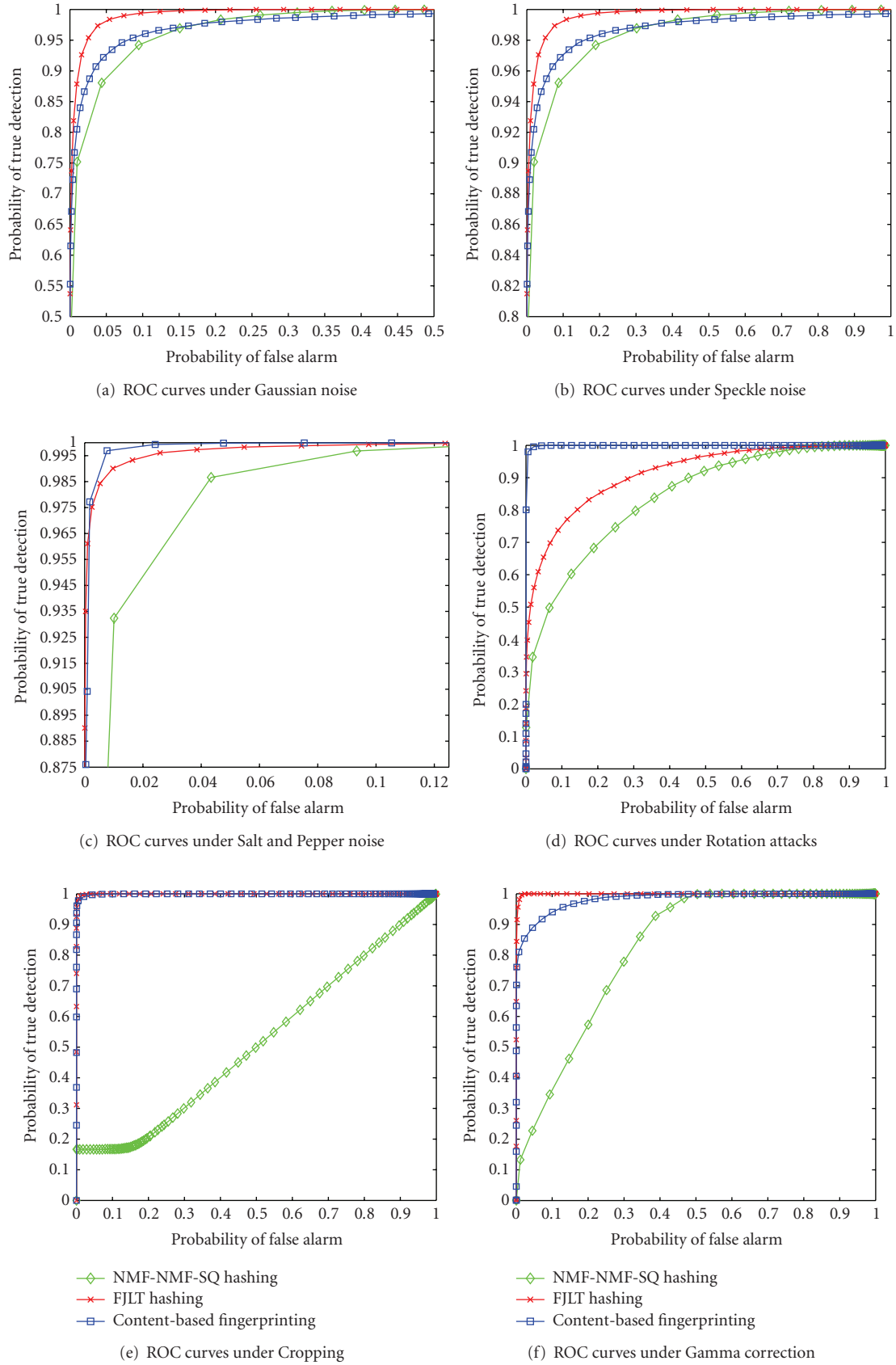


Figure 7: The ROC curves of NMF-NMF-SQ hashing, FJLT hashing, and content-based fingerprinting under six types of attacks, respectively. (a) Gaussian noise; (b) Speckle noise; (c) Salt and Pepper noise; (d) Rotation attacks; (e) Cropping; (f) Gamma correction.

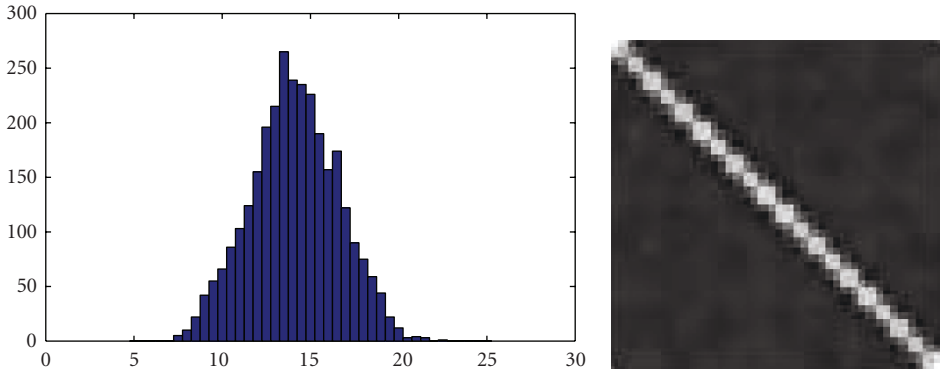


Figure 8: (a) The histogram of a typical FJLT hash vector component for image Lena from 3000 different secret keys. (b) The covariance matrix of the FJLT hash vector for image Lena from 3000 different secret keys.

the differential entropy of the FJLT hash vector, as proposed in [8]. The differential entropy of a continuous random variable X is given by

$$H(X) = \int_{\Omega} f(x) \log \frac{1}{f(x)} \mathrm{d}x, \tag{17}$$

where $f(x)$ means the probability density function (pdf) of X and Ω means the support area of $f(x)$. Since the analytical model of the pdf of the FJLT hash vector component is generally not available, we carry out the practical pdf approximation using the histograms of the hash vector components. Figure 8(a) shows the histogram of a typical component from the FJLT hash vector of image Lena resulting from 3000 different keys. It is noted that it approximately follows a Gaussian distribution. Similarly, we can obtain the histograms of other components. Based on our observations, we state that the FJLT hash vector approximately follows a multivariate Gaussian distribution. Therefore, similar to the hash in [16], we have the differential entropy of the FJLT hash vector X as

$$H(X) = \frac{1}{2} \log (2\pi e)^N |\mathrm{Cov}| \text{ bits}, \tag{18}$$

where $|\mathrm{Cov}|$ means the determinant of the covariance matrix of the hash vector, and N means the length of the FJLT hash vector.

From Figure 8(b) where an example of the covariance matrix of the FJLT hash vector is shown, we can see that the covariance matrix is approximately a diagonal matrix, meaning that the components are approximately statistically independent. Therefore, $|\mathrm{Cov}|$ can be approximately estimated as

$$|\mathrm{Cov}| = \prod_{i=1}^{N} \sigma_i^2, \tag{19}$$

where σ_i^2 means the variance of the component h_i in the FJLT hash vector. Since from information theory, the differential entropy of a random vector $X \in \mathbb{R}^n$ is maximized when X follows a multivariate normal distribution $\mathcal{N}(0, \mathrm{Cov})$ [21], we argue that the proposed FJLT hashing is highly secure (unpredictable) as it approximately follows [18]. We note that NMF-NMF-SQ hashing also was shown to approximately follow a joint Gaussian distribution and a similar statement in terms of differential entropy was given in [16]. Hence, we state that the proposed FJLT hash is comparably as secure as NMF hashing, which was shown to be presumably more secure than previously proposed schemes that are based on random rectangles alone [16].

However, the security of image hashing does not only lie on a higher differential entropy, which is only one aspect of a secure image hashing [8, 16], but also includes other factors such as key diversity and prior knowledge possessed by adversaries. Therefore, how to comprehensively evaluate the security of image hashing is still an open question. Interested readers could refer to the literatures [8, 27] regarding the security analysis issues.

6.4. Computational Complexity. We analyze the computational complexity of the proposed FJLT hashing and RI-FJLT algorithms (the computational cost of content-based fingerprinting is the sum of FJLT and RI-FJLT hashing) when compared with the NMF-NMF-SQ hashing algorithm.

(i) NMF. In [16], the computational complexity of NMF-NMF-SQ hashing has been given as follows. It does a rank r_1 NMF on $nm \times m$ matrices and then a rank r_2 approximation from the resulting $m \times 2pr_1$ matrix in [16]. At last, pseudorandom numbers are incorporated in the NMF-NMF vector of length $mr_2 + 2pr_1r_2$, and the total computation cost is

$$C_{\mathrm{NMF}} = n \cdot \mathcal{O}(m^2 r_1) + \mathcal{O}(2mnr_1r_2) + \mathcal{O}(mr_2 + 2nr_1r_2). \tag{20}$$

(ii) FJLT. Based on the analysis in [18], given a $x \in \mathbb{R}^d$, the computation cost of FJLT on x is calculated as follows. Computing $D(x)$ requires $\mathcal{O}(d)$ time and $H(Dx)$ requires $\mathcal{O}(d \log d)$. For computing $P(HDx)$,

Table 4: Identification accuracy under rotation attacks by FJLT and RI-FJLT.

Rotation degree	FJLT	RI-FJLT
$5^\circ \sim 45^\circ$	30.43%	94.57%
$50^\circ \sim 90^\circ$	0.67%	96.03%
$95^\circ \sim 135^\circ$	0.58%	94.62%
$140^\circ \sim 180^\circ$	1.13%	96.06%
Overall	8.2%	95.32%

Table 5: Computational time costs for Lena with 256×256 by FJLT, RI-FJLT and NMF-NMF-SQ hashing algorithms.

Computational cost	FJLT	RI-FJLT	NMF-NMF-SQ
time (s)	1.93	2.43	5.55

it takes $\mathcal{O}(p)$, where the p is the number of nonzeros in P, we know that the p satisfies the Binomial distribution $B(dk, q)$, therefore we take the mean value of p as dkq that equals $k\log^2 n$, where k is $\varepsilon^{-2}\log n$. Then, take the random weight incorporation into account, we have the total computation cost of the FJLT hashing as ($d = m^2$ in our case)

$$C_{\text{FJLT}} = \mathcal{O}\left(m^2(1 + 2\log m)\right) + \mathcal{O}\left(k\left(1 + \log^2 n\right)\right). \quad (21)$$

(iii) RI-FJLT. Except for the cost of FJLT hashing, we need to take the bilinear interpolation that requires $\mathcal{O}(m^2)$ and Fourier transform that takes $\mathcal{O}(m^2 \log m)$ by FFT into account. Consequently, the cost of RI-FJLT is

$$C_{\text{RI-FJLT}} = \mathcal{O}\left(m^2(2 + 3\log m)\right) + \mathcal{O}\left(k\left(1 + \log^2 n\right)\right). \quad (22)$$

Here, we specify that $k \approx 5m$ in our case and also take other parameters into account. Obviously the FJLT and RI-FJLT hashing roughly require a lower computational cost than that of NMF-NMF-SQ. To have an intuitive feeling of the computational costs required by different algorithms, we also test on a standard image Lena with size 256×256 by using a computer with Intel Core 2 CPU (2.00 GHz) and 2 G RAM. The required computational time is listed in Table 5, which shows that the FJLT and RI-FJLT hashing are much faster than NMF-NMF-SQ hashing. Note that the costs are based on a length 20 of the hash vectors in our experiments. Increasing the length of hash vectors will enhance the identification accuracy but will require more computational costs. This trade-off will be further studied in the future.

7. Discussions and Conclusion

In this paper, we have introduced a new dimension reduction technique—FJLT, and applied it to develop new image hashing algorithms. Based on our experimental results, it is noted that the FJLT-based hashing is robust to a large class of routine distortions and malicious manipulations. Compared with the NMF-based approach, the proposed FJLT hashing can achieve comparable, sometimes better, performances than that of NMF, while requiring less computational cost. The random projection and low distortion properties of FJLT make it more suitable for hashing in practice than the NMF approach. Further, we have incorporated Fourier-Mellin transform to complement the deficiency of FJLT hashing under rotation attacks. The experimental results confirm the fact that generating a hash descriptor based on a ceratin type of features to resist all types of attacks is highly unlikely in practice. However, for a particular type of distortion, it is feasible to find a specific feature to tackle it and obtain good performance. These observations motivate us to propose the concept of content-based fingerprinting as an extension of image hashing and demonstrate the superiority of combining different features and hashing algorithms.

We note that the content-based fingerprinting approach by using FJLT and RI-FJLT still suffers from some distortions, such as Gaussian noise and Gamma correction. One solution is to further find other features that are robust to these attacks/manipulations and incorporate them into the proposed scheme to enhance the performance. Future work will include how to incorporate other robust features (such as the popular SIFT-based features) and secure hashing algorithms to optimize the content-based fingerprinting framework and at the same time explore efficient hierarchical decision-making schemes for identification.

Furthermore, we plan to explore the variations of the current FJLT hashing. Similar to the NMF-based hashing approach (referred as NMF-NMF-SQ hashing in [16]) where the hash is based on a two-stage application of NMF, we can modify the proposed FJLT hashing into a two-stage FJLT-based hashing approach by introducing a second stage of FJLT as follows. Treat the intermediate hash IH as a vector with length $k \times N$, and then reapply FJLT to obtain a representation of the vector IH with further dimension reduction. Compared with our current one-stage FJLT-based hashing, the length of intermediate hash IH could be further shortened by the second FJLT and the security would be enhanced in the two-stage FJLT hashing. However, the robustness of a two-stage FJLT-based hashing under attacks such as cropping may degrade, since now each component in the modified hash vector is contributed by all the subimages by random sampling. Therefore, the distortion of local information in one subimage could affect the whole hash vector rather than a couple of hash components. The computation cost can also be a concern. We will investigate these issues in the future work.

Another concern that is of great importance in practice but is rarely discussed in the context of image hashing is automation. Automatic estimation/choice of design parameters removes the subjectivity from the design procedure and can yield better performances. For instance, algorithms for automating the design process of image watermarking have already been implemented in the literature [28–30]. However, to our knowledge, this automated solution has not yet been explored in the context of image hashing. Our preliminary study in [31] demonstrated that using

a genetic algorithm (GA) for automatic estimation of parameters of the FJLT hashing using could improve the identification performance. However, choosing the appropriate fitness function is challenging in automated image hash. We plan to investigate different fitness functions and how the GA algorithm can incorporate other factors (such as keys) and other constraints (such as the hash length).

References

[1] R. Venkatesan, S.-M. Koon, M. H. Jakubowski, and P. Moulin, "Robust image hashing," in *Proceedings of the International Conference on Image Processing (ICIP '00)*, vol. 3, pp. 664–666, Vancouver, Canada, September 2000.

[2] J. Fridrich and M. Goljan, "Robust hash functions for digital watermarking," in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC '00)*, pp. 178–183, Las Vegas, Nev, USA, March 2000.

[3] M. Wu, Y. Mao, and A. Swaminathan, "A signal processing and randomization perspective of robust and secure image hashing," in *Proceedings of the IEEE/SP 14th Workshop on Statistical Signal Processing*, pp. 166–170, Madison, Wis, USA, August 2007.

[4] C. W. Wu, "On the design of content-based multimedia authentication systems," *IEEE Transactions on Multimedia*, vol. 4, no. 3, pp. 385–393, 2002.

[5] E. Martinen and G. W. Wornell, "Multimedia content authentication: fundamental limits," in *Proceedings of the IEEE International Conference Image Processing (ICIP '02)*, vol. 2, pp. 17–20, Rochester, NY, USA, 2002.

[6] M. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: state of the art and challenges," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 2, no. 1, pp. 1–19, 2006.

[7] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Contentbased image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1349–1380, 2000.

[8] A. Swaminathan, Y. Mao, and M. Wu, "Robust and secure image hashing," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 215–230, 2006.

[9] V. Monga and B. L. Evans, "Perceptual image hashing via feature points: performance evaluation and tradeoffs," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3453–3466, 2006.

[10] V. Monga, A. Banerjee, and B. L. Evans, "A clustering based approach to perceptual image hashing," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 1, pp. 68–79, 2006.

[11] M. Johnson and K. Ramchandran, "Dither-based secure image hashing usng distributed coding," in *Proceedings of the International Conference on Image Processing (ICIP '03)*, vol. 3, pp. 751–754, Barcelona, Spain, September 2003.

[12] F. Lefbvre, J. Czyz, and B. Macq, "A robust soft hash algorithm for digital image signature," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '03)*, vol. 2, pp. 495–498, Barcelona, Spain, September 2003.

[13] K. Mihcak and R. Venkatesan, "New iterative geometric techniques for robust image hashing," in *Proceedings of the ACM Workshop in Security and Privacy in Digital Rights Management*, pp. 13–21, Philadelphia, Pa, USA, November 2001.

[14] S. S. Kozat, R. Venkatesan, and M. K. Mihcak, "Robust perceptual image hashing via matrix invariants," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '04)*, vol. 5, pp. 3443–3446, Singapore, October 2004.

[15] D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556–562, 2001.

[16] V. Monga and M. K. Mihcak, "Robust and secure image hashing via non-negative matrix factorizations," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 376–390, 2007.

[17] D. Guillamet, B. Schiele, and J. Vitria, "Analyzing non-negative matrix factorization for image classification," in *Proceedings of the International Conference on Pattern Recognition*, vol. 16, pp. 116–119, Quebec, Canada, August 2002.

[18] N. Ailon and B. Chazelle, "Approximate nearest neighbors and the fast johnson-lindenstrauss transform," in *Proceedings of the 38st Annual Symposium on the Theory of Computing (STOC '06)*, pp. 557–563, Seattle, Wash, USA, 2006.

[19] X. Lv and Z. Wang, "Fast Johnson-Lindenstrauss transform for robust and secure image hashing," in *Proceedings of the IEEE 10th Workshop on Multimedia Signal Processing*, pp. 725–729, Cairns, Australia, October 2008.

[20] C. Lin, M. Wu, J. Bloom, et al., "Rotation, scale, and translation resilient watermarking for images," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 767–782, 2001.

[21] T. Cover, J. Thomas, J. Wiley, and W. InterScience, *Elements of Information Theory*, Wiley-Interscience, New York, NY, USA, 2006.

[22] S. Dasgupta and A. Gupta, "An elementary proof of the Johnson-Lindenstrauss lemma," Tech. Rep., International Computer Science Institute, 1999.

[23] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[24] M. Alghoniemy and A. H. Tewfik, "Geometric invariance in image watermarking," *IEEE Transactions on Image Processing*, vol. 13, no. 2, pp. 145–153, 2004.

[25] C. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York, NY, USA, 2006.

[26] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognition*, vol. 38, no. 12, pp. 2270–2285, 2005.

[27] Y. Mao and M. Wu, "Unicity distance of robust image hashing," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, part 1, pp. 462–467, 2007.

[28] F. Y. Shih and Y. Wu, "Enhancement of image watermark retrieval based on genetic algorithms," *Journal of Visual Communication and Image Representation*, vol. 16, no. 2, pp. 115–133, 2005.

[29] C. S. Shieh, H. C. Huang, F. H. Wang, and J. S. Pan, "Genetic watermarking based on transform-domain techniques," *Pattern Recognition*, vol. 37, no. 3, pp. 555–565, 2004.

[30] S. Chu, H. Huang, Y. Shi, S. Wu, and C. Shieh, "Genetic watermarking for zerotree-based applications," *Circuits, Systems, and Signal Processing*, vol. 27, no. 2, pp. 171–182, 2008.

[31] M. Fatourechi, X. Lv, and Z. J. Wang, "Towards fast automated image hashing based on fast johnson-lindenstrauss transform (fjlt)," in *Proceedings of the IEEE International Workshop on Information Forensics and Security*, London, UK, December 2009.