Robust hashing for image authentication using quaternion discrete Fourier transform and log-polar transform

Junlin Ouyang, Gouenou Coatrieux, Huazhong Shu

Robust hashing for image authentication using quaternion discrete Fourier transform and log-polar transform

Junlin Ouyang[1], Gouenou Coatrieux[2], Huazhong Shu[1, 3]

[1]Laboratory of Image Science and Technology, Southeast University - Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, 210096, Nanjing, China

[2]Institut Mines-Telecom, Telecom Bretagne, INSERM U1101 Latim, Brest, F-29238, France

[3]Centre de Recherche en Information Médicale Sino-français (CRIBs), Rennes, F-35000, France

Information about the corresponding author

Huazhong Shu, Ph. D

Laboratory of Image Science and Technology

School of Computer Science and Engineering

Southeast University, 210096, Nanjing, China

Tel: 00-86-25-83 79 42 49

Fax: 00-86-25-83 79 26 98

Email: shu.list@seu.edu.cn

Abstract. In this work, a novel robust image hashing scheme for image authentication is proposed based on the combination of the quaternion discrete Fourier transform (QDFT) with the log-polar transform. QDFT offers a sound way to jointly deal with the three channels of color images. The key features of the present method rely on (i) the computation of a secondary image using a log-polar transform; and (ii) the extraction from this image of low frequency QDFT coefficients' magnitude. The final image hash is generated according to the correlation of these magnitude coefficients and is scrambled by a secrete key to enhance the system security. Experiments were conducted in order to analyze and identify the most appropriate parameter values of the proposed method and also to compare its performance to some reference methods in terms of receiver operating characteristics curves. The results show that the proposed scheme offers a good sensitivity to image content alterations and is robust to the common content-preserving operations, and especially to large angle rotation operations.

Keywords: Robust image hashing，quaternion discrete Fourier transform，log-polar transform, image authentication.

1. INTRODUCTION

With the widespread use of sophisticated image editing tools, image contents can be easily tampered or forged. Verifying image authenticity is therefore a major issue in many applications. Robust image hashing is widely applied in image authentication [1-3]. Schematically, image hashing methods extract essential image features from which a short binary or real number sequence, called hash, is generated to represent the image content. Such a hash should be robust to image content-preserving operations while being sensitive to malicious tampering ones. Because robust image hashing captures the main image characteristics, it has attracted interest for other applications like image forensic [4, 5], image retrieval [6, 7], digital watermarking [8, 9], and so on. Regarding image authentication, a robust image hash should also have good anti-collision (discriminative) capability for visually distinct images as well as a satisfactory level of security in order to make very difficult for an adversary to forge the hash value. To meet these requirements simultaneously, the construction of a robust image hashing is still a challenging task.

In general, the construction of an image hash is based on three basic steps, i.e., pre-processing, image feature extraction and construction of hash. Among them, the most critical one is certainly the image feature extraction step [10]. Existing feature extraction methods for robust image hashing can be roughly classified into the following categories.

Discrete Cosine Transform (DCT)-based image hashing methods: Fridrich and Goljan [8] used the DCT to capture the essential features of image blocks. They observed that it is very difficult to change the correlation of the low-frequency DCT coefficients without tampering the content of an image. Therefore, low-frequency DCT coefficients can be utilized as features to build the hash. De Roover *et al*. [11] defined a radial variance vector (RAV) based on the radial projection of the image pixels, and then applied the DCT to compress the RAV feature

vector and construct the image hash. This method is robust to content-preserving operations and small angle rotations.

Discrete wavelet transform (DWT) -based image hashing methods: Ahmed *et al*. [1] used a wavelet transform to extract the image features. Since the wavelet transform has a good time-frequency localization property, their method can locate tampered regions with a good accuracy but at the price of a longer image hash. Wu *et al*. [12] combined the Radon transform (RT) and the DWT to deal with print-scan attacks. Tang *et al*. [13] developed an image hashing scheme after observing that the entropy of pixel blocks, a measure used to characterize image texture, is approximately linearly changed after content-preserving operations. Then, they applied DWT to image block's entropies to realize feature compression and construct the image hash. Recently, color vector angle combined with DWT [14] were applied to robust image hashing. Their results show good robustness to common content-preserving operations and small angle rotation. Liu *et al*. [15] utilized the wave atom transform to extract image features arguing that this approach has a sparse expansion and is capable to better capture texture properties. Furthermore, they observed that the coefficients of the third scale band are more suitable to serve as image hash features than the other scale bands.

Discrete Fourier transforms (DFT)-based image hashing methods: Most DFT based approaches are combined with other transforms in order to resist to geometric distortions. For example, Swaminathan *et al*. [10] proposed a robust image hashing scheme based on the Fourier-Mellin transform. Qin *et al*. [16] introduced another scheme where a secondary image is first obtained after a rotation projection similar to the RAV of the image. A non-uniform sampling is then performed to extract the image features after applying the DFT. This method is robust to small angle rotations. Lei *et al*. [11] first carried out a RT and then computed moment features before applying DFT on these moments. The first fifteen significant DFT coefficients were then normalized and quantized to obtain the image hash value. This method shows satisfactory results when facing geometrical distortions.

Matrix decomposition-based image hashing methods: Kozat *et al*. [17] used singular value decomposition (SVD) to get robust image features and to generate an image hash. Their hashing algorithm consists of two major steps. In the first one, intermediate features are extracted from pseudo-random (PR) semi-global regions via SVD. In the second step, the SVD is again applied to the intermediate features so as to construct the final hash. Monga *et al*. [18] also introduced a new PR signal representation method using non-negative matrix factorization (NMF) to capture the features and to form the image hash. Their method shows better performance than the SVD based method. Recently, Tang *et al*. [19] proposed a robust image hashing method based on ring partition and NMF. Their results show good performance to common content-preserving operations and large angle rotation operation.

Others image hashing methods: In [20], Xiang *et al*. proposed a histogram-based image hashing scheme, which is robust to geometric distortions but not to additive noise, brightness adjustment and contrast enhancement. Battiato *et al*. [21] adopted an image representation based on a set of SIFT features (called "bag of features", BOF) to construct the hash and explored a non-uniform quantization of histograms of oriented gradients (HOG) to get tamper localization capabilities. Zhao *et al*. [3] combined global and local features to construct an image hash. The global features correspond to the Zernike moments of the luminance and chrominance components of the image, while the local ones include the positions and the texture information of salient regions. This algorithm can identify the type of image tamper as well as the modified areas' location. Other robust feature extraction methods for constructing image hashes were also reported including the random Gabor filtering [22], the ring partition [23] and shape contexts [24].

When applied to color images, most of the above schemes convert three color channels (i.e., Red, Green, Blue or RGB) into gray-scale images while discarding the chrominance information. However, exploiting the chrominance information may not only improve the detection performance [3, 16], but may also make image forgery more difficult due to the fact that the hash contains both luminance and chrominance information. Quaternions can offer

a sound way to simultaneously deal with the three color channels without discarding the chrominance information. They have already been successfully employed in color image registration [25], image analysis [26-30] and watermarking [31-33]. Recently, quaternion discrete Fourier transform (QDFT) was used to generate image hashing and applied to image retrieval [34]. By following the same path, we proposed a novel image hashing method based on QDFT and log-polar transform for image authentication. QDFT can handle simultaneously the three channels (RGB) of color image without discarding chrominance information. Similar to DFT, the low-frequency coefficients of QDFT contain the main energy of the image and represent essential image features. In addition, QDFT can be combined with the log-polar transform so as to achieve a set of features that are rotation invariant. We propose thus to take advantage of QDFT used here to build a novel and compact image hash that is robust to content-preserving operations, geometric attacks while being sensitive to malicious tampering operations.

The rest of this paper is organized as follows. Section 2 gives a brief overview of quaternion and QDFT. Section 3 introduces the proposed scheme including its pre-processing, image feature extraction, and hash construction steps as well as its image authentication procedure. Experiments and comparison results are provided in Section 4. Some concluding remarks are given in Section 5.

2. PRELIMINARIES

2.1 Quaternion

Quaternion is a generalization of complex numbers and was introduced by Hamilton in 1843 [26]. A quaternion number has four parts: one real part and three imaginary parts, and can be written as follows:

$$q = a + b\boldsymbol{i} + c\boldsymbol{j} + d\boldsymbol{k}\,, \tag{1}$$

where a , b , c and d are real numbers, $\boldsymbol{i}$, $\boldsymbol{j}$ and $\boldsymbol{k}$ are imaginary units obeying the following rules:

$$\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = -1\,, \boldsymbol{ij} = -\boldsymbol{ji} = \boldsymbol{k}, \boldsymbol{jk} = -\boldsymbol{kj} = \boldsymbol{i}, \boldsymbol{ki} = -\boldsymbol{ik} = \boldsymbol{j}\,. \tag{2}$$

From Eq. (2), it can be found that the multiplication rule of quaternion is not commutative. The conjugate and

modulus of a quaternion q are respectively defined as follows:

$$\overline{q} = a - b\boldsymbol{i} - c\boldsymbol{j} - d\boldsymbol{k} \ , \qquad |q| = \sqrt{a^2 + b^2 + c^2 + d^2} \ . \tag{3}$$

A quaternion q with a zero real part is called a pure quaternion. One pixel of a color image f at the spatial position (x, y) has three components, and can be represented as a pure quaternion [27]:

$$f(x,y) = f_R(x,y)\boldsymbol{i} + f_G(x,y)\boldsymbol{j} + f_B(x,y)\boldsymbol{k} \ , \tag{4}$$

where $f_R(x,y)$, $f_G(x,y)$ and $f_B(x,y)$ are respectively the RGB component values of $f(x, y)$.

2.2 Quaternion discrete Fourier transform

Quaternion or hypercomplex Fourier transform was first introduced in the image processing community by Ell [28]. Due to the non-commutative property of the quaternion multiplication, there are three different types of QDFT, namely, right-side, left-side and two-side [29]. Since that the right side QDFT can be processed in a similar way as the left side QDFT, and the operations are much easier with the left side DQFT or right side QDFT than that with the two-side QDFT [25, 33], we decided to use the left-side QDFT. For a color image $f(x, y)$ of size $M \times M$, the QDFT and its inverse transform IQDFT are defined as follows [30]:

$$F(u,v) = \frac{1}{M} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} e^{-2\mu\pi\left(\frac{ux}{M} + \frac{vy}{M}\right)} f(x,y) \ , \tag{5}$$

$$f(x,y) = \frac{1}{M} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} e^{2\mu\pi\left(\frac{ux}{M} + \frac{vy}{M}\right)} F(u,v) \ , \tag{6}$$

where $\boldsymbol{\mu}$ is any unit pure quaternion that can be defined as a linear combination of $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ such as: $\boldsymbol{\mu} = \alpha\boldsymbol{i} + \beta\boldsymbol{j} + \gamma\boldsymbol{k}$, $\alpha, \beta, \gamma \in \mathbf{R}$, $|\boldsymbol{\mu}| = 1$.

3. PROPOSED SCHEME

Our robust image hashing scheme is shown in Fig. 1. Like any robust hashing strategy, it includes an image pre-processing stage followed by an image feature extraction process and a hash construction. We detail each of these stages below.

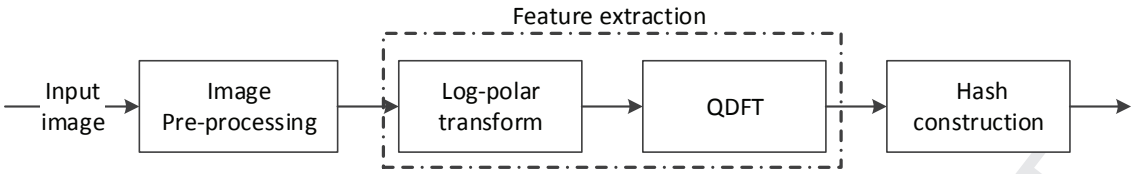


Fig. 1. Block diagram of the proposed robust image hashing scheme.

3.1. Pre-processing

The pre-processing procedure we use is illustrated in Fig. 2 for Lena test image. This one takes I_0 as an input image which is first rescaled to a fixed size $M \times M$. This step ensures that the generated image hash will be of fixed length while making it robust against image scaling operation. In this paper, the value $M = 256$ is chosen. In a second step, to make the generated image hash more robust, a smoothing (i.e. averaging filter) operation based on a $k \times k$ window is applied. Its aim is to preserve the essential structures while removing insignificant details without endangering the image content (see Fig. 2(b)). Then, considering that an image can be rotated, we propose to only consider the smoothed pixels within the image inscribed circle as illustrated in Fig. 2 (d). This is obtained by setting to zero the pixels outside the inscribed circle. As exposed in Fig. 2 (d) and Fig. 2 (e), the image and its rotated version have the same pixel values. This allows us minimizing the influence of information loss, especially the loss of pixels in the image corners, in the case of image rotations with large angles.

Fig. 2. Pre-processing procedure applied to an image and to a rotated version of an angle of 45°. (a) Image I_0, (b)-(c) Smoothed images, (d)-(e) inscribed circle in smoothed images (b) and (c).

3.2. Feature extraction

Robustness to image scaling being handled in the pre-processing stage, the image features we are looking for have to be robust to image rotation. To do so, we propose to first transform the pre-processing image into log-polar domain and then to apply the QDFT.

Let $f_1(x, y)$ be a rotated version of the original color image $f_0(x, y)$ with rotation angle θ_0:

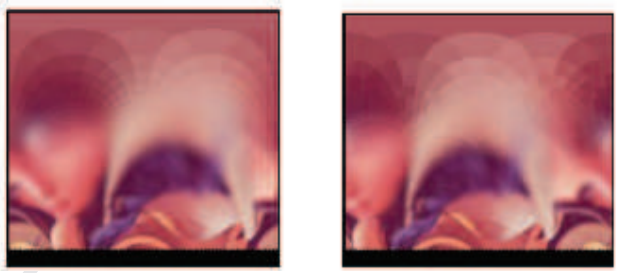
$$f_1(x,y) = f_0[(x\cos\theta_0 - y\sin\theta_0),\quad (x\sin\theta_0 + y\cos\theta_0)]\,. \tag{7}$$

If we use the log-polar coordinates:

$$\begin{aligned} x &= e^{\rho}\cos\ \theta \\ y &= e^{\rho}\sin\ \theta \end{aligned} \tag{8}$$

where $\rho = \ln\sqrt{(x-x_0)^2 + (y-y_0)^2}$ denotes the logarithm of the radial distance from the origin (x_0, y_0) of the image, $\theta = \arctan\left((y-y_0)/(x-x_0)\right)$ the polar angle in the log-polar system, e the base of the log, the origin (x_0, y_0) being located at the center of the image. Then, Eq. (7) can be rewritten as [25]:

$$f_1(\rho,\theta) = f_0[\rho,\ (\theta+\theta_0)]\,. \tag{9}$$



(a) (b)

Fig. 3. Results of Log-Polar Transform applied to Fig. 2(d) and (e)

Eq. (9) indicates that the image rotation causes a cyclical shift θ_0 along the angle axis. The results of the log-polar transform applied to the pre-processed images given in Fig. 2(d) and (e) are shown in Fig. 3(a) and (b), respectively. As it can be seen, they are similar except for a cyclical translation along the horizontal axis. When both sides of Eq. (9) are transformed by means of QDFT using Eq. (5), we obtain:

$$F_1(u,v) = e^{-2\mu\pi\left(\frac{v\theta_0}{M}\right)}F_0(u,v)\,, \tag{10}$$

where $F_0(u, v)$ and $F_1(u, v)$ respectively denote the frequency coefficients of QDFT before and after image

rotation, (u, v) denotes the frequency coordinates, and |.| represents the magnitude operator obtained by means of (3), θ_0 is rotation angle, and M denotes the size of image.

Eq. (10) indicates that $|F_1(u,v)| = |F_0(u,v)|$. In other words, the QDFT magnitudes of both $f_1(\rho, \theta)$ and its translated versions $f_0(\rho, \theta + \theta_0)$ are invariant to rotation. Due to the fact that the low-frequency components of QDFT hold the major part of the image energy and represent the main information on the image content, like for the DFT, we propose to use the magnitude of these low-frequency coefficients as features. More precisely, p magnitude coefficients of a square region from the low-frequency spectrum are selected for that purpose. For example, after the low-frequency coefficients are shifted into the center, if the center of QDFT coefficients is located at u = 129 and v = 129, and p is set to the size 5×5, then the range of selected low-frequency coefficients is: $u, v \in [127,131]$.

As stated in Ref [35], if the feature extraction part of a hashing scheme is not dependent of the knowledge of a secret key, or if it is not secured, an adversary will be able to construct a visually different image while reserving the original hash, or able to destroy the hash by introducing small distortion into the image, i.e, preserving its content. In order to secure our perceptual hash extraction, we apply the Arnold transform [33, 36], parameterized by the secret key K, so as to secretly scrambled the previously p selected QDFT coefficients.

To sum-up, both QDFT and log-polar transform are used in our scheme to capture image features onto which our image hash is built.

3.3. Hash construction

Let us represent the p QDFT magnitude coefficients of the selected and scrambled square region into a row vector $Z = [z_{(1)}, z_{(2)}, ..., z_{(p)}]$, where $z_{(i)}$ corresponds to the i^{th} QDFT magnitude coefficient of vector Z, and i =1, …, p. Each $z_{(i)}$ is determined from the square region according to the selection sequence from top to bottom and from left to right. The robust hash H_G is then generated as follows [16]:

$$H_G(i)=\begin{cases}1, & if \quad z_{(i)}-z_{(i+1)}\geq 0,\\ 0, & if \quad z_{(i)}-z_{(i+1)}<0,\end{cases} \quad i=1,...,p\text{-}1. \tag{11}$$

In order to enhance the security of H_G, this one is secretly scrambled based on a process parameterized by a secret key K_H. Without the knowledge of K_H, one will not be able to generate a valid hash. Here, we used the MatLab function randperm () to encrypt. In fact, it is a randomly permutation problem. According to the secret key, it generates a scrambled serial number. The encrypted hash is formed by the scrambled serial number.

3.4 Image authentication

In a classical scenario, the original image I_0 is transmitted along with its robust hash H_{G0} to a recipient through the Internet or a third party certification organization. At the reception, the recipient can then check the image authenticity. The received image I_1, a version of I_0 that may have undergone some content-preserving or malicious tampering operations, is utilized to generate the hash H_{G1} according to the same procedure described above. These two hashes, H_{G0} and H_{G1}, are then compared to determine whether the received image is tampered or not. The normalized Hamming distance D_1 is used in our system to measure the similarity between the two hashes:

$$D_1=d_1(H_{G0},H_{G1})=\frac{1}{p-1}\sum_{i=1}^{p-1}(Key(H_{G0}(i))\oplus Key(H_{G1}(i))), \tag{12}$$

where ' $\oplus$ ' represents the exclusive-or operation, and *Key*(.) denotes the encryption operation. If the distance D_1 is less than a pre-fixed threshold λ, the two images are considered as similar images; otherwise, they are viewed as distinct images or tampered images. The determination of the parameters of our robust image hashing scheme is analyzed in the experimental section.

Fig. 4. Examples of test image

4. EXPERIMENTAL RESULTS

4.1 Image data sets, performance criteria

To evaluate the performance of the proposed scheme, the UCID [37] image database (including 1338 color images of size 512×384 or 384×512 pixels) was selected. In all the experiments, the first 1000 images of this database were used unless stated otherwise. Some example images are displayed in Fig. 4.

The receiver operating characteristics (ROC) [38] is used to evaluate the influence of the parameters and the authentication performance in comparison with other methods. The true positive rate (P_{TPR}) and the false positive rate (P_{FPR}) of the ROC curve indicate the perceptual robustness and the discriminative capability (or anti-collision capability) of an image hash, respectively. They are defined as follows:

$$P_{TPR}(\lambda) = \frac{n_1(D_1 < \lambda)}{N_1} \quad , \qquad P_{FPR}(\lambda) = \frac{n_2(D_1 < \lambda)}{N_2} \tag{13}$$

where n_1 corresponds to the amount of pairs of visually identical images classified into similar images, n_2 is the amount of pairs of different images classified into similar images, while N_1 and N_2 denote the total number of pairs of visually identical and different images, respectively.

Table1 Content-preserving operations with different parameter values

Operations	Descriptions	Parameters
Pepper & salt noise	Noise density	0.01, 0.02, 0.03, 0.05, 0.1
Gaussian noise	Noise variance	0.001,0.005,0.01, 0.02, 0.05
Median filter	Windows size	3×3, 5×5, 7×7, 9×9, 11×11,
Average filter	Windows size	3×3, 5×5, 7×7, 9×9, 11×11,
Gaussian filter	Windows size	3×3, 5×5, 7×7, 9×9, 11×11,
Brightness adjustment	Ratio	1.5, 1.3, 1.1,0.8, 0.6
Contrast adjustment	Range of adjustment	[0.1 0.9], [0.1 0.8], [0.1 0.7], [0.2 0.9], [0.2 0.8]
Gamma correction	Gamma value	0.75, 0.8, 0.9, 1.1, 1.25
JPEG compression	Quality factor	10, 30, 50, 70, 90
Rotation	Rotation angle	2°, 10°, 30°, 45°, 210°
Scaling	Scaling factor	2, 1.5, 1.2, 0.8, 0.6
Cropping	Ratio	1%, 2%, 3%

4.2. Parameter determination

Some operations and parameters have a direct influence on the performance of our approach and need to be

determined, among which the pre-processing filter window, the number p of low-frequency coefficients used as image features, the authentication threshold λ.

Table 2 Normalized Hamming distance given in average for different sizes of pre-processing average filter window as well as for different content-preserving operations

Content-preserving operations	Para-meter	Averaging filter window of size 3×3		Averaging filter window of size 5×5		Averaging filter window of size 7×7		Averaging filter window of size 11×11	
		DD1	DD2	DD1	DD2	DD1	DD2	DD1	DD2
Pepper & salt noise	0.01	0.051	0.468	0.030	0.447	0.025	0.438	0.024	0.422
	0.02	0.073	0.466	0.042	0.445	0.035	0.439	0.031	0.423
	0.03	0.091	0.466	0.049	0.448	0.039	0.439	0.038	0.422
Gaussian noise	0.001	0.032	0.465	0.018	0.446	0.018	0.438	0.019	0.422
	0.002	0.046	0.464	0.025	0.446	0.023	0.439	0.021	0.422
	0.003	0.056	0.467	0.031	0.446	0.026	0.439	0.024	0.422
Brightness adjustment	1.3	0.058	0.466	0.059	0.445	0.059	0.440	0.061	0.421
	1.1	0.017	0.465	0.018	0.445	0.021	0.439	0.022	0.421
	0.8	0.006	0.465	0.009	0.446	0.013	0.439	0.016	0.421
Averaging filter	7×7	**0.209**	0.456	0.102	0.442	0.084	0.431	0.080	0.415
	9×9	**0.263**	0.453	0.135	0.440	0.108	0.428	0.105	0.412
	11×11	**0.305**	0.450	0.174	0.433	0.112	0.426	0.125	0.410
Median filter	3×3	0.063	0.463	0.040	0.448	0.035	0.439	0.033	0.421
	7×7	**0.195**	0.458	0.113	0.444	0.094	0.437	0.086	0.416
	11×11	**0.287**	0.455	0.183	0.438	0.105	0.431	0.133	0.413
JPEG compression	0.4	0.048	0.464	0.030	0.446	0.027	0.439	0.025	0.421
	0.2	0.070	0.464	0.047	0.447	0.039	0.439	0.035	0.422
	0.1	0.106	0.466	0.071	0.447	0.062	0.439	0.056	0.423
Rotation	5°	0.072	0.467	0.045	0.447	0.040	0.438	0.045	0.421
	30°	0.112	0.466	0.095	0.448	0.108	0.434	0.127	0.420
	45°	0.134	0.467	0.110	0.448	0.121	0.437	0.144	0.422
	120°	0.214	0.465	0.149	0.449	0.138	0.434	0.144	0.421
Average	-	0.114	0.463	0.072	0.445	0.063	0.436	0.063	0.420
DD3	-	0.349		0.367		0.373		0.357	

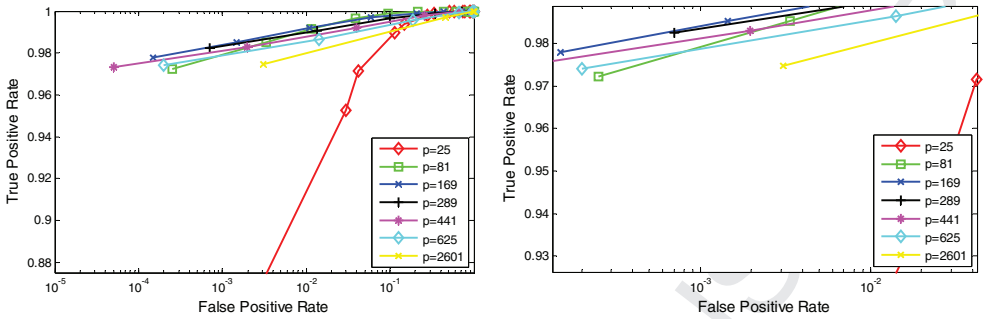
In this work, we experimentally identify the values of these parameters so as to ensure a good trade-off in terms of detection performance of our system. To do so, the first 1000 of the UCID images were used in this experiment as *original image database*, each image being submitted to 58 content-preserving operations, the parameterization of which is given in Table 1. These operations include pepper & salt noise, Gaussian noise, brightness adjustment, different filters, JPEG compression, cropping, scaling and rotation. There are thus a total of N_1=58000 visually

identical images pairs, which are looked as constituting the *visually identical image database*. The normalized Hamming distance D_1 between the original image database and the visually identical image database is computed [3]. Note that there are $N_2 = 1000\times (1000–1)/2 = 499500$ pairs of different images.

(1) *Influence of the pre-processing average filter window size (k).* To evaluate the influence of the pre-processing stage average filter and to select its suitable window size, we calculated the average Hamming distance between hashes on the whole image test set (i.e. 499500 pairs of different images and of 58000 pairs of similar images) for different window sizes k = 3, 5, 7, and 11 while considering a larger number of QDFT coefficients p (fixed to p=1000). These distances are shown in Table 2, where *DD*1 and *DD*2 denote the mean of normalized Hamming distance between hashes of visually similar images and distinct images, respectively. Small values of *DD*1 reflect a good robustness while large values of *DD*2 mean a good discriminative capability. We also introduce *DD*3 which corresponds to the absolute difference between *DD*2 and *DD*1. A large value of *DD*3 indicates that the corresponding average filter window size achieves a good trade-off between robustness and discriminative capability.

As it can be seen, if the averaging filter window is of very small size, e.g. 3×3, the value of *DD*1 is large (see bold numbers in Table 2) making our hash non-robust to content preserving operations like average and median filtering. When increasing the size of the filter window, *DD*1 values decrease whatever the image modification. The averaging filter induces an image information loss which influences the robustness of our hash. At the same time, this filtering operation also impacts the discrimination capability of our scheme as shown by *DD2* values. These values decrease when increasing the window size. However, a good tradeoff in terms of robustness and discrimination capability of our hash can be achieved by looking at *DD3*, the highest value of which *DD*3=0.373 is obtained for a window of 7×7 pixels. Based on these experiments, we decided to fix the averaging filter window size to k = 7.

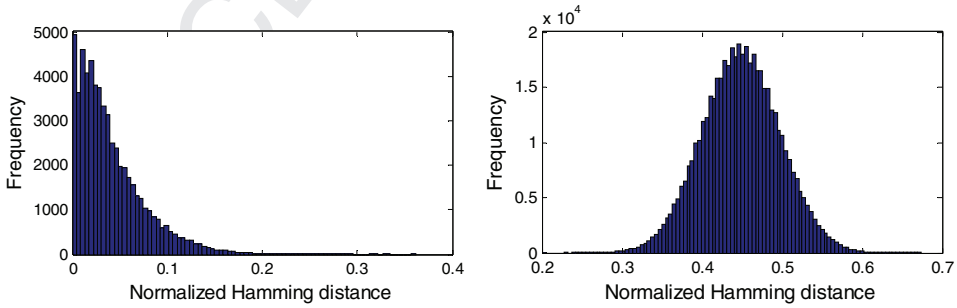
(2) *Influence of the number of selected low-frequency QDFT coefficients (p)*. As exposed in Section 3.3, the length of our robust hash depends on the parameter p. In general, the smaller the value p is, the better the perceptual robustness will be but the worse the anti-collision performance is, and vice-versa.



(a) ROC curve of different p values (b) Zoom-in view of the ROC curves

Fig. 5. ROC curves for different values p.

In order to find an appropriate trade-off, the previous test color images we built from the UCID ones were authenticated considering different values of p within the set 5^2, 9^2, 13^2, 17^2, 21^2, 25^2 and 51^2. Fig. 5 shows the corresponding ROC curves, with a zoom in Fig. 5(b) of Fig. 5(a). Slight improvements are observed when p is increasing from 81 to 289. For higher values the performance decreases. Thus, one can see that a good trade-off between the perceptual robustness and the anti-collision capability of our hash is obtained for p values in the range [169, 289]. In the rest of this paper, the value $p=15^2=225$ was retained in order to have an appropriate short hash length. Indeed, in that case, H_G is of 224 bits according to Eq. (11).



(a) Identical image pairs (b) Different image pairs

Fig. 6. Distribution of the normalized Hamming distance D_1

(3) *Authentication threshold* λ. Fig. 6(a) and (b) show respectively the Hamming distance distributions for hashes of visually identical color images and for hashes of visually different images. It can be seen that almost all

distances for different image pairs are larger than 0.2, while almost all distances between identical images are less than 0.2. This is the reason why the authentication threshold we use in the sequel is set to $\lambda = 0.2$.

4.3. Performance test

As described in the introduction, robust image hashing should contain some key properties: perceptual robustness, anti-collision capability, sensitivity and security. In this section, we examined these aspects so as to establish the performance of our hash.

(1) *Perceptual robustness.* Perceptual robustness means that both the received image and the original image have a very high probability of generating similar hash. Fig. 6(a) shows the perceptual robustness of the proposed method considering the above parameterization. It can be seen that only a few image pairs (about 0.2%) are looked as different images when λ is set to 0.2. Notice that these misclassified images are due to strong parameter attacks such as pepper and salt noise with density 0.1 and a median filter with a window size 11×11. We further use the well-known images such as Lena, Airplane, Pepper, Mandrill and Sailboat and lake to show the robustness. Almost all attacks described in Table 1 were applied. The normalized Hamming distance values corresponding to each operation are displayed in Fig. 7. It can be seen that the normalized Hamming distance $DD1$ is less than 0.2. This perceptual robustness can be achieved for three main reasons: (i) the generated hash utilizes the low-frequency QDFT coefficients whose interrelation is hardly changed by content-preserving operations; (ii) the average preprocessing filter makes the method more robust against content-preserving operations such as noise, filtering operation and JPEG compression; (iii) the magnitude of QDFT coefficients of the log-polar transformed image is invariant to rotation. The proposed method is thus perceptually robust not only to geometric attacks but also to content-preserving operations.

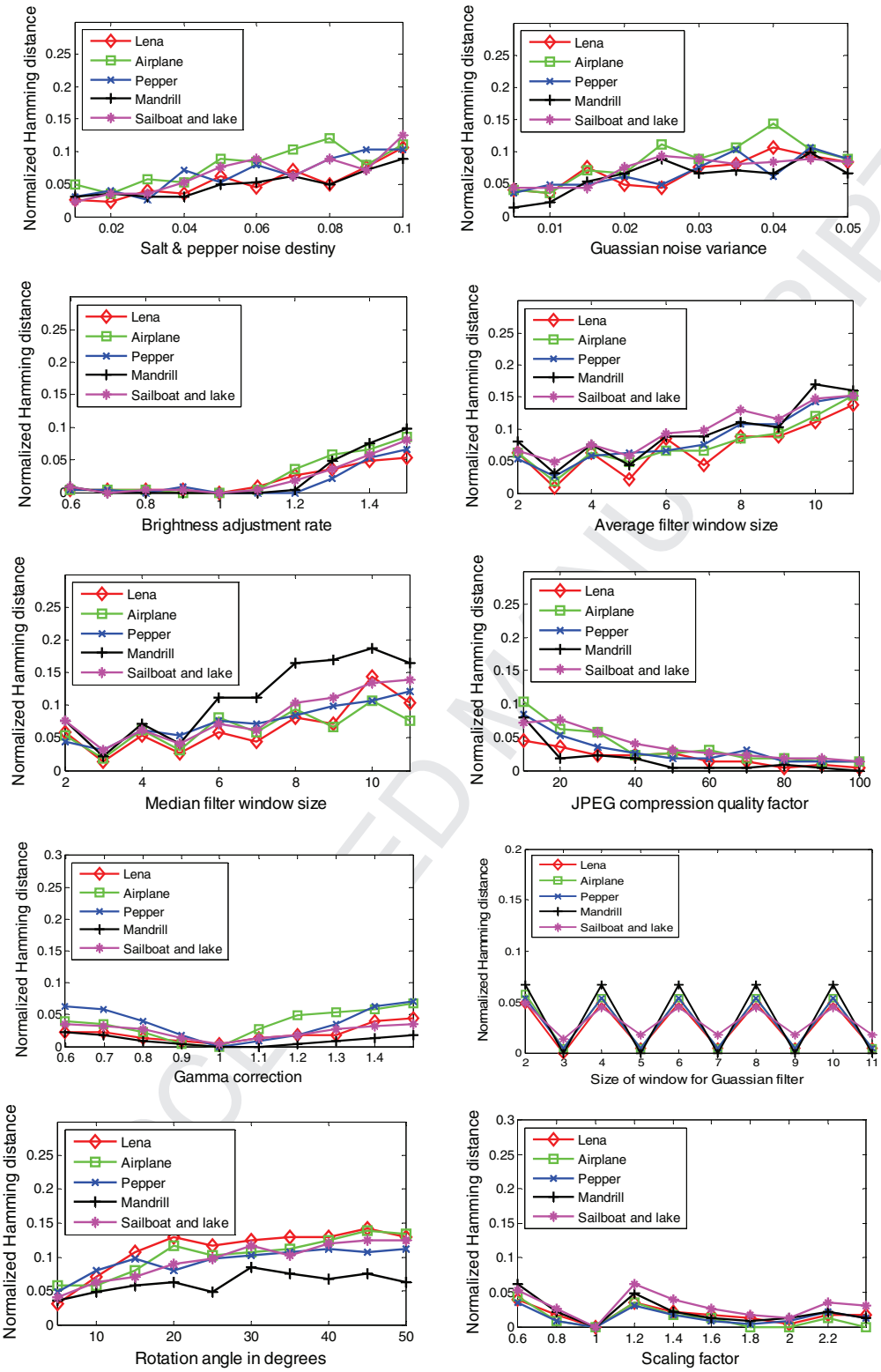


Fig. 7. The distribution of distances *DD1* in different content- preserving operations

(2) *Anti-collision capability.* Anti-collision or discriminative capability means that two visually distinct images

have a very low probability of generating similar hash. If the distance *DD*2 of two distinct images is less than the threshold λ, the collision occurs. According to Fig. 6(b), the collision probability we achieved over our whole image test set is close to zero when the threshold λ is set to 0.2 since all distance pairs of different image are more than 0.2, indicating good anti-collision performance of our scheme.

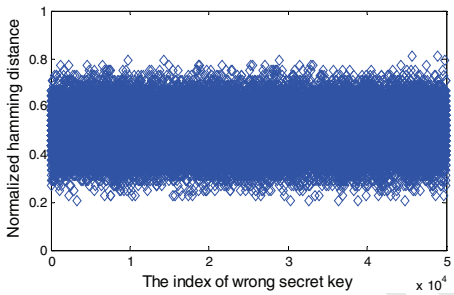


Fig. 8. Distribution of normalized Hamming distance when using 50000 wrong secret keys

(3) *Security.* To validate the security of the system, we tested 50000 wrong secret keys. The hash distance distribution of image pairs between right and wrong secret keys K_H is shown in Fig. 8. We observe that all these distances are higher than 0.2. If one does not know the secret key and the hash algorithm, the probability of generating the same hash is about $1/2^{224}$ since our hash length is 224 bits. If one does not know the secret key but know the hash algorithm, it is a special permutation problem of 224 bits. Therefore, the proposed method meets also the security requirements.

(4) *Sensitivity.* When some tampering operations are applied to an image, the corresponding image hash should be distinct from the original image one. To validate the sensitivity of the proposed method to such visual content alterations, we have simulated two kinds of tampering operations. They rely on copying a portion of a "source image" and pasting it into a "destination image" which can be same as the source image ("copy-move attack") or another image ("cut-and-paste attack"). As illustrated in the second and third row of Fig. 9, the upper left corner of the pasted area corresponds to the destination image center. We have considered different sizes of the copied area so as it correspond to 4%, 9%, 16% and 25% of the destination image dimensions. It can be seen from Fig. 9 that for such kind of attack the resulting normalized Hamming distances *DD*2 are higher than 0.2. As shown, our

method is sensitive to content alterations taking advantage of the fact that the magnitude of QDFT coefficients contains both the luminance and the chrominance information.

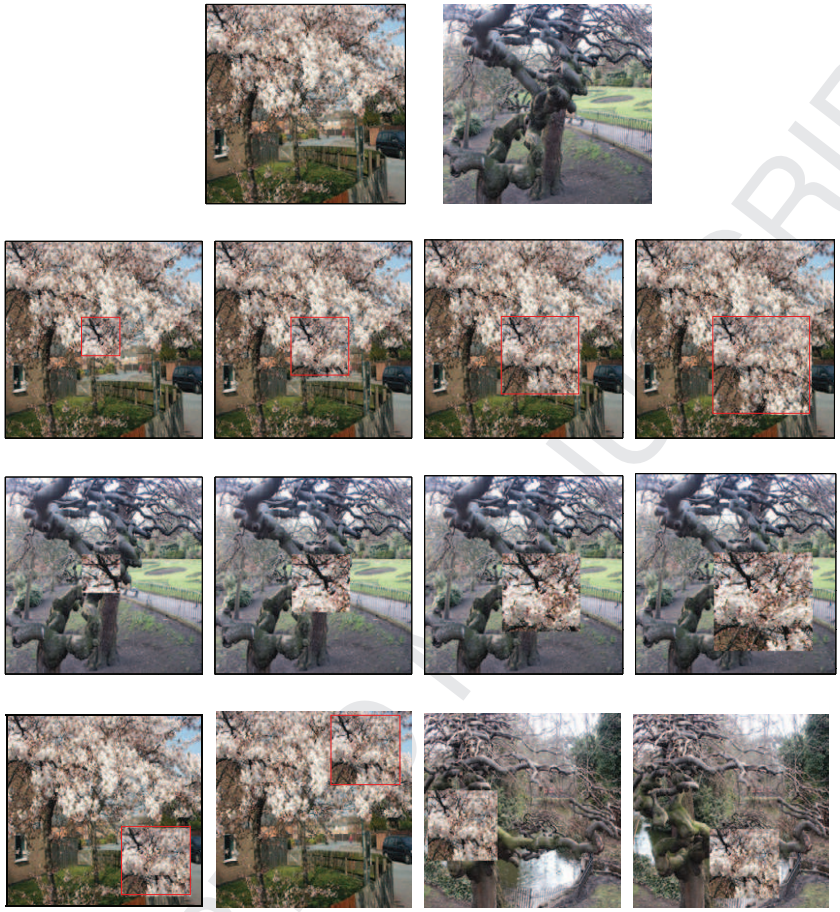


Fig. 9. Examples of original and tampered images. Top: original images. Second row: tampered images after a copy-move attack of size 4%, 9%, 16%, 25%, obtained normalized Hamming distance are *DD2*=0.3259, 0.3527, 0.3839, 0.4107, respectively. Third row: tampered image after a cut-and-paste attack of size 4%, 9%, 16%, 25%, obtained normalized Hamming distance values are *DD2*=0.2902, 0.4107, 0.4107, 0.4241, respectively. Fourth row: tampered image after a tampering rate of 10%. The resulting normalized Hamming distance values are *DD2*=0.2679, 0.2143, 0.2455, 0.3438, respectively.

To further validate the sensitivity of our hash to visual content alterations, the first 1000 images from the UCID database were also tested. Similarly to the above copy-move and cut-and-paste attacks, tampering ratios that were considered are of 4%，9%，16%，25%，36% and 49%, leading thus to 6000 normalized Hamming distances shown in Fig. 10. We observe that all distances are larger than 0.2 except for one image where the tampering ratio is equal to 4%. This confirms that the proposed method is sensitive to content alterations when the tampering ratio is higher

than 4%. In addition, we have also conducted experiments with a tampering position near the boundary (a tampering ratio about 10% was chosen) as shown in the fourth row of Fig. 9. We observe that such operation leads to a sensitivity decrease (*DD2* is higher than 0.2 but close).

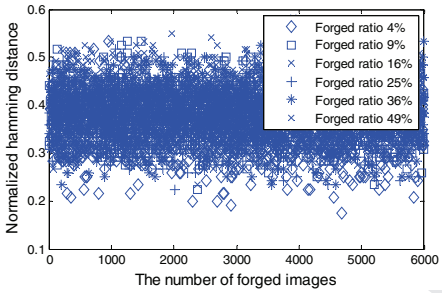


Fig. 10. Distribution of normalized Hamming distances for 6000 different tampering image pairs

(a) (b) (c)

(d) (e) (f)

Fig. 11. Examples of original images and image region color changing. Top: original images. Second row: tampered images based on color change, where the obtained normalized Hamming distance values are *DD2*=0.0982, 0.1339, 0.2634, respectively.

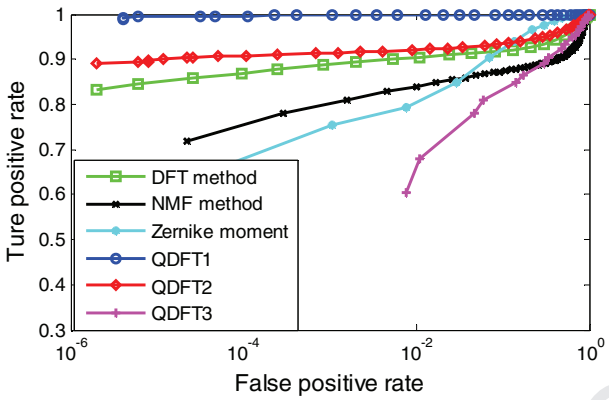
Image color change is another tampering operation. This change can operate in two ways. The first one does not change the image content as simulated in Fig. 11(d) and (e). Their normalized Hamming distance values are 0.0982 and 0.1339, respectively. The second one modifies the image content as depicted in Fig. 11(f); the corresponding normalized Hamming distance value is 0.2634. Experimental results show that our scheme is insensitive to the first kind of region color changing ways; this is not the case for the second kind of tampering.

4.4. Performance comparison

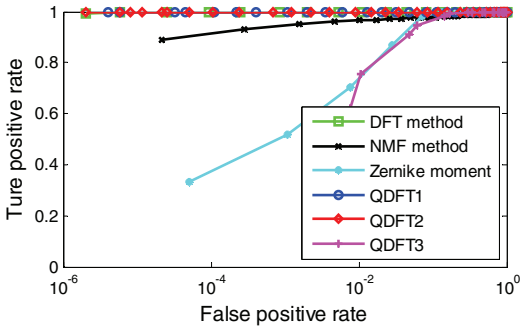
4.4.1 Robustness and anti-collision performance comparison

In order to better evaluate the performance of our QDFT based scheme, we compared it with different methods based on DFT [16], QDFT [34], Zernike moments [3] and NMF [19]. On one hand, these methods represent the current state of the art and, on the other hand, their authors suggested working on color images. In order to further examine the influence of the log-polar transform on the performance of scheme; three QDFT based schemes were compared. The first one, denoted by QDFT1, corresponds to the method already described. The second one, denoted by QDFT2, is identical to QDFT1 except that the log-polar transform is omitted. The parameters used for these two methods are the same. The QDFT method reported in [34] is referred to as QDFT3. As previously, the first 1000 images of the UCID image database were used to generate similar image pairs and distinct image pairs.

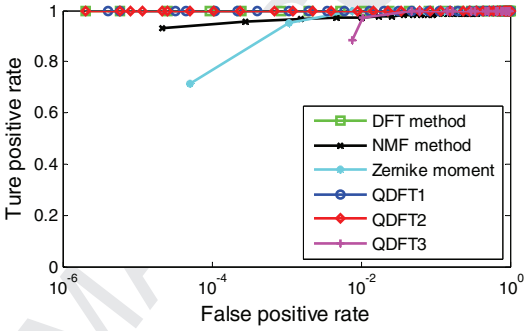
The comparison is worked out using ROC curve, which is a good measure for a fair comparison since it allows comparing methods based on different distance measurement criterion (i.e. Zernike moments [3] use Euclidean distance, NMF [19], the correlation coefficients, DFT [16], QDFT [34] and our scheme, the Normalized Hamming distance). The ROC curves are displayed in Fig. 12 (a). It can be seen that both QDFT1 and QDFT2 leads to a better behavior than the other hashing methods, QDFT3 shows the worst performance. The main reason stands on the way this hash is constructed. Basically, it directly divides the frequency domain into block of size 8×8, and then generates the hash component value by comparing the average energy of block with the one of the whole image. But due to the fact about 9% of low-frequency coefficients carry more than 99.5% of the image energy as stated in [35], most of hash component values are ‘0’, and only a few ones have value ‘1’. This leads to a very bad anti-collision capability and also impacts the overall performance of this scheme.



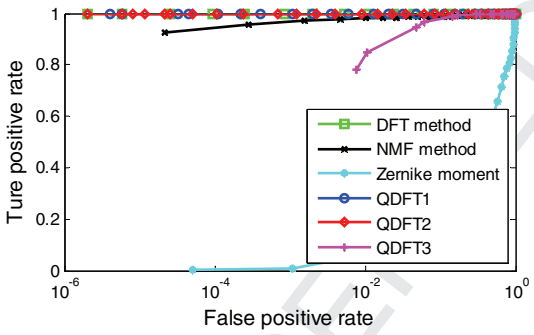
(a) Overall ROC curves



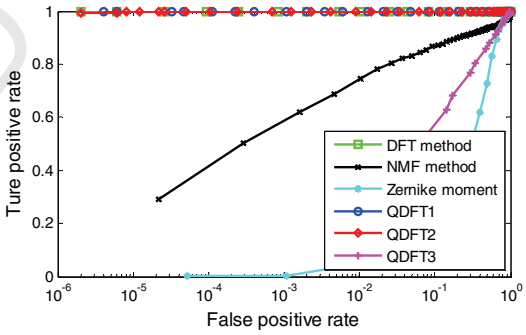
(b) Salt and Pepper noise



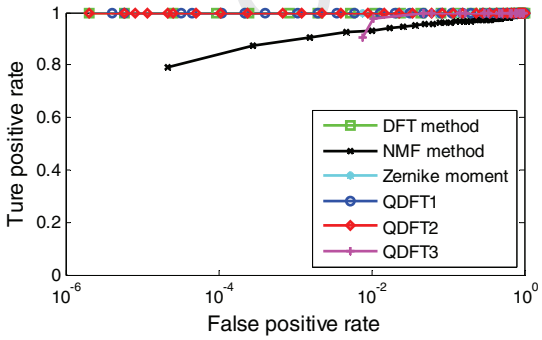
(c) Gaussian noise



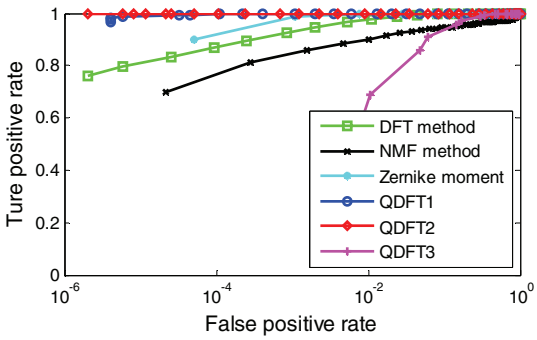
(d) Brightness adjustment



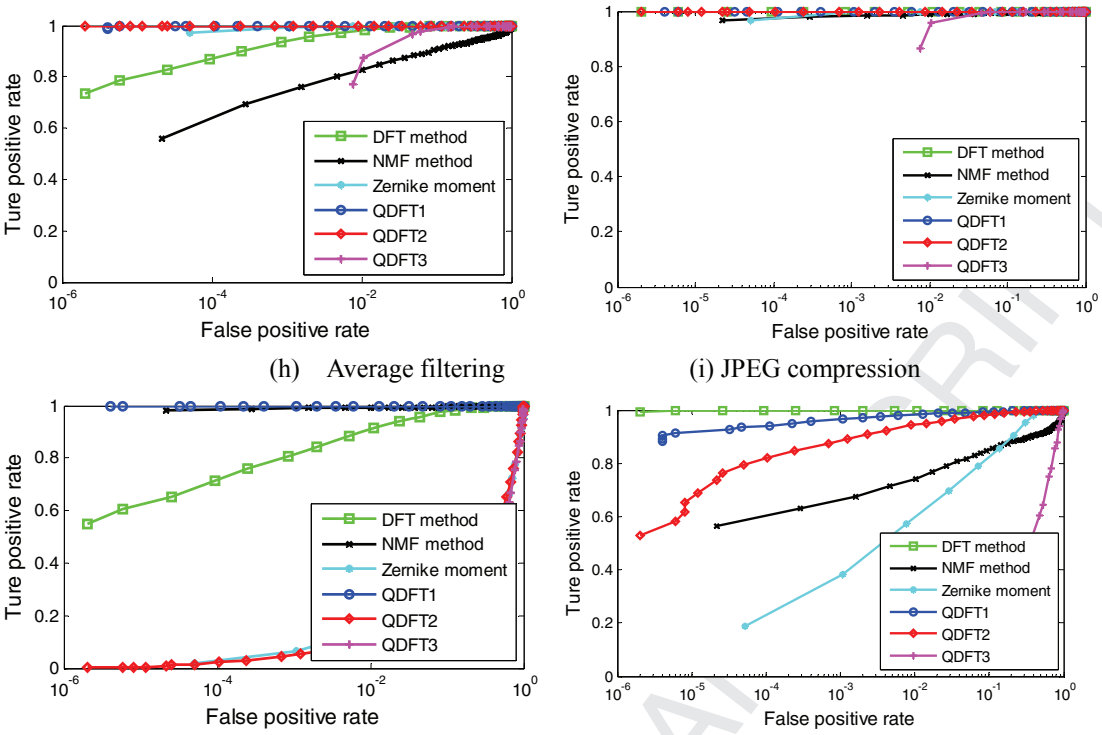
(e) Contrast adjustment



(f) Gaussian filtering



(g) Median filtering



(h) Average filtering (i) JPEG compression

(j) Rotation (k) Cropping

Fig. 12. Comparison of the methods through the ROC curves for different tampering operations

To show clearly the advantages and disadvantages of these methods for different content-preserving operations, Fig. 12(b)-(k) show the ROC curves corresponding to each operation. From these results, it appears that these methods are robust to some content-preserving attacks such as Gaussian noise, Gaussian filtering, average filtering, JPEG compressing, etc. However, our method behaves better on the overall. For instance, Fig. 12(d)-(e) shows that the Zernike moment method is sensitive to brightness and contrast adjustment. Fig. 12(e), (g), (j) and (k) show that the QDFT3 is not robust to contrast adjustment, median filtering, rotation and cropping attack. For the rotation operations illustrated in Fig. 12(j), the NMF method and QDFT1 method obtain better performance than the other four methods. When comparing QDFT1 and QDFT2, the former shows better robustness to rotation and cropping attacks since it employs the log-polar transform in feature extraction process. Thus, the proposed method is robust to almost all content-preserving operations. This is due to the joint combination of (*i*) the low-frequency magnitude coefficients of QDFT, (*ii*) the log-polar transform of the image central part, and (*iii*) the filtering operation

enhances its robustness and increases its discriminative capability.

4.4.2 Sensitivity comparison

To do that, we have built three test image sets including three tampering situations: copy-move, cut-and-pasting and region color changing. Since it is a tedious and time-consuming work to generate the tampering images using Photoshop soft, the copy-move and cut-and-pasting tampering images were automatically generated by computer. The first test set (Data 1) contains 2000 tampering images as used in subsection 4.3 for copy-move and cut-and-pasting tampering operations. The second test set (Data 2) differs from the first test set by using the 10% ratio of tampering areas with a randomly selected position of pasting. The third test set (Data 3) contains 150 tampering images. 50 region color changing tampering images were generated by Photoshop, another 100 tampering images were selected from other two data sets (each one including 50 images).

Fig. 13 summarizes the results obtained. For Data 1, the proposed method provides the best rate of recognition with a better sensitivity to slight tampering area (4%) than the other three methods. For Data 2, the DFT method [19] leads to the best results. It remains insensitive to tampering position changes when the tampering area is larger than 10%. For Data 3, the Zhao's method [3] provides the best rate of recognition in the case of region color changing.

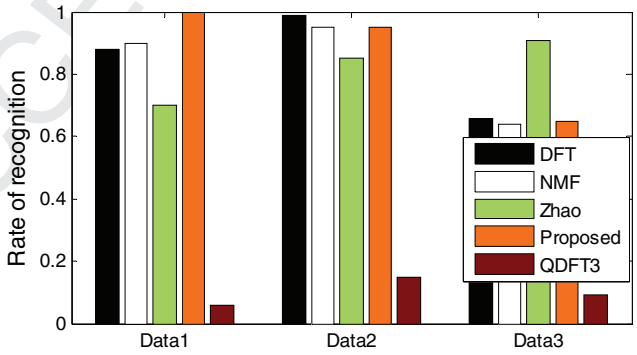


Fig. 13 Sensitivity comparison for different methods

4.4.3 Time complexity and qualitative comparison

To synthetize the comparison of our method and other hashing methods, a qualitative overview with the current

state of art is given in Table 3. We observe that the hash of our method has the shortest length and that it is robust against various content-preserving operations and any angle rotation operations. We also consider the average time required for generating a hash value in our experimental environment (i.e. CPU 3.2 GHz, 4G memory and running MatLab 7.12). Since the local features of Zhao's method [3] are mainly used to classify and locate a tampered region, the time computation of the global features generating the hash is only considered. It can be seen that the time running for the five methods is very short and less than 0.22 s. The DFT method [19] is the fastest and needs only 0.07 s.

Table 3 Overall performance comparison of hashing algorithms

	DFT [16]	Zhao [3]	NMF [19]	QDFT[34]	Proposed
Hash length	444bits	560 bits	64 digits	256bits	224bits
Robustness against slight noise, filter, JPEG	Yes	Yes	Yes	yes	Yes
Robustness against large contrast adjustment	Yes	No	No	No	Yes
Robustness against any angle rotation	No	No	Yes	No	Yes
Capability of localization and classification	No	Yes	No	No	No
Average time for generating a hash (seconds)	0.07	0.19	0.21	0.1	0.18

5. CONCLUSION

In this paper, we have developed a novel robust image hashing method devoted to color images using QDFT and log-polar transforms. By exploiting the QDFT into the log-polar domain, we better take into account the three image color planes and the resulting hash is robust to rotation attacks as well as to common image content-preserving operations. The parameters' values of our approach have been determined by means of intensive experiments on a large image data set so as to establish a good tradeoff in terms of hash robustness against content-preserving operations (JPEG compression, brightness adjustment, average and median filtering and large angle rotation attacks) and a good sensitivity to non-authorized image content alterations (copy-move and copy-paste attacks). Compared to the current state of art, for a hash length of 224 bits our scheme provides on the overall a better performance. Our future work will focus on designing a robust hash capable to locate tampered

regions of an image as well as to determine the type of the tampering.

Acknowledgments

This work was supported by the National Basic Research Program of China under Grant 2011CB707904, the National Natural Science Foundation of China under Grants 61073138, 61103141, 61271312, 61201344, the Research Fund for the Doctoral Program of Higher Education of Ministry of Education of China under Grants 20110092110023 and 2012009212003, the Qing Lan Project, DZXX-031, BY2014127-11 of Jiangsu province, and the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant 13KJB520015.

REFERENCES

[1] F. Ahmed, M. Siyal, V.U. Abbas, A secure and robust hash-based scheme for image authentication, Signal Process. 90 (5) (2010) 1456-1470.

[2] Y. Lei, Y. Wang, J. Huang, Robust image hash in Radon transform domain for authentication, Signal Process-Image Comm. 26 (6) (2011) 280-288.

[3] Y. Zhao, S. Wang, X. Zhang, H. Yao, Robust hashing for image authentication using Zernike moments and local features, IEEE Trans. Inf. Forensics Secur. 8 (1) (2013) 55-63.

[4] W. Lu, M. Wu, Multimedia forensic hash based on visual words, in: Image Processing (ICIP), IEEE International Conference on, 2010, pp. 989-992.

[5] S. Battiato, G.M. Farinella, E. Messina, G. Puglisi, A robust forensic hash component for image alignment, Image Analysis and Processing–ICIAP 2011, Springer, 2011, pp. 473-483.

[6] Y. Liu, F. Wu, Y. Yang, Y.T. Zhuang, A. G. Hauptmann, Spline regression hashing for fast image search. IEEE Trans. Image Process. 21(10) (2012) 4480-4491.

[7] X. Zhu, Z. Huang, H. Cheng, J. Cui, H. T. Shen, Sparse hashing for fast multimedia search, ACM Trans. on Information Systems. 31(2) (2013) 9:1-9:24.

[8] J. Fridrich, and M. Goljan, Robust hash functions for digital watermarking, in: Proc. IEEE International Conference on Information Technology: Coding and computing. Las Vega, Nevada, USA, 2000, pp. 178-183.

[9] J. Cannons, P. Moulin, Design and statistical analysis of a hash-aided image watermarking system, IEEE Trans. Image Process. 13 (10) (2004) 1393-1408.

[10] A. Swaminathan, Y. Mao, and M. Wu, Robust and secure image hashing, IEEE Trans. Inf. Forensics Secur. 1 (2) (2006) 215-230.

[11] C.D. Roover, C.D. Vleeschouwer, F. Lefèbvre, B. Macq, Robust video hashing based on radial projections of key frames, IEEE Trans. Signal Process. 53(10) (2005) 4020-4037.

[12] D. Wu, X. Zhou, X. Niu, A novel image hash algorithm resistant to print–scan, Signal Process. 89 (12) (2009) 2415-2424.

[13] Z. Tang, X. Zhang, Y. Dai, W. Lan, Perceptual image hashing using local entropies and DWT, Imaging Sci. J. 61(2) (2013) 241-251.

[14] Z. Tang, Y. Dai, X. Zhang, L. Huang, F. Yang, Robust image hashing via colour vector angles and discrete wavelet transform, IET Image Process. 8 (3) (2014) 142-149.

[15] F. Liu, L.M. Cheng, H.Y. Leung, Q.K. Fu, Wave atom transform generated strong image hashing scheme, Opt. Comm. 285(24) (2012) 5008-5018.

[16] C. Qin, C.C. Chang, P.L. Tsou, Robust image hashing using non-uniform sampling in discrete Fourier domain, Digital Signal Process. 23 (2) (2013) 578-585.

[17] S.S. Kozat, R. Venkatesan, M.K. Mihçak, Robust perceptual image hashing via matrix invariants, in: Proc. the IEEE International Conference on Image Processing, Singapore, 2004, pp. 3443-3446.

[18] V. Monga, M. Mhcak, Robust and secure image hashing via non-negative matrix factorizations, IEEE Trans Inf. Forensics Secur. 2 (3) (2007) 376-390.

[19] Z. Tang, X. Zhang, L. Huang, Y. Dai, Robust perceptual image hashing based on ring partition and NMF, IEEE Trans. Knowl. Data. En. 26 (3) (2014) 711-724.

[20] S. Xiang, H.J. Kim, J. Huang, Histogram-based image hashing scheme robust against geometric deformations, in: Proceedings of the 9th workshop on Multimedia & security, 2007, pp. 121-128.

[21] S. Battiato, G.M. Farinella, E. Messina, G. Puglisi, Robust image alignment for tampering detection, IEEE Trans. Inf. Forensics Secur. 7 (4) (2012) 1105-1117.

[22] Y. Li, Z. Lu, C. Zhu, X. Niu, Robust Image Hashing Based on Random Gabor Filtering and Dithered Lattice Vector Quantization, IEEE Trans. Image Process. 21 (4) (2012) 1963-1980.

[23] Z. Tang, X. Zhang, L. Huang, Y. Dai, Robust image hashing using ring-based entropies, Signal Process. 93 (7) (2013) 2061-2069.

[24] X. Lv, Z.J. Wang, Perceptual Image Hashing Based on Shape Contexts and Local Feature Points, IEEE Trans. Inf. Forensics Secur. 7 (3) (2012) 1081-1093.

[25] Q. Wang and Z. Wang, Color image registration based on quaternion Fourier transformation, Opt Eng. 51 (5) (2012) 057001-057008.

[26] I.L.v. Kantor, A.S. Solodovnikov, A. Shenitzer, Hypercomplex numbers: an elementary introduction to algebras, Springer-Verlag, New York, 1989.

[27] S.J. Sangwine, Fourier transforms of colour images using quaternion or hypercomplex, numbers, Electronics letters. 32(21) (1996) 1979-1980.

[28] T.A. Ell, Quaternion-Fourier transforms for analysis of two-dimensional linear time-invariant partial differential system, in: Proceedings of the 32nd IEEE Conference on Decision Control, 1993, pp. 1830-1841.

[29] S.C. Pei, J.J. Domg, J.H. Chang, Efficient Implementation of Quaternion Fourier Transform, Convolution, and Correlation by 2-D Complex FFT. IEEE Trans. Signal process. 49 (11) (2001) 2783-2797.

[30] T.A. Ell, S.J. Sangwine, Hypercomplex Fourier transforms of color images, IEEE Trans. Image Process. 16(1) (2007) 22-35.

[31] T.K. Tsui, X.P Zhang, D. Androutsos, Color image watermarking using multidimensional Fourier transforms, IEEE Trans. Inf. Forensics Secur. 3 (1) (2008) 16-28.

[32] F.n. Lang, J.l. Zhou, S. Cang, H. Yu, Z. Shang, A self-adaptive image normalization and quaternion PCA based color image watermarking algorithm, Expert Syst. Appl. 39 (15) (2012) 12046-12060.

[33] B.J. Chen, G. Coatrieux, G. Chen, X.M. Sun, J.L. Coatrieux, H.Z. Shu, Full 4-D quaternion discrete Fourier transform based watermarking for color images, Digital Signal Process. 28 (1) (2014) 106-119.

[34] I. H. Laradji, L. Ghouti, E-H Khiari, Perceptual hashing of color images using hypercomplex representations, in: Image Processing (ICIP), IEEE International Conference on, 2013, pp. 4402-4406.

[35] S. Wang and X. Zhang, Attacks on perceptual image hashing, in: Proceedings of the 2nd International Conference on Ubiquitous Information Technologies and Applications, 2007, pp. 199-203.

[36] V.I. Arnold, A. Avez, Ergodic problem of classical mechanics, Mathematic physics monograph series, New York, 1968.

[37] G. Schaefer, M. Stich. UCID - An Uncompressed Colour Image Database, in: Proceedings of SPIE, Storage and Retrieval Methods and Applications for Multimedia. 2004, pp. 472-480.

[38] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. 27(8) (2006) 861–874.

Junlin Ouyang is pursuing his Ph.D. Degree at School of Computer Science and Engineering, Southeast University, China. He is a lecturer in the School of Computer Science and Engineering, Hunan University of Science and Technology, China, in 2008. His main research interests include information security, image retrieve and pattern recognition.

Gouenou Coatrieux received the Ph.D. degree in Signal Processing and Telecommunication from the University of Rennes (France), in 2002. He is currently a Professor in the Information and Image Processing Department, Institute Mines-Telecom, Telecom Bretagne, Brest, France. His primary research interests concern medical information system security, watermarking, electronic patient records, and healthcare knowledge management.

Huazhong Shu received the Ph.D. degree in Numerical Analysis from the University of Rennes (France) in 1992. He is now a Professor of the Department of Computer Science and Engineering of Southeast University, China. His recent work concentrates on the image analysis, pattern recognition, and fast algorithms of digital signal processing.