CMC, vol.55, no.2, pp.331-344, 2018

Robust Image Hashing via Random Gabor Filtering and DWT

Zhenjun Tang[1, *], Man Ling[1], Heng Yao[1], Zhenxing Qian[2], Xianquan Zhang[1], Jilian Zhang[3] and Shijie Xu[1]

Abstract: Image hashing is a useful multimedia technology for many applications, such as image authentication, image retrieval, image copy detection and image forensics. In this paper, we propose a robust image hashing based on random Gabor filtering and discrete wavelet transform (DWT). Specifically, robust and secure image features are first extracted from the normalized image by Gabor filtering and a chaotic map called Skew tent map, and then are compressed via a single-level 2-D DWT. Image hash is finally obtained by concatenating DWT coefficients in the LL sub-band. Many experiments with open image datasets are carried out and the results illustrate that our hashing is robust, discriminative and secure. Receiver operating characteristic (ROC) curve comparisons show that our hashing is better than some popular image hashing algorithms in classification performance between robustness and discrimination.

Keywords: Image hashing, Gabor filtering, chaotic map, skew tent map, discrete wavelet transform.

1 Introduction

Image hashing is a useful technology of multimedia [Tang, Zhang and Zhang (2014); Qin, Chen, Dong et al. (2016)]. It can create a fixed-size string called image hash from input image. As image hash is a short representation based on visual content of image, it can be used to denote input image in practice. Actually, it has been used in various applications [Tang, Zhang, Huang et al. (2013); Qin, Ji, Zhang et al. (2017); Cao, Zhou, Sun et al. (2018)], such as image authentication, image retrieval, image indexing, image copy detection, image forensics, image quality assessment and digital watermarking. In general, image hashing should meet three properties [Qin, Chang and Tsou (2013); Tang, Zhang, Li et al. (2016)]: robustness, discrimination and security. Robustness means that image hashing should create the same or very similar hashes if two images are visually similar no matter their digital representations are same or not. This is because a digital image may be manipulated by some digital operations, e.g. JPEG compression and image enhancement. After these operations, the processed image is visually similar with its original image. But

[1] Guangxi Key Lab of Multi-Source Information Mining & Security, Guangxi Normal University, Guilin, 541004, China.

[2] Shanghai Institute of Intelligent Electronics & Systems, School of Computer Science, Fudan University, Shanghai, 200433, China.

[3] School of Information Systems, Singapore Management University, 178902, Singapore.

[*] Corresponding author: Zhenjun Tang. Email: tangzj230@163.com.

CMC. doi:10.3970/cmc.2018.02222

their digital representations are different. In other words, image hashing should be robust to digital operations. Discrimination implies that image hashing should map different images to different hashes. This property is very important because the number of different images is much bigger than that of similar images. Security means that image hashing should be key-dependent. For the same image, if the input keys are different, the generated hashes must be different. This property can avoid hash forgery and is helpful for the applications of image authentication and image forensics.

Many researchers have designed some useful image hashing algorithms. For example, Monga et al. [Monga and Evans (2006)] extracted visually significant points by end-stopped wavelet transform and used them to construct hash. This hashing is robust to JPEG compression and image rotation with small angle. Xiang et al. [Xiang, Kim and Huang (2007)] proposed to calculate hash by exploiting global histogram. This method is robust to large-angle rotation. Monga et al. [Monga and Mihcak (2007)] exploited non-negative matrix factorization (NMF) to extract image hash. This NMF-based algorithm can resist many digital operations, but it is sensitive to watermark embedding. In another work, Tang et al. [Tang, Wang, Zhang et al. (2011)] proposed a novel lexicographical framework for hash generation and designed an algorithm with DCT and NMF. A common weakness is the fragileness of large-angle rotation. Li et al. [Li, Lu, Zhu et al. (2012)] exploited random Gabor filtering (GF) and dithered lattice vector quantization (LVQ) to design hashing. The GF-LVQ hashing has good robustness and security, but its discrimination should be improved. Laradji et al. [Laradji, Ghouti and Khiari (2013)] used Quaternion Fourier Transform (QFT) to construct image hash, but the classification performance of this scheme must be improved.

Recently, Tang et al. [Tang, Lao, Zhang et al. (2016)] designed a novel image hashing with innovative use of discrete cosine transform (DCT) and local linear embedding (LLE). This algorithm can resist normal digital operations and has good discrimination. Huang et al. [Huang, Liu, Wang et al. (2016)] proposed a novel hashing method by random walk on zigzag blocking. This hashing has good security, but its classification performance must be improved. Qin et al. [Qin, Chen, Ye et al. (2016)] exploited block truncation coding to design image hashing. This hashing reaches good perceptual robustness. Vadlamudi et al. [Vadlamudi, Vaddella and Devara (2016)] divided input image into non-overlapping blocks, distributed block-based histogram bins into large containers, and calculated the ratio of pixel count between two neighboring containers. The hashing is robust against digital operations, but its discrimination is not good enough. Tang et al. [Tang, Li, Song et al. (2017)] exploited multidimensional scaling (MDS) to design robust hashing. This method is also robust against digital operations. Qin et al. [Qin, Sun and Chang (2018)] proposed to calculate image hash with hybrid feature extraction. In another study, Qin et al. [Qin, Chen, Luo et al. (2018)] exploited dual-cross pattern encoding and salient structure detection to construct image hash. This method can resist JPEG compression and rotation with small angle.

In this paper, we exploit random Gabor filtering and wavelet transform [Gurusamy and Subramaniam (2017)] to design a robust image hashing. Our hashing not only has good robustness and discrimination, but also reaches good security. Many experiments are carried out to evaluate our performance. Receiver operating characteristic (ROC) curve comparisons show that our hashing is better than some popular hashing algorithms in classification between robustness and discrimination. The rest of this paper is organized as follows. Section 2 introduces our image hashing and Section 3 discusses experimental results. Conclusions are given in the Section 4.

2 Proposed image hashing

Our image hashing includes three parts: preprocessing, random Gabor filtering and discrete wavelet transform (DWT), as shown in Fig. 1. The step of preprocessing is to generate a normalized image. The step of random Gabor filtering is to extract robust and secure image features. The step of DWT is to compress the extracted features and make a short hash. The following sections will explain these steps in detail.

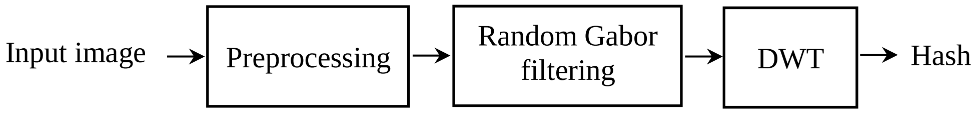


Figure 1: Block diagram of the proposed hashing

2.1 Preprocessing

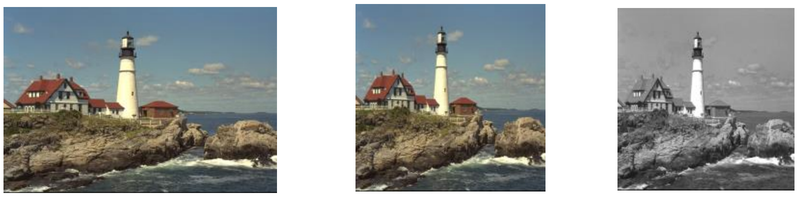
The step of preprocessing consists of two operations: bi-linear interpolation and color space conversion. The bi-linear interpolation converts input image to a fixed size *M*×*M*. This operation is to ensure that our hashing can resist image scaling and hashes of digital images with different sizes have the same length. For RGB color image, color space conversion is conducted. Here, the RGB color space is mapped to HSV color space. HSV color space is also called hexagonal cone model, which is introduced by A. R. Smith in 1978. In this model, a pixel is described by its hue, saturation and value. Let *H*, *S*, and *V* be the hue, saturation and value of a pixel, respectively. Thus, they can be determined by the following equations.

$$H = \begin{cases} \dfrac{(-B+G)*\pi/3}{\mathrm{Max}(R,G,B)-\mathrm{Min}(R,G,B)}, & \text{If } R = \mathrm{Max}(R,G,B) \\ \dfrac{(-R+B)*\pi/3}{\mathrm{Max}(R,G,B)-\mathrm{Min}(R,G,B)}, & \text{If } G = \mathrm{Max}(R,G,B) \\ \dfrac{(-G+R)*\pi/3}{\mathrm{Max}(R,G,B)-\mathrm{Min}(R,G,B)}, & \text{If } B = \mathrm{Max}(R,G,B) \\ \text{Undefined}, & \text{If } R = G = B \end{cases} \tag{1}$$

$$S = \begin{cases} \dfrac{\mathrm{Max}(R,G,B)-\mathrm{Min}(R,G,B)}{\mathrm{Max}(R,G,B)}, & \text{If } \mathrm{Max}(R,G,B) \neq 0 \\ 0, & \text{If } \mathrm{Max}(R,G,B) = 0 \end{cases} \tag{2}$$

$$V = \mathrm{Max}\ (R, G, B) \tag{3}$$

where *R*, *G* and *B* are the red, green and blue components of a pixel, Max(*R*, *G*, *B*) and Min(*R*, *G*, *B*) represent the maximum and minimum values of *R*, *G* and *B*, respectively. In this paper, we exploit the *V* component of HSV color space to calculate image hash. This is based on the following consideration. For many reported hashing algorithms, HSV color space outperforms YCbCr color space and CIE L*a*b* color space in making good classification performance [Tang, Li, Song et al. (2017)]. Fig. 2 is an example our preprocessing, where (a) is an RGB color image, (b) is the resized image and (c) is the **V** component.



(a) Input image (b) Resized image (c) **V** component

Figure 2: An example of the preprocessing

2.2 Random Gabor filtering

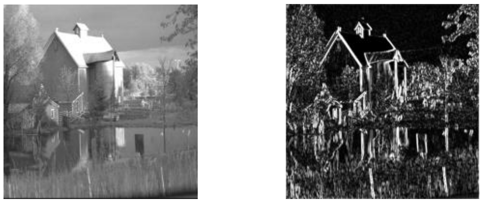
Gabor filter is a useful technique of image processing. Since its frequency and orientation representations are similar with those of human visual system, it has been successfully used in many applications, such as edge detection, texture analysis, face recognition, and retina recognition. In this paper, we select Gabor filter to conduct feature extraction. This is based on the following considerations. For the feature extraction, Gabor filter can simultaneously provide inspection of images in the spatial and frequency domains [Kamarainen, Kyrki and Kälviäinen (2002)]. In addition, Gabor filter has high robustness against distortion, which helps to make a robust hash. In the spatial domain, the 2-D Gabor filter is a Gaussian kernel function modulated by a sine wave. The kernel function of 2-D Gabor filter is defined as follows.

$$G(x,y,\lambda,\theta,\sigma)=\frac{1}{2\pi\sigma^2}\exp\left[-\pi\left(\frac{x_\theta}{\sigma}\right)^2-\pi\left(\frac{y_\theta}{\sigma}\right)^2\right]\exp(\frac{2\pi i x_\theta}{\lambda}) \tag{4}$$

$$x_\theta = x\cos\theta - y\sin\theta \tag{5}$$

$$y_\theta = x\sin\theta + y\cos\theta \tag{6}$$

where *x* and *y* denote the size of the filter window in the *x*-coordinate and *y*-coordinate directions, λ is the wavelength of the sine wave, σ is the standard deviation of the Gaussian function in the *x* and *y* directions, and θ is the direction of the sine wave. Fig. 3 shows an example of Gabor filtering, where (a) is a grayscale image and (b) is the result of Gabor filtering.



(a) Grayscale image (b) Filtered result

Figure 3: An example of Gabor filtering

To generate secure image features, the **V** component is first divided into non-overlapping blocks sized *m*×*m*. For simplicity, let *M* be an integer multiple of *m*. Thus, there are *N*=(*M*/*m*) blocks along the *x*-axis and the *y*-axis directions, respectively. Next, we apply the 2-D Gabor filtering to every block with a random direction θ. Clearly, the result of our Gabor filtering is secure. This is because different θ values will lead to different filtered results and the correct guess of the used θ values of all blocks are almost impossible without knowledge of secret key. In this paper, a chaotic map called Skew tent map [Tang, Zhang and Lan (2015)] is used to generate the random direction θ. The well-known Skew tent map is defined as follows.

$$F(x) = \begin{cases} \frac{x}{p}, x \in [0, p] \\ \frac{1-x}{1-p}, x \in (p, 1] \end{cases} \tag{7}$$

where $x \in [0,1]$ is the initial state of the chaotic system and $p \in (0,1)$ is the control parameter. In this paper, x and p are used as the secret keys to control the Eq. (7) for iteratively generating N_2 random numbers. Let **A** be the two-dimensional array storing these N_2 random numbers. As the output of Eq. (7) is a floating-point number in the interval [0, 1], we map the numbers of the array **A** to the interval [0, 2π] by the following equation.

$$C[i][j] = A[i][j]\times 2\pi \tag{8}$$

where $A[i][j]$ and $C[i][j]$ are the elements of **A** and **C**, $1 \le i \le N$ and $1 \le j \le N$. Suppose that $\mathbf{B}_{i,j}$ represents the block of **V** in the *i*-th row and the *j*-th column, where $1 \le i \le N$ and $1 \le j \le N$. We apply Gabor filtering with the random direction $C[i][j]$ to $\mathbf{B}_{i,j}$ and select the variance of the filtered result as the block feature. Let $p_{i,j}$ be the feature of $\mathbf{B}_{i,j}$. Thus, a feature matrix **P** is available as follows.

$$\mathbf{P} = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,N} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,N} \\ \cdots & \cdots & \cdots & \cdots \\ p_{N,1} & p_{N,2} & \cdots & p_{N,N} \end{bmatrix} \tag{9}$$

Note that the use of random Gabor filtering in our paper is different from that of Li et al. [Li, Lu, Zhu et al. (2012)], in which Gabor filtering is applied to the whole image and the random filter is designed by summing up some Gabor filters with random directions. Moreover, the hashing algorithm reported in Li et al. [Li, Lu, Zhu et al. (2012)] does not reach a desirable classification performance. Section 3.4 will validate this.

2.3 DWT

To derive a short hash, we compress the feature matrix **P** by a single-level 2-D DWT and take the DWT coefficients in the LL sub-band for representation. DWT is a useful technique of image processing and a single-level 2-D DWT can decompose an input image into four sub-bands, i.e. LL sub-band, LH sub-band, HL sub-band and HH sub-band. Fig. 4 is the schematic diagram of 2-D DWT. In this paper, we select DWT coefficients in the LL sub-band to denote the feature matrix. This is based on the following considerations.

(1) DWT coefficients in the LL sub-band are the approximation coefficients of the feature matrix. (2) Our selection of the DWT coefficients in the LL sub-band can reduce about 75% coefficients. Consequently, our image hash is formed by concatenating all DWT coefficients in the LL sub-band. As the size of **P** is $N \times N$, the size of LL sub-band is $t \times t$ where $t = \lceil N/2 \rceil$ and $\lceil \cdot \rceil$ is the upward rounding. Therefore, our hash length is $L = t^2$ decimal digits.

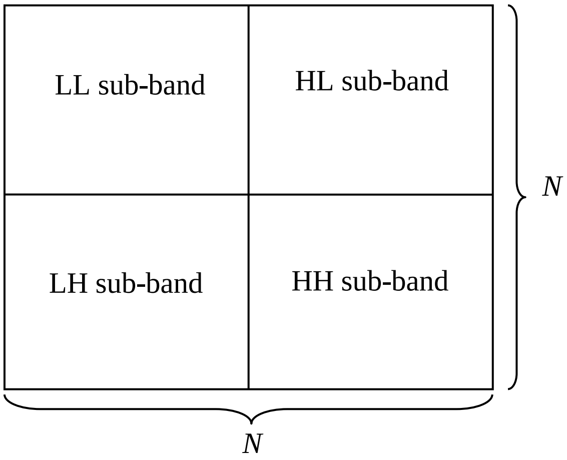


Figure 4: Schematic diagram of 2-D DWT

2.4 Hash similarity

As the element of our hash is decimal digit, we exploit the L2 norm to measure similarity of two image hashes. Let $\mathbf{h}_1=[h_1(1), h_1(2), \ldots, h_1(L)]$ and $\mathbf{h}_2=[h_2(1), h_2(2), \ldots, h_2(L)]$ be two image hashes. Thus, their L2 norm is defined as follows.

$$d(\mathbf{h}_1, \mathbf{h}_2) = \sqrt{\sum_{l=1}^{L}[h_1(l) - h_2(l)]^2} \tag{10}$$

where $h_1(l)$ and $h_2(l)$ are the l-th elements of $\mathbf{h}_1$ and $\mathbf{h}_2$, respectively. If the L2 norm of two hashes is smaller than a pre-defined threshold T, their corresponding images are viewed as similar images. Otherwise, they are considered as different images.

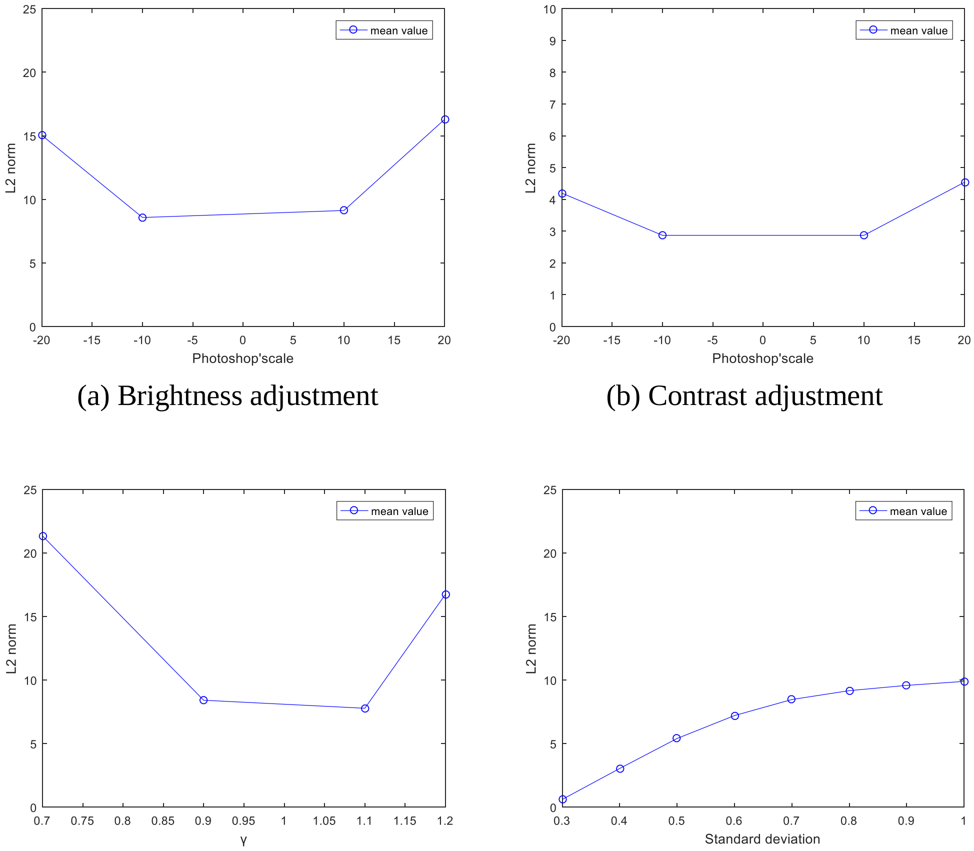
3 Experimental results

In the following experiments, the used parameters of our image hashing are as follows. The normalized image size is 512×512, the block size is 32×32, the window size of Gabor filter is $x=y=1$, the wavelength is $\lambda=16$, the parameters of skew tent map are $x=0.3$ and $p=0.4$. Therefore, our hash length is $L=64$ decimal digits. Section 3.1 and Section 3.2 discuss robustness and discrimination. Security is tested in Section 3.3 and performance comparison is finally presented in Section 3.4.

3.1 Robustness

The Kodak Lossless True Color Image Suite [Franzen (2017)] is taken as the image dataset.

This image dataset contains 24 color images and the sizes of these images are 768×512 or 512×768. To generate similar images of these 24 images, some robustness attacks are conducted by using Photoshop, MATLAB and StirMark. The used robustness attacks include brightness adjustment, contrast adjustment, gamma correction, 3×3 Gaussian low-pass filtering, speckle noise, salt and pepper noise, JPEG compression, watermark embedding, image scaling, and combinational attack of rotation, cropping and rescaling. For each attack, different parameters are used and thus every original image has 74 similar images. Fig. 5 presents the mean L2 norm under different attacks, where the *x*-axis is the used parameter and the *y*-axis is the L2 norm. It can be seen that most L2 norms are smaller than 20, except several cases in Figs. 5 (c), (h) and (j). Therefore, if the threshold is selected as *T*=20, our hashing can resist most of these attacks.

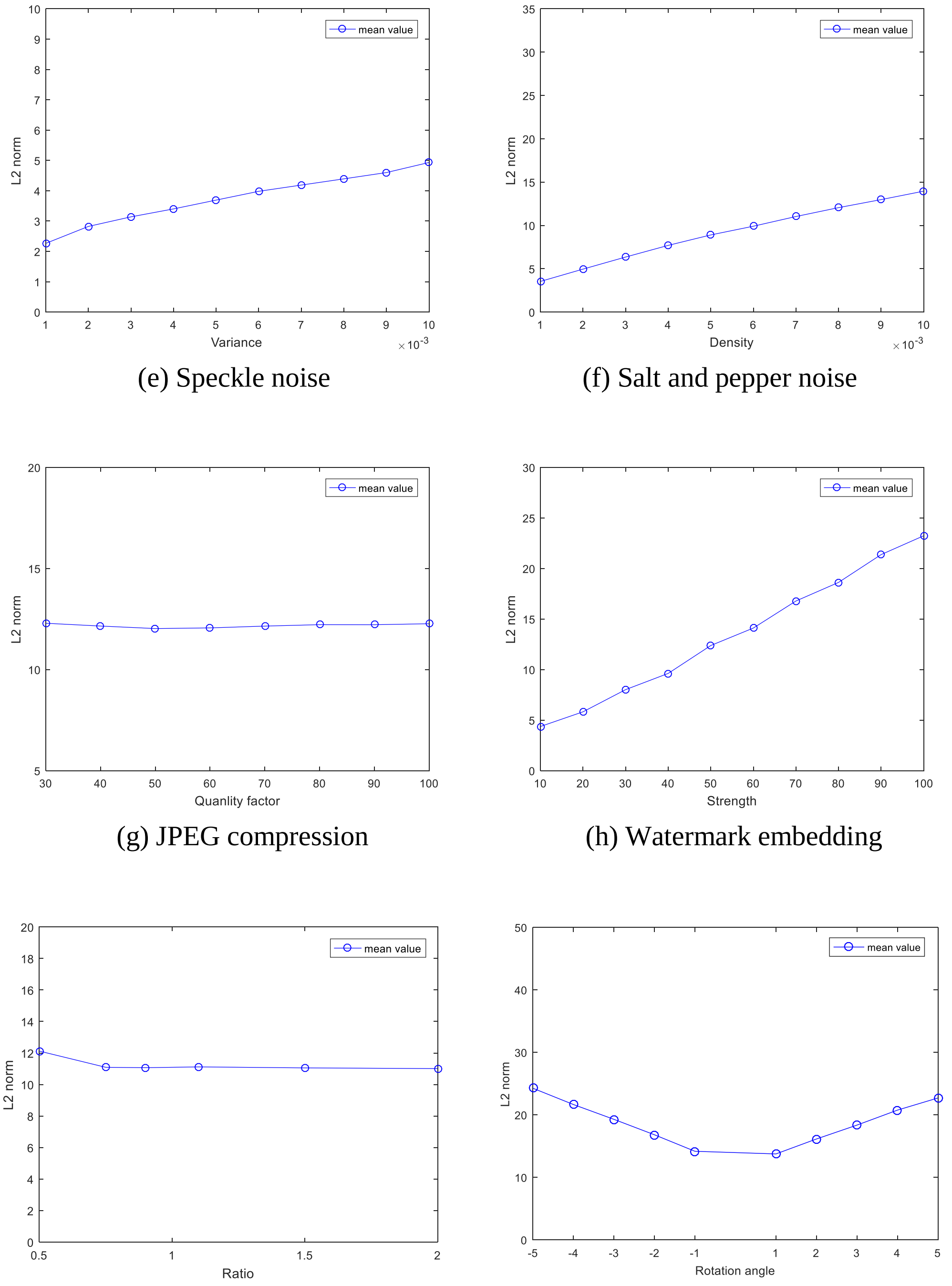


(a) Brightness adjustment

(b) Contrast adjustment

(c) Gamma correction

(d) 3×3 Gaussian low-pass filtering



(e) Speckle noise

(f) Salt and pepper noise

(g) JPEG compression

(h) Watermark embedding

(i) Image scaling

(j) Rotation, cropping and rescaling

Figure 5: Robustness test based on Kodak image database

3.2 Discrimination

The Uncompressed Colour Image Database (UCID) [Schaefer and Stich (2004)] is taken as the dataset for validating discrimination. This dataset includes 1338 true color images, whose sizes are 384×512 or 512×384. We extract hashes of these 1338 color images, calculate L2 norm between each pair of hashes, and then obtain 1338×(1338-1)/2=894453 distances. Fig. 6 is the distance distribution of different images, where the *x*-axis is the L2 norm and the *y*-axis is its frequency. It is observed that the minimum L2 norm and the maximum L2 norm are 10.15 and 222.94, respectively. In addition, the mean and standard deviation of these distances are 82.33 and 22.99, respectively. If the threshold is selected as *T*=20, there are only 0.00257% different images mistakenly judged as similar images.

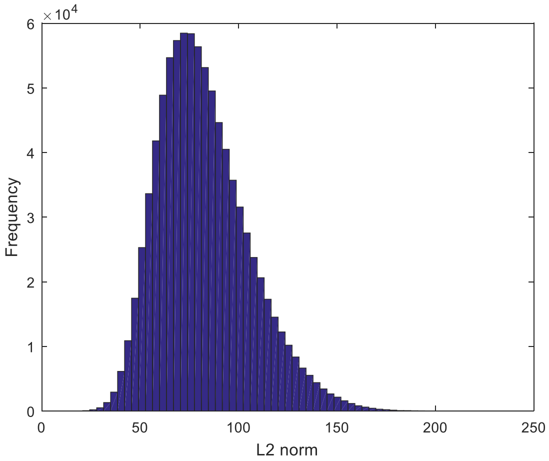


Figure 6: Hash distance distribution of different images

3.3 Security

To validate our security, we also take the images of Kodak image database as test images, use different secret keys (i.e. initial *x* of Skew tent map) to generate hashes of every test image and calculate L2 norm between the hashes of test image. It is observed that all L2 norms are big. For space limitation, a typical example is illustrated. Fig. 7 is the used test image. Fig. 8 presents L2 norm between hashes of the test image controlled by different keys, where the *x*-axis is the index of the wrong key and the *y*-axis is the L2 norm. Clearly, the minimum L2 norm is 41, which is much bigger than the above-mentioned threshold *T*=20. This illustrates that our hashing is key-dependent.

Figure 7: A test image of Kodak image database

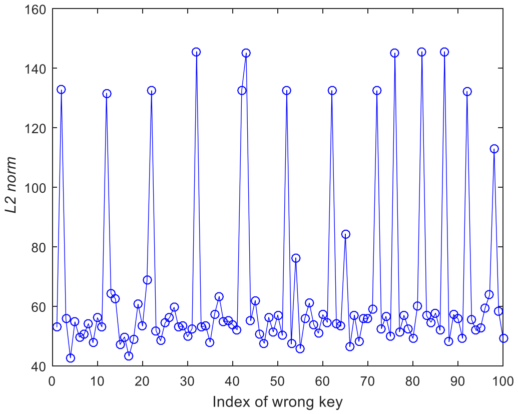


Figure 8: L2 norm between hashes of an image controlled by different keys

3.4 Performance comparison

To show advantages of the proposed hashing, we compared it with some popular image hashing algorithms. The selected algorithms include the GF-LVQ hashing [Li, Lu, Zhu et al. (2012)], the random walk based hashing [Huang, Liu, Wang et al. (2016)], the local histogram based hashing [Vadlamudi, Vaddella and Devara (2016)] and the hybrid feature based hashing [Qin, Sun and Chang (2018)]. In the comparison, the image datasets used in Sections 3.1 and 3.2 are also exploited to validate robustness and discrimination of the evaluated algorithms, and all images are resized to 512×512 during hash generation. To make comparisons as fair as possible, the original similarity metrics of the compared algorithms are also adopted here, i.e. L2 norm for the hybrid feature based hashing, and the normalized Hamming distance for the GF-LVQ hashing, the random walk based hashing and the local histogram based hashing.

The receiver operating characteristic (ROC) graph [Fawcett (2006)] is exploited to theoretically analyze classification performance between robustness and discrimination. In the ROC graph, the curve is formed by a set of points (P_{FPR}, P_{TPR}), where P_{FPR} and P_{TPR} represent the false positive rate (FPR) and the true positive rate (TPR), respectively. The FPR and TPR are defined as follows.

$$P_{\mathrm{FPR}}(d \leq T) = \frac{\text{Number of the pairs of different images judged as similar images}}{\text{Total pairs of different images}} \tag{11}$$

$$P_{\mathrm{TPR}}(d \leq T) = \frac{\text{Number of the pairs of similar images judged as similar images}}{\text{Total pairs of similar images}} \tag{12}$$

It is clear that FPR and TPR are the indicators of discrimination and robustness, respectively. A small FPR means good discrimination and a big TPR implies good robustness. As the *x*-axis is FPR and *y*-axis is TPR in the ROC graph, it can be intuitively concluded that the ROC curve near the top-left corner is better than the curve far away from it. In practice, the Area Under the ROC Curve (AUC) [Fawcett (2006)] is calculated to conduct quantitative analysis. The range of AUC is [0, 1], and a bigger AUC means a better classification performance. Fig. 9 is the ROC curve comparisons among different hashing algorithms. Clearly, the curve of the proposed hashing is above those curves of the compared algorithms. The AUCs of the GF-LVQ hashing, the random walk based hashing, the local histogram based hashing and the hybrid feature based hashing are 0.99009, 0.95932, 0.83577 and 0.99469, respectively. The AUC of our hashing is 0.99997, which is bigger than those of all compared algorithms. This means that our hashing is better than the compared algorithms in classification between robustness and discrimination.

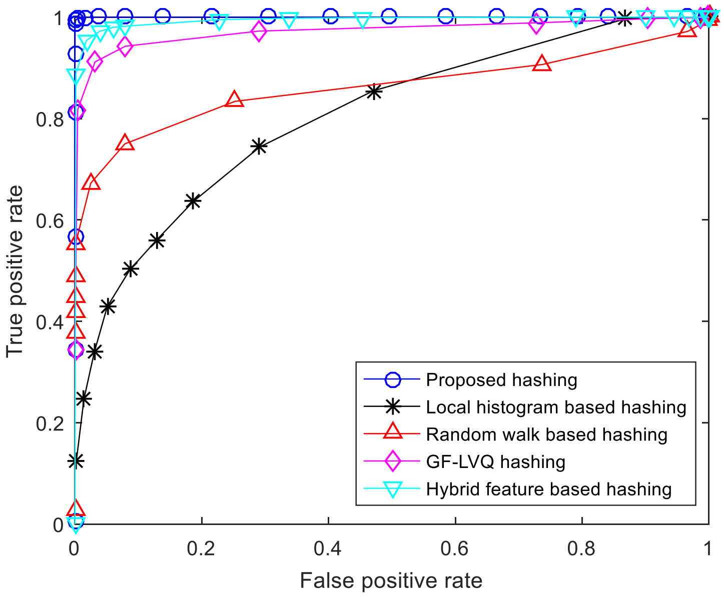


Figure 9: ROC curve comparisons among different hashing algorithms

In addition, the running time of the compared algorithms is also compared. This is done by recording the total consuming time of extracting hashes in the discrimination test and calculating the average value for generating a hash. All algorithms are implemented with MATLAB 2016a. The configurations of the used computer are as follows: the CPU is Intel Core i7-6700 with 3.40 GHz and the capacity of RAM is 8.0 GB. It is found that the average time of the GF-LVQ hashing, the random walk based hashing, the local histogram based hashing, hybrid feature based hashing and the proposed hashing is 0.32, 0.03, 0.02, 35 and 0.18 seconds, respectively. Our hashing is faster than the GF-LVQ hashing and the hybrid feature based hashing, but is slower than the random walk based hashing and the local histogram based hashing. As to hash length, the hashes of the hybrid feature based hashing and the proposed hashing are 104 and 64 decimal numbers, and those of the GF-LVQ hashing, the random walk based hashing and the local histogram based hashing are 120, 144 and 448 bits, respectively. Tab. 1 summarizes performance comparisons among different hashing algorithms.

Table 1: Performance comparisons among different hashing algorithms

Algorithm	AUC	Average time (s)	Hash length
GF-LVQ hashing	0.99009	0.32	120 bits
Random walk based hashing	0.95932	0.03	144 bits
Local histogram based hashing	0.83577	0.02	448 bits
Hybrid feature based hashing	0.99469	35	104 decimal digits
Proposed hashing	0.99997	0.18	64 decimal digits

4 Conclusions

We have proposed a robust image hashing based on random Gabor filtering and DWT. Our proposed hashing extracts robust and secure image features by Gabor filtering, Skew tent map and compress the image features via 2-D DWT. Experiments with open image datasets have been carried out to validate our performances and the results have illustrated that our proposed hashing is robust, discriminative and secure. ROC comparisons have shown that our proposed hashing is better than some popular image hashing algorithms in classification performance between robustness and discrimination.

Acknowledgement: This work is partially supported by the National Natural Science Foundation of China (Nos. 61562007, 61762017, 61702332), National Key R&D Plan of China (2018YFB1003701), Guangxi "Bagui Scholar" Teams for Innovation and Research, the Guangxi Natural Science Foundation (Nos. 2017GXNSFAA198222, 2015GXNSFDA139040), the Project of Guangxi Science and Technology (Nos. GuiKeAD17195062), the Project of the Guangxi Key Lab of Multi-source Information Mining & Security (Nos. 16-A-02-02, 15-A-02-02), the Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing, and the Innovation Project of Guangxi Graduate Education (No. XYCSZ 2018076).

References

Cao, Y.; Zhou, Z.; Sun, X.; Gao, C. (2018): Coverless information hiding based on the molecular structure images of material. *Computers, Materials & Continua*, vol. 54, no. 2, pp.197-207.

Fawcett, T. (2006): An introduction to ROC analysis. *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861-874.

Franzen, R. (2017): *Lossless True Color Image Suite*. http://r0k.us/graphics/kodak/.

Gurusamy, R; Subramaniam, D. V. (2017): A machine learning approach for MRI brain tumor classification. *Computers, Materials & Continua*, vol. 53, no. 2, pp. 91-108.

Huang, X.; Liu, X.; Wang, G.; Su, **M.** (2016): A robust image hashing with enhanced randomness by using random walk on zigzag blocking. *IEEE TrustCom/BigDataSE/ISPA*, pp. 14-18.

Kamarainen, J. K.; Kyrki, V.; Kälviäinen, H. (2002): Noise tolerant object recognition using Gabor filtering. *IEEE International Conference on Digital Signal Processing*, vol. 2, pp. 1349-1352.

Li, Y.; Lu, Z.; Zhu, C.; Niu, X. (2012): Robust image hashing based on random Gabor filtering and dithered lattice vector quantization. *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1963-1968.

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

Monga, V.; Evans, B. L. (2006): Perceptual image hashing via feature points: performance evaluation and tradeoffs. *IEEE Transaction on Image Processing*, vol. 15, no. 11, pp. 3453-3466.

Monga, V.; Mihcak, M. K. (2007): Robust and secure image hashing via non-negative matrix factorizations. *IEEE Transaction on Information Forensics and Security*, vol. 2, no. 3, pp. 376-390.

Qin, C.; Chang, C. C.; Tsou, P. L. (2013): Robust image hashing using nonuniform sampling in discrete Fourier domain. *Digital Signal Processing*, vol. 23, no. 2, pp. 578-585.

Qin, C.; Chen, X.; Dong, J.; Zhang, X. (2016): Perceptual image hashing with selective sampling for salient structure features. *Displays*, vol. 45, pp. 26-37.

Qin, C.; Chen, X.; Luo, X.; Zhang, X.; Sun, X. (2018): Perceptual image hashing via dual-cross pattern encoding and salient structure detection. *Information Sciences*, vol. 423, pp. 284-302.

Qin, C.; Chen, X.; Ye, D.; Wang, J.; Sun, X. (2016): A novel image hashing scheme with perceptual robustness using block truncation coding. *Information Sciences*, vol. 361-362, pp. 84-99.

Qin, C.; Ji, P.; Zhang, X.; Dong, J.; Wang, J. (2017): Fragile image watermarking with pixel-wise recovery based on overlapping embedding strategy. *Signal Processing*, vol. 138, pp. 280-293.

Qin, C.; Sun, M.; Chang, C. C. (2018): Perceptual hashing for color images based on hybrid extraction of structural features. *Signal Processing*, vol. 142, pp. 194-205.

Schaefer, G.; Stich, M. (2004): UCID-An uncompressed colour image database. *SPIE, Storage and Retrieval Methods and Applications for Multimedia*, pp. 472-480.

Tang, Z.; Huang, Z.; Zhang, X.; Lao, H. (2017): Robust image hashing with multidimensional scaling. *Signal Processing*, vol. 137, pp. 240-250.

Tang, Z.; Lao, H.; Zhang, X.; Liu, K. (2016): Robust image hashing via DCT and LLE. *Computers & Security*, vol. 62, pp. 133-148.

Tang, Z.; Li, X.; Song, J.; Wei, M.; Zhang, X. (2017): Colour space selection in image hashing: An Experimental Study. *IETE Technical Review*, vol. 34, no. 4, pp. 440-447.

Tang, Z.; Wang, S.; Zhang, X.; Wei, W.; Zhao, Y. (2011): Lexicographical framework for image hashing with implementation based on DCT and NMF. *Multimedia Tool and Application*, vol. 52, no. 2-3, pp. 325-345.

Tang, Z.; Zhang, X.; Huang, L.; Dai, Y. (2013): Robust image hashing using ring-based entropies. *Signal Processing*, vol. 93, no. 7, pp. 2061-2069.

Tang, Z.; Zhang, X.; Lan, W. (2015): Efficient image encryption with block shuffling and chaotic map. *Multimedia Tools and Applications*, vol. 74, no. 15, pp. 5429-5448.

Tang, Z.; Zhang, X.; Li, X.; Zhang, S. (2016): Robust image hashing with ring partition and invariant vector distance. *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 1, pp. 200-214.

Tang, Z.; Zhang, X.; Zhang, S. (2014): Robust perceptual image hashing based on ring partition and NMF. *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 711-724.

Vadlamudi, L. N.; Vaddella, R. P. V.; Devara, V. (2016): Robust hash generation technique for content-based image authentication using histogram. *Multimedia Tools and Applications*, vol. 75, no. 11, pp. 6585-6604.

Xiang, S.; Kim, H. J.; Huang, J. (2007): Histogram-based image hashing scheme robust against geometric deformations. *ACM Multimedia and Security workshop*, pp. 121-128.