Compact and Robust Image Hashing [⋆]

Sheng Tang[1,2], Jin-Tao Li[1], and Yong-Dong Zhang[1]

[1] Institute of Computing Technology, Chinese Academy of Sciences,
100080, Beijing, China
[2] Graduate School of the Chinese Academy of Sciences,
100039, Beijing, China
{ts,jtli,zhyd}@ict.ac.cn

Abstract. Image hashing is an alternative approach to many applications accomplished with watermarking. In this paper, we propose a novel image hashing method in the DCT Domain which can be directly extended to MPEG video without DCT transforms. A key goal of the method is to produce randomized hash signatures which are unpredictable for unauthorized users, thereby yielding properties akin to cryptographic MACs. This is achieved by encryption of the block DCT coefficients with chaotic sequences. After applying Principal Components Analysis (PCA) to the encrypted DCT coefficients, we take the quantized eigenvector matrix (8 × 8) and 8 eigenvalues together as the hash signature, the length of which is only 72 bytes for any image of arbitrary size. For image authentication, we also present an algorithm for locating tampering based on the hashing method. Experiments on large-scale database show that the proposed method is efficient, key dependent, pairwise independence, robust against common content-preserving manipulations.

1 Introduction

With the rapid growth of multimedia applications, protection of intellectual property is becoming more prominent. Image hashing (also known as fingerprinting, digital signature, and passive or noninvasive watermarking) is useful in protection of intellectual property. It can be used for multimedia authentication, indexation of content, and management of large database [2, 15]. It is an emerging research area that is receiving increased attention [2]. An image hash function maps an image to a short binary signature based on the image's appearance to the human eye [20]. In general, an image hash function requires the following desirable properties [2, 4]:

1. Robustness (Invariance under perceptual similarity): Images can be represented equivalently in different forms, and undergo various manipulations during distribution that may carry the same or similar perceptual information. Therefore, the signatures resulting from degraded versions of an image should result

[⋆] This work was supported by National Nature Science Foundation of China under grant number 60302028.

in the same or at least similar signatures with respect to that of the original image. This renders traditional cryptographic schemes using bit-sensitive hash algorithms, such as MD5 and SHA-1 not applicable [1, 2, 18], since even one bit change of the input will alter the output signature dramatically.

2. Pairwise independence (Discriminability or collision free): If two images are perceptually different, the signatures from the two images should be considerably different.

3. Key dependence: In some applications such as image authentication, it is required that the hash function $H()$ depends on a key K, i.e., for two different keys K_1 and K_2, $H_{K_1}(C) \neq H_{K_2}(C)$ for any image C.

4. Short bit length: The hash function should map an input image of arbitrary size to an output signature of short bit length. In some cases such as the method in [4] and our proposed method, the fixed bit length of the signature is preferable for its convenience in signature matching.

Significant attention has been given to robust hashing techniques. Up to now, many image hashing methods have been proposed [2–20]. These methods can be roughly classified into statistics-based [9–12], relation-based [6–8], edge or feature point based [12–15], coarse representation based [16–18], radon-based [2–5], mesh-based [19], and clustering-based [20]. As to statistics-based methods, some are not secure because their signatures can be easily forged due to the easiness of modifying images maliciously without changing the signatures such as block-histogram-based method in [9], and block-mean-based method in [10], and moment-based in [12]. Most current feature-point-based approaches have limited utility as they have poor robustness properties [15]. Additionally, the signature lengthes of many existing methods such as those in [6, 7, 9–12] etc., are not short and depend on image sizes. Although the signature length of the method proposed by [4] is only 180 real numbers regardless of image size, it is not very short compared with our method (only 72 bytes). Recently, several radon-based signatures have been proposed [2–5], which take the advantage of invariant features of the transform to provide robustness, but few address the problem of how to locate tampered regions, and can not directly extended and applied to MPEG video for real-time processing, which is the key motivation of this work. The method in [19] is somewhat complex in that most of the time is consumed in mesh normalization, and can not directly extended to MPEG video either. Additionally, most existing methods have focussed extensively on the problem of capturing image characteristics but randomization of the hash are not explicitly analyzed [15].

In this paper, we present a novel image hashing scheme in the DCT domain. To increase the signature's discriminability, we use the DC and 7 low-frequency terms of block DCT coefficients as the distinguishing features of an image called DCT data matrix. On the other hand, for the purpose of making the signature robust to minor pixel modifications that arise from blurring and compression operations, we apply PCA to the matrix, and quantize the eigenvector matrix and eigenvalues to get compact signature. Before PCA, we encrypt the DCT coefficients with chaotic sequences to achieve the randomization or key dependence of the hash. Based on the new scheme, we also present an algorithm for locating tampering. Experimental results show that our proposed method is effective. The paper is organized as follows. Section 2 and 3 describes signature generation and matching respectively. Section 4 addresses how to locate tampering. Section 5 reports experimental results, and conclusions are drawn in the last section.

2 Hash algorithms

We propose two algorithms, Algorithm A and Algorithm B. We present Algorithm A first as it is simpler and deterministic, and forms the backbone of the main, and it is aimed for image indexing which requires no motivation to randomization [15]. The second algorithm uses randomization to increase the output entropy and achieve key dependence of the hash function for authentication.

2.1 Algorithm A - Deterministic

The procedure for generating hash signature is described as follows. First, we transform the image C into 8×8 block-DCT domain. Then, we prepare the DCT data matrix A for PCA: divide the DC and 7 low-frequency AC terms (as shown in Fig.10.10 in [21]) by the corresponding values of the quantization matrix used in JPEG, and place the 8 quantized coefficients of each block into the N×8 matrix A in row or column order, where N is the total number of blocks. This can be represented as (1), where D_{ij} denotes the j^{th} quantized DCT coefficients of the i^{th} block of the image C.

$$A = \begin{pmatrix} D_{11}\,,\, D_{12}\,,\cdots,\, D_{18} \\ D_{21}\,,\, D_{22}\,,\cdots,\, D_{28} \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ D_{N1}, D_{N2}, \cdots, D_{N8} \end{pmatrix} \tag{1}$$

Before PCA, for the purpose of achieving high speed, instead of using the covariance matrix of A adopted by conventional PCA algorithm, we adopt the centered and scaled matrix B of A [22], i.e., standardizing A by removing the mean of each column and dividing each column by its standard deviation.

Finally, we apply PCA to the standardized matrix B [22], and use the resultant 8×8 eigenvector matrix V and 8 eigenvalues $\lambda_i (i = 1, \ldots, 8)$ as signature.

To get more compact signature, we quantize each element $a \in [-1, 1]$ of V and λ_i to an one-byte integer a_q and λ_{qi} according to (2) and (3) respectively.

$$a_q = \lfloor 127(1 + a) \rfloor \tag{2}$$

$$\lambda_{qi} = \lfloor \frac{255\lambda_i}{\sum_{j=1}^{8} \lambda_j} \rfloor \tag{3}$$

2.2 Algorithm B - Randomized

For image authentication, the security of the hash algorithm is an issue. More precisely, it is required that the hash function depends on the private key K [2]. We propose a novel method by using chaotic sequences to achieve this as follows.

Chaotic systems are very sensitive to initial conditions, have noise-like behaviors and compact description [23]. So we use chaotic sequence to randomize (encrypt) the DCT data matrix A before PCA. The logistic map for generating the chaotic sequence is:

$$x_{n+1} = 1 - 2x_n^2. \tag{4}$$

where $x_n \in (-1, 1)$ is a real number, $n \in [0, 8N - 1]$, and the initial value $x_0 \in (0, 1)$ is returned by a random function using the key K as its seed. Thus, we can easily convert the chaotic sequence (column vector) $\{x_n\}$ to an $N \times 8$ encryption matrix G in row major order. Therefore we can calculate the encrypted matrix E from the DCT data matrix A by:

$$E = A .* G \tag{5}$$

where the operator ".∗" means the scalar multiplication of two matrices. Finally, we substitute E for A in the deterministic algorithm. The remainder of the algorithm is the same as the deterministic algorithm, that is, applying PCA to the standardized matrix of E and subsequent quantization of eigenvector matrix and eigenvalues.

Because it is impossible to deduce E from the signature, it is very hard to find the private key to forge the signature after encryption, even if the original image is available. The adoption of encryption is to ensure that only the right source can generate the authentication signature, i.e., the hash function depends on the private key. Therefore, different signatures generated with different keys do not match due to their different eigenvectors and eigenvalues. In the extreme hypothetical case, the private key used by the original source may be known to the attacker. This is is a general problem for any secure communication and is out the scope of this paper.

3 Signature matching

Two images are declared similar (for indexation) or authentic (for authentication) if the similarity S between their signatures is above a certain threshold T, which can be determined by experiments or by user's demands according to various applications. The main idea of signature matching is that if two images are considered similar or authentic, corresponding eigenvectors from the two signatures should be high correlative. Thus, S can be calculated by computing correlation between corresponding pairs of eigenvectors, that is, the cosine of the angle between them since the two eigenvector matrices are orthogonal matrices.

After dequantizing each element a_q of eigenvector matrices by:

$$a = \frac{a_q}{127} - 1 \tag{6}$$

we let $V_o = (\alpha_{o1}, \alpha_{o2}, \ldots, \alpha_{o8})$, $\lambda_o = (\lambda_{o1}, \lambda_{o2}, \ldots, \lambda_{o8})$ and $V_t = (\alpha_{t1}, \alpha_{t2}, \ldots, \alpha_{t8})$, $\lambda_t = (\lambda_{t1}, \lambda_{t2}, \ldots, \lambda_{t8})$ be the dequantized eigenvector matrices and quantized eigenvalue vector of the original image C_o and the image C_t to be tested respectively. S can be calculated by computing the eigenvalue-weighted summation of the correlations of all the pairs as:

$$S = \sum_{i=1}^{8} \omega_i |\alpha'_{oi} \alpha_{ti}| \tag{7}$$

where α'_{oi} denotes the transpose of column vector α_{oi}, and ω_i is the eigenvalue factor defined as (8). The factor is used for considering different contribution of each compared pair of eigenvectors.

$$\omega_i = \frac{\lambda_{oi} + \lambda_{ti}}{2 \times 255} \tag{8}$$

4 Locating tampered blocks

For authentication, locating of tampering, such as detecting modification of licence plate is useful [1,21]. Based on the randomized hash algorithm, we present an algorithm for locating tampering if the calculated S between the images C_o and C_t is below the authentic threshold T. If a malicious tampering is occurred, the DCT coefficients of tampered blocks changes significantly, hence remarkable altering of corresponding HTS values. Therefore, by comparing the HTS values of the corresponding blocks of the image C_o and C_t, we can easily determine which block is most possibly tampered. The algorithm is described as follows.

First, after PCA, according to [22], $\mathrm{HTS_t}$ vector of the image C_t can be calculated from the $N \times 8$ standardized matrix B_t , 8×8 eigenvector matrix V_t, and 8×8 diagonal eigenvalue matrix λ_t as:

$$\mathrm{HTS_t} = |\tfrac{1}{\sqrt{\lambda_t}}(B_t V_t)'|' \tag{9}$$

where the operator "||" returns a row vector of the Euclidian length of each column.

For authentication without original images, since the B_o can not be accessed, we use B_t to estimate $\mathrm{HTS_o}$ of the original image C_o. So we substitute λ_o and V_o for λ_t and V_t in (9) to estimate $\mathrm{HTS_o}$:

$$\mathrm{HTS_o} = |\tfrac{1}{\sqrt{\lambda_o}}(B_t V_o)'|'. \tag{10}$$

Finally, determine which block is the region most possibly tampered by computing the difference vector δ between the $\mathrm{HTS_t}$ with $\mathrm{HTS_o}$ according to:

$$\begin{cases} \delta = (\mathrm{HTS_t} - \mathrm{HTS_o})^2 \\ (i,j) = argmax\{\delta\} \end{cases} \tag{11}$$

where the returned (i, j) denotes the indices (or location) of the required block.

5 Experimental results

In evaluating our proposed method, we tested it on the well-known image "Lena" (512×512) and "bmw" (800 × 600) downloaded from www.bmw.com, and 10000 test images randomly selected from the Corel Gallery database (www. corel.com) including many kinds of images (256 × 384 or 384 × 256). All the colour images are transformed into 8 bits/pixel gray level images. To do experiments, we first extracted signatures from all the 10000 images. Although we implemented the method with Matlab C++ Math Library, it took only about 350 seconds for Algorithm B to extract the 10000 signatures on the PC of Pentium IV 2.4G, which shows the method is efficient.

5.1 Robustness Test

To test robustness of the method, the original images were subjected to various image processing steps which are detailed in [24]. We first compressed the 10000 images to various JPEG images with different quality levels Q ranging from 20% to 90%, and calculated S between images and their corresponding JPEG images. The means and standard deviations (Std) of the measured S were shown in Table.1. Compared with the mean and Std of S in the following pairwise dependence test, this table shows that two proposed algorithms are fairly robust against compression.

Table 1. Means and Std of the measured S between 10000 images and corresponding JPEG images

JPEG	Algorithm A		Algorithm B	
Compression	Mean	Std	Mean	Std
JPEG(Q=20%)	0.9620	0.0555	0.9158	0.0879
JPEG(Q=30%)	0.9738	0.0453	0.9325	0.0750
JPEG(Q=40%)	0.9753	0.0469	0.9404	0.0732
JPEG(Q=50%)	0.9670	0.0542	0.9387	0.0758
JPEG(Q=60%)	0.9867	0.0304	0.9647	0.0515
JPEG(Q=70%)	0.9890	0.0290	0.9713	0.0473
JPEG(Q=80%)	0.9949	0.0179	0.9849	0.0340
JPEG(Q=90%)	0.9950	0.0160	0.9915	0.0197

For image authentication, we set the mean S (0.9158) of JPEG(Q=20%) as the threshold T for authentication.

As to Algorithm A, we added noise to the image "Lena" in various noise levels ranging from 1 to 5, and the calculated S between the original image and noised ones are 0.9980, 0.9514, 0.9848, 0.8939, 0.8370 respectively. We rotated the image "Lena" with small angles varying from 1 to 6 degree. The calculated S between the original image and rotated ones are 0.7792, 0.7459, 0.8031, 0.7950, 0.8378, 0.6670 respectively. We also scaled the image “Lena” with scaling factors ranging from 20% to 200%. The mean and Std of the calculated S between the original image and scaled ones are 0.7337 and 0.1360 respectively. The above results show that the Algorithm A is fairly robust against noising when noise levels is less than 4, while not robust against geometric manipulations such as scaling and rotation.

As to Algorithm B and the same noised image “Lena”, the calculated S between the original image and noised ones are 0.9456, 0.9203, 0.8946, 0.8735, 0.7512, which shows that Algorithm B is robust against noising when noise levels is less than 3.

The Algorithm B is sensitive to geometric manipulations which can be clearly specified by users [6]. The reason is that the encryption matrix G is sensitive to the altering of image sizes.

5.2 Pairwise independence test

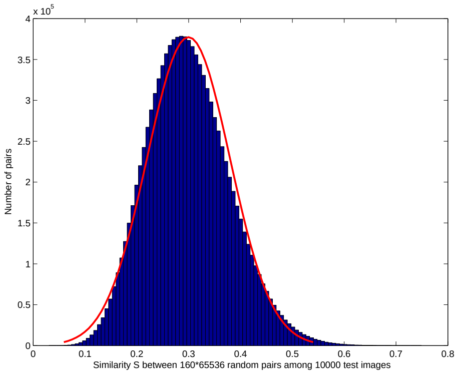


Fig. 1. Pairwise independence test of Algorithm B: Histogram of the measured S between 10×2^{20} pairs of signatures randomly selected from the 10000 images. The red line represents the ideal random i.i.d. case N(0.2995, 0.0801).

As to Algorithm B, we randomly selected 10×2^{20} pairs of signatures from the 10000 images, and calculated S between each pair. As shown in Fig.1, all the measured S were within the range between 0.0305 and 0.7491. The mean μ and Std σ were 0.2995 and 0.0801. As the histogram closely approaches the ideal random i.i.d. case N(μ, σ), we can conclude that the proposed Algorithm B is pairwise independent, and can calculate the false alarm rate $\mathrm{P_{FA}}$ (the probability that declare different images as authentic) according to:

$$\mathrm{P_{FA}} = \int_T^\infty \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}} = \frac{1}{2} erfc(\frac{T-\mu}{\sqrt{2}\sigma}). \tag{12}$$

Substituting μ=0.2995, σ=0.0801, T=0.9158, we got very low false alarm rate: $P_{FA} = erfc(5.4406)/2 = 7.1229 \times 10^{-15}$. It shows that our method is fairly discriminative, i.e., collusion-free.

As to Algorithm A, we got the similar result. The mean μ and Std σ were 0.3648 and 0.1004. So we can conclude that the two proposed Algorithms are pairwise independent.

5.3 Key dependence test

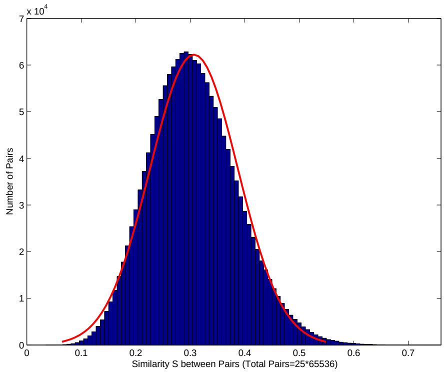
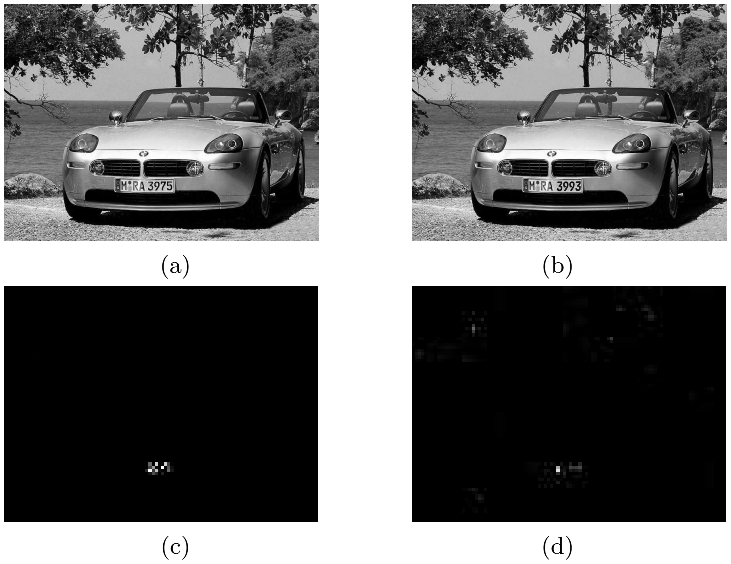


Fig. 2. Key dependence test of Algorithm B: Histogram of the measured S between 25 × 65536 pairs of signatures randomly selected from 65536 signatures generated by different keys from the image "Lena". The red line represents the ideal random i.i.d. case N(0.3067, 0.0807).

To test key dependence of the proposed Algorithm B, we used the image "Lena" to generate 65536 different signatures by different keys ranging from 0 to 65535, and randomly selected 25 × 65536 pairs of signatures to calculate S between each pair. As shown in Fig.2, all the measured S were within the range between 0.0349 and 0.7542. The mean and Std were 0.3067 and 0.0807. According to (12), we got the probability P_F that declare different signatures generated by different keys as same: $P_F = erfc[(0.9158 - 0.3067)/(\sqrt{2} \times 0.0807)]/2 = erfc(5.3362)/2 = 2.2347 \times 10^{-14}$, which shows the method is key dependent.

5.4 Authentication test

We made modifications within the region of the license plate in "bmw" as shown in Fig.3(b). The measured S between the original and tampered images was 0.5614 ($< T$), so we successfully detected that the image was tampered, and located the tampered regions as shown in Fig.3(d). The row and column indices of the most possibly tampered block returned by the propose method are 59 and 47 respectively. It shows that Algorithm B can detect malicious modifications and locate tampering.



(a) (b)

(c) (d)

Fig. 3. Authentication test: (a)original image (800 × 600); (b) tampered image (800 × 600) with changing licence "3975" to "3993"; (c) the highlights indicate the real changes of block DCT coefficients between (a) and (b); (d) block-based HTS difference vector δ map (100 × 75) between (a) and (b), the highlight intensity is proportional to the possibility of being tampered.

6 Conclusion

In this paper, we present a compact image hashing method based on PCA of block DCT coefficients. Experiments show that the proposed method is efficient, pairwise independent, and robust against common content-preserving manipulations. The randomized algorithm is key dependent, and can detect malicious modifications and locate tampering. It is convenient to extend our method to verify MPEG video streams without DCT transforms. Additionally, since the signature length is only 72 bytes long regardless of image size, it is of great importance to embed the signature into the image itself (such as into the middle-frequency terms of block DCT coefficients) for providing solutions to self-authentication watermarking system [7] in that watermarking capacity is greatly limited [12].

References

1. B.B.Zhu, M.D.Swanson, A.H.Tewfik, "When seeing isn't believing [multimedia authentication technologies]", *Signal Processing Magazine, IEEE,* 21(2):40-49, 2004.
2. J.S.Seoa, J.Haitsmab, T.Kalkerb, C.D.Yoo, "A robust image fingerprinting system using the Radon transform," *Signal Processing: Image Communication*, 19(4):325-339, April 2004.
3. Z.Yao, N.Rajpoot, "Radon/Ridgelet Signature for Image Authentication", *Proc. IEEE ICIP 2004, Singapore*, October 2004

4. F. Lefebvre, J. Czyz, and B. Macq, “A robust soft hash algorithm for digital image signature”, *In: Proceedings of IEEE ICIP 2003, Barcelona*, II:495-498, Sept. 2003.
5. F. Lefebvre, B. Macq, and JD Legat, “RASH: RAdon Soft Hash Algorithm”, *In 11th European Signal Processing Conference*, Toulouse, France, Sept. 2002.
6. C.-Y. Lin and S.-F. Chang, “A robust image authentication method distinguishing JPEG compression from malicious manipulation”, *IEEE Trans. Circuits Syst. Video Technol.*, 11(2): 153-168, 2001.
7. C.-Y. Lin and S.-F. Chang, “Robust digital signature for multimedia authentication”, *Circuits and Systems Magazine, IEEE,* 3(4):23-26, 2003.
8. C.-S.Lu, H.-Y. M.Liao, Structural digital signature for image authentication:an incidental distortion resistant scheme”, IEEE Trans. Multimedia,pp.161-173,June 2003.
9. M. Schneider, S.-F. Chang, “A robust content based digital signature for image authentication”, *In: Proceedings of IEEE ICIP 96, Lausanne, Switzerland,* Vol.3:227-230, October 1996.
10. Lou DC, Liu JL. “Fault resilient and compression tolerant digital signature for image authentication”, *IEEE Trans. Consumer Electronics,* 46(1):31-39, 2000.
11. C. Kailasanathan and R. Safavi Naini, “Image authentication surviving acceptable modifications using statistical measures and k-mean segmentation”, IEEE-EURASIP Work. Nonlinear Sig. and Image Proc., June 2001.
12. M.P. Queluz, “Authentication of digital images and video: generic models and a new contribution”, *Signal Processing: Image Communication,* 16(5):461-475, 2001.
13. S. Bhattacharjee, M. Kutter, “Compression tolerant image authentication”, *In: Proceedings of the IEEE ICIP 1998, Chicago, IL,* October 1998.
14. J. Dittman, A. Steinmetz, and R. Steinmetz, Content based digital signature for motion picture authentication and content-fragile watermarking, Proc. IEEE Int. Conf. Multimedia Comp. and Sys., pp. 209-213, 1999.
15. Vishal Monga and Brian L. Evans, “Robust Perceptual Image Hashing Using Feature Points”, *Proc. IEEE ICIP 2004, Singapore,* 3:677-680, Oct. 2004
16. J. Fridrich and M. Goljan, “Robust hash functions for digital watermarking”, *Proc. IEEE Int. Conf. Info. Tech.: Coding and Comp.*, Mar. 2000.
17. R. Venkatesan, S. M. Koon, M. H. Jakubowski, and P. Moulin, ”Robust image hashing”, *Proc. IEEE Conf. Image Proc.*, Sept. 2000.
18. K. Mihcak and R. Venkatesan, New iterative geometric techniques for robust image hashing, *Proc. ACM Work. Security and Privacy in Dig. Rights Man.*, Nov. 2001.
19. Chao-Yong Hsu and Chun-Shien Lu, “Geometric Distortion-Resilient Image Hashing System and Its Application Scalability”, *In Proceedings of the ACM 2004 multimedia and security workshop on Multimedia and security, Magdeburg, Germany*, pages:81-92, Sept., 2004.
20. Vishal Monga, Arindam Banerjee, and Brian L. Evans, “Clustering Algorithms for Perceptual Image Hashing”, *Proc. IEEE Work. on Digital Signal Processing*, Aug. 1-4, 2004, pp. 283-287, Taos, NM.
21. I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking,* NewYork: Morgan Kaufmann, 2001.
22. J. Edward Jackson, *A User’s Guide to Principal Components,* John Wiley & Sons, Inc., pp. 1-25, 1991.
23. Hui Xiang, Lindong, Wang Hai Lin, Jiaoying Shi, ”Digital Watermarking Systems with Chaotic Sequences”, *SPIE Conference on Security and Watermarking of Multimedia Contents*, SPIE Vol.3657, pp.449-457, San Jose, California,January,1999.
24. Fabien A. P. Petitcolas, “Watermarking schemes evaluation”, *IEEE Signal Processing,* 17(5):58-64, September 2000.