Digital Authentication Technique using Content-based Watermarking in DCT Domain

Hyun Lim, Myung Eun Lee, Soon Young Park, Wan Hyun Cho*

Dept. of Electronics Engineering, Mokpo National Univ.,
*Dept. of Statistics, Chonnam National Univ., Korea
Email: hlim,melee,sypark@mokpo.ac.kr, whcho@chonnam.ac.kr

Abstract

In this paper, we present a digital authentication technique using content-based watermarking in digital images. To digest the image contents, Hopfield network is employed on the block-based edge image. The Hopfield function extracts the same bit for similarly looking blocks so that the values are unlikely to change to the innocuous manipulations while being changed for malicious manipulations. By inputting the extracted bit sequence with secret key to the cryptographic hash function, we generate a watermark for each block by seeding a pseudo random number generator with a hash output. Therefore, the proposed authentication technique can distinguish between malicious attacks and innocuous attacks. Watermark embedding is based on the block-based spread spectrum method in DCT domain and the strength of watermark is adjusted according to the local statistics of DCT coefficients in a zig-zag scan line in AC subband. The numerical experiments show that the proposed technique is very efficient in the performance of robust authentication.

1. Introduction

Digital watermarking schemes are designed for two purposes: digital copyright protection and digital authentication. While digital copyright protection is to detect copyright violations of multimedia data, the goal of digital authentication technique is to verify that the original contents have not been modified up to a particular level [1,2]. For the satisfying authentication tasks, it is required to localize modifications occurred to an image or to discriminate between malicious and innocuous manipulations. Here the innocuous manipulations mean the legitimate distortions such as noise addition and high-quality lossy compression. On the other hand, malicious manipulations are the illegitimate distortions such as low-quality lossy compression and removing image objects so that the original contents are likely to be destroyed.

Recently image authentication by means of robust watermarking becomes increasingly popular in the sense that it can distinguish between malicious and innocuous manipulations unlike the fragile watermarking techniques. In this method, any manipulations do not affect the embedded watermark itself but generate a different watermark which is not correlated with the embedded one by using the concepts of visual hash function [3]. Fridrich proposes a robust block-based visual hash function to embed hash outputs into each block by using a robust watermarking technique. The proposed hash function produces almost the same bit sequence for similarly looking blocks while producing uncorrelated sequence for two different blocks [4]. Lu et al. quantize a host image's wavelet coefficients as masking threshold units and embeds robust and fragile watermarks for image authentication and protection [5].

In this paper, we describe a technique that uses a robust watermark by using image description vectors extracted from sub-blocks. To digest the image content, a Hopfield network is employed on the block-based edge image [6]. After training the Hopfield network with predefined reference patterns, we input the block-based edge image into the network to achieve a stable state corresponding to one of the learned patterns. The Hopfield function makes the test pattern converge to the same state for similarly looking blocks while converges to the spurious states for differently looking ones. After converting the reference pattern number extracted from a block-based image to the bit sequence, the bit sequence concatenated with secret key is inputted into the cryptographic hash function. Then a watermark for each block is obtained by seeding a pseudo random number generator with a hash output. Next we embed watermarks in block-based DCT domain. For the watermark detection, image description vectors are first extracted from the block-based image whether they are watermarked, unwatermarked or attacked to obtain hash output and the hash values are used for the seed of watermark generation. Depending upon the correlation between a set of watermarked DCT coefficients and the generated watermark, content authentication is determined.

2. Extraction of Image Description Vectors

The main idea in using image description vectors for the content-based image authentication steps from the fact that the important information in an image exists in edge components. The feature vector that describes the visual contents can be extracted as Fig. 1.

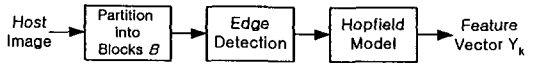


Fig. 1 Generating a content-based feature vector

We partition a gray scale image into blocks of $I \times J$ pixels and compute the $I \times J$ edge image from each block by using edge operator. The block image is first preprocessed by a median filter to suppress the noise and reject the isolated components. Then we use the Canny edge operator to digest the block-based image contents [7]. The edge detection process serves to simplify the analysis of images by drastically reducing the amount of data to be processed, while at the same time preserving useful structural information about object boundaries. In order to quantify the distribution of edge pattern E, we use a Hopfield network for feature vectors encoding.

For Hopfield network, what we want to find is based on the recognition of the edge shapes when a given block contains an edge. After training the Hopfield network with predefined edge patterns, we input the block-based edge image into the network to achieve a stable state corresponding to one of the learned patterns. The Hopfield function makes the test pattern converge to the same state for similarly looking blocks while converges to the spurious states for differently looking ones. After converting the edge pattern number extracted from a block-based image to the bit sequence Y_k, the bit sequence concatenated with secret key is inputted into the cryptographic hash function.

A cryptographic hash function maps an arbitrary-length string into a fixed-length hash value. One of the properties of the hash function is, given a desired output, computationally impossible to find a corresponding input string [3]. We compute the hash for each block as follows.

$$H(K \circ Y_k \circ k) = (d_1, \Lambda, d_p) \tag{5}$$

where K is a secrete key, k is the block number consisting of a string of bits and o string concatenation operation. Here, the hash function produces the output string of p bits with the input of the string concatenation of K, Y_k, and k. In this paper, we will use the well-known MD5 as the hash function where $p = 128$ [8]. The hash output for each block is used as the seed for the generation of a watermark after reducing the 128 bits to 32 bits as the exclusive OR operation 4 bits in number order.

3. Watermark Framework

Fig. 2 shows the block diagram of the proposed watermark embedding and detection algorithms for image authentication. After partitioning the host image into blocks, a set of feature vectors which adequately describe the content of an image are extracted to be used as the input of the hash function. To make the proposed watermarking system secure against unauthorized detection, hash function with a secrete key are used and the watermark sequence for each block is produced by seeding a pseudo random number generator with a hash output. Watermark embedding process is similar to the previous spread spectrum method [9], but we use the strength α-curve which adapts to the local statistics instead of fixing the strength to the certain constant value to enhance the robustness while preserving the imperceptibility.

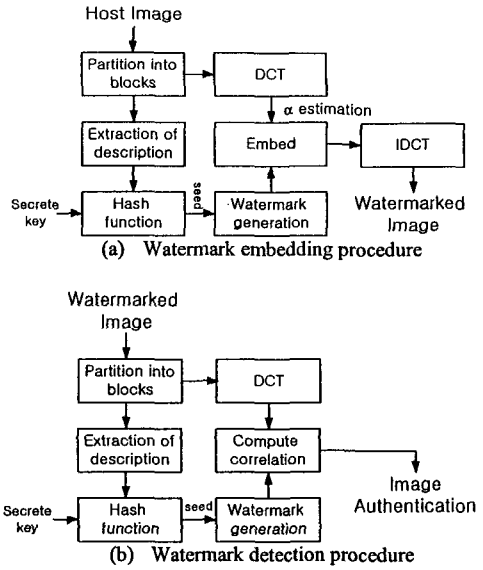


Fig. 2. A block diagram of a proposed watermarking system.

To embed the watermarks, we partition a gray scale image into blocks of $I \times J$ pixels and compute the $I \times J$ DCT coefficients for each block. Let v_{ij}^k be the DCT coefficients at the location (i, j) of any block k and Ω_i the locations of the DCT coefficients in the $i-th$ zig-zag scan line. For example $\Omega_0 = \{(0,0)\}$, $\Omega_1 = \{(0,1),(1,0)\}$ and $\Omega_i = \{(0,i),\Lambda,(i,0)\}$.

The watermark signal which is obtained by seeding a pseudo random generator follows a normal distribution with zero mean and unit variance:

$$W = \{w_0, \Lambda, w_{m-1}\}, w_i \sim N(0,1). \tag{6}$$

For the simplicity of the notation, let $X = \{x_0, \Lambda, x_{m-1}\}$ be the DCT coefficients sequence of length m by selecting a zig-zag ordering in AC frequency subband of the 2-dimensional DCT $\{v_{ij}^k\}$ of block k.

The watermark embedding coefficients is obtained by inserting the watermark into DCT coefficients.

$$x_i^w = x_i + \alpha |x_i| w_i, i = 0, \Lambda, m-1 \tag{7}$$

where α is a properly chosen parameter tuning the watermark strength. By selecting the higher α the watermark becomes more robust while the watermark being more visible. Instead of fixing this quantity by an empirical value as in the previous methods, we adjust the parameter adaptively according to the local statistics of DCT coefficients as follows.

$$\alpha_i = \frac{s}{\log(\sum_{v_{ij}^k \in \Omega_i} | v_{ij}^k |)} \tag{8}$$

where s is the scaling factor to control the bias of the curve and is in proportion to $v_{00}^k / \bar{v}_{00}$, where $\bar{v}_{00}$ is the mean of the DC coefficients over all blocks. After modifying the DCT coefficients by a watermark and applying the inverse DCT for each block, the watermarked image I^w is obtained.

To extract the watermark from possibly corrupted image I^*, the $I \times J$ block-based DCT transform is applied to this image. The image description vector Y is first extracted in a similar manner as embedding process to obtain the hash output and the hash values are used for the seed of watermark generation. Detection is based on the correlation between a set of watermarked DCT coefficients and the generated watermark:

$$R_k = \frac{1}{m} \sum_{i=0}^{m-1} x_i^* w_i \tag{9}$$

where R_k is the correlation for the $k-th$ block. Correlation R_k is used to verify the content authentication by comparing it to a threshold T_a. The statistical properties for the correlation detector are described in the next section.

4. Statistical Properties of a Watermark Detector

One way to insist the authentication is to divide the image into several blocks and embed the different watermarks in each block. Then it is sufficient that we may verify whether these watermarks exist in the observed image. In order to solve this problem statistically, we consider the following two hypotheses.

H_0: No change of content for each block of a given image occurs.

H_1: Some manipulation for each block of a given image occurs.

Next, we divide the observed image into the L blocks of size $(l \times l)$ and compute the DCT coefficients for each divided block image using the DCT transformation. Then we can use the following correlation detector as the statistics that we may detect if some fabrication or manipulation at any block k of observed image occurs.

$$R_k = \frac{1}{m} \sum_{i=0}^{m-1} x_i^* w_i^* \tag{10}$$

Here m is the total number of DCT coefficients of each block where the watermark is embedded.

If the null hypothesis H_0 is true, the DCT coefficients computed from observed block image contain the same as the embedded watermark. They are given as:

$$x_i^* = x_i + \alpha | x_i | w_i + \varepsilon_i \tag{11}$$

where w_i denote the embedded watermark signal and ε_i is the measurement error. If the observed image is consistent with the original given image, two kinds of watermarks are equal exactly.

$$w_i^* = w_i, \; i = 1, \Lambda \; , m-1 \tag{12}$$

Hence the detector statistic is given as follows:

$$R_k = \frac{1}{m} \sum_{i=0}^{m-1} (x_i + \alpha | x_i | w_i + \varepsilon_i) w_i \tag{13}$$

In this case, we can compute the mean and variance of given detector statistic. They are given by

$$\begin{aligned} \mu_k &= E(R_k) = E[\frac{1}{m} \sum_{i=0}^{m-1} (x_i + \alpha | x_i | w_i + \varepsilon_i) w_i] \\ &= \frac{1}{m} \sum_{i=0}^{m} \alpha E[| x_i |] \end{aligned} \tag{14}$$

$$\begin{aligned} \sigma_k^2 &= V(R_k) \\ &= \frac{1}{m^2} E(\sum_{i=0}^{m-1} (x_i + \alpha | x_i | w_i + \varepsilon_i) w_i)^2 - (\frac{1}{m} \sum_{i=0}^{m-1} \alpha E[| x_i |])^2 \\ &= \frac{1}{m^2} \sum_{i=0}^{m-1} ((1 + 3\alpha^2) E(| x_i |^2) + N\sigma_\varepsilon^2) - \frac{1}{m^2} \sum_{i=0}^{m-1} \alpha^2 E[| x_i |]^2 \end{aligned} \tag{15}$$

Also by the central limit theorem, the distribution of detector statistic approximates the normal distribution, and so the reject region of the null hypothesis is given at significant level γ as follows.

$$R_k \le \mu_k - z_\gamma \frac{\sigma_k}{\sqrt{m}} \tag{16}$$

5. Experiments

To verify the theoretical results and evaluate the performance of the proposed watermark technique, we apply 64×64 DCT transform to the 512×512 traffic intersection image shown in Fig. 3 [10]. The binary Hopfield function with a 64-neuron network is trained with 8 edge patterns, which are horizontals, verticals, and diagonal, respectively. We embed the watermark by using strength α -curve in the DCT domain.

The watermarked image produce the invisible watermark added to the original image by using block-based spread spectrum technique and hash function. Our image description vector Y_k behaves well in an image under lossy compression and

additive noise image. On the other hand, if some of part for watermarked image is modified from editing, then image description feature vectors are changed.

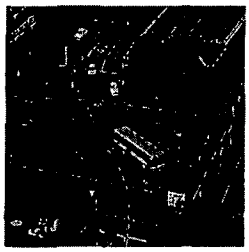
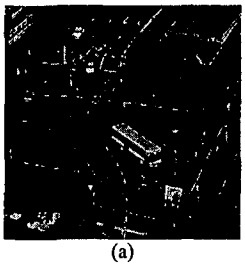
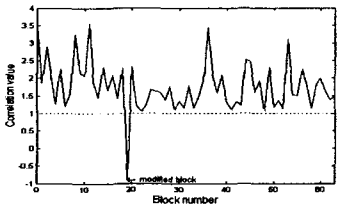


Fig. 3 Original test image.

Fig. 4 shows the watermarked image and its correlation result when the portion of the original image is modified by malicious manipulation: one of the small car is erased in Fig. 4(a). Here, the dotted line represents the threshold value determined from Eq. (16) with $z_\gamma = 0.1$. It can be noted that the corresponding correlation values drop under the threshold value. Therefore we can verify the content authentication by comparing correlation for each block to the predefined threshold.



(a)



(b)

Fig. 4 (a) The watermarked image with manipulation. (b) The results of correlations indicating the block number where changes have occurred.

6. Conclusions

In this paper, we have presented a robust watermarking based-authentication technique using Hopfiled network. Image description vectors are estimated from block-based edge image so that the values are unlikely to change to the innocuous manipulations while being changed for malicious manipulations. By inputting the image description vectors concatenated with secret key to the cryptographic hash function, we have generated a watermark for each block-based embedding by seeding a pseudo random number generator with a hash output. We have also shown that the robustness is enhanced by using the watermark strength which adapts to local statistics. For the statistical analysis of the watermark detection, the detector statistics is estimated and the threshold value is determined with moderate significant level. The experiments show that the proposed technique is very efficient in the performance of content authentication.

References

[1] F. Bartolini, A. Tefas, M. Barni and I. Pitas, "Image Authentication Techniques fir Surveillance Applications," *IEEE Proceedings*, 89(10), pp.1403-1418, Oct. 2001.

[2] M. M. Yeung and F. Minzer, "An invisible watermarking technique for image verification," *Proc. ICIP97, IEEE Int. Conf. Image Proceedings*, pp.680-683, Santa Barbara, CA, Oct. 1997.

[3] I. J. Cox, M. L. Miller and J.A. Bloom, *Digital Watermarking*, Morgan Kaufmann, 2001.

[4] J. Fridrich, "Image watermarking for tamper detection," *Proc. ICIP98 IEEE Int. Conf.* Image processing, II, pp.404-408, Chicago, IL., Oct. 1998.

[5] C. S. Lu and H. Y. Mark, "Multipurpose Watermarking for image Authentication and Protection," *IEEE Trans. on Image Processing*, 10(10), pp.1579-1591, Oct. 2001.

[6] Simon S. Haykin, *Neural Networks: A Comprehensive Foundation*, Pretice Hall(Sd), 1998.

[7] J. F. Canny, "A computational approach to edge detection," IEEE Trans. On Pattern analysis and Machine intelligence, vol. PAMI-8, Nov. 1986.

[8] R. L. Rivest, *The MD5 message digest algorithm*, Tech. Rep., 1992.

[9] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure Spread Spectrum Watermarking for Images," *Audio and Video. Proc. ICIP'96*, III, pp.243-246, 1996.

[10] Traffic Image Sequences, http://i21www.ira.uka.de/image_sequences/bad/bad_000.gif.