BINARY IMAGE AUTHENTICATION USING ZERNIKE MOMENTS

Hongmei Liu, Wei Rui, Jiwu Huang*

Dept. of Electronics and Communication, Sun Yat-Sen University
Guangzhou 510275, P. R. China

ABSTRACT

In this paper we propose a content-based binary image authentication scheme. At first, we use Zernike moments magnitudes (ZMM) to generate the feature vector and demonstrate that this feature vector can represent the binary image and decide its authenticity effectively. Then the watermark is generated by quantizing ZMMs and embedded into the image. The authentication doesn't need the original watermark. The decision depends on the distance between the extracted watermark and the feature vector of the test image and a metric measure. To decrease the influence of watermarking on the feature vector, we split the binary image into two parts by a random mask, one for generating feature vector and the other for embedding watermark. Zernike moments are usually computationally expensive, so we propose a fast algorithm. Extensive experiments show that our scheme can detect malicious attacks effectively.

Index Terms—binary image authentication, Zernike moments, watermark, feature vector

1. INTRODUCTION

The wide use of binary images has brought great interest for authentication. Digital watermarking technology seems to be a promising way and some schemes have been proposed. In [1], the authors proposed a block-based data-hiding scheme for binary images that can be used for authentication. They shuffle the image and compute the flippability score of each pixel. Watermark bits are embedded by flipping pixels that have relatively higher flippability scores. The authenticity of the image can be decided by comparing the extracted watermark with the original watermark. In [3], the authors proposed a data-hiding algorithm for binary document images based on Distortion-Reciprocal Distortion Measure [4]. Pixels that cause less distortion are chosen as candidates for flipping. To embed watermark, the authors enforce the odd-even feature of non-uniform blocks and employ a 2-D shifting to provide security for tamper proofing and authentication. Most of existing schemes use content independent watermark, thus the sender and the appraiser need to transmit the original watermark.

For the sake of convenience, we propose a new scheme combining content-based watermark with the embedding method in [1]. The authentication process doesn't need the original watermark. We study the characteristics of the ZMMs for binary images. Experimental results show that ZMMs can be used to represent the binary image and to decide the authenticity of the image effectively. In order to decrease the influence of watermarking on feature vector, we split a binary image into two parts by a random mask, one for generating feature vector and the other for watermark embedding. Zernike moments are usually computationally expensive; hence we propose a fast algorithm. Extensive experiments show that our scheme can detect malicious manipulations effectively.

The paper is organized as follows. In section 2, we describe the Zernike moments, the properties of ZMMs-based feature vector, and a fast computation algorithm. The outline of the proposed system, the splitting of the binary image, the authentication process and experimental results are given in section 3. Section 4 concludes the paper.

2. ZERNIKE MOMENTS

In content-based authentication scheme, extraction of feature vector is the most challenging issues. We identify Zernike moments to generate feature vector. The definition of Zernike moments in the following are based on [7].

2.1. Zernike moments and their characteristics

Zernike moments are based on a set of complex polynomials that form a complete orthogonal set over the interior of the unit circle, $x^2 + y^2 = 1$. Let the set of these polynomials be denoted by $\{V_{nm}(x, y)\}$:

$$V_{n,m}(x,y) = V_{n,m}(\rho,\theta) = R_{n,m}(\rho)\exp(jm\theta) \qquad (1)$$

where n is a non-negative integer and m is an integer such that $n-|m|$ is non-negative and even. ρ and θ represent polar coordinates over the unit circle and R_{nm} are polynomials of ρ (Zernike polynomials) given by

$$R_{n,m}(\rho) = \sum_{s=0}^{n-|m|/2} \frac{(-1)^s\left[(n-s)!\right]\rho^{n-2s}}{s!(\frac{n+|m|}{2}-s)!(\frac{n-|m|}{2}-s)!} \qquad (2)$$

Note that $R_{n,-m}(\rho) = R_{n,m}(\rho)$. These polynomials are orthogonal and satisfy

$$\iint_{x^2+y^2\le 1}\left[V^*_{n,m}(x,y)\right]\times V_{p,q}(x,y)dxdy = \frac{\pi}{n+1}\delta_{n,p}\delta_{m,q} \qquad (3)$$

with

Supported by NSF of China (60325208, 60633030, 90604008), NSF of Guangdong(04205407,06023191), Key Project of Science and Technology of Guangzhou, China (2005Z3-D0391)，973 Program(2006CB303104).

*Contact author (isslhm@mail.sysu.edu.cn)

$$\delta_{a,b} = \begin{cases} 1 & a = b \\ 0 & otherwise \end{cases} \quad (4)$$

The Zernike moment of order n with repetition m for a digital image is

$$A_{n,m} = \frac{n+1}{\pi} \sum_x \sum_y f(x,y) V^*_{n,m}(\rho,\theta), x^2 + y^2 \le 1 \quad (5)$$

Note that $A^*_{n,m} = A_{n,-m}$.

Suppose that one knows all moments $A_{n,m}$ up to order $N_{\max}$ of $f(x,y)$. Using orthogonality of the Zernike basis, we can reconstruct the image $f(x,y)$

$$\hat{f}(x,y) = \sum_{n=0}^{N_{\max}} \sum_m A_{n,m} V_{n,m}(\rho,\theta) \quad (6)$$

As $N_{\max}$ approaches infinity, $\hat{f}(x,y)$ will approach $f(x,y)$.

The reconstruction process is illustrated in Fig.1. For a 64*64 binary image of letter E, the reconstructed images are generated by using Equ.(6). It shows that the lower order moments capture gross shape information and the details are filled in by higher order moments. According to experiments, 12-order ZMMs with each quantized into 8 bits is enough for the authentication. In the following sections, we adopt 12-order and 49 ZMMs to generate the feature vector.

(a) (b) 6 order (c) 8 order (d) 12 order (e) 14 order

Fig.1. Reconstruction of a binary image. From left to right: the original image, the reconstructed image with order 6, 8, 12 and 14, respectively.

From Fig.1 and the research in [8], we can see that ZMMs can provide effective representation of the binary image. If a binary image is manipulated, we can measure the distance between the ZMM feature vectors of the original image and the manipulated image, then decide the authenticity of the image. We define the distance as:

$$D(f_1(x,y), f_2(x,y)) = D(Z_1, Z_2) = \sum_{i=1}^{N} (ZMM_{1,i} - ZMM_{2,i})^2 \quad (7)$$

where Z_1 and Z_2 are feature vectors of $f_1(x,y)$ and $f_2(x,y)$.

$$Z_i = (ZMM_{1,i}, ZMM_{2,i}, \cdots, ZMM_{N,i}) = (|A_{00}|, |A_{11}|, |A_{20}|, \cdots, |A_{N_{\max}N_{\max}}|)$$

where $ZMM_{k,i}$ is the k^{th} ZMM of the feature vector Z_i. Smaller distance means better match of two binary images. For binary images, the incidental manipulations are usually lossless, so we can decide the authenticity of the image by the following rule:

$$decision = \begin{cases} Incidental & D(f_1(x,y), f_2(x,y)) = 0 \\ Malicious & otherwise \end{cases} \quad (8)$$

We address the influence of malicious manipulations on ZMM feature vector. The experiments are performed on 100 256*256 binary images downloaded from the Internet. Some of them are shown in Fig.2. For each image, we choose randomly a block with different size of $l \times l$, $l = 1,2,\cdots,32$, and reverse the values of pixels within the block. The distance between the original and the tampered images are computed using Equ.(7) and shown in Fig.3, where x-axis and y-axis represent the tampered block size l and the average distance between the tampered and original images, respectively. From Fig.3, we can see that the distance increases as the tampered block size increases. So, the distance of feature vectors can reflect the degree of content change of binary images.

Fig.2. Sample binary images

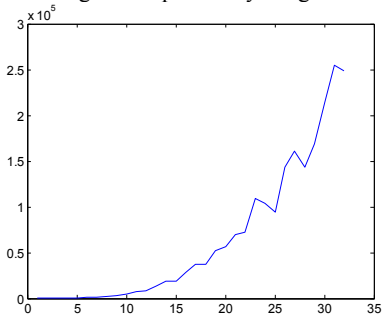

Fig.3. The relationship between distance and l

2.2. A fast algorithm for computing ZMMs

Zernike moments are usually computationally expensive. According to the invariant watermarking algorithm in [9], it takes about 5 minutes to compute the Zernike moments of a 256*256 gray image with $N_{\max} = 5$ using Matlab on a 1.5-GHz Pentium PC. By using their look-up table method, it still needs about 1 minute [9]. So we propose a fast algorithm using Matlab.

Firstly, we compare the times taken by different methods using Matlab to compute the following expression:

$$C = \sum_x \sum_y A(x,y) B(x,y) \quad (9)$$

where A and B are both 2-dimension matrices. The first method uses two nested loops, one for x and one for y. In each loop, compute $A(x,y)B(x,y)$ and add it to C. The second method uses matrix operations in Matlab by rewriting the Equ.(9) as $C = SUM(A .* B)$. In Matlab, $SUM(A)$ means sum of all elements in matrix A and $A .* B$ means multiplying corresponding elements of A and B one by one.

We test these two methods on two matrices, A and B, with size 256*256 and data-type double. The time taken to compute C in Equ.(9) is 0.2188s for the first method and 0.0023s for the second method, respectively. We can see that the second method is much faster than the first method. Because the expression to compute $A_{n,m}$ in Equ.(5) is similar to that in Equ. (9), the idea of our fast algorithm is using matrix method to replace the usually used loop method.

In Equ.(5), $f(x,y)$ can be regarded as a 2-dimension matrix. So we need to get a same-size 2-dimension matrix for $V^*_{n,m}(\rho,\theta)$. For a given image $f(x,y)$ of $M\times N$, $(1\le x\le M, 1\le y\le N)$, order n and repetition m, the method is as follows:

1) Convert the image into polar coordinates and get two matrices to contain ρ and θ. Get two matrices to contain $\exp(jm\theta)$ and $R_{n,m}(\rho)$.
2) Rewrite Equ.(1) as $mat(V_{n,m}) = mat(\exp(jm\theta)) .* mat(R_{n,m}(\rho))$, where $mat(A)$ represents the $M\times N$ matrix that contains A.
3) Rewrite Equ.(5) as $A_{n,m} = ((n+1)/\pi)SUM(f .* mat(V^*_{n,m}))$.

Table.1 shows the times taken by the proposed fast algorithm, the conventional loop method and the look-up table method mentioned in [9]. Zernike moments are computed on images in Fig.2 using Matlab on a 1.8-GHz Pentium PC, where $N_{\max} = 12$. We can see that under the same condition, our algorithm is much faster than the look-up table method mentioned in [9]. That's important for us to test the property of Zernike moments and the performance of the authentication scheme on large amount of images.

Table.1. The seconds taken by different methods to compute ZMMs of the images in Fig.2

Images	Loop method	Method in [9]	Our method
1	139.86	38.13	6.60
2	142.59	37.18	7.13
3	157.81	38.93	6.48
4	103.80	41.44	6.48

3. PROPOSED AUTHENTICATION SCHEME

3.1. Outline of the proposed algorithm

The block diagram of our scheme is shown in Fig.4.

The watermarking process is as follows:

1) Divide the original image $f_1(x,y)$ into two parts, P_1 and P_2, by a secret key K_1.
2) Generate feature vector Z_1 from P_1 and quantize it.
3) Embed quantized feature vector into P_2 using embedding method in [1] and get the watermarked image.

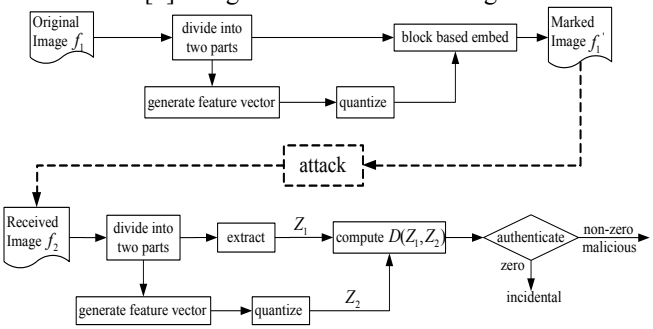

Fig.4. Embedding and authentication processes

The authentication process doesn't need the original feature vector. The process is as follows:

1) Divide the test image $f_2(x,y)$ into two parts by key K_1.
2) Generate feature vector Z_2 of $f_2(x,y)$ from P_1.
3) Extract watermark from P_2 and estimate the feature vector Z_1 of $f_1(x,y)$.
4) Compute distance between feature vectors in step 2 and 3 using Equ.(7).
5) Decide the authenticity of $f_2(x,y)$ by Equ.(8).

3.2. Image splitting and authentication

In our scheme, the quantized feature vector is embedded into the original image. If the feature vector of the watermarked image is not equal to that of the original image, then the watermarked image itself will be regarded as inauthentic by the decision rule in Equ.(8). So we have to guarantee that the feature vectors of the original image and the watermarked image are equal. We divide the binary image into two subspaces [10], one for feature generation and the other for watermark embedding.

We assume that $f_1(x,y)$ and $f_2(x,y)$ are the original and watermarked image, respectively. For $f_1(x,y)$, we generate a 2-D mask image $m(x,y)\in\{0,1\}$ controlled by a secret key K_1. By using the mask $m(x,y)$, the image $f_1(x,y)$ is divided into two parts randomly, P_1 and P_2. If $m(i,j)=0$, $f_1(i,j)\in P_1$; otherwise $f_1(i,j)\in P_2$. We demonstrate the division result in Fig.5, where Fig.5 (a), (b), and (c) are the original image, P_1 and P_2, respectively. We can see that both of P_1 and P_2 can be regarded as coarse versions of the original image. P_1 is used to generate the feature vector Z_1 and P_2 is used to embed watermark. In authentication process, the test image $f_2(x,y)$ is also divided into two parts using K_1. P_1 is used to generate feature vector Z_2 and P_2 is used to extract watermark.

Using such strategy, we can see that watermarking affects only P_2, while P_1 will not be altered. If $f_2(x,y)$ is the watermarked image or the losslessly manipulated watermarked image, the feature vectors of $f_1(x,y)$ and $f_2(x,y)$ both generated from P_1 are the same. The watermark can be extracted correctly from P_2 of $f_2(x,y)$ and restored as the feature vector of $f_1(x,y)$. The distance computed in the step 4 of the authentication process will be zero and $f_2(x,y)$ will be regarded as authentic. On the other hand, from Fig.5, we can also observe that P_1 and P_2 of a binary image are closely neighboring to each other. If the attacker wants to tamper the watermarked image, he will have to change pixels from both P_1 and P_2, thus both of the feature vector and the watermark would be modified. The distance computed in the authentication process will not be equal to zero and the tampered image will be regarded as inauthentic.

Fig.5. Image splitting

Fig.6 shows an example of authentication, where images (a), (b) and (c) are the original image, the watermarked image and the tampered watermarked image, respectively. *D* is the distance computed in step 4 of the authentication process. We can see that the watermarked image in (b) can be passed as authentic by Equ.(8) with *D*=0, while the tamper in (c) can be detected with *D*>0.

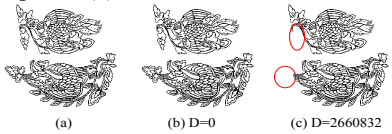

Fig.6. (a) original image (b) watermarked image (c) tampered watermarked image. The ovals point out the tampered areas.

3.3. Experimental results

We test the performance on 100 binary images downloaded from Internet. And 49 ZMMs are used to generate the feature vector. Each ZMM is quantized into 8 bits. For each image, we choose randomly a block with size of $l \times l$, $l = 1,2,\cdots,32$ and modify these blocks. The results are shown in Fig.7, where x-axis and y-axis represent l and the distance computed in the authentication process. From Fig.7, we can see that when the block size increases, the distance increases. The distance can be used to reflect the degree of the tampering.

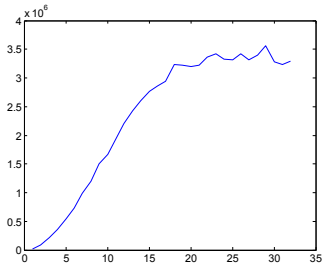

Fig.7. Experimental results

In Fig.8, we give some examples of authentication results when different types of malicious manipulations are applied to the watermarked image. From Fig.8, we can observe that the proposed scheme can detect the tamper by using Equ.(7) and Equ.(8) effectively.

4. CONCLUSION AND FUTURE WORK

In this paper, we propose a content-based watermarking scheme for binary image authentication. Our main contributions are as follows:

1) Propose to use Zernike moments to generate feature vector for binary image authentication.

2) Split the binary image into two subspaces by a random mask controlled by a secret key, thus we can decrease the influence of watermarking on the feature vector.

3) Propose a fast algorithm for computation of Zernike moments using Matlab.

4) The authentication process doesn't need the original watermark and the decision is based on a metric measure.

Our future directions include: (1) algorithm that can locate the tampered areas of binary images; (2) algorithm that can differ non-malicious operations, such as transmission errors, from malicious attacks.

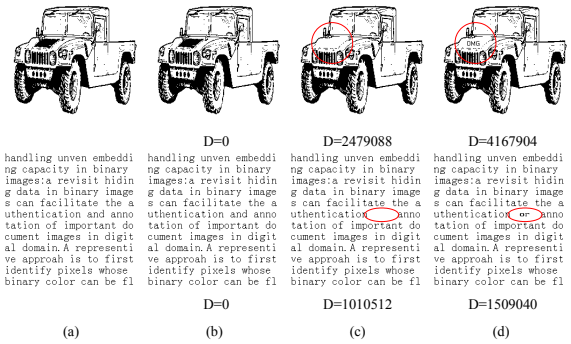

Fig.8. (a) original images (b) watermarked images (c) erasing (d) replacing. The areas in the ovals are tampered

5. REFERENCES

1. M.Wu and B.Liu, "Data hiding in binary image for authentication and annotation", *IEEE Trans.on Multimedia*, 6(4): 528-538, Aug.2004.
2. M.Wu, E.Tang and B.Liu, "Data hiding in digital binary image," *Proc.ICME'02,* pp.393-396, 2002.
3. H.Lu, A.C.Kot and J.Cheng, "Secure data hiding in binary document images for authentication", *Proc. ISCAS'04*, Vol.3, pp.806-809, 2004.
4. H.Lu, J.Wang, A.C.Kot and Y.Q.Shi, "An objective distortion measure for binary document images based on human visual perception", *Proc.ICPR' 02*, vol.IV, pp.239-242, 2002.
5. H. Liu, J. Lin and J.Huang, "Image authentication using content based watermark", *Proc.ISCAS'05*, vol.4, pp.4014-4017, 2005.
6. H.Liu, L.Zhu and J.Huang, "A hybrid watermarking scheme for video authentication", *Proc.ICIP'*06, pp.2569-2572, 2006.
7. Khotanzad and Y.H.Hong, "Invariant image recognition by Zernike moments", *IEEE Trans. PAMI.*, vol.12, no.5, pp. 489-497, 1990.
8. C.H.Teh and R.T.Chin, "On image analysis by the methods of moments", *IEEE Trans. PAMI*, vol.10, no.4, pp.496–513, Apr.1988.
9. H.S.Kim and H.K.Lee "Invariant image watermark using Zernike moments", *IEEE Trans.CSVT*, vol.13, no.8, pp.766-775, Aug.2003.
10. H.KIM and A.AFIF, "Secure authentication watermarking for binary images", *Proc.SIBGRAPI'03,* pp.199-206.
11. Qing Chen; Xiaoli Yang; Jiying Zhao, "Robust image watermarking with Zernike moments", Electrical and Computer Engineering , pp.1340 – 1343, 2005.