2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE)
December 16-18, Changchun, China

Object based self-embedding watermarking for video authentication

Yanjiao Shi, Miao Qi [*], Yinghua Lu, Jun Kong [*]
School of Computer Science and Information Technology
Northeast Normal University
Changchun, China
e-mail:{qim801, kongjun}@nenu.edu.cn

Danying Li
Changchun Rural Commercial Bank
Changchun, China

***Abstract*—Video cameras have been widely installed in public facilities for the surveillance applications. So, video authentication has become more and more important. This paper presents an object based self-embedding watermarking for video authentication. The principal content of entire video and details of moving objects are protected by a reference sharing mechanism. At the receiver, if the stego-video is authenticated as un-tampered, the details of moving objects can be restored completely. If the stego-video is judged as tampered and the tampered area is not too extensive, the principal content of tampered regions and details of moving objects can be restored.**

Keywords-Self-embedding; Video authentication; Watermarking; Reference sharing

I. INTRODUCTION

Digital watermarking techniques have recently been utilized to protect the integrity and validity of digital multimedia[1,2]. They allow users to embed watermarks, such as logos, trademarks or copyright information, into the host images or videos. The watermarks can later be extracted and used for digital multimedia authentication. In some methods referred to fragile watermarking, the watermark is referenced to the content of the original image, which is also described as "self-embedding". Information extracted from the intact regions can be used to restore the tampered regions of the received multimedia.

Nowadays, video cameras have been widely installed in public facilities for the surveillance applications. The integrity and validity of video play an important role in applications such as intelligence information gathering, criminal evidence, security surveillance, and insurance claims. However, this trustworthiness could no longer be granted since users can easily manipulate, modify or forge digital content without causing noticeable traces using easy-to-use editing software abounded on the Internet. The edited videos do not have any value for legal proof. Therefore, video authentication has become an important issue.

Many watermarking methods have been proposed for image and video authentication [3-8]. Literature [3-5] introduced a novel reference sharing mechanism for image authentication. In literature [3], The watermark consisting reference-bits and check-bits was embedded using a lossless watermarking method. At the receiver end, the reliable reference-bits were used to recover. Video can provide more information than image, so some video authentication methods also have been proposed. Literature [6] proposed a content based authentication watermarking scheme for H.264/AVC video. It could discriminate the malicious tampering from the mild signal processing and could locate the tampered location. Literature [7] embedded the timing information into the frames by chaotic system. At the receiving end, temporal and spatial tampering could be detected by the observed demodulated information. However, these video authentication methods do not have the ability to restore the tampered regions. They also neglect the distortion of the moving objects caused by embedding. In the surveillance video, what we pay more attention are the moving objects. For a stego-video, the moving objects should be in their original state as much as possible for further video analysis. In this paper, a self-embedding watermarking method taking the moving objects into consideration is proposed. In our method, the moving objects are detected first using the background subtraction technique. For each frame, the reference data derived from its principal content and details of the moving objects are scattered into itself. The reference sharing mechanism used here can improve the ability of recovering when tampering occurs. At the receiver, the integrity and validity of stego-video should be verified firstly. If there is no tamper to the stego-video, the details of the moving objects can be restored completely. Otherwise, the tampered regions can be located and their principal content can be restored first, then the moving objects are detected using the restored video frames and the details are recovered finally.

The rest of this paper is organized as follows. Section 2 describes the generation of the content to be protected and the procedure of embedding. In Section 3, recovery of tampered regions and moving objects are described. Experimental results are described in Section 4. Finally, the conclusions are drawn in Section 5.

II. WATERMARK GENERATION AND EMBEDDING

A. *Extraction of the moving objects*

The background subtracion method is employed to extract the moving objects. For a color video frame, we first decompose it's three color channels *r*, *g* and *b* to eight bit-plans b_{c8}, b_{c7}, b_{c6}, b_{c5}, b_{c4}, b_{c3}, b_{c2}, b_{c1}, respectively, $c=r$, g, b. We call b_{c8}, b_{c7}, b_{c6}, b_{c5}, b_{c4} the Most Significant Bit (MSB) dataset and b_{c3}, b_{c2}, b_{c1} the Least Significant Bit (LSB) dataset. Then set b_{c3}, b_{c2} and b_{c1} as '100' so as to approximate to the original frame. The true MSB planes and artificial LSB planes are used to extract the moving objects of every video frame by background subtraction method [8]. In the embedding procedure, the LSB planes are replaced by watermark. In order to extract the moving objects accurately and recover them losslessly, the unchanged MSB planes but not the original frame is used to extract the moving objects.

To get the final object, the pixels that any value of their *r*, *g* and *b* is larger than a threshold *T* are reserved and others are set to 0. Then we partition the foreground pixels

[*] Corresponding author

into blocks sized 8×8. If half of the pixels of a block are regarded as foreground pixels, the block is judged as foreground block. Fig. 1 shows the tiny differences of results using original frame and MSB. It illustrates that the MSB can represent the principal content of original frame. The threshold T in this paper is set $0.15\times m_c$, where m_c is the mean value of c color channel of current frame.

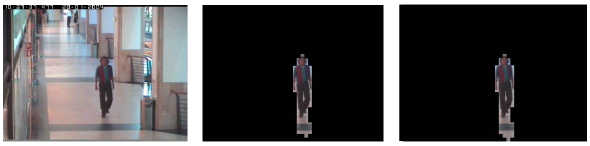


Figure 1. Results of 100th frame by background subtraction. (a) Original frame. (b) Extraction of moving objects using original frame. (c) Extraction of moving objects using MSB.

B. Generation of watermark

The MSB of the frame and LSB of the moving objects are collected to form a binary string B which is regarded as the important content to be protected. Motived by literature [3], the data embedded into the host image is reference data but not the dataset itself.

Denote the width and height of each frame as w and h, the number of blocks of moving objects as N. Collect MSB of the entire frame and LSB of the moving objects, and use L_{LSB} and L_{MSB} to represent the length of them respectively, so the length of B is $L_{LSB}+L_{MSB}$, where $L_{LSB}=N\times192\times3$. We permute and divide the B dataset into M subsets each of which contains 192 bits, that is, $192\times M=L_{LSB}+L_{MSB}$. Denote the bits in k-th subset as $c_{k,1},c_{k,2},\cdots,c_{k,192}$ and its reference bits $r_{k,1},r_{k,2},\cdots,r_{k,L}$ are generated as the following

$$\begin{bmatrix} r_{k,1} \\ r_{k,2} \\ \vdots \\ r_{k,L} \end{bmatrix} = A\cdot\begin{bmatrix} c_{k,1} \\ c_{k,2} \\ \vdots \\ c_{k,192} \end{bmatrix}, k=1,2,\cdots,M, \tag{1}$$

where L is the length of each group of reference bits, A is a pseudo-random binary matrix sized $L\times192$ .The arithmetic in (1) is modulo-2. Because of the permutation of B dataset, the L reference bits are generated from the 192 bits that scattered into the entire frame. In this step, the number of reference bits we actually want to obtain is N_{ref} ($N_{ref}=15/2\times w\times h$), so the length of reference bits of each group is $L=\lfloor N_{ref}/M\rfloor$. Then we permute the N_{ref} reference bits as a part of the watermark.

C. Embedding of watermark

For each frame, partition it into blocks sized 8×8. Accordingly, divide the N_{ref} reference bits into $(w\times h)/64$ groups. Therefore, a one-to-one relationship is established. For a color block, its MSB and corresponding reference bits are fed to hash function to get 96 hash bits. Hash function possesses several special properties and the most useful one in this paper is that any change in input will cause a very different output. Then the 480 reference bits and 96 hash bits are permuted and regarded as watermark to be embedded into the LSB planes of the block. So, a watermarked video is obtained.

The matrix A in each group can be the same or different. The permutation operation and generation of matrix A are key dependent and different permuting keys will enhance the security of the system. Moreover, different A for each group can also improve the security performance. We can use one private key to produce a pseudo-random number sequence and every number of the sequence will be used as key to permute or generate A.

III. TAMPER LOCATION AND CONTENT RESTORATION

A. Tampered Block Identification

At the receiving end, for each frame, partition it into blocks sized 8×8 first, and then extract the LSB of every block. For one block, the extracted LSB dataset is de-permuted and decomposed into reference bits and hash bits. We feed the MSB of the current block and the extracted reference bits into hash function. If the generated hash bits are equal to the extracted hash bits, the block is judged as reserved, otherwise, the block is tampered.

B. Restoration

Case 1: If there is no tamper to the video, background subtracion method is used to locate the moving objects. Then the LSB data of each frame can be solved using MSB and extracted reference bits as following:

$$\begin{bmatrix} r_{k,1} \\ r_{k,2} \\ \vdots \\ r_{k,L} \end{bmatrix} - A_M\cdot C_M = A_L\cdot C_L, \tag{2}$$

where C_M and C_L are MSB and LSB, A_M and A_L are matrices whose columns are corresponding to MSB in C_M and LSB in C_L. The left side and A_L are known and the purpose is to find C_L. If the linear equations have a unique solution, the LSB can be solved successfully. Denote the length of C_L as n_L, if and only if the rank of A_L equals to n_L, that is, the columns of A_L are linearly independent, a unique solution to equation (2) exists. For a random binary matrix sized $i\times j$, probability of its columns being linearly dependent is denoted $q(i,j)$, which can be worked out[3]:

$$q(i,1)=\frac{1}{2^i}, \tag{3}$$

$$q(i,j+1)=q(i,j)+[1-q(i,j)]\cdot\frac{2^j}{2^i}, j=1,2,\cdots,i-1, \tag{4}$$

$$q(i,j)=1, \text{ if } j>i. \tag{5}$$

Denote LSB rate $\frac{L_{LSB}}{L_{MSB}+L_{LSB}}$ as α, and n_L obeys a binomial distribution

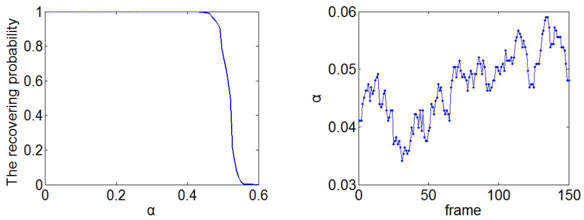
$$P_{n_L}(j)=\binom{192}{j}\cdot\alpha^j\cdot(1-\alpha)^{(192-j)}, j=0,1,\cdots,192. \tag{6}$$

The probability of all columns of A_L being linearly independent is

$$P_{LI}^{L} = \sum_{j=0}^{192} \left\{ P_{n_L}(j) \cdot \left[1 - q(L,j) \right] \right\}. \quad (7)$$

All blocks in a frame can be restored with the probability

$$P^{L} = P_{LI}^{L(L_{LSB}+L_{MSB})/192}. \quad (8)$$



(a) (b)

Figure 2. (a) The recovering probability under different α . (b) The α values of each frame.

Figure 2(a) shows the recovering probability under different α , and Fig. 2(b) shows the α values of each frame. We can see that the α values of all frames are from 0.03 to 0.06 which are in the range that can be successfully recovered with probability 1.0.

Case 2: If the stego-video has been tampered and the tampered blocks of each frame have been located, the MSB of the tampered blocks should be restored as

$$\begin{bmatrix} r_{k,e(1)} \\ r_{k,e(2)} \\ \vdots \\ r_{k,e(v)} \end{bmatrix} = A^{E} \cdot \begin{bmatrix} C_{(R,M)} \\ C_{(T,M)} \\ C_{L} \end{bmatrix}, \quad (9)$$

where $r_{k,e(1)}, r_{k,e(2)}, \cdots, r_{k,e(v)}$ are extractable reference bits, A^{E} is a matrix whose rows are corresponding to extractable reference bits, $C_{(R,M)}$, $C_{(T,M)}$ and C_L are reserved MSB, tampered MSB and LSB respectively. We can reformulate (9) as

$$\begin{bmatrix} r_{k,e(1)} \\ r_{k,e(2)} \\ \vdots \\ r_{k,e(v)} \end{bmatrix} - A_{(R,M)}^{E} \cdot C_{(R,M)} = \left[A_{(T,M)}^{E} A_{L}^{E} \right] \cdot \begin{bmatrix} C_{(T,M)} \\ C_{L} \end{bmatrix}, \quad (10)$$

where $A_{(R,M)}^{E}$, $A_{(T,M)}^{E}$ and A_{L}^{E} are matrix whose columns are corresponding to reserved MSB, tampered MSB and LSB respectively. The left side of Eq. (10) and $A_{(T,M)}^{E}$, A_{L}^{E} are known and the purpose is to find $C_{(T,M)}$ and C_L. Denote tampered rate β as the ratio between tampered blocks and all blocks, then the probability of successfully recovering can be estimated. The number of extractable reference bits in one group v obeys a binomial distribution

$$P_{v}(i) = \binom{L}{i} \cdot (1-\beta)^{i} \cdot \beta^{(192-i)}, i = 0,1,\cdots,L. \quad (11)$$

The columns of matrix $A_{(T,M)}^{E}$ n_T also obeys a binomial distribution

$$P_{n_T}(j_1) = \binom{192}{j_1} \cdot (\beta)^{j_1} \cdot (1-\beta)^{(192-j_1)}, j_1 = 0,1,\cdots,192. \quad (12)$$

Denote the columns of matrix $\left[A_{(T,M)}^{E} A_{L}^{E} \right]$ as n_{TL}, and

$$P_{n_{TL}}(j) = \sum_{k=0}^{j} P_{n_T}(k) \cdot P_{n_L}(j-k), j = 0,1,\cdots 192. \quad (13)$$

So the probability of all columns of $\left[A_{(T,M)}^{E} A_{L}^{E} \right]$ being linearly independent is

$$P_{LI}^{T} = \sum_{i=0}^{L} \sum_{j=0}^{192} \left\{ P_{v}(i) \cdot P_{n_{TL}}(j) \left[1 - q(i,j) \right] \right\}. \quad (14)$$

All blocks in one frame can be restored with the probability

$$P^{T} = P_{LI}^{T(L_{LSB}+L_{MSB})/192}. \quad (15)$$

The values of P^T depend on LSB rate α and tampered rate β as shown in Fig. 3.

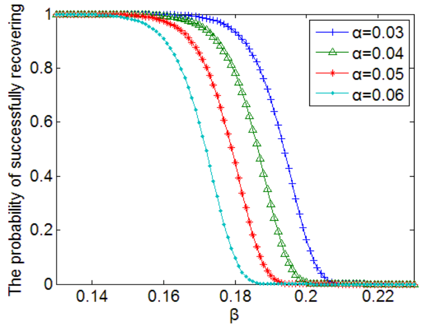


Figure 3. The values of PT with different α and β .

If the MSB of all frames can be recovered, the LSB of the moving objects of each frame also can be obtained. However, what have been got just are the values of LSB but not the positions of moving objects, so the restored video is used to get the moving objects by background subtraction and restore them losslessly.

IV. EXPERIMENTAL RESULTS

The test video used in the experiment is a color surveillance video in a hall with 150 frames sized 288×384 , and three of which are shown in Fig. 4.

(a) (b) (c)

Figure 4. Three frames of the test video. (a) 50th frame. (b) 100th frame. (c) 150th frame.

The Peak Signal to Noise Ratio (PSNR) is used to evaluate the quality of stego-video. Fig. 5 shows the PSNR of all frames. It can be seen that the PSNR values are from 37.85 dB to 37.93 dB, which are acceptable for the imperceptibility requirement.

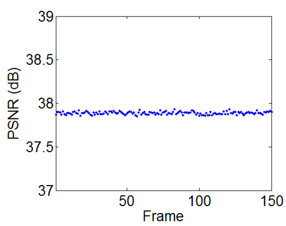


Figure 5. The PSNR values of each stego-frame.

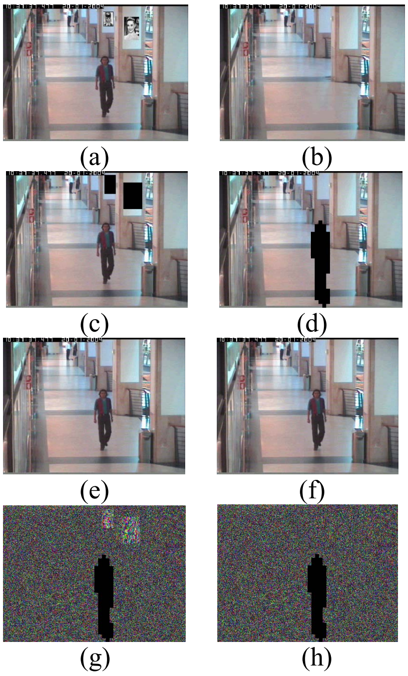


Figure 6. (a) Tampered vision of 50th frame by replacing parts of background. (b) Tampered vision of 50th frame by deleting moving objects. (c) Location of fake regions for (a). (d) Location of fake regions for (b). (e) Restored vision of (a). (f) Restored vision of (b). (g) Difference magnified by a factor 30 of original 50th frame and (e). (h) Difference magnified by a factor 30 of original 50th frame and (f).

To test the ability of recovering, the stego-video is tampered by two ways. One is to replace the regions of background with two other images (shown as Fig. 6(a)), and the other is to delete the moving objects (shown as Fig. 6(b)). The LSB rate α is 0.0231, and the tampered rates β are 0.0394 and 0.1152 respectively. The results of tamper location are shown in Fig. 6(c) and Fig. 6(d) where black regions are tampered. Fig. 6(e) and Fig. 6(f) show the results of recovering, wherein the moving objects are restored completely. Fig. 6(g) and Fig. 6(h) are the difference between original frame and restored frames from which we can see that the moving objects have been restored completely.

The PSNR values of restored video frames are shown in Fig. 7. It can be seen that the values are from 37.81 dB to 37.98 dB which indicate that this method can achieve satisfactory restoration quality.

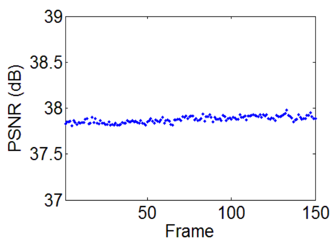


Figure 7. The PSNR values of each restored frame.

V. CONCLUSIONS

This paper proposes an object based self-embedding watermarking for video authentication. The five valid MSB plans of each frame are used to extract the moving objects by background subtraction method. Then the MSB of each frame and LSB of moving objects are collected to form a B dataset. What is embedded into the frame is the reference data but not the dataset itself. On the receiver side, if the stego-video is authenticated as un-tampered, the moving objects can be restored losslessly. If the video has been tampered, the extracted reference data and reserved MSB are used to recover the MSB of tampered regions, and then the moving objects can be obtained. Finally, the solved LSB is mapped to the moving objects.

Experimental results show that the proposed method can recover the moving objects in a video losslessly both in un-tampered case or tampered case. Moreover, if video has been tampered and the tampered area is not too extensive, the tampered video can be restored with a satisfying quality.

ACKNOWLEDGMENT

This work is supported by the Key Project of MOE (No. 109052), the Jilin Provincial Natural Science Foundation (No. 201115003), the Fundamental Research Funds for the Central Universities (No. 10QNJJ004, No. 10JCXK008), the Science Foundation for Young Teachers of Northeast Normal University (No. 09QNJJ006), the Fund of Jilin Provincial Science & Technology Department (No. 20111804) and National Nature Science Foundation of China (No.10871037, No. 11071046)

REFERENCES

[1] Deepa Kundur, Dimitrios Hatzinaoks, Digital watermarking for telltale tamper proofing and authentication, Proceedings of the IEEE, vol. 87, pp. 1167-1180, 1999.

[2] Shaohui Liu, Hongxun Yao, Wen Gao, Yongliang Liu, An image fragile watermark scheme based on chaotic image pattern and pixel-pairs, Applied Mathematics and Computation, vol. 185, pp. 869-88, 2007.

[3] Xinpeng Zhang, Shuozhong Wang, Fragile watermarking with error-free restoration capability, IEEE Transaction on Multimedia, vol. 10, pp. 1490-1499, 2008.

[4] Xinpeng Zhang, Shuozhong Wang, Zhenxing Qian and Guorui Feng, Reference Sharing Mechanism for Watermark Self-Embedding, IEEE Transactions on Image Processing, vol. 20, pp. 485-495, 2011.

[5] Xinpeng Zhang , Shuozhong Wang, Fragile watermarking scheme using a hierarchical mechanism, Signal Processing, vol. 89, pp. 675-679, 2009.

[6] Dawen Xu, Rangding Wang, Jicheng Wang, A novel watermarking scheme for H.264/AVC video authentication, Signal Processing: Image Communication, vol. 26, pp. 267-279, 2011.

[7] Siyue Chen, Henry Leung, Chaotic watermarking for video authentication in surveillance applications, IEEE Transactions on Circuits and Systems for Video Technology, vol. 18, pp. 704-709, 2008.

[8] Jun Kong , Ying Zheng, Yinghua Lu, Baoxue Zhang, A novel background extraction and updating algorithm for vehicle detection and tracking, Fuzzy Systems and Knowledge Discovery, vol. 3, pp. 464-468, 2007.