Fragile Image Watermarking for Medical Images

Jasni Mohamad Zain[1] **Nor Amizam Jusoh[2]**

[1] *Image Processing &Multimedia Group, Faculty of Computer Systems &Software Engineering*
University College of engineering & Technology Malaysia, Locked Bag12,25000 Kuantan, Pahang, Malaysia
Tel: +60-9-5492113, Fax: +60-9-5492144, E-mail: jasni@kuktem.edu.my

[2] *Image Processing &Multimedia Group, Faculty of Computer Systems &Software Engineering*
University College of engineering & Technology Malaysia, Locked Bag12,25000 Kuantan, Pahang, Malaysia
Tel: +60-9-5492130, Fax: +60-9-5492144, E-mail: amizam@kuktem.edu.my

Abstract

This paper will discuss security of medical images and review some work done regarding medical images. We will then propose a fragile watermarking scheme that can detect tamper and recover the image. Our method requires a secret key and a public chaotic mixing algorithm to embed and recover a tampered image. The scheme is also resilient to VQ attack. The purpose is to verify the integrity and authenticity of medical images. We used 800x600x8 bits ultrasound (US) greyscale images in our experiment. We test our algorithm for up to 50% tampered block. We obtained 100% recovery for spread-tampered block.

Keywords

Information hiding and watermarking, medical images.

Introduction

Security of medical images, derived from strict ethics and legislative rules, gives rights to the patient and duties to the health professionals. This imposes three mandatory characteristics: confidentiality, reliability and availability:

- Confidentiality means that only the entitled persons have access to the images;
- Reliability which has two aspects; Integrity: the image has not been modified by non-authorized person, and authentication: a proof that the image belongs indeed to the correct patient and is issued from the correct source;
- Availability is the ability of an image to be used by the entitled persons in the normal conditions of access and exercise.

Security risks of medical images can vary from random errors occurring during transmission to lost or overwritten segments in the network during exchanges in the intra- and inter-hospital networks. One must also guarantee that the header of the image file always matches that of the image data. In addition to these unintentional modifications one can envision various malicious manipulations to replace or modify parts of the image, called tampering [1].

The studies that are specifically directed to watermarking of medical images are few. Anand and Niranjan [2] propose to embed an encrypted version of the Electronic Patient Record (EPR) in the least significant bit (LSB) plane of the image. Miaou et al [3] similarly propose a LSB technique where the host image authenticates the transmission origin with an embedded message composed of various patient data (e.g ECG record), the diagnosis report and the doctor's seal. [4], propose a trusted header scheme by embedding the hash of the file header of medical standard image in the image raw data. Coatrieux et al [5] propose Region of Interest (ROI) to preserve the diagnostic zone and Region of Non Interest (RONI) whose integrity needs not be preserved and serves as the watermark carrier. Previous researchers working in the area of medical imaging have not included tamper detection and recovery in their work.

P. Wong describes a fragile marking technique in [6], which obtains a digest using a hash function. The image, image dimensions, and marking key are hashed during embedding and used to modify the least-significant bit plane of the original image. This is done in such a way that when the correct detection side information and unaltered marked image are provided to the detector, a bi-level image chosen by the owner (such as a company logo or insignia), is observed. This technique has localization properties and can identify regions of modified pixels within a marked image. However, Holliman and Memon [7] soon presented a vector quantization (VQ) counterfeiting attack that can construct a counterfeit image from a VQ codebook generated from a set of watermarked images. To solve the problem of VQ counterfeiting attack, several enhanced algorithms were proposed [8][9]. Nonetheless, they either fails to effectively address the problem or sacrifice tamper localization accuracy of the original methods [10]. Celik et al.[10] then presented an algorithm based on Wong's scheme and demonstrated that their algorithm can thwart the VQ codebook attack while sustaining the localization property.

In this paper, we propose a watermarking method for image tamper detection and recovery. We are interested in local manipulation such as additional or removal of part of an image. Our method is efficient as it only uses simple

operations such as parity checks and comparison between average intensities as compared to method proposed by Celik et. al. [10].

Methodology

We describe the watermarking embedding procedure in this section. Each image is of size M x N pixels where M and N are assumed to be a multiple of eight and the number of greylevels is 256.

Preparation

We need to prepare a one to one block (8x8 pixels) mapping sequence A → B→ C→ D → … → A for watermarking embedding, where each symbol denotes an individual block. The intensity feature of block A will be embedded in block B, and the intensity feature of block B will be embedded in block C, etc. Voyatzis and Pitas [11] presented a two dimensional, discrete Torus automorphism for creating a unique and random mapping of the pixels within an image. We use a 1D transformation based on [11] to get a one-to-one mapping:

$$\vec{B} = [(k \times B) \bmod N_b] + 1, \quad (1)$$

where $B, \vec{B}, k \in [1, N_b]$, k is a secret key (prime number), and N_b is the total number of blocks in the image.
The generation algorithm of the block-mapping sequence is as follows:

- Divide the image into non-overlapping blocks of 8x8 pixels.
- Assign a unique integer $B \in \{1,2,3,..., N_b\}$ to each block from left to right and top to bottom, where N_b= (M/8) x (N/8).
- Randomly pick a prime number $k \in [1, N_b]$.
- For each block number B, apply equation (1) to obtain $\vec{B}$, the number of its mapping block.
- Record all pairs of B and $\vec{B}$ to form the block mapping sequence.

Table 1 - Mapping Of Blocks With K=23,26 And Nb=40

k	B	1	2	3	4	5	6	21	22	23	24
23	$\vec{B}$	24	7	30	13	36	19	4	27	10	33
26	$\vec{B}$	27	13	39	25	11	37	27	13	39	25

Note that the secret key, k, must be a prime in order to obtain a one to one mapping; otherwise, the period is less than N_b and a one to many mapping may occur. Table 1 lists some parts of the mapping sequence generated with Nb=40, k=23 (prime) and 26 (not prime) respectively. In this table, $\vec{B}$ starts to repeat at B=21 when k=26, which is not a prime.

Embedding

For each block B of 8x8 pixels, we further divide it into four sub-blocks of 4x4 pixels. The watermark in each sub-block is a 3-tuple (v, p, r), where both v and p are 1-bit authentication watermark, and r is a 7-bit recovery watermark for the corresponding sub-block within block A mapped to B. The following algorithm describes how the 3-tuple watermark of each sub-block is generated and embedded:

- Set the LSB of each pixel within the block to zero and compute the average intensity of the block and each of its four sub-blocks, denoted by avg_B and avg_Bs, respectively.
- Generate the authentication watermark, v, of each sub-block as:

$$v = \begin{cases} 1 & if\ avg\_Bs \geq avg\_B, \\ 0 & otherwise, \end{cases} \quad (2)$$

- Generate the parity check bit, p, of each sub-block as:

$$p = \begin{cases} 1 & if\ num\ is\ odd, \\ 0 & otherwise, \end{cases} \quad (3)$$

where num is the total number of 1s in the seven MSBs of avg_Bs.

- From the mapping sequence generated in the preparation step, obtain block A whose recovery information will be stored in block B.
- Compute the average intensity of each corresponding sub-block As within A, and denote it avg_As.
- Obtain the recovery intensity, r, of As by taking 7 MSB in avg_As. Seven bits is used as we are using one bit for watermarking.
- Embed the 3-tuple watermark (v, p, r), 9 bits in all, onto the LSB of of each pixel in a 3x3 block within Bs as shown in fig. 1, where r1 is the MSB, e.g. if the intensity of As is 155, r1, r2, r3, r4, r5, r6 and r7 is 1, 0, 0, 1, 1, 0 and 1 respectively.

v	p	r1
r2	r2	r4
r5	r6	r7

Figure 1- Watermark positioned in the LSB of 3x3 blocks

Tamper Detection

The test image is first divided into non-overlapping blocks of 8x8 pixels, as in watermarking embedding process. For each block denoted as $\vec{B}$, we first set the LSBs of each pixel in $\vec{B}$ to zero and compute its average intensity, denoted as avg_$\vec{B}$. We then perform 2-level detection. In level-1 detection, we examine each 4x4 sub-block within one block. In level-2 detection, we treat an 8x8 block as one unit. Level-3 detection is for VQ attack resilience only. The procedure of our hierarchical tamper detection scheme is described in the following:

Level-1 detection

- For each sub-block $\vec{B}$s of 4x4 pixels within the block $\vec{B}$, perform the following steps:
- Extract v and p from $\vec{B}$s.
- Set the LSBs of each pixel within each $\vec{B}$s to zero and compute the average intensity for each sub-block $\vec{B}$s, denoted as avg_$\vec{B}$s.
- Count the total number of 1s in avg_$\vec{B}$s and denote it as Ps.
- Set the parity check bit p' of $\vec{B}$s to 1 if Ps is odd, otherwise, set it to 0.
- Compare p' with p. If they are not equal, mark $\vec{B}$s as tampered and complete the detection for $\vec{B}$s.
- Set the algebraic relation v'=1 if avg_$\vec{B}$s>=avg_$\vec{B}$, otherwise, set it to 0.
- Compare v' with v. If they are not equal, mark $\vec{B}$s as tampered and complete the detection for $\vec{B}$s; otherwise mark it valid.

Level-2 detection

For each block of size 8x8 pixels, mark this block tampered if any of its sub-block is marked tampered; otherwise mark it valid.

Level-3 detection

For each valid block $\vec{B}$ of size 8x8 pixels, perform the following steps:

- Find the block number of block C, where block C is the one in which the intensity feature of block $\vec{B}$is embedded.
- Locate block C.
- If block C is marked tampered, assume block $\vec{B}$ is valid and complete the test.
- If block C is valid, perform the following steps:
- Obtain the 7-bit should-be intensity of each $\vec{B}$s by extracting the LSBs from each pixels in the corresponding block within block C, padding one zero to the end to make an 8-bit value.
- Compare with avg_$\vec{B}$s and mark $\vec{B}$ tampered if they are different.

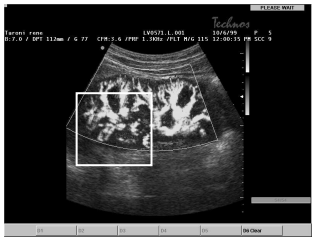
Image Recovery

After the detection stage, all the blocks are marked either valid or tampered. We only need to recover the tampered blocks and leave those valid blocks as they are. For convenient, we call the tampered block, block B and the block embedded with its intensity, block C. The restoration procedure for each tampered block is described as follows:

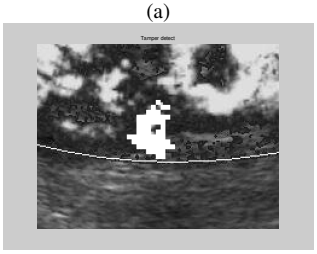
- Calculate the block number for block C.
- Locate block C
- Obtain the 7-bit intensity of each sub-block within block B, padding one zero to the end to make an 8-bit value.
- Replace the new intensity of each pixel within the sub-block with this new 8-bit intensity.
- Repeat step 3 and 4 for all sub-blocks within block B.
- Replace the new intensity of each pixel within the sub-block with this new 8-bit intensity.
- Repeat step 3 and 4 for all sub-blocks within block B.

Results

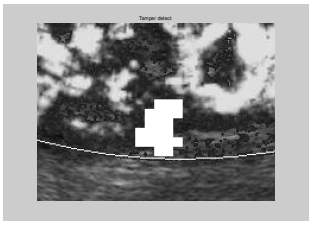
We carried out two experiments to test our algorithms. We watermarked our test image with peak signal to noise ratio of 54.8 dB. In the first experiment, we tamper a watermarked image by adding a clone of part of the original image as in Figure 2 (a). Level-1 detection left some



(a)



(b)



(c)

Figure 2. (a) Tampered image (b) level-1 detection with some areas undetected (c) level-2 with 100% detection

areas undetected as seen in Figure 2(b). 100% tamper was detected using level-2.

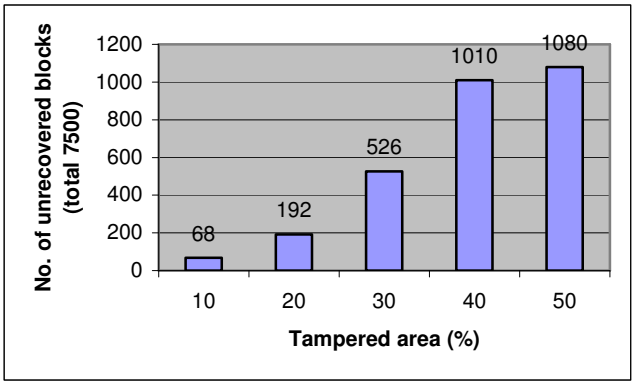


Figure 3 - Unrecovered blocks for single tampered block

We used spread tampering and single block tampering ranging from 10% to 50% as shown in fig. 3 with k=3739 as shown in Fig. 4 for our second experiment to determine recovery rate of our method. Our results showed that we could recover all tampered areas for spread-tampered blocks and the result for single tampered block is shown in Fig. 2.

Conclusion

This paper has discussed the security of medical images and review some work done regarding medical images. We also proposed a watermarking scheme that can detect tamper and recover the image. The purpose is to verify the integrity and authenticity of images. The experimental results demonstrate that the precision of tamper detection and localization is close to 100% after level-2 detection. The tamper recovery rate is better than 86% for a less than half tampered image.

In keeping distortion level low, we could make sure that the watermarked image can still be valuable for other purpose, such as case study in school, with the patient's confidential information is not being disclosed.

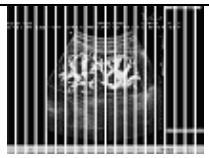
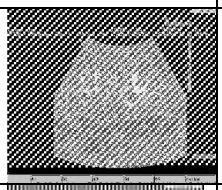
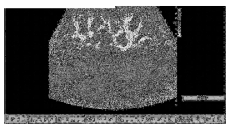
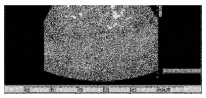
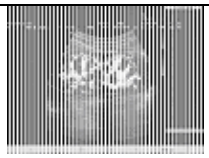
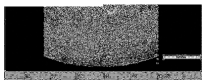
Tamper rate	Spread Tampered blocks	Single tampered block
10%		
20%		
30%		
40%		
50%		

Figure 4 - Tampered Images

References

[1] M.L. Miller, I.J. Cox, J.M.G. Linnartz and T. Kalker, "A Review of Watermarking Principles and Practices," in Digital Signal Processing for Multimedia Syatems, K.K. parhi and T. Nishitani Eds. New York: Marcel Dekker Inc., 1999, pp. 461-485.

[2] D. Anand and U. Niranjan, "Watermarking Medical Images with Patient Information," in IEEE/EMBS Conference, 1998, pp. 703-706.

[3] S.-. Miaou, C.-. Hsu, Y.-. Tsai and H.-. Chao, "A secure data hiding technique with heterogeneous data-combining capability for electronic patient records," in 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Jul 23-28 2000, 2000, pp. 280-283.

[4] B. Macq and F. Dewey, "Trusted Headers for Medical Images," in DFG VIIII-DII Watermarking Workshop, 1999.

[5] G. Coatrieux, B. Sankur and H. Maitre, "Strict Integrity Control of Biomedical Images," in SPIE Conf. 4314: Security and Watermarking of Multimedia Contents III, 2001.

[6] P.W. Wong, “A public key watermark for image verification and authentication”, in Proceedings of the IEEE International Conference on Image Processing, Chicago, IL, October 1998, pp. 455-459.

[7] M. Holliman, N. Memon, “Counterfeiting attacks on oblivious block-wise independent invisible watermarking schemes, IEEE Trans. Image Processing, 9(2000), pp. 432-441.

[8] J. Fridrich, M. Goljan, A.C. Baldoza, “New fragile authentication watermark for images”, in Proceedings of the IEEE International Conference on Image Processing, Vancouver, BC, Canada, September 2000, pp. 10-13.

[9] P.W. Wong, N. Memon, “Secret and public key authentication schemes that resist vector quantization attack”, Proceeding SPIE 3971 (75), 2000, pp. 417-427.

[10] M. U. Celik, G. Sharma, E. Saber, A.M. Tekalp, “Hierarchical watermarking for secure image authentication with localization”, IEEE Trans. Image Processing, 11(6) 2002, pp.585-594.

[11] G. Voyatzis, I. Pitas, “ Applications of toral automorphisms in image watermarking”, in Proceedings of the International Conference on Image Processing, vol. II, 1996, pp. 237-240.