Two-Layer Binary Image Authentication With Tampering Localization

Huijuan Yang and Alex C. Kot

School of Electrical and Electronic Engineering,
Nanyang Technological University, Singapore 639798.
Email: ehjyang, eackot@ntu.edu.sg

Abstract— In this paper, a novel two-layer blind binary image authentication scheme is proposed, in which the first layer is targeted at the overall authentication and the second layer is targeted at identifying the tampering locations. The "flippability" of a pixel is determined by the three "Connectivity-Preserving" transition criterions described in [4]. The image is partitioned into multiple Micro-Blocks and the Micro-Blocks are classified into eight categories. The Block Identifier is defined adaptively for each class and embedded in those "Qualified" and "Self-Detecting" Micro-Blocks in order to identify the tampered locations. The Block Identifier is defined in such a way that any changes occurred either to the "Qualified" or its neighboring "Un-qualified" Micro-Blocks will render the retrieved Block Identifier different from the one embedded. Experimental results validate the arguments made. Discussions on the accuracy of localization of tamperings are provided.

I. INTRODUCTION

Authentication of digital documents has aroused great interest due to many important documents are digitized and stored, e.g., fax documents, insurance documents and personal documents. It is becoming important on how to ensure the authenticity and integrity of these documents? On the other hand, the availability of the powerful image editing software has made copying and editing an image much more easier. Authentication and detection of tamperings and forgery is thus of primary concern. Data hiding for binary images authentication has been a promising approach to alleviate these concerns.

In the past few years, many papers proposed new techniques for document watermarking and data hiding, e.g., text line, word or character shifting, boundary modifications, fixed partitioning the image into blocks, modification of character features, modification of run-length patterns and modification of halftone images [1]. Recently, several pattern-based methods are proposed in [2][3][4][5]. Wu *et al*. propose to employ a visual distortion table to assess the "flippability" of pixels in 3×3 blocks. Shuffling technique is applied to equalize the uneven embedding capacity of the image. However, it is not easy to find a good shuffling key to ensure that each block in the shuffled domain has at least one "flippable" pixel, therefore, a larger block size is required so that the capacity is not large. Pairs of contour patterns are used to trace the contour to find the data embedding locations in [3]. The Connectivity-Preserving pattern based approach [4] handles the uneven "embeddability" of the host image by embedding the watermark only in those "embeddable" blocks. Small overlapping blocks are employed to achieve large capacity.

The tampering localization and authentication for gray or color images based on public and private key encryption technique is reported in [6]. To the best of our knowledge, very few papers reported in literature discuss about the problem of tampering localization for binary images. This maybe due to the facts that the capacity of binary images normally is not as large as that of gray or color images and the "flippable" pixels are unevenly distributed. Therefore, it is difficult to identify the tampered locations. In [7], the image is divided into multiples of 128×128 block and each block is shuffled. The whole image is divided into three regions: region A is used to store the second layer Digital Signature (DS). The remaining region is further divided into regions B and C, where the first level DS is computed on region C and inserted into region B. Finally, compute the fingerprint of region B and C and insert it into region A. It is noticed that any modifications which maintain the parities of blocks in region A cannot be detected. Further improvements are reported in [8], in which the blocks are chained and the DS computed from previous block is fed to the next block for the DS calculation. However, the last block still suffers a parity attack and from the first to the last block, the probability of detecting the parity attack decreases. In addition, the all white or black uniform area does not participate the block chaining process, which makes it easy to be tampered.

In this paper, we propose a two-layer authentication technique for binary images. The overall authentication is achieved in the first layer by hiding the Cryptographic Signature (CS) of the image. The localization of tampering is achieved in the second layer by embedding the Block Identifier (BI) in the "Qualified" or "Self-Detecting" Micro Blocks (MBs). The total Micro-Blocks are classified and the Block Identifier is defined adaptively for each class. The proposed method can be applied for binary images authentication and tampering localization. This paper is organized as follows. The "flippability" criterion and the block signature generation is described in Section II. The detection of tampering location is discussed in Section III. The experimental results and discussions are presented in Section IV and Section V concludes the paper.

II. THE "FLIPPABILITY" CRITERION AND BLOCK SIGNATURE GENERATION

A. The "Flippability" Criterion

Recall that in [4], the "flippability" of a pixel is determined by three transition criterions defined in a 3×3 block, which are calculated from the center pixel to its eight neighbors, i.e., the horizontal and vertical (VH), interior right angle (IR) and sharp corners (C) transitions are calculated before and after flipping the center pixel. If the transition numbers do not change, it implies that flipping the pixel will not destroy the connectivity between pixels and does not create extra clusters as well [4]. This invariant feature can be utilized to locate the "flippable" pixels in the watermarked image. Patterns that satisfy and are excluded by the criterions are listed in Fig. 1.

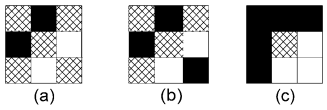

Fig. 1. Patterns that (a) satisfy VH transition, (b) are excluded by IR transition and (c) are excluded by C transition, excluding the patterns that differ only by rotation, complement and mirroring. Pixels in grid represent "don't care" pixels.

B. The Block Signature Generation

The block signature generation is proposed in [5]. Each Micro-Block is divided into 9 regions and each region consists of multiple Finer Blocks (FB) as is shown in Fig. 2. Each Finer

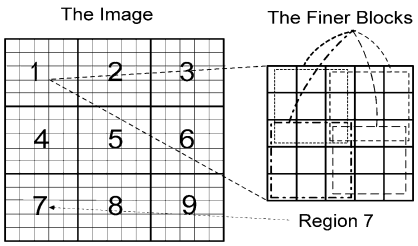

Fig. 2. The Micro-Blocks and the Finer Blocks.

Block is of size $m \times n$ and centered at $f(x, y)$, where $m = 2a+1$, $n = 2b+1$, a and b are nonnegative integers. The feature code (fc) of the Finer Block is

$$fc(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} g(s,t) \otimes f(x+s, y+t); \qquad (1)$$

where the weightage $g(s,t)$=1, 2, 4, ..., 2^{r-1} depends on the locations of the pixels in the Finer Block and $\otimes$ represents element wise multiply. The mean value of the feature code $\bar{fc}$ of the (i,j)th region is given by

$$\bar{fc}(i,j) = \frac{1}{u \times v} \sum_{x=1}^{u} \sum_{y=1}^{v} fc(x,y) \qquad (2)$$

where u and v denote the number of the Finer Blocks in each region of the Micro-Block in x and y direction respectively.

The "Block Signature" (BS) for each region of the Micro-Block is obtained by mapping the $\bar{fc}$ to a binary bit via a Look Up Table (LUT), that is

$$BS(i,j) = LUT(\bar{fc}(i,j)) \qquad (3)$$

This mapping will create an uncertainty of $2^{m \times n}$ for each Finer Block. In the proposed scheme, the size of the Finer Block is chosen to be 3 × 3.

III. THE TAMPERING DETECTION AND LOCALIZATION

A. Micro-Block Classification

The "flippability" of candidate pixels in each overlapping 3 × 3 block is determined by the transition criterions described in [4]. Thereafter, the Micro-Blocks are classified into eight categories based on the number of "flippable" pixels N_f as is shown in Table I, where "∧" is the logical "and" operation.

The reason for classifying the Micro-Blocks based on N_f lies in the facts that once a location is tampered, the "flippability" condition will most probably change and N_f will change. Hence, the class which it belongs to will change as well.

TABLE I
The Classification of The Micro-Blocks

CLASS	CONDITIONS
C_0	the MB is all white
C_1	the MB is all black
C_2	$(\text{MB} \notin \{C_0, C_1\}) \wedge (N_f = 0)$
C_3	$(N_f \geq 1) \wedge (N_f < T_0)$
C_4	$(N_f \geq T_0) \wedge (N_f < T_1)$
C_5	$(N_f \geq T_1) \wedge (N_f < T_2)$
C_6	$(N_f \geq T_2) \wedge (N_f < T_3)$
C_7	$N_f \geq T_3$

B. The Distance Computation for Consecutive "Qualified" Micro-Blocks

For $MBs \in \{C_4, C_5, C_6, C_7\}$, these blocks are named as the "Qualified" Micro-Blocks ($QMBs$) which will participate the block chaining process. The distance between two consecutive "Qualified" Micro-Blocks is computed as follows

a) Record the index of the Micro-Block for both x and y directions if the "Qualified" Micro-Block is the first block. Otherwise, calculate the relative distance D_x and D_y between the current and its previous "Qualified" Micro-Block for vertical and horizontal directions respectively.
b) Assign "0" and "1" to sign bit S_y for the horizontal distance D_y to be negative and positive respectively.
c) Calculate the relative distance for the horizontal direction as $D_y = (K_c - 1) + (N_y - K_p)$, if $(S_y = 0) \wedge (D_y > \lfloor N_y/2 \rfloor)$. where N_y is the number of the Micro-Blocks along the horizontal direction, K_c and K_p are the index of the current and its previous "Qualified" Micro-Block.

C. The Block Identifier Formation

The Block Identifier consists of Block Signature BS, the distances between two consecutive "Qualified" Micro-Blocks D_x and D_y, the sign bits S_y of D_y and the Inter Block Features (F_{IB}) between two consecutive "Qualified" Micro-Blocks. The strategy for formulating the Block Identifier is detailed as

1) Divide the image into multiple Micro-Blocks.
2) Determine the "flippability" in each Finer Block and compute N_f in each Micro-Block.
3) Generate the Block Signature for each Micro-Block by setting the "flippable" locations to a fixed value, e.g., 0s, followed the steps discussed in Section II-B.
4) Classify the Micro-Blocks into eight classes based on Table I.
5) Compute the distance between the two consecutive "Qualified" Micro-Blocks. While for $MBs \in \{C_3\}$, the Micro-Block does not participate the block chaining process, however, the Block Signature will be embedded for "Self-Detecting" changes.
6) Represent the horizontal distance D_y and vertical distance D_x by a fixed length binary sequence, e.g., currently $D_x = 3\ bits$ and $D_y = 4\ bits$ are chosen respectively.
7) Generate the Inter Block Feature (F_{IB}). The Inter Blocks (IBs) are those blocks which lie between two "Qualified" Micro-Blocks, i.e., $IBs \in \{C_0, C_1, C_2, C_3\}$. the number of Micro-Blocks in each class, e.g., C_0, C_1, C_2, C_3 is computed. The F_{IB} consists of 3 bits by randomly mapping the number of the minority, the number of the majority of the IBs, and the value of the minority and majority blocks to binary bits respectively.

8) Form the Block Identifier. The Block Identifier defined in Fig. 3 is given by

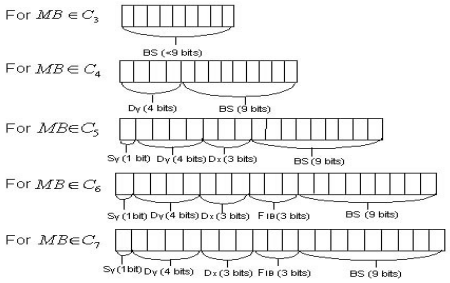

Fig. 3. The Block Identifier for different classes.

- If $MB \in C_3$, $BI = BS$.
- If $MB \in C_4$, $BI = D_y \parallel BS$.
- If $MB \in C_5$, $BI = S_y \parallel D_y \parallel D_x \parallel BS$.
- If $MB \in C_6$, $BI = S_y \parallel D_y \parallel D_x \parallel F_{IB} \parallel BS$.
- If $MB \in C_7$, $BI = S_y \parallel D_y \parallel D_x \parallel F_{IB} \parallel BS$.

where "$\parallel$" denotes the "concatenation" operation.

D. The Data Embedding Process

The Micro-Blocks are employed for data hiding unit in order to provide tampering localization information. The data embedding process is detailed as follows

1) Divide the image into multiple Micro-Blocks.
2) Determine the "flippability" in each Finer Block and compute N_f in each Micro-Block.
3) Classify the Micro-Blocks into eight categories and form the Block Identifier.
4) Embed the Block Identifier on the "flippable" pixels in each Micro-Block to enforce the odd-even feature of the corresponding 3×3 blocks.
5) For $MB \in \{C_7\}$, embed the Cryptographic Signature in the "flippable" pixels to carry the overall authentication data after embedding the Block Identifier. The Cryptographic Signature generation process is detailed in [4].

It is worthwhile noticing that the intermediate image is generated by setting all the "flippable" locations, which can be determined both in the embedding and extraction process, to fixed values, e.g., 0s. The data embedding process is shown in Fig. 4, where the process shown in dashed lines is used to generate Cryptographic Signature.

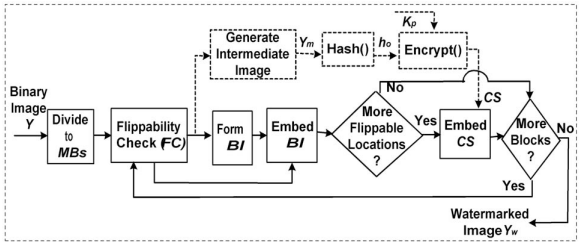

Fig. 4. The block diagram of data embedding process.

E. The Data Extraction, Authentication and Tampering Location Detection

Since the number of "flippable" pixels in each Micro-Block does not change in the data hiding process, the same process can be carried out to determine the "flippable" pixels. Thereafter, the Micro-Blocks are classified, the distance between two consecutive "Qualified" Micro-Blocks and the "Inter-Block Features" are computed. Hence, the Block Identifier can be created. Detection of the tampering location is done by comparing the extracted BI'_e with those calculated from the watermarked image BI_w and authentication of the watermarked image is done by comparing the hash value of the watermarked image h_w with the one extracted h'_e. The data extraction and tampering localization process is detailed as follows

1) Perform the same process as that in the data embedding process to form the BI_w for the watermarked image.
2) Identify the current block as "tampered" if the new computed Block Signature BS_w is different from the one extracted from the watermarked image BS_e.
3) Identify the Inter-Blocks between two consecutive "Qualified" Micro-Blocks as "tampered" if the new calculated sign S_{yw}, the vertical distance D_{xw}, the horizontal distance D_{yw} or the F_{IBw} are different from those extracted. The tampered area lies between the previous and the current "Qualified" Micro-Blocks.
4) Verify the integrity and authenticity.
   - Extract CS_w from those blocks containing more "flippable" pixels than those required to embed the BIs, e.g., $MBs \in \{C_7\}$ by computing the odd-even features of the corresponding 3×3 blocks.
   - Decrypt CS_w by providing the public key K_{pub} of the authorized user or owner to $Decrypt()$ to obtain the hash value of the original image h'_e.
   - Perform the same process to generate h_w of the watermarked image, e.g., find the "flippable" locations, clear those "flippable" locations to generate the intermediate image Y'_w. Apply the hash function $Hash()$ to Y'_w to obtain the hash value of the watermarked image h_w.
   - Compare h'_e with h_w gives the authentication results.

The data extraction, authentication and the tampering detection and localization process is shown in Fig. 5.

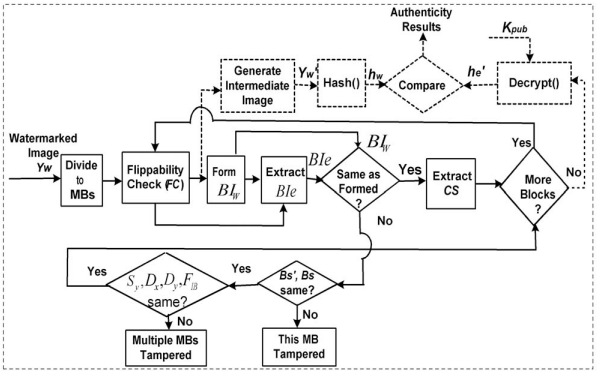

Fig. 5. The block diagram of data extraction, authentication and tampering detection and localization process.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

Extensive experiments are carried out to test the efficiency of tampering detection and localization. The results are shown in Fig. 6.

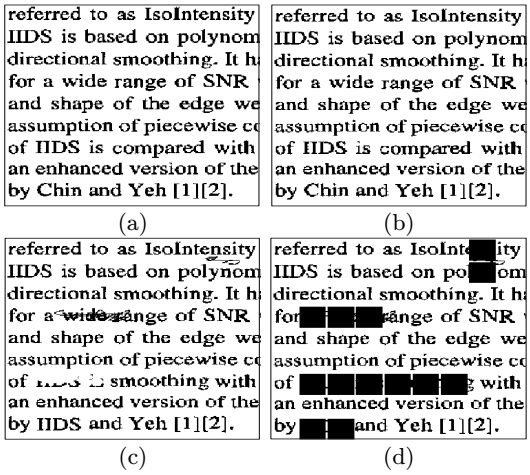

Fig. 6. The data hiding and tampering detection and localization results. (a) the original image of size 280×280, (b) the watermarked image with 1438 bits embedded (MB size of 31×31), (c) the tampered image. Several regions in the image are tampered by erasing, cropping, cut and paste, and (d) the tampering detection and localization results. The blocks painted in black are those blocks which have been tampered.

In the experiment, the parameters are chosen as $T_0 = 9$, $T_1 = 13$, $T_2 = 17$ and $T_3 = 20$ based on the definition of the Block Identifier. The overall authentication results are similar to that described in [4]. It is easy to see that good localization can be achieved by employing the small size of Micro-Blocks. Generally, the accuracy of the localization is related to the size of the tampered regions and the block size of the Micro-Blocks. The smaller the size of the Micro-Blocks, the more accurate the localization results are. Embedding Block Identifier consumes most of the capacity, however, if the block size of the Micro-Blocks is chosen to be sufficiently large, the decrease of the capacity is not obvious. Therefore, a good compromise should be made between the capacity for embedding the other payload watermark, e.g., the Cryptographic Signature and the accuracy of identifying the tampered locations. Miss detection of the tamperings may occur for those MBs that do not have enough "flippable" pixels to embed the BS, e.g., $MBs \in \{C_3\}$ or those "Qualified Blocks", e.g., $MB \in \{C_4, C_5\}$, which do not have enough "flippable" pixels to carry a complete BI. To evaluate the performance of the proposed scheme, the Miss Detection Rate (MDR) defined as the ratio of the Number of Miss Detected Tamperings ($NMDT$) and the Total Number of Tamperings (TNT) is employed and given by

$$MDR = \frac{NMDT}{TNT} \times 100\% \tag{4}$$

Forty text images of different sizes and resolutions are used in the experiments. The tamperings include: Erasure-erase characters or words; Insertion-insert characters or words; Cut and Paste-cut the characters and words and paste them in other locations; Tamper In Blank-tamper the black or white uniform regions. Among total 242 tamperings, only 4 cannot be correctly detected and localized, which gives $MDR = 1.65\%$. All these tamperings belong to very small erasure.

The proposed scheme for identifying the tampered locations is based on the observation that any tamperings occurred to the "Qualified" Micro-Block will change its class type or its Block Signature. On the other hand, any tamperings occurred between two consecutive "Qualified" Micro-Blocks may render a new "Qualified" Micro-Block be generated and hence the distance between the two consecutive "Qualified" Micro-Blocks will change. In addition, even if the distance between two consecutive "Qualified" Micro-Blocks may not change, the features of the "Un-qualified" Micro-Blocks between them will change. The secrecy of the random sequence which is used to map the features will further make it difficult for a tampering to go undetected. However, if the image contains large uniform or "non-flippable" regions between the two "Qualified" Micro-Blocks, the number of "Qualified" Micro-Blocks will become less and therefore the localization results may not be so accurate and localized. For very small images, the capacity may not be large enough to embed both Block Identifier and Cryptographic Signature. In this case, only Cryptographic Signature may be embedded. Generally, a Message Authentication Code of length 128 is considered to be secure.

In order to tackle the sensitivity of the proposed scheme to random noise, Error Correction Coding (ECC) can be applied to the watermark bits, which consists of the Block Identifier and Cryptographic Signature, e.g., $BCH(31, 26, 1)$ can be used to encode the watermark for each Micro-Block. Of course, in using the ECC, the total capacity will drop significantly. However, robustness has increased at the cost of the decrease of the security, e.g., some of the tamperings may not be detected.

V. CONCLUSIONS

In this paper, a novel blind two-layer data hiding for binary images authentication and tampering localization scheme is proposed. The "flippability" of a pixel is determined based on the "Connectivity-Preserving" criterions in a 3×3 block. The novel way of dividing the images into Micro-Blocks and embed Block Identifier in each Micro-Block is effective in detecting tamperings occurred to the watermarked image, both in "Qualified" Micro-Blocks and the "Un-qualified" Micro-Blocks. The Block Signature, the distance between two "Qualified" Micro-Blocks and the features of the "Un-qualified" Micro-Blocks are effective in tracking the changes. The proposed two-layer authentication: the first layer which is for the overall image authentication and the second layer which is for tampering detection and localization, is effective in detecting any changes and in the meantime the locations being tampered can be identified. Experimental results enforce the arguments made.

References

[1] M. Chen, E. K. Wong, N. Memon and S. Adams, "Recent Development in Document Image Watermarking and Data Hiding", *Proc. SPIE* vol. 4518, pp. 166-176, August 2001.

[2] Min Wu, and Bede Liu, "Data Hiding in Binary Images for Authentication and Annotation", *IEEE Trans. on Multimedia*, vol. 6, no. 4, pp. 528-538, August 2004.

[3] Q. Mei, E. K. Wong and N. Memon, "Data Hiding in Binary Text Document," *Proc. of SPIE*, Vol. 4314, pp. 369-375, 2001.

[4] H. Yang and Alex C. Kot, "Date Hiding for Text Document Image Authentication by Connectivity-Preserving", *Proc. of the IEEE ICASSP'*2005, vol. 2, pp. 505-508, March 2005.

[5] H. Yang, Alex C. Kot and Jun Liu, "Semi-fragile Watermarking For Text Document Images Authentication", *IEEE Int. Symp. on Circuits and Systems*, vol. 4, pp. 4002-4005, May 2005.

[6] P. W. Wong, Memon, N. , "Secret and Public Key Image Watermarking Schemes for Image Authentication and Ownership Verification," *IEEE Trans. on Image Processing*, vol. 10, no. 10, pp. 1593-1601, Oct. 2001.

[7] H. Y. Kim and de Queiroz, R. L., "Alteration-Locating Authentication Watermarking for Binary Images," in *Proc. Int. Workshop on Digital Watermarking*, pp. 125-136, 2004.

[8] H. Y. Kim and de Queiroz, R. L., "A public-key authentication watermarking for binary images", *Proc. of the IEEE ICIP'*2004, vol. 5, pp. 3459 - 3462, Oct. 2004.