A New Approach to Authentication of Binary Images for Multimedia Communication With Distortion Reduction and Security Enhancement

Chih-Hsuan Tzeng and Wen-Hsiang Tsai, *Senior Member, IEEE*

***Abstract*—A new approach to binary image authentication in multimedia communication with distortion reduction and security enhancement is proposed. Special codes are embedded into the blocks of given images and verified to accomplish the authentication purpose. Enhancement of security in detecting tampered images is achieved by randomly generating the codes and embedding them into randomly selected locations in the image blocks. The reduction of image distortion coming from pixel value replacement in code embedding is carried out by allowing multiple locations for embedding the codes. Security analysis and experimental results are also included to show the effectiveness of the proposed approach.**

***Index Terms*—Authentication codes, binary images, code embedding and verification, distortion reduction, image authentication, security protection.**

I. Introduction

IMAGE transmission is a major activity in today's communication. With the advance of digital technologies, it is now easy to modify digital images without causing noticeable changes, resulting possibly in illicit tampering of transmitted images. It is so desirable to design effective algorithms for *image authentication*, aiming at checking the fidelity and integrity of received images.

This authentication problem is difficult for binary images because of their simple binary nature. Embedding of authentication signals into binary images will cause destruction of image contents, and so arouses possible suspect from invaders. Therefore, a good solution should take into consideration not only the security issue of reducing the possibility of being tampered with imperception but also the effectiveness of reducing image distortion resulting from authentication signal embedding. In this study, we propose an authentication method for binary images with a good balance between the mutually conflicting goals of distortion reduction and security enhancement.

So far, there are very few researches about binary image authentication, though some works on hiding data in binary images have been reported. Tseng *et al.* [1] mapped block contents into the data to be hidden. Wu *et al.* [2] embedded bits in image blocks by pattern matching. The method can be used for image authentication. In [3] and [4], secret data were embedded into binary images by manipulating dithering patterns. In [5], [6], spaces in textural document images were used for embedding copyright-protecting watermarks.

The method proposed in this letter for binary image authentication is based on the idea of embedding randomly-generated codes, called *authentication codes*, into the blocks of a given *cover image*, resulting in a *stego-image*. Authentication is achieved by verifying the codes in the blocks of a given stego-image. Tampering of a stego-image block will destroy the code in the block and cause an erroneous verification result. To reduce image distortion resulting from embedding codes in a cover image, a new technique of allowing more than one pixel group, called a *code holder*, for embedding a code is proposed, and the *optimal code holder* whose pixel values are minimally different from those of the authentication code is chosen to embed the code. Detailed analysis of the probability for a tampered image to be verified to be authentic is also conducted. The result may be utilized to decide strategies for choosing proper authentication code lengths as well as appropriate numbers of code holders.

In the remainder of this letter, we first describe the proposed authentication code embedding and verification processes in Section II, followed by the security analysis in Section III. Some experimental results are given in Section IV, followed by a conclusion in Section V.

II. Proposed Authentication Method

The input to the proposed authentication code embedding process includes a cover image I with L blocks, two keys K_1 and K_2, and two random number generators f_1 and f_2. The output is a stego-image I' in which authentication codes are embedded. The steps of the algorithm are as follows.

1) Use f_1 with K_1 as the seed to generate a sequence of L random numbers $c_1, c_2, \ldots, c_L$, each with m bits, as the authentication codes.
2) Embed each c_i into a corresponding block B_i in I in the following way to yield I'.
    2.1) Use f_2 with K_2 as the seed to select randomly in B_i a certain number, say n, of code holders,

Manuscript received January 28, 2003. The associate editor coordinating the review of this letter and approving it for publication was Dr. C.-K. Wu. This work is supported in part by the MOE Program for Promoting Academic Excellency of Universities under Grant 89-1-FA04-1-4.

The authors are with the Department of Computer and Information Science, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: chtzeng@cis.nctu.edu.tw; whtsai@cis.nctu.edu.tw).

Digital Object Identifier 10.1109/LCOMM.2003.815656

1089-7798/03$17.00 © 2003 IEEE

each including a certain number, say m, of *ordered* pixels.

2.2) Compare c_i with each code holder g_k by matching respectively the m corresponding bits in c_i and g_k, and find the *optimal* code holder $g_{\rm opt}$ with the minimum number of different bit values.

2.3) Replace, if necessary, the value of each bit in g_{opt} with the corresponding one in c_i to complete the code embedding work.

As an example, let the pixels in a given 3×3 image block B in a raster scanning order be denoted as P_1 through P_9 whose contents, when concatenated, are 011 110 010. A 2-bit authentication code $c = 01$ generated by f_1 is to be embedded in B. Assume that three code holders H_1, H_2, and H_3 are allowed, which are decided by f_2 to be the pixel pairs (P_1, P_5), (P_2, P_9), and (P_5, P_1), respectively. Note that the two pairs $H_1 = (P_1, P_5)$ and $H_3 = (P_5, P_1)$ are *distinct* as the pixels in each pair are regarded to be *ordered*. Accordingly, since the contents of the three code holders in sequence are (0, 1), (1, 0), and (1, 0), the optimal code holder is just $H_1 = (P_1, P_5)$ because the bits of c match those in H_1 exactly. And so no bit replacement is necessary when c_i is embedded into H_1.

On the other hand, the proposed authentication code verification process is as follows. The input includes a stego-image I' with L blocks, as well as the two keys K_1 and K_2 and the two random number generators f_1 and f_2 used in the authentication code embedding process. The output is an authentication report for I'.

1) Re-generate the L m-bit authentication codes $c_1, c_2, \ldots, c_L$ using f_1 and K_1.

2) Verify each c_i in a corresponding block B_i in I' in the following way:

   2.1) Use f_2 and K_2 to reselect in B_i the n m-pixel *code holders*.

   2.2) Compare c_i with each code holder g_k by matching the m corresponding bits in c_i and g_k, and label B_i as *tampered* if there exists no code holder with content identical to c_i; or as *authentic*, otherwise.

3) If there exists any block being labeled tampered, report negative fidelity and output all tampered blocks; otherwise, regard the entire image of I' untampered.

III. Security Analysis

First, we want to analyze the probability that a tampered image block B is erroneously verified to be authentic. Recall that we embed in each block an authentication code c with m bits into one of n allowed code holders. If c is found in any of the n holders, we decide the authentication to be successful. The probability for a tampered bit (0 or 1) in B found in a code holder H to be authentic, i.e., to be with a value identical to the corresponding bit in the stego-image, is obviously 1/2, and so the probability for all the m tampered bits in H to be authentic is $(1/2)^m$. This means that the reverse probability for a code holder to be *safe* is $1 - (1/2)^m = (2^m - 1)/2^m$. There are n code holders in B, therefore the probability for all the n code holders to be safe is $((2^m - 1)/2^m)^n$. Inversely, the probability for the tampered block B to be verified to be authentic is accordingly $1 - ((2^m - 1)/2^m)^n$.

Now, we can compute the probability for a tampered image with L blocks to be verified to be authentic, which is just $[1 - ((2^m - 1)/2^m)^n]^L$ (denoted as P_w in the sequel). This probability, called *erroneous authentication probability* and abbreviated as EAP subsequently, has nothing to do with the size of the image block. But it is noted that the above analysis is valid under the case that the involved code holders have no pixel in common. In other cases, the EAP can be figured out to be smaller than P_w, so P_w may be regarded as the EAP value for *the worst case*.

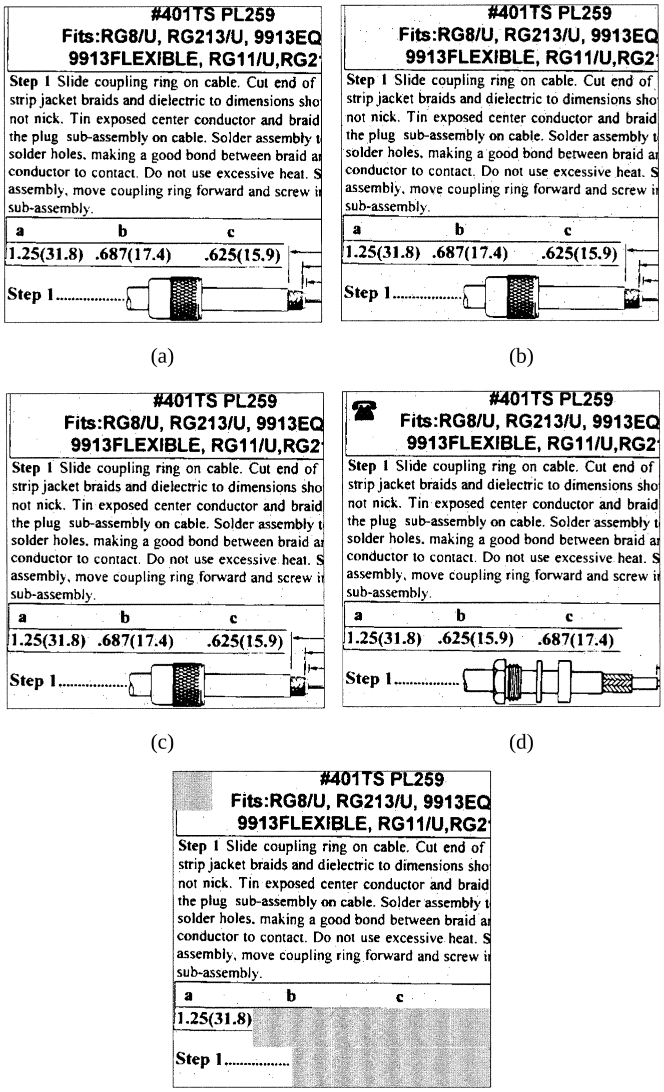
For example, if we choose 6 bits as the authentication code length and allow eight code holders for a tampered image with $L = 64$ blocks, then the EAP is $[1 - ((2^6 - 1)/2^6)^8]^{64} \approx (0.118\,37)^{64} \approx 4.79 \times 10^{-60}$ which is negligibly small. That is, the safety of the proposed scheme is no problem in this case.

The tradeoff between distortion reduction and security enhancement can be checked from the formula $[1 - ((2^m - 1)/2^m)^n]^L$ for computing the EAP. To reduce image distortion, we have to allow more code holders (i.e., to allow large n values), use less bits in the authentication code (i.e., to decrease the value of m), and partition the image into less blocks to restrain code embedding (i.e., to make the value of L smaller). These activities all result in an increase of the EAP as can be seen from the formula $[1 - ((2^m - 1)/2^m)^n]^L$. On the other extreme, we may take m to be as large as possible [so that $(2^m - 1)/2^m$ approaches 1] and allow just a single code holder ($n = 1$). Then the EAP will approaches 0, meaning that an invader has almost no chance to pass authentication with a tampered image. But the price we pay is that the content of each image block has almost been destructed. A compromise obviously must be considered, and the optimal choice is of course problem-dependent.

In case it is desired to protect the image to the block detail from *partial* image tampering, we have to consider the EAP value for a single block, which is $1 - ((2^m - 1)/2^m)^n$. Then, selections of n and m become critical. For $n = 8$, $m = 6$, and the EAP is $1 - [(2^6 - 1)/(2^6)]^8 \approx 0.118\,37$. If this is unsatisfactory to applications with higher security requirements, selections of a smaller n, say 6, and a larger m, say 10, may be considered, if the block size is larger enough to hold the code. This will result in an EAP value of $1 - [(2^{10} - 1)/(2^{10})]^6 \approx 0.005\,85$ which is reasonably small and safe, but at the sacrifice of increasing image distortion due to less code holder choices and more bit replacements.

IV. Experimental Results

An example of experimental results of the proposed method is shown in Fig. 1. Fig. 1(a) shows a cover image of size 512 × 512. Fig. 1(b) and (c) show two stego-images resulting from embedding authentication codes into Fig. 1(a) with $m = 8$, $n = 2$ and with $m = 8$ and $n = 5$, respectively. The block size was selected to be 64 × 64. It can be seen that the image in Fig. 1(c) includes less distortion than Fig. 1(b) because of the use of more code holders. Fig. 1(d) shows an image resulting from tampering



#401TS PL259
Fits:RG8/U, RG213/U, 9913EQ
9913FLEXIBLE, RG11/U,RG2

Step 1 Slide coupling ring on cable. Cut end of
strip jacket braids and dielectric to dimensions sho
not nick. Tin exposed center conductor and braid
the plug sub-assembly on cable. Solder assembly t
solder holes. making a good bond between braid ai
conductor to contact. Do not use excessive heat. S
assembly. move coupling ring forward and screw i
sub-assembly.

a	b	c
1.25(31.8)	.687(17.4)	.625(15.9)

Step 1..................

(a)

#401TS PL259
Fits:RG8/U, RG213/U, 9913EQ
9913FLEXIBLE, RG11/U,RG2

Step 1 Slide coupling ring on cable. Cut end of
strip jacket braids and dielectric to dimensions sho
not nick. Tin exposed center conductor and braid
the plug sub-assembly on cable. Solder assembly t
solder holes. making a good bond between braid ai
conductor to contact. Do not use excessive heat. S
assembly. move coupling ring forward and screw i
sub-assembly.

a	b	c
1.25(31.8)	.687(17.4)	.625(15.9)

Step 1..................

(b)

#401TS PL259
Fits:RG8/U, RG213/U, 9913EQ
9913FLEXIBLE, RG11/U,RG2

Step 1 Slide coupling ring on cable. Cut end of
strip jacket braids and dielectric to dimensions sho
not nick. Tin exposed center conductor and braid
the plug sub-assembly on cable. Solder assembly t
solder holes. making a good bond between braid ai
conductor to contact. Do not use excessive heat. S
assembly. move coupling ring forward and screw i
sub-assembly.

a	b	c
1.25(31.8)	.687(17.4)	.625(15.9)

Step 1..................

(c)

☎ #401TS PL259
Fits:RG8/U, RG213/U, 9913EQ
9913FLEXIBLE, RG11/U,RG2

Step 1 Slide coupling ring on cable. Cut end of
strip jacket braids and dielectric to dimensions sho
not nick. Tin exposed center conductor and braid
the plug sub-assembly on cable. Solder assembly t
solder holes. making a good bond between braid ai
conductor to contact. Do not use excessive heat. S
assembly. move coupling ring forward and screw i
sub-assembly.

a	b	c
1.25(31.8)	.625(15.9)	.687(17.4)

Step 1..................

(d)

#401TS PL259
Fits:RG8/U, RG213/U, 9913EQ
9913FLEXIBLE, RG11/U,RG2

Step 1 Slide coupling ring on cable. Cut end of
strip jacket braids and dielectric to dimensions sho
not nick. Tin exposed center conductor and braid
the plug sub-assembly on cable. Solder assembly t
solder holes. making a good bond between braid ai
conductor to contact. Do not use excessive heat. S
assembly. move coupling ring forward and screw i
sub-assembly.

a	b	c
1.25(31.8)		

Step 1..................

(e)

Fig. 1. An experimental result: (a) a cover image; (b) a stego-image resulting from embedding authentication codes into (a) with $m = 8$ and $n = 2$; (c) another stego-image resulting from embedding authentication codes into (a) with $m = 8$ and $n = 5$; (d) a tampered version of (c); (e) authentication result of (d) with tampered blocks marked in gray.

Fig. 1(c), and Fig. 1(e) shows the result of authentication in which tampered blocks are marked in gray.

V. Conclusion

A new approach to binary image authentication for distortion reduction and security enhancement has been proposed. It may be employed to design application-dependent schemes for embedding, from the viewpoint of enhance the security of the image, proper amounts of authentication codes into image blocks and then verifying them for fidelity and integrity checks without introducing unacceptable distortion. The security analysis and the experimental results show the feasibility of the proposed approach. Extensions of the approach to other image types may be tried in the future.

References

[1] Y.-C. Tseng, Y.-Y. Chen, and H.-K. Pan, "A secure data hiding scheme for binary images," *IEEE Trans. Commun.*, vol. 50, no. 8, pp. 1227–1231, Aug. 2002.

[2] M. Wu, E. Tang, and B. Liu, "Data hiding in digital binary images," in *Proc. IEEE Int. Conf. on Multimedia and Expositions*, vol. 1, New York, 2000, pp. 393–396.

[3] K. Matsui and K. Tanaka, "Video-steganography: How to secretly embed a signature in a picture," in *Proc. IMA Intellectual Property Project*, vol. 1, 1994.

[4] H. C. Wang, "Data hiding techniques for printed binary images," in *Proc. Int. Conf. on Information Technology: Coding and Computing*, Las Vegas, NV, Apr. 2001, pp. 55–59.

[5] S. H. Low, N. F. Maxemchuk, and A. M. Lapone, "Document identification for copyright protection using centroid detection," *IEEE Trans. Commun.*, vol. 46, pp. 372–383, Mar. 1998.

[6] D. Huang and H. Yan, "Interword distance changes represented by sine waves for watermarking text images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 1237–1245, Dec. 2001.