Authentication Based on Secret-Sharing-System for Images with a Data Repair Capability

Maneesha E.K.
Dept. of Computer Science and Engineering
MEA Engineering College
Perinthalmanna, Kerala, India

Harish Binu K.P.
Dept. of Computer Science and Engineering
MEA Engineering College
Perinthalmanna, Kerala, India

***Abstract*—It is very necessary to design effective methods to solve image authentication problem, particularly images whose security must be protected. An enhanced blind authentication method based on secret sharing with self-repairing capability for tiff image is proposed. Image divided in to number of blocks and authentication signal is generated for each block. The authentication signal combined with binarized block content which is transformed into several shares using Shamir secret sharing scheme. The generated shares embedded in to alpha channel plane. The alpha channel plane is combined with original image to form a tiff image. During embedding process, computed share values are mapped into a range of alpha channel values to yield a transparent stego image. During image authentication, image block is marked as tampered if authentication signal computed from current block does not match with authentication signal from corresponding alpha channel block. Data repairing is applied to each tampered block by using Reverse Shamir scheme. Method for protection of data hidden in the alpha channel is also proposed. Experimental result shows effectiveness of proposed method for real time application.**

Keywords—Image Authentication, Shamir secret sharing scheme, alpha channel, TIFF image.

I. INTRODUCTION

The open availability of powerful digital image processing tools allows open access, changes to original data and reuse of visual materials. In fact, nowadays many people could easily create illegal copies and change images in such a way that may lead to big economic or human lives losses. Digital image is used to preserve important information. Integrity and authenticity of digital image make new challenges. It is desirable to design effective methods to solve this kind of image authentication problem [1]-[3], especially for document images must be protected. It is also hoped that if part of a document image is verified to have been altered illicitly, the destructed content can be repaired. Such image content authenticationand self-repair capabilitiesare useful for security protection of digital documents in many fields, such as important certificates, signed documents, scanned checks, circuit diagrams, art drawings, design drafts, and so on

Images, which include texts, tables, line arts, etc. as main contents, are often digitized into grayscale images with twomajor gray values, one being of the background (including mainly blank spaces) and the other of the foreground (including mainly texts). It is noted that such images, though gray-valued in nature, look like binary. For example, the two major gray values in the document image are 174 and 236, respectively. It seems that such binary-like grayscale document images may be thresholded into binary ones for later processing, but such a thresholding operation often destructs the smoothness of the boundaries of text characters, resulting in visually unpleasant stroke appearances with zigzag contours. Therefore, in practical applications text documents are often digitized and kept as grayscale images for later visual inspection.

In general, the image authentication problem is difficult for a binary document image because of its simple binary nature which leads to perceptible changes after authentication signals are embedded in the image pixels. Such changes will arouse possible suspicions from attackers. A good solution to such binary image authentication thus should take into account not only the security issue of preventing image tampering, but also the necessity of keeping the visual quality of the resulting image. In this study, we propose an authentication method which deals with binary-likegrayscale document images instead of pure binaryones, and solves simultaneously the problems of image tampering detection and visual quality keeping.

Several methods for binary image authentication have been proposed in the past. Wu and Liu [4] manipulated the so-called flippable pixels to create specific relationships to embed data for authentication and annotation of binary images. Yang and Kot [5] proposed a two-layer binary image authentication method in which one layer is used for checking the image fidelity and the other for checking image integrity. In the method, a connectivity-preserving transition criterion for determining the flippability of a pixel is used for embedding the cryptographic signature and block identifier. Later, Yang and Kot [6] proposed a pattern-based data hiding method for binary image authentication in which three transition criteria are used to determine the flippabilities of pixels in each block, and the watermark is adaptively embedded into embeddable blocks to deal with the uneven embeddability condition in the host image. In the method proposed by Kim et al. [7], a set of pseudo-random pixels in a binary or halftone image are chosen and cleared, and authentication codes are computed accordingly and inserted into selected random pixels. In Tzeng and Tsai's method [8], randomly-generated authentication codes are embedded into image blocks for use in image

authentication, and a so-called code holder is used to reduce image distortion resulting from data embedding. Lee et al. [9] proposed a Hamming-code-based data embedding method which flips one pixel in each binary image block for embedding a watermark, yielding small distortions and low false negative rates. Lee et al. [10] improved the method later by using an edge line similarity measure to select flippable pixels for the purpose of reducing the distortion.

In this study, a method for authentication of document images with an additional self-repair capability for fixing tampered image data is proposed. The input cover imageis assumed to be any type image with two major gray values. After the proposed method is applied, the cover image is transformed into a stego-imagein the TIFF format with an additional alpha channelfor transmission on networks or archiving in databases. The stego-image, when received or retrieved, may be verified by the proposed method for its authenticity. Integrity modifications of the stego-image can be detected by the method at the block level and repaired at the pixellevel. In case that the alpha channel is totally removed from the stego-image, the entire resulting image is regarded as inauthentic, meaning that the fidelity check of the image fails. The proposed method is based on the so-called (k, n)-threshold secret sharing scheme proposed by Shamir [12] in which a secret message is transformed into *n* sharesfor keeping by *n* participants; and when kof the n shares, not necessarily all of them, are collected, the secret message can be recovered losslessly. Such a secret sharing scheme is useful for reducing the risk of incidental partial data loss.

Conventionally, the concepts of "secret sharing" and "data hiding for image authentication" are two irrelevant issues in the domain of information security. But in the proposed method, we combine them together to develop a new image authentication technique. The secret sharing scheme is used in the developed technique not only to carry authentication signals and image content data, but also to help repair tampered data through the use of shares.

An issue in self-repairing of tampered data at attacked image parts is that after the original data of the cover image are embedded into the image itself for use in later data repairing, the cover image is destructed in the first place and the original data are no longer available for data repairing, resulting in a contradiction. A solution to this problem is to embed the original image data somewhere elsewithout altering the cover image itself. The way proposed in this study to implement this solution is to utilize the extraalpha channel in a TIFF image to embed the original image data. However, the alpha channel of the TIFF image is used originally for creating a desired degree of transparency for the image. And embedding of data into the alpha channel will create randomtransparency in the resulting TIFF image, producing an undesirable opaque effect. One way out, as proposed in this study, is to map the resulting alpha channel values into a small range near their extreme value of 255, yielding a nearly imperceptibletransparency effect on the alpha channel plane.

Another problem encountered in self-repairing of the original image data is that the data to be embedded in the carrierare often large-sized. For our case here with the alpha channel as the carrier, this is not a problem because the cover image we deal with is essentially binary-like, and so we may just embed into the carrier a binary version of the cover image, which includes much less data. Furthermore, through a careful design of authentication signals, a proper choice of the basic authentication unit (i.e., the unit of 2×3 image block), and a good adjustment of the parameters in Shamir's scheme, we can reduce the data volume of the generated shares effectively so that more shares can be embedded into the alpha channel plane. It is noted that by the proposed method, the larger the number of shares is, the higher the resulting data repair capability becomes, as can be seen in the subsequent sections. Finally, we distribute the multipleshares randomlyinto the alpha channel to allow the share data to have large chances to survive attacks and so to promote the data repair capability. To the best of our knowledge, this is the first secret-sharing-basedauthentication method for binary-like grayscale document images. It is also the first authentication method for such document images through the use of the TIFF image. Note that this method is not a secret-sharing technique, buta document image authentication method.

II. RELATED WORKS

Data Hiding in Binary Image for Authentication and Annotation [4] provides a new method to embed data in binary images, including scanned text, figures, and signatures. The method manipulates "flippable" pixels to enforce specific block based relationship in order to embed a significant amount of data without causing noticeable artifacts. Shuffling is applied before embedding to equalize the uneven embedding capacity from region to region. The hidden data can be extracted without using the original image, and can also be accurately extracted after high quality printing and scanning with the help of a few registration marks. The proposed data embedding method can be used to detect unauthorized use of a digitized signature, and annotate or authenticate binary documents.

Binary Image Authentication with Tampering Localization by Embedding Cryptographic Signature and Block Identifier [5] proposes a novel two-layer blind binary image authentication scheme, 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 "connectivity-preserving" transition criterion. The image is partitioned into multiple macro-blocks that are subsequently classified into eight categories. The block identifier is defined adaptively for each class and embedded in those "qualified" and "self-detecting" macro-blocks in order to identify the tampered locations. The overall authentication is achieved in the first layer by hiding the cryptographic signature (CS) of the image. The localization of the tampering is achieved in the second layer by embedding the block identifier (BI) in the "qualified" or "self-detecting" macro-blocks (MBs).Specifically, we group multiple overlapping 3X3 blocks to form an MB and classify the MBs to "qualified" macro-blocks (QMBs) and "unqualified" macro-blocks (UMBs) based on the number of "flippable" pixels . The QAMs are chained, and that is used to identify the

tampering occurred both to the and its neighboring is embedded. Since the "flippability" of the pixels in each MB does not change in the data hiding process, the same process can be carried out to form the BI for the watermarked images Yw. Identify the current MB as "tampered" if the new computed BSw is different from the one extracted BSe. Identify the UMBs between two consecutive QMBs as "tampered". The tampered area lies between the previous and the current QMBs.

Pattern-Based Data Hiding for Binary Image Authentication by Connectivity-Preserving [6] provides a data hiding method for binary image authentication provided. The "flippability" of a pixel is determined by imposing three transition criteria in a 3x3 moving window centered at the pixel. The "embeddability" of a block is invariant in the watermark embedding process; hence the watermark can be extracted without referring to the original image. The "uneven embed ability" of the host image is handled by embedding the watermark only in those "embeddable" blocks. The locations are chosen in such a way that the visual quality of the watermarked image is guaranteed. Different types of blocks are studied and their abilities to increase the capacity are compared. The problem of how to locate the "embeddable" pixels in a block for different block schemes is addressed which facilitates the incorporation of the cryptographic signature as the hard authenticator watermark to ensure integrity and authenticity of the image.

A new Approach to Authentication of Binary Images for Multimedia Communication with Distortion Reduction and Security Enhancement [7] provides binary image authentication. 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.

III. PROPOSED SYSTEM

A detailed description on the implementation aspects of the proposed work is discussedin this chapter.

The proposed method is based on the (2, 6) secrete sharing scheme. The alpha channel space can be utilized to embed the extra information in the image. The alpha channel embedded in the TIFF image is used for creating required transparency for the image. In the proposed system the resulting alpha channel values are mapped into a range of 238 to 255. The authentication signal is generated using the optimal block size of 2 by 3. Then these authentication signals are mapped as shares in the image. This is an image authentication method, not a secrete sharing method. These embedded shares are then used for checking the integrity of the image at receiver side. For this system, input will be any type of image and output will be the Secured image/ original image without any tampering. If tampered, the system should recover it. For better performance encryption and decryption is provided. For security purpose image encryption is performed which can't be decrypted without a key.

The given approach to secret image sharing is based on the (k, n)-threshold secret sharing method proposed by Shamir(1979). For a group of n secret sharing participants n shares are generated from a secret integer value y for the threshold k. In next part the algorithm for image authentication and repairing are explained with results.

Two block diagrams describing the proposed method are shown in fig.1 and 2.

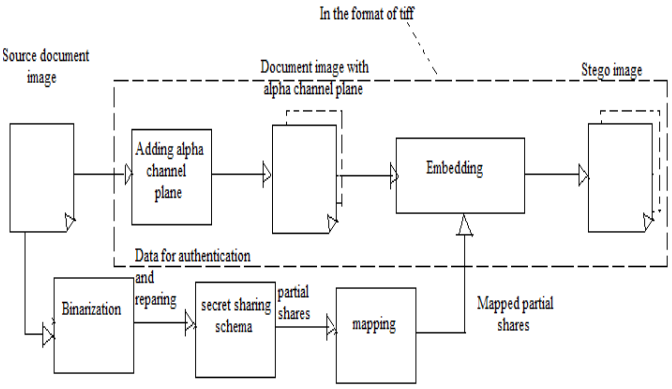

Fig. 1 Illustration of creating a Stego- image from a source image and an alpha channel.

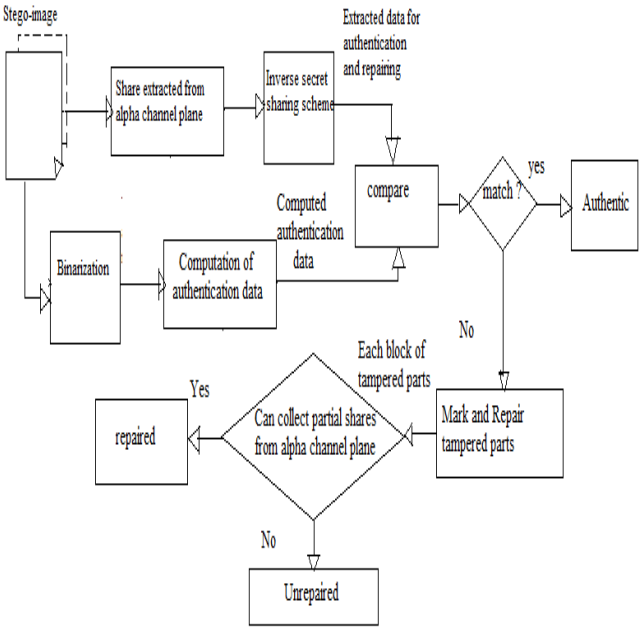

Fig. 2 Authentication process including verification and self-repairing of a stego-image in TIFF format.

IV. IMPLEMENTATION DETAILS

A. Algorithm for Generation of Stego Image

A detailed algorithm for describing the generation of a stego-image in the TIFF format of the proposed method is presented in the following

*Input:*An image Iwith two major gray values, and a secret key *K*.
*Output:*a stego-image I′ in the TIFF format with relevant data embedded, including the authentication signals and the data used for repairing.

Step1. (Input image binarization) Apply moment-preserving thresholding [13] to Ito obtain two representative gray values *g*1 and *g*2; compute $T = (g1 + g2)/2$; and use *T* as a threshold to binarize I, yielding a binary version Ib with "0" representing *g*1 and "1" representing *g*2.

Step2. (Transforming the cover image into the PNG format) Transform Iinto a PNG image with an alpha channel plane Iαby creating a new image layer with 100% opacity and no color as Iαand combining it with I using an image processing software package.

Step3. (Beginning of looping) Take in an unprocessed raster-scan order a 2×3 block Bbof Ibwith pixels p1,p2, …, p6.

Step4. (Creation of authentication signals) Generate a 2-bit authentication signal $s = a1a2$ with $a1 = \text{xor}(p1,p2, p3)$ and $a2 = \text{xor}(p4,p5,p6)$ where xordenotes the exclusive-OR operation.

Step5. (Creation of data for secret sharing) Concatenate the eight bits of *a*1, *a*2, and *p*1 through *p*6 to form an 8-bit string; divide the string into two 4-bit segments; and transform the segments into two decimal numbers m1 and m2, respectively.

Step6. (Partial share generation) Set *p*, *ci*, and *xi* in Eqs. (1) of Algorithm 1 to be the following values: (a) $p = 17$ (the smallest prime number larger than 15); (b) $d = m1$, $c1 = m2$; (c) $x1 = 1$, $x2 = 2,\ldots, x6 = 6$; and perform Algorithm 1 as a (2, 6)-threshold secret sharing:

$$qi=F(xi)=(d+c1xi)modp, \quad (1)$$

where $i = 1, 2, \ldots, 6$.

Step7. (Mapping of the partial shares) Add 238 to each of *q*1 through *q*6, resulting in the new values of *q*1' through *q*6', respectively, which fall in the nearly total transparency range of 238 through 254 in the alpha channel plane Iα.

Step8. (Embedding two partial shares in the current block) Take the block Bαin Iα corresponding to Bbin Ib, select the first two pixels in Bα in the raster-scan order, and replace their values by *q*1' and *q*2', respectively.

Step9. (Embedding remaining partial shares at random pixels)Use the key *K* to select randomly four pixels in Iα but outside Bα, which are unselected yetin this step and notthe first two pixels of any block; and in the raster scan order replace the four pixels' values by the remaining four partial shares *q*3' through *q*6' generated above, respectively.

Step10. (End of looping) If there exists any unprocessed block in Ib, then go to Step 3; otherwise, take the finalIin the TIFF format as the desired stego-image I′.

The possible values of q1 through q6 yielded by Eq. (1) above are between 0 and 16 because the prime number p used there is 17. After performing Step7 of the above algorithm, they become q1' through q6', respectively, which fall in to a small interval of integer ranging from 238 to 254. Subsequent embedding of q1' through q6' in such a narrow interval into the alpha channel plane means that very similar values will appear everywhere in the plane, resulting in a nearly uniform transparency effect, which will not arouse notice from an attacker.

Fig. 3 to illustrate Steps 8 and 9 of Algorithm for authentication where a core idea of the proposed method is presented: two shares of the generated six are embedded at the current block and the other four embedded at four randomly-selected pixels outside the block, with each selected pixel not being the first two ones in any block.

B.Algorithm for Stego-Image Authentication

A detailed algorithm describing the proposed stego-image authentication process, including both verification and self-repairing of the original image content, is presented in the following.

Input: a stego image I'; the representative gray value g1 and g2, and the secret key K used in the algorithm IV (A)
Output: an image Ir with tampered blocks marked, and their data repaired if possible.

Step1. (Binarization of the stego-image) Compute T=(g1+g2)/2 and use it as a threshold to binarize I', yielding a binary version Ib' of I' with "0" representing g1 and "1" representing g2.

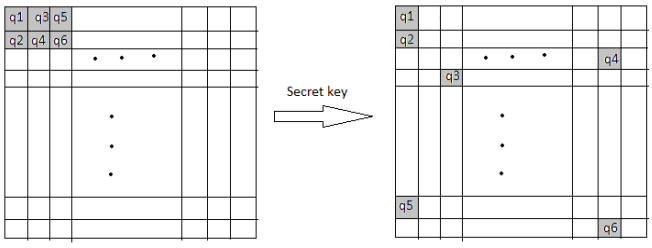

Fig. 3. Illustration of embedding six shares created for a block- two shares embedded at the current block and the other four in four randomly-selected pixels outside the block, with each selected pixel not being the first two ones in any block

Step2. (Beginning of looping) Take in a raster-scan order an unprocessed block Bb′ from Ib′ with pixel values p1 through p6, and find the

six pixels' values q1' through q6' of the corresponding block Bα′ in the alpha channel plane Iα′ of I′.

Step3. (Extraction of the hidden authentication signal) Perform the following steps to extract the hidden 2-bit authentication signal $s = a1a2$ from Bα′.

(1) Subtract 238 from each of $q1$` and $q2$` to obtain two partial shares $q1$ and $q2$ of Bb', respectively.

(2) With the shares (1, $q1$) and (2, $q2$) as input, perform Shamir reconstruction to extract the two values d and $c1$ (the secret and the first coefficient value, respectively as output.

(3) Transform d and $c1$ into two 4-bit binary values, concatenate them to form an 8-bit string S, and take the first two bits $a1$ and $a2$ of S to compose the hiddenauthentication signal $s = a1a2$.

Step4. (Computation of the authentication signal from the current block content) Compute a two-bit authenticationsignal $s' = a1a2$from the values $p1$ through $p6$ ofthe six pixels of Bb' by $a1$' = xor ($p1$,$p2$, $p3$) and $a2$' = xor ($p4$, $p5$,$p6$).

Step5. (Matching of the hidden and computed authentication signals and marking of tampered blocks) Match s and s' by checking if $a1 = a1'$ and $a2 = a2'$; and if any mismatchoccurs, mark Bb', the corresponding block B' inI', and all the partial shares embedded in $B\alpha$' as *tampered*.

Step6. (*End of looping*) If there exists any *unprocessed* block in Ib′, then go to Step 2; otherwise, continue.

Step7. (Extraction of the remaining partial shares) For each block Bα'in $I\alpha'$, perform the following steps to extract the remaining four partial shares $q3$ through $q6$ of the corresponding block Bb' in Ib' from blocks in $I\alpha'$ other than $B\alpha'$.

(1) Use the key K to collect the four pixels in $I\alpha'$ in the same order as they were randomly selected for Bb′ in Step 9 of Algorithm 3, and take out the respective data $q3$', $q4$', $q5$', and $q6$' embedded in them.

(2) Subtract 238 from each of $q3$' through $q6$' to obtain $q3$ through $q6$, respectively.

Step8. (Repairing the tampered regions)

For each block B' in I' marked as tampered previously, perform the following steps to repair it if possible.

(1) From the six partial shares $q1$ through $q6$ of the block Bb′ in Ib′ corresponding to B'(two computed in Step 3(1) and four in Step 7(2) above), choose two of them, say qk and ql, which are notmarked as tampered, if possible.

(2) With the shares (k, qk) and (l, ql) as input, perform Algorithm 2 to extract the values of d and $c1$ (the secret and the first coefficient value) as output.

(3) Transform d and $c1$ into two 4-bit binary values and concatenate them to form an 8-bit string S′.

(4)Take the last six bits $b1'$, $b2'$, …, $b6'$ from S' and check their binary values to repair the corresponding tampered pixel values $y1'$, $y2'$, …, $y6'$ of block B' by the following way:

if$bi' = 0$, set $yi' = g1$; otherwise, set $yi' = g2$ where i =1,2, …, 6.

Step9. Take the final I' as the desired self-repaired image Ir.

V. RESULTS AND OBSERVATIONS

A. Result

We conducted image-modification attacks to the stego images using two common image editing operations, namely superimposing and painting. Fig. 4(c) shows the result of superimposing a white rectangular shape with a fake signature "Simo" on the genuine signature "C. W. Lee" in the stego-image of Fig. 4(b). Fig. 4(a) is the original image. Fig. 4(d) shows the authentication result yielded by Algorithm IV (B), with the gray blocks indicating the detected tampered image parts. As can be seen, the superimposing rectangular part on the signature C. W. Lee has been detected completely. For each of the detected tampered blocks, if at least two untampered shares of it can be collected, its original content can be repaired, yielding the result shown in Fig. 4(e)

In Fig. 5(a), a text line under the signature in the signed paper disappeared after a white rectangular band was superimposed on it. The results of image authentication and repairing are shown in Figs. 5(b) and 5(c), respectively. The repair result of the text line is visually well recognizable.

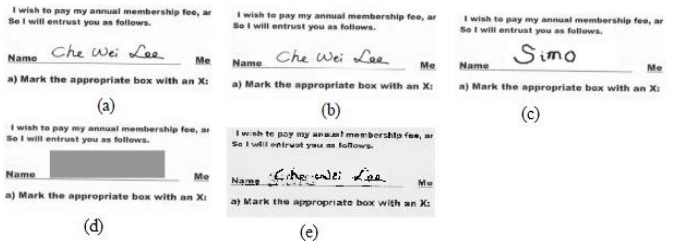

Fig. 4. Authentication result of a TIFF document image of a signed paper attacked by superimposing a white rectangular shape on the signature. (a) Original cover image. (b) A stego-image with embedded data. (c)Tampered image yielded by the superimposing operation. (d) Result with tampered blocks detected and marked as gray. (e) Data repair result.

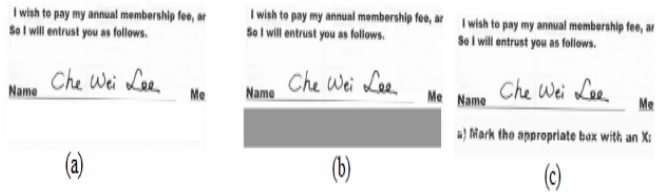

Fig. 5. Authentication result of the document image of a signed paper attacked by superimposing a white rectangular shape on a piece of text in Fig. 4(b). (a) Tampered image yielded by the superimposing operation. (b) Result with tampered blocks detected and marked as gray. (c) Data repair result.

B. Observation

1) *Merits of the Proposed Method*

In addition to being capable of data repairing and being blindin nature (requiring no overhead other than the stego-image), the proposed method has several other merits, which are described in the following.

a) Providing pixel-level repairs of tampered image parts
b) Having higher possibility to survive image content attacks
c) Making use of a new type of image channel for data hiding
d) Causing no distortion to the input image
e) Enhancing data security by secret sharing

2) Measures for Security Enhancement

The secret key *K*, which is used to randomize the pixel positions for embedding the mapped partial shares *q*3' through *q*6' mentioned in Step 9 of Algorithm 4.1, provides a measure to protect the shares. More specifically, as described in Algorithm 4.1, each block in the alpha channel plane may be regarded to consist of two parts, the first partincluding the first two pixels and the secondincluding the remaining four. The first part of each block is used for keeping the first two partial shares $q1'$ and $q2'$, and the second part for keeping the last four partial shares $q3'$ through $q6'$ of other blocks located at random positions. Therefore, the probability of correctly guessing the locations of all the embedded partial shares in a stego-image is 1/[(*m*x*n*) - (*m*x*n*/6)×2]! where *m*x*n* is the size of the cover image; *m*x*n*/6 is the total number of blocks, each with six pixels; and (*m*x*n*) - (*m*x*n*/6)×2 is the total number of pixels in the blocks other than those in the first parts of all the blocks. This probability obviously is very small for common image sizes, meaning that a correct guess of the embedded partial shares is nearly impossible.

To enhance further the security of the data embedded in the stego-image, one additional measure is adopted in the proposed method (but not included in the previously-proposed algorithms for clarity of algorithm descriptions). It is randomization of the constant values of *x*1 through *x*6 used in Step 6 of Algorithm 4.1 and Step 3(2) in Algorithm 4.2. Specifically, in Step 3(2) in Algorithm 4.2 we can see that the input shares into Algorithm 2, (1, *q*1) and (2, *q*2), can be forged easily, leading to the possibility of creating fake authentication signals. To remedy this weakness, with the help of another secret key we may choose these values of *x*1 through *x*6 for each block to be randomwithin the allowed integer range of $0 \leq xi < p$ (= 17) [11]. Then, the probability of guessing correctly all these values for all the *m*x*n*/6 blocks in a stego-image can be figured out to be $1/[(17\times16\times15\times14\times13\times12)]^{mxn/6} \approx 1/(8.911\times106)^{mxn/6}$ which is also very small for common image sizes *m*x*n*.

3) Comparison Of Performances With Other Methods

TABLE I Comparison of document image authentication methods

	Distortionin stego image	**Tampering localization capability**	**Repair capability**	**Reported Authentication pecistion**	**Distribution of authenticated image part**	**Manipulation of data emedding**
Wu & Liu	Yes	No	No	Macro block	Non blank part	Pixel flippability
Yan & Kot	Yes	Yes	No	33X33 block	Non blank part	Pixel flippability
Yan & Kot	Yes	No	No	Macro block	Non blank part	Pixel flippability
Tzeng & Tsai	Yes	Yes	No	64X64 block	Entire image	Pixel replacement
Proposed method	No	Yes	Yes	2X3 block	Entire image	Alpha channel replacement

A comparison of the capabilities of the proposed method with those of four existing methods is shown in Table 1. All but the proposed method will create distortion in the stego-image during the authentication process. More importantly, only the proposed method has the capability of repairing the tampered parts of an authenticated image.

Furthermore, among the methods with tampering localization capabilities at the block level like Yang and Kot [5], Tzeng and Tsai [8], and the proposed method, the proposed method provides a finer authentication precision with the block size of 2×3. Specifically, the method in [5] needs larger macro-blocksto yield pixel flippabilities for embedding authentication data. In the case of using smaller blocks, Tzeng and Tsai's method [8] has a high possibility to generate noise pixels as mentioned in [6], and so they conducted experimental results with the larger block size of 64×64.

CONCLUSION AND FUTURE WORKS

A new authentication based on secret-sharing-system for images with a data repair capabilityhas been proposed. Both the generated authentication signal and the content of a block are transformed into partial shares by the Shamir method, which are then distributed in a well-designed manner into an alpha channel plane to create a stego-image in the TIFF format. The undesired opaque effect visible in the stego-image coming from embedding the partial shares is eliminated by mapping the share values into a small range of alpha channel values near their maximum transparency value of 255.

In the process of image block authentication, a block in the stego-image is regarded as having been tampered with if the computed authentication signal does not match that extractedfrom corresponding partial shares in the alpha channel plane. For self-repairing of the content of a tampered block, the reverse Shamir scheme is used to compute the original content of the block from any two untampered shares. Measures for enhancing the security of the data embedded in the alpha channel plane were also proposed. Experimental results have been shown to prove the effectiveness of the proposed method. Future studies may be directed to choices of other block sizes and related parameters (prime number, coefficients for secret sharing, number of authentication signal bits, etc.) to improve data repair effects. Applications of the proposed method areto repair imagein original color may also be tried.

REFERENCES

[1] C. S. Lu and H. Y. M. Liao, "Multipurpose watermarking for image authentication and protection," *IEEE Trans. on ImageProcessing*, vol. 10, no. 10, pp. 1579–1592, Oct. 2001..
[2] M. U. Celik, G. Sharma, E. Saber and A. M. Tekalp, "Hierarchical watermarking for secure image authentication with localization," *IEEE Trans.on Image Processing*, vol. 11, no. 6,
pp. 585–595, June 2002.
[3] Z. M. Lu, D. G. Xu and S. H. Sun, "Multipurpose image, watermarking algorithm based on multistage vector quantization" *IEEE Trans. on Image Proc.*, vol. 14, pp. 822–831, June
2005.
[4] M. Wu and B. Liu, "Data hiding in binary images for authentication and annotation," *IEEE Trans.on Multimedia*, vol. 6, no. 4, pp. 528–538, Aug. 2004.
[5] H. Yang and A. C. Kot, "Binary image authentication with tampering localization by embedding cryptographic signature and block identifier," *IEEE Signal Processing Letters*, vol. 13, no. 12, pp. 741–744, Dec. 2006.
[6] H. Yang and A. C. Kot, "Pattern-based data hiding for binary images authentication by connectivity-preserving," *IEEETrans. on Multimedia*, vol. 9, no. 3, pp. 475–486, April 2007.
[7] H. Y. Kim and A. Afif, "Secure authentication watermarking for halftone and binary images," *Int'l Journal of ImagingSystems and Technology*, vol. 14, no. 4, pp. 147–152, 2004.
[8] C. H. Tzeng and W. H. Tsai. "A new approach to authentication of binary images for multimedia communication with distortion reduction and security enhancement," *IEEE CommunicationsLetters*, vol. 7, no. 9, pp. 443–445.
[9] Y. Lee, J. Hur, H. Kim, Y. Park and H. Yoon, "A new binary image authentication scheme with small distortion and low false negative rates," *IEICE Trans. on Communications*, vol.E90-B, no. 11, Nov. 2007.
[10] Y. Lee, H. Kim, and Y. Park, "A new data hiding scheme for binary image authentication with small image distortion," *Information Sci.*, vol. 179, no. 22, pp. 3866–3884, Nov. 2009.
[11] A. Shamir, "How to share a secret," *Communication of ACM*, vol. 22, pp. 612–613, 1979.
[12] C. C. Lin and W. H. Tsai, "Secret image sharing with steganography and authentication," *Journal of Systems & Software*, vol. 73, pp. 405–414, 2004.
[13] W. H. Tsai, "Moment-preserving thresholding: a new approach," *Computer Vision, Graphics, and ImageProcessing*, vol. 29, no. 3, pp. 377-393, 1985.