IOSR Journal of Computer Engineering (IOSRJCE)
ISSN : 2278-0661 Volume 1, Issue 6 (July-Aug 2012), PP 09-16
www.iosrjournals.org

Authentication of Document Image with Data Repairing

P. Rajitha Nair, Dr. Sipi Dubey
4th Semester M.Tech (Software Engineering) in Department of Computer Science & Engineering,
Professor& Dean in Department of Computer Science & Engineering,
[1,2]RCET, Bhilai

Abstract: *In this paper, we are introducing a blind authentication method which is based on the secret sharing technique with a data repair capability for document images with the use of the PNG image. We generate an authentication signal for each block of a document image which, together with the block content in binary form, is transformed into several shares using the Shamir secret sharing scheme. These parameters are carefully chosen so that a large number of shares possible are generated and embedded into an alpha channel plane. Now the alpha channel plane is combined with the original image to form a PNG image. In the process of image authentication, an image block is marked as tampered if the authentication signal computed from the current block content does not match the one extracted from the shares embedded in the alpha channel plane. Repairing of data is now done to each tampered block by a reverse Shamir technique after collecting any two or more shares from unmarked blocks. Also a measure to protect the security of the data hidden in the alpha channel is proposed.*

Index Terms: *Image authentication, secret sharing, data repair, data hiding, and PNG (Portable Network Graphics) image.*

I. Introduction

Important information can be preserved easily in form of digital images. But, with the fast advance of digital technologies, it is easy to make visually alterations to the contents of digital images. Digital image can now be reproduced and spread easily. Therefore, preserving important images secretly is a major issue. How to ensure the integrity and authenticity of a digital image is a major challenge. It is very necessary to design effective methods to solve this type of image authentication problem, especially for images of documents whose security must be protected. Secret image sharing has been recently presented to solve this problem. Secret image sharing techniques generate several shared images from the protected image, and the protected image is reconstructed by enough different shared images. We are also hoping that if part of a document image is verified to be altered illicitly, the destructed content can be repaired. These image content authentication and self-repair capabilities are useful for security protection of digital documents for important certificates, signed documents, scanned checks, circuit diagrams, art drawings, design drafts, last will and testaments, etc. Document images can also include texts, tables, line arts, as main contents, are often digitized into grayscale images with two major gray values, one being of the background (including mostly blank spaces) and the other of the foreground (including mostly texts). It is noted that such images, though gray-valued in nature, look like binary. It seems that such binary-like grayscale document images may have threshold of binary ones for later processing, but such an operation often destructs the smoothness of the boundaries of text characters, resulting in visually obnoxious stroke appearances with zigzag contours. Therefore, in practical applications text documents are often digitized and kept as grayscale images for later visual inspection. The image authentication crisis 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 doubts 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. Here, we propose an authentication method which deals with binary-like grayscale document images as a replacement for of pure binary ones, and solves concurrently the problems of image tampering detection and visual quality keeping. In this study, a method for authentication of document images with a supplementary self-repair capability for fixing tampered image data is proposed. The input cover image is assumed to be a binary-like grayscale image with 2 major gray values. After the proposed method is applied, the cover image is transformed into a stego-image in the PNG format with an supplementary alpha channel for transmission on networks or archiving in databases. The stego-image, when received or retrieved, may be verified by the proposed technique for its authenticity. Integrity modifications of the stego-image can be detected by the method at the block level and repaired at the pixel level. In case that the alpha channel is totally removed from the stego-image, the intact 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 in which a secret message is transformed into n shares for keeping by *n* participants; and when k of the *n* shares, not necessarily all of them, are collected, the secret message can be recovered without any loss. Such a secret sharing scheme is useful for reducing the risk of incidental partial data loss.

In *Section 2,* we are discussing the Shamir secret sharing algorithm. Our proposed plan is presented in *Section 3*. Performance comparison is done in *Section 4.* At last, our conclusions are offered in *Section 5.*

II. Use Of The Shamir Method For Secret Sharing

The proposed approach to secret image sharing is based on the *(k, n)*-threshold secret sharing method proposed by Shamir (1979). In this section we describe how to use the Shamir method for conventional secret sharing before describing our approach in the next section. By the Shamir method, to generate *n* shares for a group of *n* secret sharing participants from a secret integer value y for the threshold k, we can use the following *(k-1)*-degree polynomial in the following way-

Algorithm 1: (k,n)**-threshold secret sharing**
Input: Secret *d* in the form of an integer, number of participants, and threshold.
Output: Shares in the form of integers for the participants to keep
.
Step 1*:* Choose randomly a prime number that is larger than *d*.
Step 2*:* Select *k-1* integer values within the range of 0 through *p-1*.
Step 3*:* Select *n* distinct real values $x_1, x_2, \ldots, x_n$.
Step 4*:* Use the following *(k-1)*-degree polynomial to compute *n* function values, $F(x_i)$ called *partial shares* for $i=1,2,\ldots,n$, *i.e.*,

$$F(x_i)=(d+c_1x_i+c_2x_i^2+\ldots+c_{k-1}x_i^{k-1})_{modp} \quad \ldots\ldots(1)$$

Step 5*:* Deliver the 2-tuple $(x_i,F(x_i))$ as a *share* to the i^{th} participant where $i=1,2,\ldots,n$.
The *k* coefficients, namely *d* and $c_1$through c_{k-1} in *Eqn. (1)* above, it is necessary to gather at least shares from the *n* participants to form *k* equations of the form of *Eqn. (1)* to solve these *k* coefficients in order to recover secret *d*. This explains the term *threshold* for *k* and the name *(k,n) -threshold* for the Shamir method [7]. Below is a description of the just-mentioned equation-solving process for secret recovery

Algorithm 2: Secret recovery
Input: *k* shares collected from the *n* participants and the prime number *p* with both *k* and *p* being those used in Algorithm 1.
Output: secret *d* hidden in the shares and coefficients c_i used in *Eqn. (1)* in Algorithm 1, where $i=1, 2\ldots k-1$.
Step 1*:* Use the *k* shares $(x_1, F(x_1)), (x_2, F(x_2))\ldots (x_k, F(x_k))$ to setup

$$F(x_j) = (d+c_1x_j+c_2x_j^2+\ldots+c_{k-1}x_j^{k-1})_{modp} \quad \ldots\ldots(2)$$

where $j=1, 2 \ldots k$.
Step 2*:* Solve the *k* equations above by Lagrange's interpolation to obtain *d* as

$$d=(-1)^{k-1}[F(x_1)\frac{x_2x_3\ldots x_k}{(x_1-x_2)(x_1-x_3)\ldots(x_1-x_k)}+F(x_2)\frac{x_1x_3\ldots x_k}{(x_2-x_1)(x_2-x_3)\ldots(x_2-x_k)}$$
$$+\ldots+F(x_k)\frac{x_1x_2\ldots x_{k-1}}{(x_k-x_1)(x_k-x_2)\ldots(x_k-x_{k-1})}]_{\mathrm{mod}\,p}.$$

follows:

Step 3*:* Compute c_1 through c_{k-1} by expanding the following equality and comparing the result with *Eqn. (2)* in *Step 1* while regarding variable *x* in the equality below to be x_j in (2):

$$F(x)=\Big[F(x_1)\frac{(x-x_2)(x-x_3)\ldots(x-x_k)}{(x_1-x_2)(x_1-x_3)\ldots(x_1-x_k)}$$
$$+F(x_2)\frac{(x-x_1)(x-x_3)\ldots(x-x_k)}{(x_2-x_1)(x_2-x_3)\ldots(x_2-x_k)}+\cdots$$
$$+F(x_k)\frac{(x-x_1)(x-x_2)\ldots(x-x_{k-1})}{(x_k-x_1)(x_k-x_2)\ldots(x_k-x_{k-1})}\Big]_{\mathrm{mod}p}.$$

In the above algorithm *Step3* is in addition included for the purpose of computing the values of parameters c_i in the proposed method. In other applications, if only the secret value need be recovered, this step may be eliminated.

Note that according to *Shamir (1979),* if fewer than k secret shares are collected, the *k* unknowns cannot be solved and the desired *y* value cannot be reconstructed.

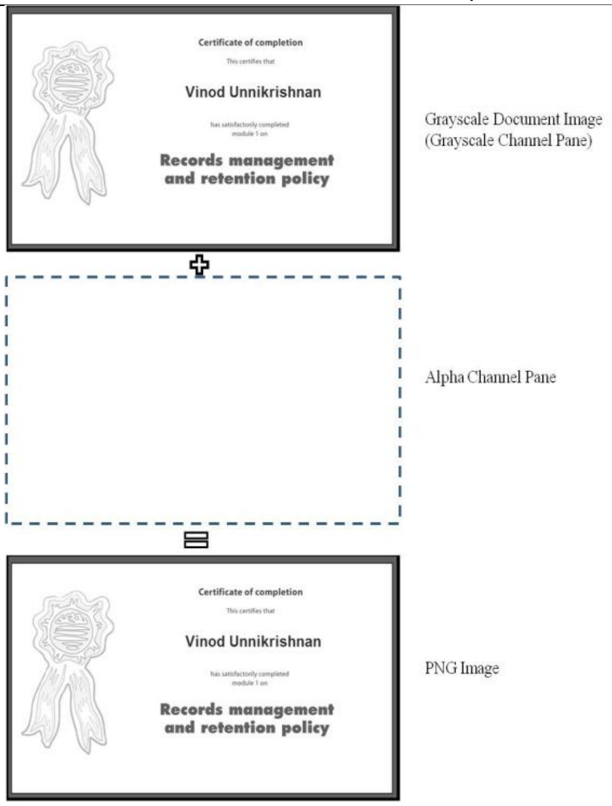

Fig. 1. Pictorial representation of a PNG image from a grayscale document image and an additional alpha channel plane

III. Image Authentication And Data Repairing

We create a PNG image from a binary-type grayscale document image *S* with an alpha channel plane. The actual image *S* may be assumed as a grayscale channel plane of the PNG image. Then, *S* is converted to binary form with moment-preserving threshold, yielding a binary version of *S*, which we denote as S_b. Data image for authentication and repairing are then computed from S_b and taken as an input to Shamir's secret sharing scheme, to generate *n* secret shares of the data. The share values are mapped subsequently into a small range of alpha channel values near the maximum transparency value to create an imperceptibility effect. Lastly, the mapped secret shares are randomly entrenched into the alpha channel for the function of promoting the security, protection and data repair capability.

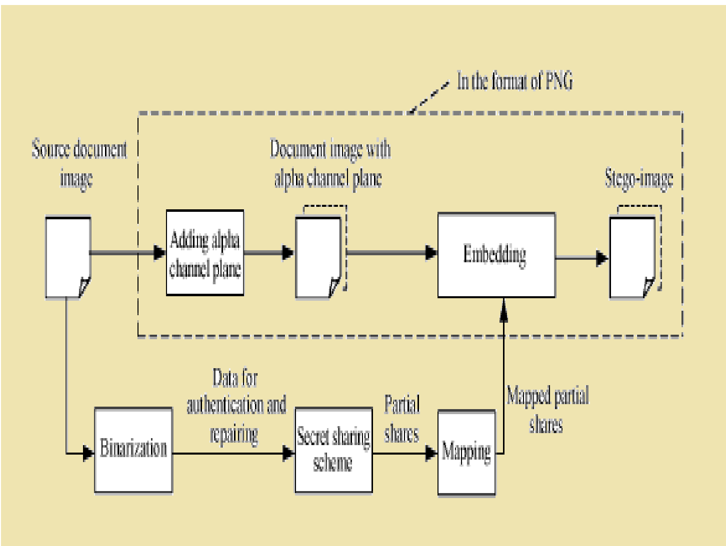

Fig. 2 Pictorial representation of creating a PNG image from gray scale document image and alpha channel.

The alpha channel plane is used for carrying data for authentication and repairing, so no demolition will occur to the input image in the process of verification. On the contrary, traditional image authentication methods often sacrifice part of image contents, such as LSB's or pixels that can be flipped, to provide accommodation to data used for authentication. Additionally, once stego-image generated from a conventional method like an LSB-based one is unintentionally compressed by a lossy compression method, the stego-image might cause fake positive alarms in the authentication system. In comparison, the anticipated method yields a stego-image in the PNG format which in usual cases will not be compressed further, reducing the opportunity of invalid authentication caused by imposing undesired compression operations on the stego-image.

3.1 Generating stego-image

A comprehensive algorithm for describing the generation of a stego-image in the PNG format of the anticipated method is presented as follows:

Algorithm 3: Generating a stego-image in PNG format from a given grayscale image.

Input: A image document in grayscale S with two major gray values, and a secret key K.

Output: A stego-image S' in the PNG format with relevant data embedded, including the authentication signals and the data used for repairing

Step A: **Generating authentication signals**

(i) (Conversion of Input image to Binary form) Apply *moment-preserving threshold [6]* to S to obtain two representative gray values g_1 and g_2, compute

$T = (g_1 + g_2)/2$; And use T as a threshold to convert S *into binary form*, yielding the binary version S_b with *"0"* representing g_1 and *"1"* representing g_2.

(ii) (Convert the cover image into the PNG format) Convert S into a PNG image with an alpha channel plane S_α by creating a new image layer with *100%* opacity and no color as S_α and combining it with S using an image processing software package.

(iii) Take in an unrefined raster-scan order a 2×3 *block* B_b of S_b with pixels $p_1, p_2 \dots p_6$.

(iv) (Creating authentication signals) Create a 2-bit authentication signal $Z = a_1a_2$ with $a_1 = p_1 \text{ XOR } p_2 \text{ XOR } p_3$ and $a_2 = p_4 \text{ XOR } p_5 \text{ XOR } p_6$.

Step B: Design and Embedding of Shares

(v) (Creating data for secret sharing) concatenate the *8* 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 convert the segments into 2 decimal numbers m_1 and m_2, respectively.

(vi) (Generation of Partial Share) Set p, c_i, and x_i in *Eqn. (1)* of Algorithm 1 to be the following values: *(a)* $p = 17$ (the smallest prime number larger than15); *(b)* $d = m_1, c1 = m_2$; *(c)* $x_1 = 1, x_2 = 2, \dots, x_6 = 6$; and execute Algorithm 1 as *a (2, 6)-threshold secret sharing scheme* to generate six partial shares q_1 through q_6 using the following equations:

$$q_i = F(x_i) = (d + c_1x_i)_{\text{mod } p} \quad \dots\dots\dots\dots (3)$$

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

(vii) (Map of the partial shares) Adding 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 S_α.

(viii) (Embedding two fractional shares in the current block) receive the block B_α in $\underline{S_\alpha}$ corresponding to B_b in S_b, select the first two pixels in B_α in the raster-scan order, and substitute their values by q_1' and q_2', respectively.

(ix) (Embedding remaining incomplete shares at random pixels) Use the key K to select randomly 4 pixels in S_α but outside B_α, which are unselected yet in this step and not the first 2 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.

(x) If there exists any unprocessed block in S_b, then go to (iii), if not, take the final S in the PNG *format* as the preferred stego-image S'.

The promising values of q_1 through q_6 yield by *Eqn. (3)* above are between 0 and 16 because the prime number p used there is 17. After executing (vii) of the above algorithm, they become q_1' through q_6', respectively, which all fall into a small interval of integers ranging from 238 to 254 with a width of 17 (the value of the prime number). Consequent embedding of q_1' through q_6' in such a narrow interval into the alpha channel plane means that very alike values will appear everywhere in the plane, resulting in a nearly uniform transparency effect, which will not stimulate notice from an attacker.

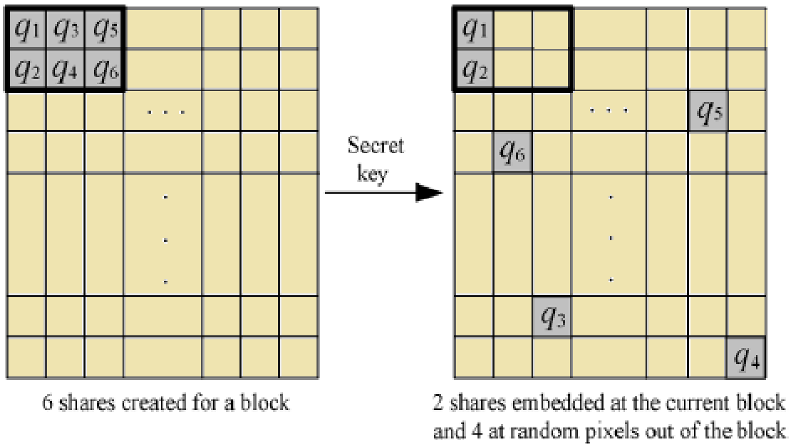

Fig. 3. Pictorial representation of embedding 6 shares generated for a block, 2 shares embedded in current block and other 4 in 4 randomly selected pixels outside the block, with each selected pixel not being the first 2 ones in any block.

The motivation why we choose the prime number to be 17 in the above algorithm is that if it was chosen instead to be larger than 17, then the above-mentioned interval will be enlarged and the values of q_1' through q_6' will become possibly lesser than 238, creating visually whiter stego-image. In contrast, the *8* bits mentioned in (v) and (vi) above are transformed into 2 decimal numbers m_1 and m_2 with their maximum values being 15 (notice (v) above), which are forced to lie in the range of *0* through *p-1* (notice Step 2 in Algorithm 1). Therefore, *p* should not be chosen to be smaller than 16. In short, $p = 17$ is a best possible choice.

3.2 Stego-Image Authentication

A complete algorithm describing the proposed stego-image authentication process, including both verification and self-repairing of the original image content, is described below.

Algorithm 4: Authentication of a given stego-image in the PNG format.
Input: A stego-image *S′,* the representative gray values g_1 and g_2, and the secret key *K* used in Algorithm 3.
Output: An image S_r with tampered blocks marked, and their data repaired if possible.

Part 1: Extraction of the embedded two representative gray values.
Step 1: (Conversion of the stego-image to Binary form) Compute $\boldsymbol{T = (g_1 + g_2)/2}$ And use it as a threshold to convert *S′ into Binary Form*, yielding the binary version S_b' of *S′* with *"0"* representing g_1 and *"1"* **representing $\boldsymbol{g_2}$.**
Part 2: Authentication of the stego-image.
Step 2: (Start looping) Take in a raster-scan order an unprocessed block B_b' from S_b' with pixel values p_1 through p_6, and find the 6 pixel values q_1', through q_6' of the corresponding block B_b' in the alpha channel plane S_α' of *S′.*
Step 3*:* (Drawing out of the secreted authentication signal) to extract the hidden 2-bit authentication signal $Z = a_1a_2$ from B_α' we will follow the steps:
(1) Subtract 238 from each of q_1' and q_2' to obtain the 2 respective partial shares q_1 and q_2 of B_b'.
With the shares $(1, q_1)$ and $(2, q_2)$ as input, perform Algorithm 2 to extract the 2 values *d* and c_1 (the secret and the first coefficient value, respectively) as output.
(2) Transform *d* and c_1 into two 4-bit binary values, concatenate them to form an 8-bit string *W*, and take the first two bits a_1 and a_2 of W to compose the hidden authentication signal $Z = a_1a_2$.
Step 4: (Computation of the authentication signal from the current block content) Compute a two-bit authentication signal $Z' = a_1'a_2'$ from the values p_1 through p_6 of the six pixels of B_b' by $a_1' = p_1\ XOR\ p_2\ XOR\ p_3$ and $a_2' = p_4\ XOR\ p_5\ XOR\ p_6$.
Step 5: (Harmonizing the hidden and computed authentication signals and marking of tampered blocks) Match *Z* &*Z′* by checking if $a_1 = a_1'$& $a_2 = a_2'$*,* and if any variance occurs, mark B_b', the corresponding block *B′* in *S′,* and all the partial shares embedded in B'_α as tampered.
Step 6: (Close loop) if there exists any unprocessed block in S_b'*,* then go to *Step 2*; otherwise, *go on.*

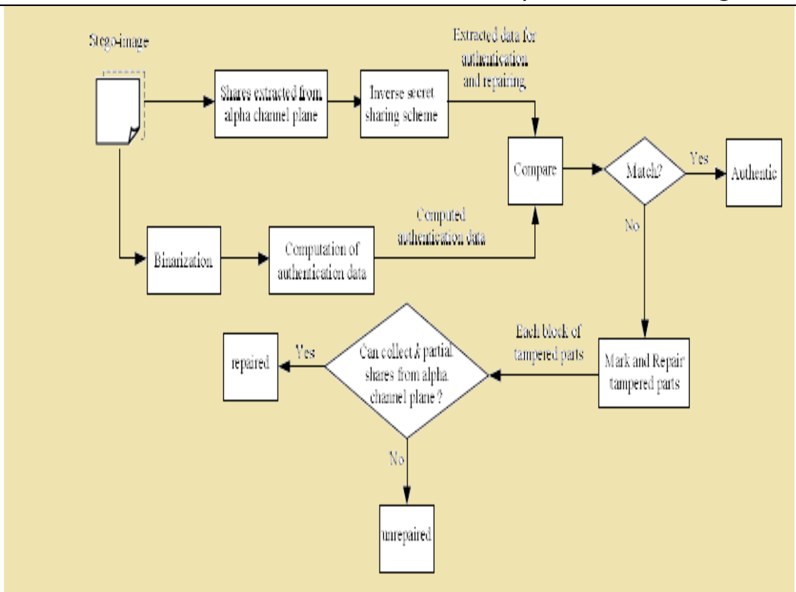

Fig. 4. Verification and self-repairing of stego-image in PNG format for the process of image authentication

Part 3: Self-repairing the original image content

Step 7: (Drawing out of the remaining partial shares) For each block B'_α in S_α', execute the following steps to extract the remaining 4 partial shares q_3 through q_6 of the corresponding block B_b' in S_b' from blocks in S_α' other than B'_α.

(1) Use the key K to collect the 4 pixels in S'_α in the same order as they were randomly selected for B_b' in Step 9 of Algorithm 3, and take out the respective data q_3', q_4', q_5', and q_6' embedded in them.

(2) Subtract 238 from each of q_3' through q_6' to obtain q_3 through q_6, respectively.

Step 8: (Repair the tampered regions) On behalf of each block B' in S' marked as tampered previously, execute the following steps to repair it if possible.

(1) From the 6 partial shares q_1 through q_6 of the block B_b' in S_b' corresponding to B'(two computed in *Step 3(1)* and four in *Step 7(2)* above), select 2 of them, say q_k and q_l, which are not marked as tampered, if possible.

(2) With the shares (k, q_k) and (l, q_l) as input, execute Algorithm 2 to mine the values of d and c_1 (the secret and the first coefficient value) as output.

(3) Transform d and c_1 into two 4-bit binary values and concatenate them to form an 8-bit string W'.

(4) Take the last 6 bits $b_1', b_2', \ldots b_6'$ from W' and check their binary values to repair the corresponding tampered pixel values $y_1', y_2', \ldots, y_6'$ of block B' by the following way: if $b_i' = 0$, set $y_i' = g_1$; otherwise, set $y_i' = g_2$; where $i = 1, 2 \ldots 6$.

Step 9: Take the final S' as the desired self-repaired image S_r.

IV. Experimental Results

Experimental Results Using a Document Image of a Cheque

Experimental results yielded by the use of a document image of a Cheque are shown in Figs.5 through 7. In Figure 5 an authentication result of an image of a Cheque in PNG format is shown where Fig.5(a) Original cover image. Fig.5(b) Binary-like of original image. Fig.5(c) Alpha plane of original image. Fig.5(d) Original image in Stego PNG format.Figure 6 shows a document image of a Cheque in the form of PNG tampered with image editor. Fig.6(a) Original cover image-(edited one). Fig.6(b) Binary-like of edited image. Fig.6(c) Alpha plane of edited image. Fig.6(d) Edited image in Stego PNG format.

Fig. 5 Authentication result of an image of a Cheque in PNG format (a) Original cover image. (b) Binary-like of original image. (c) Alpha plane of original image. (d) Original image in Stego PNG format.

Fig. 6 Authentication result of a document image of a Cheque in the form of PNG tampered with image editor. (a) Original cover image-(edited one). (b) Binary-like of edited image. (c) Alpha plane of edited image. (d) Edited image in Stego PNG format.

Fig. 6: Authentication result of the document image of a Cheque (a) Original cover image-(edited one). (b) Retrieved PNG image. (c) Retrieved original image(cover image)

V. Merits Of The Proposed Method

Along with being capable of data repairing and being blind in nature), the proposed method has several other qualities, which are as follows:

(A) It has higher possibility to survive image content attacks - By the combination of the Shamir scheme, authentication signal generation, and random embedding of multiple shares, the proposed method can survive malicious attacks of common content modifications, such as superimposition, painting, etc., as demonstrated by experimental result.

(B) It provides pixel-level repairs of tampered image parts – If we could collect two non-tampered partial shares, a tampered block can be repaired at the pixel level by this method. This method yields a better repair outcome for texts in images because text characters or letters are smaller in size with many arched strokes and need finer pixel-level repairs when tampered.

(C) Enhancing data security by secret sharing - As a replacement for of hiding data directly into document image pixels, the proposed method embeds data in the form of shares into the alpha channel of the PNG image. Its effect may be regarded as double-fold security protection, one fold contributed by the shares as a form of disguise of the original image data and the authentication signals, and the other fold contributed by the use of the alpha channel plane which is created to be nearly transparent.

(D) Causing no distortion to the input image – Usual image authentication methods usually embed authentication signals into the cover image itself will unavoidably cause damage to the image content to a certain extent. Other than such methods, the proposed method utilizes the pixels values of the alpha channel for the purpose of image authentication and data repairing, leaving the original image (i.e., the grayscale channel) undamaged and so causing no alteration to it. The alpha channel plane may be removed after the authentication process to get the original image.

(E) Use of a new type of image channel for data hiding – Rather than common types of images, a PNG image has the extra alpha channel plane which normally is used to produce transparency to the image. As a comparison, many other methods use LSB's as the carriers of hidden data.

VI. Performance Comparison

Comparison of the capability of the proposed method with those of four existing methods is shown in Table1. All the proposed method will create alteration in the stego-image during the authentication process. More significantly, only the proposed method has the capability of repairing the tampered parts of an authenticated image.

Table 1 Comparison of different document image authentication methods.

	Distortion in stego-image	Tamperring Localization Capability	Repair Capability	Reported Authentication precision	Distribution Of authenticated image parts	Manipulation of data embedding
Wu & Liu [8]	Yes	No	No	Macro-block	Non-blank part	Pixel flippability
Yang & Kot [9]	Yes	Yes	No	33×33 block	Non-blank part	Pixel flippability
Yang& Kot [10]	Yes	No	No	Macro-block	Non-blank part	Pixel flippability
Tzeng & Tsai [11]	Yes	Yes	No	64×64 block	Entire image	Pixel replacement
Propo-sed method	No	Yes	Yes	2×3 block	Entire image	Alpha channel Pixel replacement

VII. Conclusions And Future Enhancement

We have proposed an image authentication method along with a data repair capability for binary-like grayscale document images based on secret sharing. 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 an elegant manner into an alpha channel plane to create a stego-image in the PNG format. For self-repairing the content of a tampered block, the reverse Shamir scheme is used to compute the original content of the block from any 2 un-tampered shares. A measure for enhancing the protection of the data embedded in the alpha channel plane is also proposed. Experimental results have shown to prove the effectiveness of the proposed method. Upcoming studies may be aimed at choices of other block sizes and associated parameters (prime number, coefficients for secret sharing, number of authentication signal bits, etc.) to advance data repair effects. Applications of the proposed method for authentication and repairing of attacked color images may also be tried.

References

[1] A Secret-Sharing-Based Method For Authentication Of Grayscale Document Images Via The Use Of The PNG Image With A Data Repair Capability By Che-Wei Lee, And Wen-Hsiang Tsai,At IEEE Transactions On Image Processing, Vol. 21, No. 1, January 2012.

[2] A Geometry-Based Secret Image Sharing Approach By Chien-Chang Chen And Wen- Yin Fu Department Of Computer Science Hsuan Chuang University Hsinchu, 300 Taiwan Journal Of Information Science And Engineering 24, 1567-1577(2008).

[3] Secret Sharing And Information Hiding By Shadow Images Chin-Chen Chang,The Duc Kieu Department Of Information Engineering And Computer Science, Feng Chia University, Taichung 40724, Taiwan, R.O.C.

[4] Secret Image Sharing With Steganography And Authentication Chang-Chou Lin, Wen-Hsiang Tsai,Department Of Computer And Information Science, National Chiao Tung University, Hsinchu 300,Taiwan,ROC Received 24 October 2002;Received In Revised Form 30 May 2003;Accepted 20 July 2003.

[5] Improvements Of Image Sharing With Steganography And Authentication Ching-Nung Yang , Tse- Shih Chen, Kun Hsuan Yu, Chung-Chun Wang Department Of Computer Science And Information Engineering, National Dong Hwa University, Sec. 2, Da Hsueh Rd., Hualien, Taiwan Received 22 October 2005; Received In Revised Form 18 November 2006; Accepted 26 November 2006.

[6] W. H. Tsai, "Moment-Preserving Thresholding:
A New Approach," Comput. Vis. Graph. Image Process. Vol. 29, No. 3, Pp. 377–393, Mar.1985.

[7] A. Shamir, "How To Share A Secret," Commun.ACM,Vol.22, No.11,Pp.612–613,Nov. 1979.

[8] 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, 2004.

[9] 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.

[10] H. Yang And A. C. Kot, "Pattern-Based Data Hiding For Binary Images Authentication By Connectivity-Preserving," IEEE Trans. On Multimedia, Vol. 9, No. 3, Pp. 475–486, April 2007.

[11] 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 Communications Letters, Vol. 7, No. 9, Pp. 443–445.