Grayscale Image Authentication with Data Repair Capability

ARJUN NICHAL
Electronics and Tele. Engineering Department
Rayat Institute of Research and Development
Satara, INDIA
arjunnichal@gmail.com

BHALCHANDRA GODBOLE
Electronics
Engineering Department
Karmaveer Bhaurao Patil College of Engineering
Satara, INDIA
bhalchandra.godbole@kbpcoes.edu.in

Abstract - Image Authentication technique has gained importance in now days. The digital revolution in Image Processing has made it possible to create manipulate and transmit digital images in a simple and fast manner. Therefore most of the important images such as military, Medical, Companies secret data must be protected against manipulation. So to protect originality and authenticity of multimedia images and important scanned documents various authentication methods are evolved. This paper proposes image authentication scheme with data repair capability by the use of portable network graphics (PNG) image with bitplane slicing method. Experimental results shows that for different attacks authentication system withstand and repair original data as it is. It also shows the effectiveness of the work.

Keywords—Authentication, Fragile Watermarking, Semi Fragile watermarking, Tamper Detection.

1. Introduction

The well-known adage that “the photograph doesn't lie” is no longer true due to the availability of powerful image manipulation software. Digital images have been adopted because of their ease of manipulation, processing, and storage. It is almost impossible to distinguish subjectively which images are original, and which have been manipulated. This technical development has decreased the credibility that photography used to achieve. Image authentication techniques protect images from malicious manipulation at every stage of transmission and storage. Reliable image authentication technology must be able to protect an image from the time it was first produced until the final stage of use. Digital image is a form for preserving important information. However, with the fast advance of digital technologies, it is easy to make visually imperceptible modifications to the contents of digital images. How to ensure the integrity and the authenticity of a digital image is thus a challenge. It is desirable to design effective methods to solve this kind of image authentication problem [1]–[2], particularly for images of documents whose security must be protected. It is also hoped that if part of a document image is verified to have been illicitly altered, the destroyed content can be repaired. Such image content authentication and self-repair capabilities are useful for the security protection of digital documents in many fields, such as important certificates, Important signed digital images, signed documents, scanned cheques, circuit diagrams, art drawings, design drafts,[3] last will and testaments, and so on. In general, authenticity is a relative concept: whether an item is authentic or not is relative to a reference or certain type of representation that is regarded as authentic. Authentication is usually done by checking whether certain rules and relationship which are supposed to be found for an authentic copy are still hold in the test material. Following is the basic image authentication algorithm with data repair capability.

```
If (Received watermark ~ = Original Watermark)
      Statement = Manipulation Occurs.
else
            Statement = Manipulation Not occurs.
end
```

2. Literature Review

In this we discuss different techniques of image authentication. Fragile watermarking systems are very sensitive to all kind of malicious attacks. This system is designed because when manipulation occurs then watermarks are expected to be destroyed completely. Therefore they are useful in various fields such as military, companies’ secret data circulation etc. Fig 1 shows us classification of digital image authentication. It is classified into two main categories that is signature based and digital watermarking based. Digital watermarking based authentication gains importance in the field of information security. It’s further divided into fragile watermarking based, semi-fragile watermarking based and robust watermarking based. In further writing we just focus on all these types of image authentication techniques. Authentication is the process or action of verifying the identity of user or process. Image authentication technique have a recently gained great attention due to its importance for a large number of multimedia application. With the fast development of information technology, the digital image has become an important way of preserving and communicating important information; however, the wide application of image editing software makes it easy to modify the contents of digital images without visual perception. Therefore, how to ensure the credibility of image content has become a challenge. Image authentication technology is an efficient method of overcoming this challenge[4]-[6].

Strict image authentication considers an image as non-authentic when just an image pixel or even one bit of data has been changed. There are applications that need such service[7]-[10]. However, this is not the desired authentication method for most practical cases.

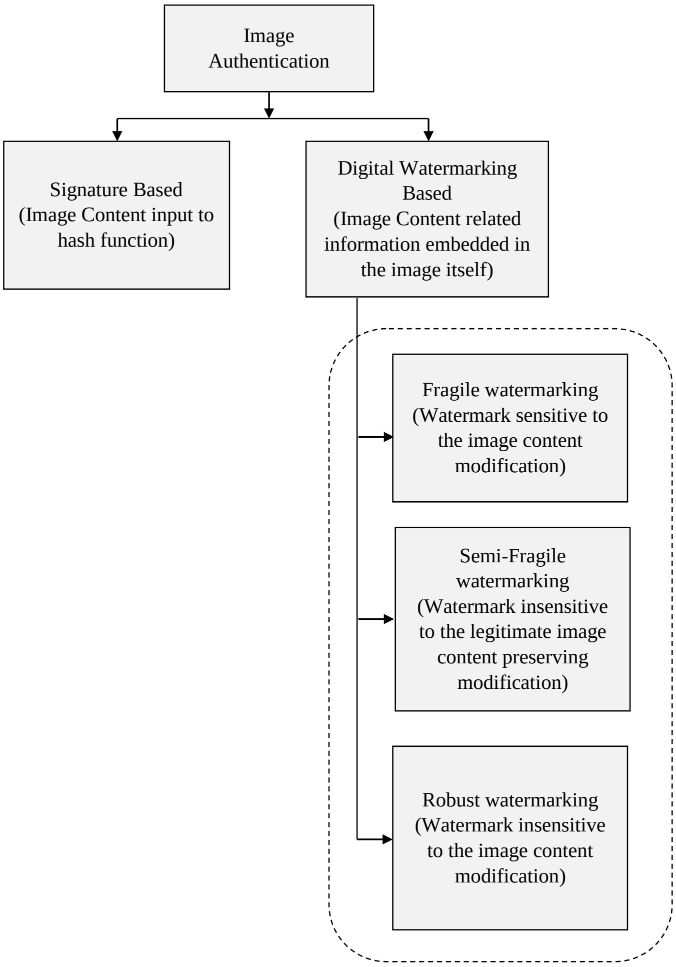


Image Authentication
Signature Based (Image Content input to hash function)
Digital Watermarking Based (Image Content related information embedded in the image itself)
Fragile watermarking (Watermark sensitive to the image content modification)
Semi-Fragile watermarking (Watermark insensitive to the legitimate image content preserving modification)
Robust watermarking (Watermark insensitive to the image content modification)

Fig 1. Classification of Image Authentication Systems

3. Methodology

This describes the overview of the system that is developed. The Grayscale image authentication with data repair capability is proposed. This contains the authentication data embedding and authentication checking algorithms.

The proposed technique first convert input grayscale image into Portable network graphics (PNG) format by adding alpha channel. At the same time binarization of data for authentication and repairing is done. This data is embedded into alpha channel by keeping transparency at highest level. Stego image is generated in PNG format. Following steps are involved in the authentic data embedding process. Fig 2. shows the details of authentic data embedding process.

3.1 Stego Image Formation

Step 1: Cover Image
The input grayscale image is an 8 bit image. That grayscale image we can use for embedding authentic data. The input grayscale image has an extension of PNG, JPG, BMP, etc. The input grayscale image represented using following expression.
Let A be an input grayscale image having size $m * n$ and represented as:

$$A=\left\{ x(i,j) \,\middle|\, \begin{matrix} 1 \leq i \leq m, 1 \leq j \leq n \\ x(i,j)\epsilon\ \{0,1,2,3,4,\ldots\ldots..,255\} \end{matrix} \right\}$$

Where, m is the number of rows present in an image and n is the number of columns present in an image. Pixel values of an image are lies between 0 to 255.

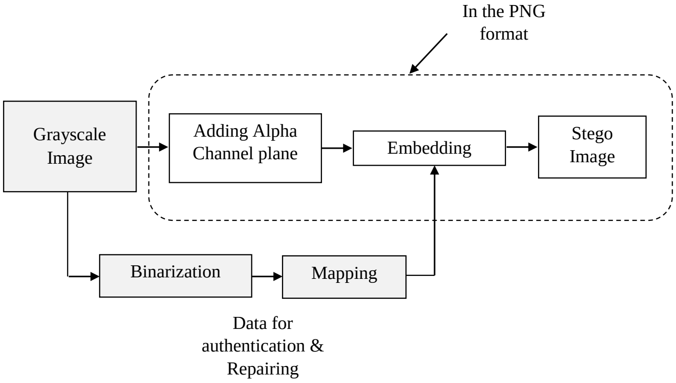


Fig 2. Authentication data Embedding process

Step 2 Alpha Channel Addition
alpha channel will create random transparency in the resulting PNG image, producing an undesirable opaque effect. One way out, is to map the resulting alpha channel values into a small range near their extreme value of 255, yielding a nearly imperceptible transparency effect on the alpha channel plane. The alpha channel is defined as below.

$$Alpha = \left\{ x(i,j) \,\middle|\, \begin{matrix} 1 \leq i \leq m, 1 \leq j \leq n \\ x(i,j) = 255 \end{matrix} \right\}$$

The alpha channel used as a carrier for embedding authentic data. At the same time transparency of the image is also maintained by alpha channel.

Algorithm 1: Preparation of the cover image i.e. Creation of PNG image

Input: Select Cover Image

Output: Modified cover image.png

```
Step 1: Select cover image
        if (cover image is PNG)
             {
              There is no need to add alpha channel
             }
            else
            {
             Add alpha channel to form PNG image
            }
Step 2: End
```

Step 3: Binarization and mapping of authentic data for embedding purpose.
In this step Binarization of cover image is done. For Binarization of cover image for embedding purpose bitplane slicing is used and using bitplane replacement embedding operation is done in cover image.

For grayscale images have 8 bit-planes, this can be represented as follows:

$Pl_k = \left\{ x(i,j,k) \middle| \begin{matrix} 1 \le i \le r, 1 \le j \le c \\ x(i,j,k) \in \{0,1\} \end{matrix} \right\}$ ... ... Where: 1 ≤ k ≤ 8Input cover image is a grayscale image has 8 bits/pixels. It also has 8 biplanes.

Algorithm 2: Binarization (Bitplane slicing)

Input: Cover Image

Output: 8 bitplanes of cover image

Step 1: Select cover image
Step 2: Slice the cover image into 8 bitplanes

$$Pl_k = \left\{ x(i,j,k) \middle| \begin{matrix} 1 \le i \le r, 1 \le j \le c \\ x(i,j,k) \in \{0,1\} \end{matrix} \right\}$$

$Where: 1 \le k \le 8$
$Pl_1, Pl_2, Pl_3, Pl_4, Pl_5, Pl_6, Pl_7 \text{ and } Pl_8$
Step 3: For embedding purpose we only select
$Pl_4, Pl_5, Pl_6, Pl_7 \text{ and } Pl_8$ because these bitplanes has highest
information as compared with other bitplanes.
Step 4: End

Step 4: Mapping and Embedding

This is the most important step in the whole process. In this step bitplane 4,5,6,7,8 are embedded into alpha channel and alpha channel will combined with grayscale image. Figure 4.6 shows a clear picture about embedding of authentic data in alpha channel. While embedding authentic data in alpha channel we only consider bitplane number 4 to bitplane number 8. Bitplane number 4 to 8 has highest information of the cover image. The size of alpha channel and size of the input grayscale image should be same.

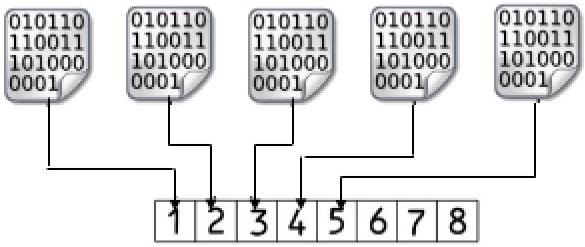


Fig 3. Replacing Alpha channel bitplanes with Authentic Data

1st five bitplanes of alpha channel are replaced by authentic data to get final stego alpha channel. The following algorithm 3 is about formation of stego alpha channel. In this algorithm embedding of bitplanes in alpha channel is done

Algorithm 3: Mapping and Embedding bitplanes in alpha channel

Input: Alpha Channel and bitplanes
$Pl_1, Pl_2, Pl_3, Pl_4, Pl_5, Pl_6, Pl_7 \text{ and } Pl_8$
Output: Stego Alpha Channel ($Stego_{alpha}$)

Step 1: Take Alpha Channel (the size of alpha channel and cover
grayscale image should be same)
Step 2: Take $Pl_4, Pl_5, Pl_6, Pl_7 \text{ and } Pl_8$ from slicing of grayscale
image

$$Pl_k = \left\{ x(i,j,k) \middle| \begin{matrix} 1 \le i \le r, 1 \le j \le c \\ x(i,j,k) \in \{0,1\} \end{matrix} \right\}$$

$Where: 4 \le k \le 8$
$Pl_4, Pl_5, Pl_6, Pl_7 \text{ and } Pl_8$
Step 3: Alpha channel is a matrix of all 255, slice alpha channel
as well

$$Al_k = \left\{ alpha(i,j,k) \middle| \begin{matrix} 1 \le i \le r, 1 \le j \le c \\ alpha(i,j,k) \in \{0,1\} \end{matrix} \right\}$$

$Where: 1 \le k \le 8$
$Al_1, Al_2, Al_3, Al_4, Al_5, Al_6, Al_7 \text{ and } Al_8.$
Bitplanes of alpha channels consist of all ones.
Step 4: Stego alpha channel is formed by multiplying binary equivalent multiplier.

$$Stego_{alpha} = Pl_8 * 2^0 + Pl_7 * 2^1 + Pl_6 * 2^2 + Pl_5 * 2^3 + Pl_4 * 2^4 + Al_3 * 2^5 + Al_2 * 2^6 + Al_1 * 2^7$$

Step 5: End

Step 5: Formation of stego grayscale image

Final stage of this process is of forming stego grayscale image. The stego alpha channel is combined with the input cover image for getting stego grayscale image. Algorithm 4 is about combining stego alpha channel with cover image. The output of this algorithm is stego grayscale image.

Algorithm 4: Forming Stego Grayscale Image

Input: Cover Image and Stego alpha
Channel
Output: Stego Grayscale Image S

Step 1: Take Cover Image and Stego alpha channel
Step 2: Combine cover image and stego alpha channel together
Step 3: Stego Grayscale Image S= Cover Image + Stego Alpha Channel
Step 4: End

3.2 Stego Image Authentication

This includes stego image verification and self-repairing process. In this process first authentic data is extracted from stego image and it is matched with the computed authentication data. If both data's are matched with each other then it is authentic image. If it is not matched then it is not authentic data. If second case occurs then move for further processes of verification and self-repair.

The stego image authentication includes three important stages.
Stage 1: Extraction of authentic data from alpha channel
Stage 2: Verification of the stego image
Stage 3: Self repairing of original image content.
Figure 4 shows all three stages in detail. Stage 1 is about extraction of authentic data from alpha channel. Stage 2 is about verification of stego image and stege 3 is about self-recovery of original image content.

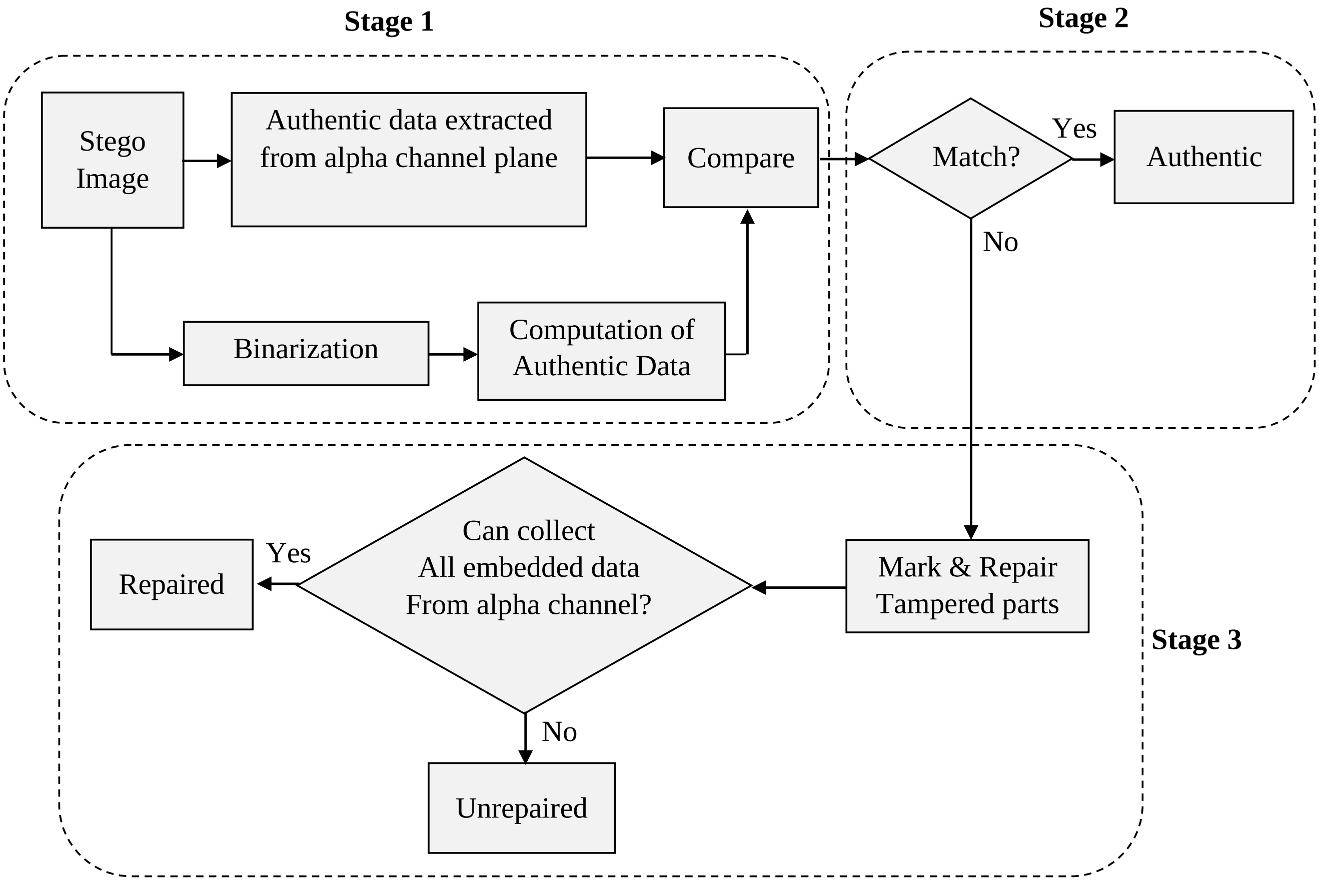

Stage 1
Stage 2
Stage 3
Stego Image
Authentic data extracted from alpha channel plane
Compare
Binarization
Computation of Authentic Data
Match?
Yes
No
Authentic
Mark & Repair Tampered parts
Can collect All embedded data From alpha channel?
Yes
No
Repaired
Unrepaired

Fig 4. Authentication process including verification and self-repairing of a stego grayscale image in PNG format.

Above 3 stages are briefly discussed in algorithm 5. In this algorithm 5 stego image S is the input and self-repaired image R is the output.

```
Algorithm 5: Authentication of stego grayscale image
Input: Stego image $S$
Output: Self repaired image $R$

Stage 1: Extraction of authentic data from alpha channel
Step 1: Take Stego image
Step 2: Extract alpha channel from stego PNG image
Step 3: Take bitplanes one to five from alpha channels by bitplane slicing process and indicate it as a $Sl_1, Sl_2, Sl_3, Sl_4$ and $Sl_5$
Step 4: Now combine all above alpha channel bitplanes to form a authentication signal for matching and repairing, indicate it as a $A_{signal}$
$A_{signal} = Sl_5 * 2^3 + Sl_4 * 2^4 + Sl_3 * 2^5 + Sl_2 * 2^6 + Sl_1 * 2^7$   (4.8)

Stage 2: Verification of the stego image
Step 5: For verification of the authentic data bitplane slicing (Binarization) of stego image (Grayscale data) is done.
Step 6: Take last five bitplanes from above process and indicated as $Pl_4, Pl_5, Pl_6, Pl_7$ and $Pl_8$
Step 7: Now combine all above bitplanes to form a Verification signal for matching, indicate it as a $V_{signal}$
$V_{signal} = Pl_4 * 2^3 + Pl_5 * 2^4 + Pl_6 * 2^5 + Pl_7 * 2^6 + Pl_8 * 2^7$   (4.9)
Step 8: if ($A_{signal}$= $V_{signal}$ )
        {
        Image is Authentic,
        }
    else
      {
      Image is not Authentic, Need to repair.
      }
Stage 3: Self repairing of original image content
Step 9: In this step data is repaired by using $A_{signal}$ and $V_{signal}$
 if ($A_{signal}(i,j) \sim = V_{signal}(i,j)$)
{
$R(i,j) = A_{signal}(i,j)$
}
Step 10: Take the final $R$ as the desired self-repaired image.
```

4. Results

In this, simulation results are given to demonstrate the performance of the Developed algorithms. The main idea of these algorithm is to find image or scanned document is authentic or not and if it not authentic then find alteration mode and repair tampered data. The main objectives of developing these algorithms are to embed authentication data with host file not in the separate file, to tampered area on image, to repair original content of tampered area and keep visual quality of image high after embedding authentication data in host file. Different types of objective parameters are used to analyze quality of the stego image and quality of the repaired image. Objective image quality assessment (IQA) comprises two categories one is full reference IQA and second is No reference IQA. For analyzing above algorithm parameters from full reference IQA have been chosen. The following parameters are used from FR-IQA for analyzing quality of the stego and repaired image. Mean Square Error (MSE), Peak-Signal to Noise Ratio (PSNR), Normalized Cross Correlation (NCC), Average Difference (AD), Structural content (SC). Two more parameters are defined specifically for analyzing developed algorithms are Embedding capacity (EC), Number of bits embedded (NBE).

The results are taken for different images like grayscale image, Grayscale document image, color image and color document image. This algorithm is also tested for criminal face authentication. Algorithm can take any size of image. There is no restriction for image size.

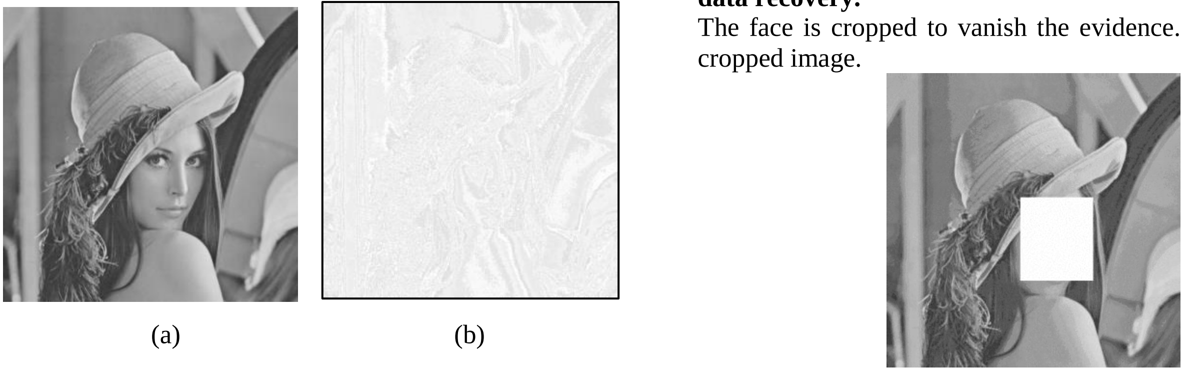
The results are taken in following sequence.

1. First authentication data is embedded into the alpha channel
2. Various quality parameters are recorded in a table
3. Different attacks are applied over a stego image
4. Attacked image is repaired and quality parameters are recorded.

Results for Grayscale image authentication

1. Authentication Data embedding

The standard Lena image is taken as a cover image. The size of the image is 512*512.



(a) (b) (c)

Fig 5 Authentication Data Embedding Process of Grayscale Image (a) Cover Image (b) Alpha Channel + Authentication Data (c) Stego image

The quality parameters for above embedding process is listed below. The parameters consist of PSNR, MSE, SC, NCC, AD, EC and NBE.

Table 1: Image quality parameter values for Grayscale Image

Image Quality Parameter grayscale image	Values
PSNR in dB of Grayscale Image	100
PSNR in dB of Alpha Channel	22.7994
MSE of Alpha Channel	338.634
SC of Alpha Channel	0.880012
AD of Alpha Channel	15.9625
NCC of Alpha Channel	1.06522
EC of Alpha Channel	2097152
NBE in Alpha Channel	1310720

2. External Attacks on Stego Image, Authenticity verification and data recovery

The images are attacked by several techniques. In this study we have focused on intentional attacks. Intentional attacks comprises following manipulation activities.

i) Image Cropping
ii) Text attack

Now algorithm 5 from chapter 4 is checked for all above attacks for authenticity verification and data recovery.

i) Image Cropping Attack, Authenticity verification and data recovery.

The face is cropped to vanish the evidence. Fig 6 shows the cropped image.

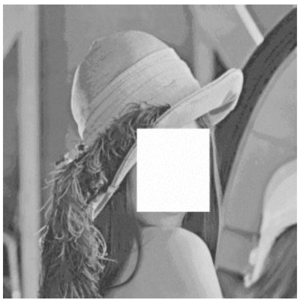
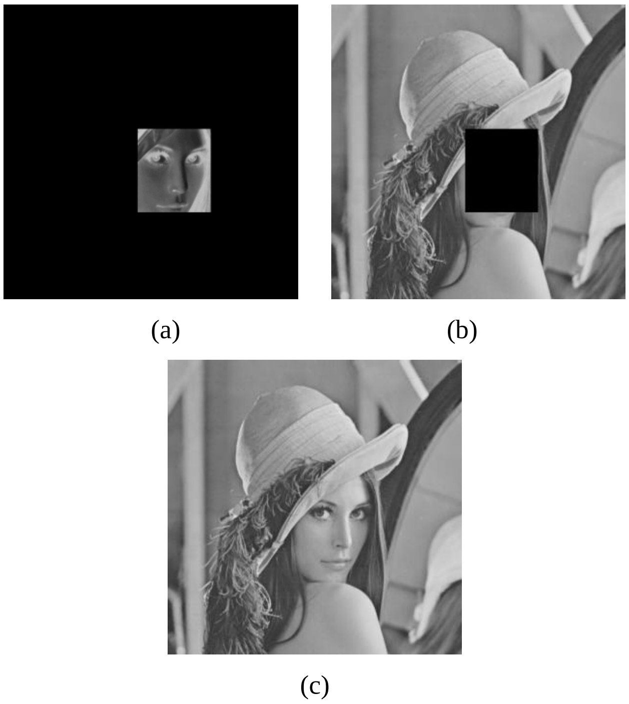


Fig 6. Cropping attack on Grayscale Image

Image cropping is applied on stego image to vanish the identity of image. Now authenticity verification and data recovery is applied over cropped image to recover original content of the lena image.



(a) (b)

(c)

Fig 7. Authentication, verification and Data recovery of Grayscale Image (a) Subtraction of Authentication data and verification data (b) Tampered area identification (c) Repaired Image

Fig 7 shows detailed Authentication verification and Data recovery outputs. Fig 7 (a) is a subtraction of authentication data embedded into alpha channel and data extracted for verification from grayscale image. Figure 7 (b) is an identification of the tampered area and Figure 7 (c) is a repaired image. This image is obtained by mapping authentication data to tampered area.

The following quality parameters are recorded for checking quality of the repaired image. Quality checking is done between repaired image and input cover image. Following table 2 comprises all the quality parameters.

Table 2: Repaired image quality parameters Grayscale Image for cropping attack

Image Quality Parameter Grayscale Image	Values
PSNR in dB	46.7197
MSE	1.22589
NCC	1.00176
SC	0.99643
AD	0.245876

ii) Text Attack, Authenticity verification and data recovery

In this attack externally text is inserted on image to claim ownership of the image. Figure 5.4 shows the text attack.

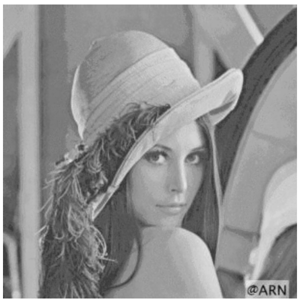
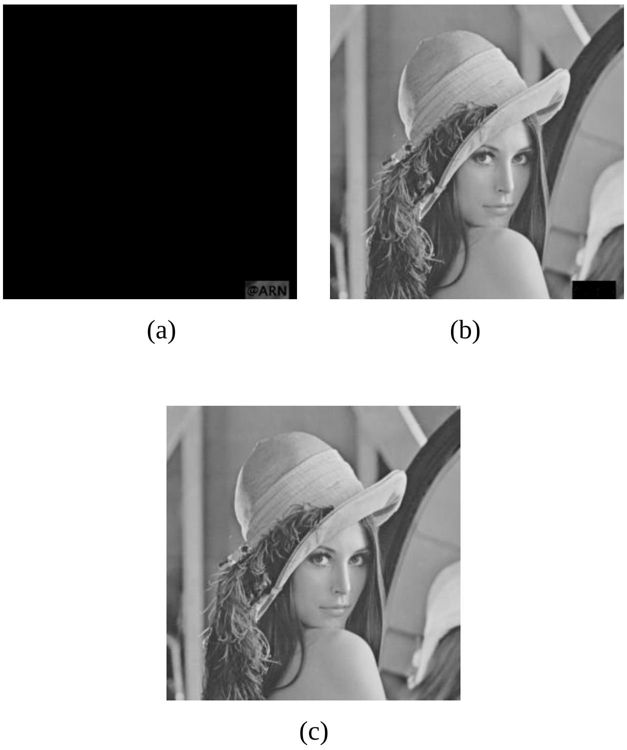


Fig 8. Text attack on Grayscale Image

Anyone can claim ownership of the image by inserting the text on image. In above Figure 8 @ARN text is inserted externally to claim the ownership of the image.



(a) (b)

(c)

Fig 9. Authentication, verification and Data recovery of Grayscale Image (a) Subtraction of Authentication data and verification data (b) Tampered area identification (c) Repaired Image

Figure 9 shows detailed Authentication verification and Data recovery outputs. Figure 9 (a) is a subtraction of authentication data embedded into alpha channel and data extracted for verification from grayscale image. Figure 9 (b) is an identification of the tampered area and Figure 9 (c) is a repaired image. This image is obtained by mapping authentication data to tampered area.

The following quality parameters are recorded for checking quality of the repaired image. Quality checking is done between repaired image and input cover image. Following table 5.3 comprises all the quality parameters.

Table 3: Repaired image quality parameters Grayscale Image for text attack

Image Quality Parameter Grayscale Image	**Values**
PSNR in dB	55.5631
MSE	0.159996
NCC	1.00017
SC	0.999651
AD	0.03228

5. Conclusion

The algorithms developed for authentication of grayscale image embeds authentication data in the host file rather than in a separate data file. If authentication data embeds in a separate data file and if it is lost due to manual mistakes then it’s a huge loss. In this case no one can check whether given image is authentic or not. This embedding approach increase complexity at the authentication checking. The developed algorithms embeds authentication data in alpha channel not in the grayscale image pixel. This embedding approach in an alpha channel keeps grayscale image or color image pixels unchanged. The results shows the quality of the stego image after embedding authentication data is high. The developed algorithms embeds authentication data in an alpha channel. Alpha channel produces transparency effect to the image. Authentication data is embedded into the alpha by using bitplane slicing in the highest bitplanes to reduce the opaque effect visible in the stego-image. The opaque effect visible in the stego-image when authentication data embedded into the lower bitplanes of an alpha channel. There are only few techniques in the research which works for authentication and data recovery of grayscale image. Most of the techniques in the literature embeds binary like data to check authenticity and for recovery. The proposed techniques embed five bitplanes of grayscale image in an alpha channel. These five bitplanes has highest information of the grayscale image. At the time of authenticity checking and recovery, maximum grayscale data is repaired. The result shows the quality of repaired image is 90%.

References

[1] Rafeal Gonzalez et al., “Digital image processing”, *3rd edition, published by Pearson India Education services Pvt Ltd, 2016.*

[2] Anand, A., Raj, A., Kohli, R., & Bibhu, V., “Proposed symmetric key cryptography algorithm for data security”, *International Conference on Innovation and Challenges in Cyber Security (ICICCS-INBUSH), pp. 159-162, 2016.*

[3] Omar Farook Mohammad et al., “A Survey and Analysis of the Image Encryption Methods”, *International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, pp. 13265-13280, Number 23, 2017.*

[4] H. B BasanthKumar, “Digital Image Watermarking: An Overview”, *Oriental Journal of Computer Science & Technology, Vol. 9, No. (1): pp. 07-11, April 2016.*

[5] Naina Choubey and Mahendra Kumar Pandey, “Transform based Digital Image Watermarking: An Overview”, *International journal of Computer Trends and Technology (IJCTT), 24(2); pp. 80-83, 2015.*

[6] R. English, "Comparison of High Capacity Steganography Techniques *", IEEE, International Conference of Soft Computing and Pattern Recognition, pp.448-453, December 2010.*

[7] H. Sajedi, and M. Jamzad, " Secure steganography based on embedding capacity *", Springer Verlag, International Journal of Information Security, Vol.8, Issue 6, pp.433-445, August 2009.*

[8] T. Morkel, " Image Steganography Applications for Secure Communication ", *M.Sc. thesis, Faculty of Engineering, Built Environment and Information Technology University of Pretoria, Pretoria, pp.126-132, May 2012.*

[9] Y. Lee, H. Kim, and Y. Park, “A new data hiding scheme for binary image authentication with small image distortion”, *Inf. Sci., vol. 179, no. 22, pp. 3866–3884, Nov. 2009.*

[10] H. Yang and A. C. Kot, “Pattern-based data hiding for binary images authentication by connectivity-preserving”, *IEEE Trans. Multimedia, vol. 9, no. 3, pp. 475–486, Apr. 2007.*