***2021 International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON) Pune, India. Oct 29-30, 2021***

Grayscale Image Authentication with Data Repair Capability

Arjun Nichal

*Research Scholar,*

*Electronics and Tele. Engineering Department Rayat Institute of Research and Development* Satara, India

[arjunnichal@gmail.com](mailto:arjunnichal@gmail.com)

Dr. Bhalchandra Godbole *Associate Professor, Electronics Engineering Department*

*Karmaveer Bhaurao Patil College of Engineering* Satara, India [bhalchandra.godbole@kbpcoes.edu.in](mailto:bhalchandra.godbole@kbpcoes.edu.in)

***Abstract* - The technique of image authentication has become increasingly popular in recent years. The digital revolution in image processing has made it feasible to quickly and easily create, manipulate, and transport digital images.**

copy are still present in the test material. Following is the basic image authentication algorithm with data repair capability.

**Therefore most of the vital photos such as military, Medical,**

**Companies secret data must be protected against alteration. So to safeguard originality and validity of multimedia images and essential scanned documents many authentication methods are evolved. The usage of portable network graphics (PNG) images with bitplane slicing method in this study provides an image authentication scheme with data repair capability. Experiments indicate that the authentication system can withstand and repair original data in the face of various attacks. It also demonstrates the work's effectiveness.**

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

1. INTRODUCTION

Due to the advent of advanced image editing tools, the saying "the photograph doesn't lie" is no longer accurate. Because of their ease of manipulation, processing, and storage, digital photographs have become popular. It's nearly hard to tell which photographs are genuine and which have been modified on a subjective level. The credibility that photography used to have has been eroded as a result of technological advancements. At every level of transmission and storage, image authentication algorithms safeguard images from malicious alteration. Image authentication system that is reliable must be able to protect an image from the time it is created until it is used. A digital image is a method of archiving vital data. With the rapid advancement of digital technologies, it is now possible to make virtually undetectable changes to the contents of digital photos. As a result, ensuring the integrity and authenticity of a digital image is a difficult task. Effective approaches for solving this type of image authentication problem [1]–[2] are desirable, especially for photos of documents whose security must be preserved. It's also intended that if a portion of a document image is found to have been tampered with, the damaged content can be restored. These image content authentication and self-repair capabilities are useful for the security protection of digital documents in a variety of fields, including important certificates, important signed digital images, signed documents, scanned cheques, circuit diagrams, art drawings, design draughts, and last will and testaments, among others. Authenticity is a relative concept in general: whether or not an object is authentic depends on a reference or a specific sort of representation that is considered authentic. Authentication is typically accomplished by determining if certain rules and relationships that are expected to be present in an authentic

*If (Received watermark ~ = Original Watermark) Statement = Manipulation Occurs.*

*else*

*Statement = Manipulation Not occurs.*

*end*

1. 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 confirming a user's or process's identity. Due to its usefulness in a wide range of multimedia applications, image authentication techniques have recently received a lot of attention. The digital image has become an important way of keeping and sharing significant information due to the rapid development of information technology; nevertheless, the widespread use of image editing software makes it easy to modify the contents of digital images without visual perception. As a result, ensuring the credibility of visual content has become a difficult task. Image authentication technology is a good way to get around this problem. [4]-[6].

When a single image pixel or even a single piece of data is modified, strict image authentication considers the image non-authentic. There are some applications that require this type of service [7]-[10]. In most circumstances, however, this is not the preferred authentication mechanism.

Where, 𝑚 is the number of rows present in an image and

Digital Watermarking Based

(Image Content related information embedded in the image itself)

Signature Based (Image Content input to

hash function)

Image Authentication

𝑛 is the number of columns present in an image. Pixel values of an image are lies between 0 to 255.



In the PNG format

Mapping

Grayscale Image

Binarization

Embedding

Stego Image

Adding Alpha Channel plane

Data for authentication & Repairing

Fig 2. Authentication data Embedding process

# Step 2 Alpha Channel Addition

The alpha channel will cause unpredictable transparency in the final PNG image, resulting in an opaque look that is undesired. One solution is to translate the resulting alpha channel values into a tiny range near 255, resulting in an almost undetectable transparency effect on the alpha channel plane. The alpha channel is described in the following way.

1 ≤ i ≤ 𝑚, 1 ≤ j ≤ 𝑛

𝐴𝑙𝑝ℎ𝑎 = { 𝑥(i, j) |

𝑥(i, j) = 255 }

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.

Fig 1. Classification of Image Authentication Systems

|  |
| --- |
| **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 |

1. METHODOLOGY

This describes the overview of the system that is

Robust watermarking (Watermark insensitive to the image content modification)

Semi-Fragile watermarking (Watermark insensitive to the legitimate image content preserving modification)

Fragile watermarking (Watermark sensitive to the image content modification)

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.

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

# 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:

represented as:

𝑃𝑙𝑘

1 ≤ i ≤ 𝑟, 1 ≤ j ≤ 𝑐

= {𝑥(i, j, 𝑘)| 𝑥(i, j, 𝑘) c {0,1} } … … Where: 1 ≤

A= { 𝑥(i, j) | 1 ≤ i ≤ 𝑚, 1 ≤ j ≤ 𝑛 }

𝑥(i, j)c {0,1,2,3,4, ,255}

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

Wℎ𝑒𝑟𝑒: 4 ≤ 𝑘 ≤ 8

𝑃𝑙4, 𝑃𝑙5, 𝑃𝑙6, 𝑃𝑙7 𝑎𝑛𝑑 𝑃𝑙8

Step 3: Alpha channel is a matrix of all 255, slice alpha

channel as well

1 ≤ i ≤ 𝑟, 1 ≤ j ≤ 𝑐

( )

𝑃𝑙𝑘 = {𝑥 i, j, 𝑘 | }

𝐴𝑙𝑘

= {𝑎𝑙𝑝ℎ𝑎(i, j, 𝑘)|1 ≤ i ≤ 𝑟, 1 ≤ j ≤ 𝑐}

𝑎𝑙𝑝ℎ𝑎(i, j, 𝑘) c {0,1}

Wℎ𝑒𝑟𝑒: 1 ≤ 𝑘 ≤ 8

𝑥(i, j, 𝑘) c {0,1}

Wℎ𝑒𝑟𝑒: 1 ≤ 𝑘 ≤ 8

𝐴𝑙1, 𝐴𝑙2, 𝐴𝑙3, 𝐴𝑙4, 𝐴𝑙5, 𝐴𝑙6, 𝐴𝑙7 𝑎𝑛𝑑 𝐴𝑙8.

𝑃𝑙1, 𝑃𝑙2, 𝑃𝑙3, 𝑃𝑙4, 𝑃𝑙5, 𝑃𝑙6, 𝑃𝑙7 𝑎𝑛𝑑 𝑃𝑙8

Step 3: For embedding purpose we only select

𝑃𝑙4, 𝑃𝑙5, 𝑃𝑙6, 𝑃𝑙7 𝑎𝑛𝑑 𝑃𝑙8 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 image about embedding of authentic data in alpha channel. While embedding authentic data in

Bitplanes of alpha channels consist of all ones.

Step 4: Stego alpha channel is formed by multiplying binary equivalent multiplier.

𝑆𝑡𝑒𝑔𝑜𝑎𝑙𝑝ℎ𝑎 = 𝑃𝑙8 \* 20 + 𝑃𝑙7 \* 21 + 𝑃𝑙6 \* 22 +

𝑃𝑙5 \* 23 + 𝑃𝑙4 \* 24 + 𝐴𝑙3 \* 25 + 𝐴𝑙2 \* 26 + 𝐴𝑙1 \* 27

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.

alpha channel we only consider bitplane number 4 to bitplane number 8. Bitplane number 4 to 8 has highest information of

|  |
| --- |
| **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 |

the cover image. The size of alpha channel and size of the input grayscale image should be same.

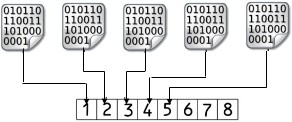


Fig 3. Replacing Alpha channel bitplanes with Authentic Data

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

1. *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

|  |
| --- |
| **Algorithm 3: Mapping and Embedding bitplanes in alpha channel** |

further processes of verification and self-repair.

The stego image authentication includes three important

**Input:** Alpha Channel and bitplanes

𝑃𝑙1, 𝑃𝑙2, 𝑃𝑙3, 𝑃𝑙4, 𝑃𝑙5, 𝑃𝑙6, 𝑃𝑙7 𝑎𝑛𝑑 𝑃𝑙8

**Output:** Stego Alpha Channel (𝑆𝑡𝑒𝑔𝑜𝑎𝑙𝑝ℎ𝑎)

Step 1: Take Alpha Channel (the size of alpha channel and cover

grayscale image should be same)

Step 2: Take 𝑃𝑙4, 𝑃𝑙5, 𝑃𝑙6, 𝑃𝑙7 𝑎𝑛𝑑 𝑃𝑙8 from slicing of grayscale

image

1 ≤ i ≤ 𝑟, 1 ≤ j ≤ 𝑐

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.

𝑃𝑙𝑘 = {𝑥(i, j, 𝑘)|

𝑥(i, j, 𝑘) c {0,1} }

**Stage 1**

**Stage 2**



Yes

Match?

No

Yes

Can collect All embedded data

From alpha channel?

**Stage 3**

No

Computation of Authentic Data

Stego Image

Unrepaired

Mark & Repair Tampered parts

Repaired

Binarization

Authentic

Compare

Authentic data extracted from alpha channel plane

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.

Step 7: Now combine all above bitplanes to form a Verification signal for matching, indicate it as a 𝑉𝑠ig𝑛𝑎𝑙

𝑉𝑠ig𝑛𝑎𝑙 = 𝑃𝑙4 \* 23 + 𝑃𝑙5 \* 24 + 𝑃𝑙6 \* 25 + 𝑃𝑙7 \* 26 + 𝑃𝑙8 \* 27 (4.9)

|  |
| --- |
| **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  𝑆𝑙1, 𝑆𝑙2, 𝑆𝑙3, 𝑆𝑙4 𝑎𝑛𝑑 𝑆𝑙5  Step 4: Now combine all above alpha channel bitplanes to  form a authentication signal for matching and repairing, indicate it as a 𝐴𝑠ig𝑛𝑎𝑙  𝐴𝑠ig𝑛𝑎𝑙 = 𝑆𝑙5 \* 23 + 𝑆𝑙4 \* 24 + 𝑆𝑙3 \* 25 + 𝑆𝑙2 \* 26 + 𝑆𝑙1 \* 27 (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 𝑃𝑙4, 𝑃𝑙5, 𝑃𝑙6, 𝑃𝑙7 𝑎𝑛𝑑 𝑃𝑙8 |

Step 8: if (𝐴𝑠ig𝑛𝑎𝑙= 𝑉𝑠ig𝑛𝑎𝑙 )

{

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 𝐴𝑠ig𝑛𝑎𝑙 and

𝑉𝑠ig𝑛𝑎𝑙

if (𝐴𝑠ig𝑛𝑎𝑙(i, j) ∼ = 𝑉𝑠ig𝑛𝑎𝑙(i, j))

{

𝑅(i, j) = 𝐴𝑠ig𝑛𝑎𝑙(i, j)

}

Step 10: Take the final 𝑅 as the desired self-repaired image.

1. 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.



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

TABLE I. 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 |

1. *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.

* 1. Image Cropping
  2. Text attack

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

1. *Image Cropping Attack, Authenticity verification and data recovery.*

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

(a)

(b)

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.

(c)

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

* 1. (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 II. 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 |

1. *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.

* 1. (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 III. 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 |

1. CONCLUSION

The algorithms developed for authentication of grayscale and color images 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,*](http://link.springer.com/journal/10207) *Vol.8,* [*Issue 6,*](http://link.springer.com/journal/10207/8/6/page/1) *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.*