Reversibility and Security Preservation of Images with Embedded Data

Reswin M R, Gopakumar G

***Abstract*— In image data hiding methods where the distortion of the host image after the data extraction is intolerable and the availability of the original image is in high demand such as in applications such as medical image sharing, reversible data hiding come into scene. Reversible data hiding (RDH) is a method in which the host image can be fully restored after data extraction from the host image .The method here presents a reversible data hiding method accompanied with an encryption method so as to ensure the security of the host image which is in high demand as well as the security of the message or data hidden in the host image respectively. For reversible data hiding, an efficient extension of the histogram modification technique by considering the differences between adjacent pixels instead of simple pixel value is used. A binary tree structure is used to solve the problem of communicating pairs of peak points. A histogram shifting technique is used to prevent overflow and underflow.To further ensure the security of the host image as well as the message inside it, chaos encryption method which combines the features of discrete chaos encryption and logistic chaos encryption is used. With the above method high capacity and low distortion can be achieved efficiently.**

***Index Terms*— Chaos encryption, Histogram shifting Reversible data hiding.**

I. INTRODUCTION

Data hiding are a group of techniques used to put a secure data in a host media (like images) with small deterioration in host. In most cases of data hiding, the cover media becomes distorted due to data hiding and cannot be inverted back to the original media. That is, cover media has permanent distortion even after the hidden data have been removed. In some applications, such as medical diagnosis and law enforcement it is desired that the original cover media must be recovered efficiently with no loss. The marking techniques satisfying this requirement are referred to as reversible, lossless, distortion-free or invertible data hiding techniques.

With the development of medical instruments, it gets much easier for the medical doctors to make diagnoses of the patient's diseases by using medical images. Due to the digital nature and enormous amount of medical images, several issues may arise. First of all, to protect the privacies of the patient, embed data into medical images secretly. Next, during the data extraction phase, extract the data without inducing any alterations in the host image. And thirdly, due to the vast amount of images, authentication of patient's images should be done in order to avoid the probability of mistakenly diagnosing another patient's image. For the three kinds of cases observed, a data hiding scheme is employed to meet the goals for the reception of correct information for the medical doctors, and the proper treatment to the patients.

Reversible data hiding is employed to reach the goals mentioned above. Reversible data hiding is a technique to embed additional message into some distortion-unacceptable cover media, such as military or medical images, with a reversible manner so that the original cover content can be perfectly restored after the extraction of the hidden message. Reversible data hiding is a hybrid method which combines various techniques to ensure the reversibility. Its feasibility is mainly due to the lossless compressibility of natural images.

The proposed method presents a reversible data hiding method accompanied with an encryption method so as to further increase the embedding capacity and to ensure the security of the host image which is in high demand as well as the security of the message or data hidden in the host image respectively. For reversible data hiding, an efficient extension of the histogram modification technique by considering the differences between adjacent pixels instead of simple pixel value is used. Distribution of pixel differences is used to achieve large hiding capacity while keeping the distortion low. A histogram shifting technique is used to prevent overflow and underflow.

To further ensure the security of the host image the host image is encrypted using an encrypting algorithm that ensures reversibility .Chaos-based image encryption can provide a class of very promising methods that can fulfill many of the requirements that conventional encryption methods cannot provide particularly with a good combination of speed, security, and flexibility. The encryption-decryption process used in this proposed method combines the benefits of two chaotic systems namely discrete chaotic encryption and logistic map encryption.

II. METHODS

A. PROPOSED METHOD

The proposed method presents a reversible data hiding method accompanied with a compression and encryption method so as to further increase the embedding capacity and to ensure the security of the host image which is in high demand as well as the security of the message or data hidden in the host image respectively. For reversible data hiding, an efficient extension of the histogram modification technique by considering the differences between adjacent pixels instead of simple pixel value is used. A binary tree structure is

Manuscript received April 17, 2014
Reswin M R, Dept. of Computer Science,College of Engineering,Chengannur,Kerala , India
Gopakumar G, Dept. of Computer Science,College of Engineering,Chengannur,Kerala,India

used to solve the problem of communicating pairs of peak points. Distribution of pixel differences is used to achieve large hiding capacity while keeping the distortion low. A histogram shifting technique is used to prevent overflow and underflow. To further ensure the security of the host image the host image is encrypted using an encrypting algorithm that ensures reversibility.

The encryption-decryption process used in this proposed method combines the benefits of two chaotic systems namely discrete chaotic encryption and logistic map encryption. First the properties of both the above said systems are discussed and finally the benefits of the encryption scheme used in the proposed method by combining the discrete chaotic encryption and logistic map encryption are discussed.

Fig.1 shows the block diagram of the proposed method at the sender side. The procedure is as follows.

- The cover image or the host image is read as input image.
- Input the data that is to be hidden in the host image.
- Hide the data inside the host image.
- The host image together with the hidden data is encrypted as a whole to get the output image.

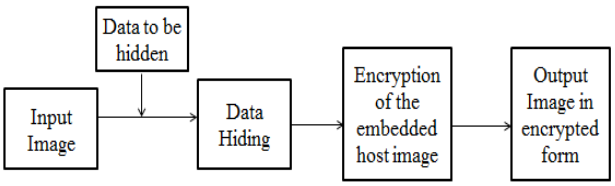


Fig. 1. Work flow diagram, Sender side

Fig. 2 shows the block diagram of the proposed method at the receiver side. The procedure is as follows.

- The encrypted image is read as the input image.
- The image is decrypted.
- The data inside the image is extracted and the original host image is restored.

.

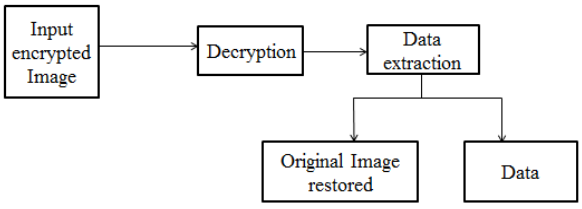


Fig. 2. Work flow diagram, Receiver side

B. Data Hiding Method

In the proposed method, for reversible data hiding, an efficient extension of the histogram modification technique by considering the differences between adjacent pixels instead of simple pixel value is used. Since image neighbour pixels are strongly correlated, the distribution of pixel difference has a prominent maximum, that is, the difference is expected to be very close to zero.The differences have almost a zero-mean and Laplacian-like distribution.Laplacian data can be applied to data hiding schemes to improve their embedding ability. This observation leads toward designs in which the embedding is done in pixel differences.

The histogram modification technique does not work well when an image has an equal histogram. While multiple pairs of peak and minimum points can be used for embedding, the pure payload is still a little low. Moreover, the histogram modification technique carries with it an unsolved issue in that multiple pairs of peak and minimum points must be transmitted to the recipient via a side channel to ensure successful restoration. Thus a binary tree structure is presented in the following subsection that deals with communication of multiple peak points.

Figure below shows an auxiliary binary tree for solving the issue of communication of multiple peak points. Each element denotes a peak point. Assume that the number of peak points used to embed messages is 2^L , where L is the level of the binary tree. Once a pixel difference d_i that satisfies $d_i < 2^L$ is encountered, if the message bit to be embedded is 0, the left child of the node d_i is visited. Otherwise, the right child of the node d_i is visited. Higher payloads require the use of higher tree levels, thus quickly increasing the distortion in the image beyond acceptable levels. However, all the recipient needs to share with the sender is the tree level L, because an auxiliary binary tree is proposed that predetermines multiple peak points used to embed messages.

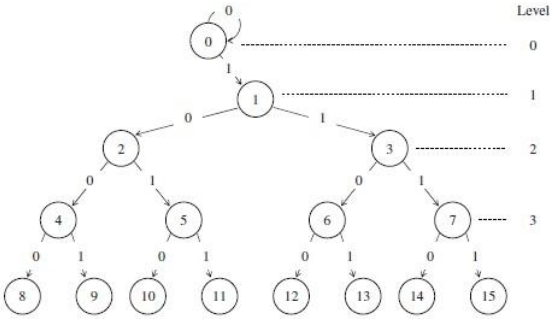


Fig. 3. Auxiliary binary tree

Modification of a pixel may not be allowed if the pixel is saturated (0 or 255). To prevent overflow and underflow, a histogram shifting technique that narrows the histogram from both sides, as shown in figure below is adopted. Assume that the number of peak points used to embed messages is 2^L ,where L is the level of the proposed binary tree structure. Thus, we shift the histogram from both sides by 2^L units to prevent overflow and underflow since the pixel x_i that satisfies $d_i \geq 2^L$ will shift by 2^L units after embedding takes place.

After narrowing the histogram to the range 2^L, 255-2^L, record the histogram shifting information as overhead bookkeeping information. For this purpose, create a one bit map as the location map, which is equal in size to the host image. For a pixel having grayscale value in the range 2^L, 255

-2^L ,assign a value 0 in the location map.Otherwise,assign a value 1.The location map is losslessly compressed by the run-length coding algorithm, which will yield a large increase in compression ability since pixels out of the range 2^L, 255 -2^L are few and are almost always contiguous. The overhead information will be embedded into the host image together with the embedded message. The maximum modification to a pixel is limited to 2^Laccording to the proposed tree structure. As a result, shifting the histogram from both sides by 2^L units enables us to avoid the occurrence of overflow and underflow.

1) Embedding Process: The embedding process involves several steps. For an N-pixel 8-bit grayscale host image H with a pixel value x_i, where x_i denotes the grayscale value of the i^{th} pixel , $0 \le i \le N-1$, $x_i \in Z$, $x_i \in 0, 255$.

1. Read the host image. Determine the level L of the binary tree.

2. Shift the histogram of the host image from both sides by 2^L units. The histogram shifting information is recorded as overhead bookkeeping information that will be embedded into the image itself with payload.

3. Scan the image host image in an inverse s-order. Calculate the pixel difference d_i between pixels x_i- 1 and x_i.
4. Create location map using difference image same size as that of difference image(which is same size as that of host image).

$$\text{Location map} = \begin{cases} 1 & \text{if } d_i < 2^L \text{ or } d_i > 255 - 2^L \\ 0 & \text{Otherwise} \end{cases}$$

5. Compress the location map using run length encoding.
6. Convert the compressed location map to binary.
7. Read the message to be hidden and convert to binary.
8. Combine the message and location map in binary form.
9. Embed the combination of message and location map into histogram shifted image using pixel difference image as follows. Scan the whole image in the same inverse s-order. If $d_i \ge 2^L$, shift x_i by 2^L units

$$y_i = \begin{cases} x_i \text{ if } i = 0 \\ x_i + 2^L \text{ if } d_i \ge 2^L \text{ and } x_i \ge x_{i-1} \\ x_i - 2^L \text{ if } d_i \ge 2^L \text{ and } x_i < x_{i-1} \end{cases}$$

where y_i is the watermarked value of pixel i.

10. If $d_i < 2^L$, modify x_i according to the message bit

$$y_i = \begin{cases} x_i + (d_i + b) & \text{if } x_i \ge x_{i-1} \\ x_i - (d_i + b) & \text{if } x_i < x_{i-1} \end{cases}$$

where b is a message bit to be embedded and b ϵ {0, 1}.

2) Extraction Process : This process extracts both overhead information and payload from the watermarked image and losslessly recovers the host image. Let L be the level of the proposed binary tree. For an N-pixel 8-bit watermarked image W with a pixel value y_i, where y_i denotes the grayscale value of the i^{th} pixel,$0 \le i \le N-1$, $y_i \in Z$,$y_i \in 0, 255$.

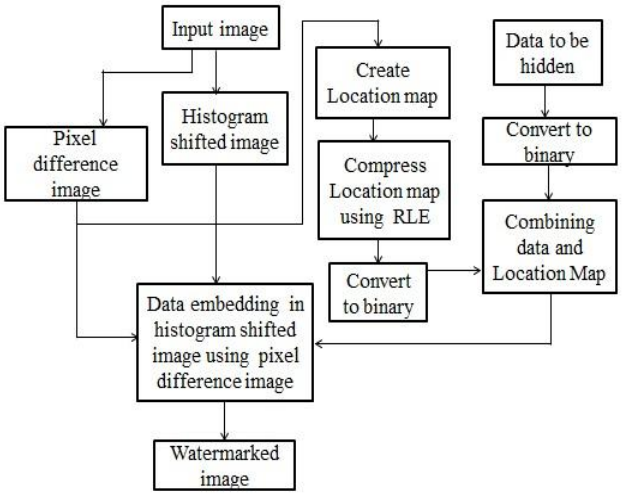


Fig. 4. Flow diagram for data embedding

1. Scan the watermarked image W in an inverse s-order.
2. If $|y_i - x_{i-1}| < 2^{L+1}$, extract message bit b by

$$b = \begin{cases} 0 \text{ if } |y_i - x_{i-1}| \text{ is even} \\ 1 \text{ if } |y_i - x_{i-1}| \text{ is odd} \end{cases}$$

Where x_{i-1} denotes the restored value of y_{i-1}

3. Restore the original value of host pixel x_i by

$$x_i = \begin{cases} y_i + \left\lceil \frac{|y_i - x_{i-1}|}{2} \right\rceil & \text{if } |y_i - x_{i-1}| < 2^{L+1} \text{ and } y_i < x_{i-1} \\ y_i - \left\lceil \frac{|y_i - x_{i-1}|}{2} \right\rceil & \text{if } |y_i - x_{i-1}| < 2^{L+1} \text{ and } y_i > x_{i-1} \\ y_i + 2^L & \text{if } |y_i - x_{i-1}| \ge 2^{L+1} \text{ and } y_i < x_{i-1} \\ y_i - 2^L & \text{if } |y_i - x_{i-1}| \ge 2^{L+1} \text{ and } y_i > x_{i-1} \\ y_i & \text{Otherwise} \end{cases}$$

4. Repeat Step 2 until the embedded message is completely extracted.
5. Extract the overhead information from the extracted message. If a value 1 is assigned in the location i, restore x_i to its original state by shifting it by 2^Lunits. Otherwise, no shifting is required

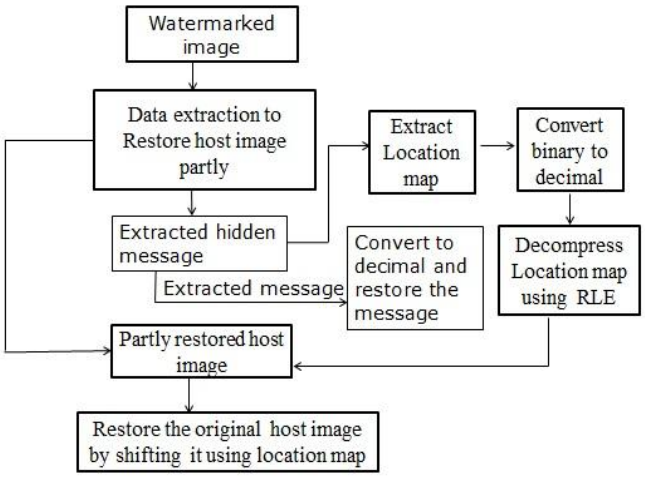


Fig.5. Flow diagram for data extraction

C. *Encryption and Decryption*

The advent of internet has paved the way for transmitting more and more information over the internet in a very less time. Audio, images other multimedia are being transmitted over the internet. The traditional encryption schemes appear not to be ideal for image applications, due to some intrinsic features of images such as bulk data capacity and high redundancy, and high correlation among pixels, and usually are huge in size. Along with it sometimes image applications also have their own requirements like real-time processing, image format consistence, and data compression for transmission. Simultaneous fulfilments of these requirements, along with high security and high quality demands, have presented great challenges to real-time imaging practice. The traditional encryption schemes require extra operations on image data, thereby demanding long computational time and high computing power.

Chaos-based image encryption can provide a class of very promising methods that can fulfil many of these requirements and can prove its superiority over the conventional encryption methods, particularly with a good combination of speed, security, and flexibility. Chaotic systems has proved its significance and its vitality for secure communication and information encryption like non-periodicity, randomness, turbulence, good statistic characteristic , easy ,regeneration and wondrous sensitivity for initial values.

The encryption - decryption process used in this proposed method combines the benefits of two chaotic systems namely discrete chaotic encryption and logistic map encryption. First the properties of both the above said systems are discussed and finally the benefits of the encryption scheme used in the proposed method by combining the discrete chaotic encryption and logistic map encryption are discussed.

1)Encryption - Decryption Process in Proposed Method:
The encryption- decryption process combines the benefits provided by both the discrete chaotic encryption and logistic map encryption. The first system based on Chebyshev chaotic sequence, is relatively simple and hence the time taken for the encryption process is very less. The proposed encryption system has two encryption stages. The first encryption stage uses a chaotic system that is based on the Logistic Map. The second encryption stage uses a discrete chaotic encryption.

a)Encryption Algorithm: The key is denoted as k = (k_1, k_2),where k_1 = (μ, x_1) and k_2 = (x_2, y_2). The parameter μ is selected such that 3.569< μ < 4.0 and $x_1 \in (0, 1)$. The initial Conditions x_2 and y_2 for the chaotic system of the second stage lie in [-1, 1].Consider the plain image to be represented by A of size M × N, and A (i, j) stands for an individual pixel in the image. The encrypting process consists of following steps

1. Generate a chaotic sequence using the sub-key k_1 as the initial conditions of the first chaotic system using the equation
f (x) = μx(1 - x)
2. Transform the chaotic sequence into a binary stream by a threshold function.
3. Modify pixel values of the plain image A (i, j) using the binary stream as a key stream and get the image A_1(i, j). The operation is bit-wise XOR.
4. Generate two sufficiently long Chebyshev chaotic sequence using Equation
$$x_{k+1} = 16x_k^5 - 20x_k^3 + 5x_k$$
5. And x_2 and y_2 as the initial conditions. The lengths of the two sequences shouldbe much larger than M and N respectively.
6. Arbitrarily intercept two sequences of length M and N from the above two sequences respectively i.e., $\{x_1, x_2, \ldots x_M\}$ and $\{y_1, y_2, \ldots, y_N\}$
7. Rearrange the two sequences obtained in step 5 either in ascending or descending order to get two new discrete chaotic sequences of length M and N respectively i.e. $\{x_1', x_2', \ldots, x_M'\}$and $\{y_1', y_2', \ldots, y_N'\}$
8. Decide the position of each $x_i \in \{x_1, x_2, \ldots x_M\}$in sequence $\{x_1', x_2', \ldots, x_M'\}$and generate replacement address set s_1= $\{a_1, a_2, \ldots, a_M\}$
9. Similarly decide the position of each $y_i \in \{y_1, y_2, .., y_N\}$ in the sequence $\{y_1', y_2', \ldots, y_N'\}$ to generate replacement address set s_2 = $\{b_1, b_2, \ldots, b_N\}$
10. The address set s_1 is used in row scrambling of the pixels in the image encrypted in stage I. Similarly, the second sequence s_2 is used to scramble the columns of the image encrypted in stage I.

The encryption scheme based on discrete chaotic encryption is the fastest encryption scheme. The encryption scheme based on logistic Map has higher decorrelating ability. The proposed scheme is thus able to successfully combine the benefits of the two chaos based encryption schemes namely faster execution time of the first technique and higher de-correlating ability of the second technique. These kind of chaos-based image encryption schemes are very practical with respect to operational speed, computational cost, and implementation simplicity. Chaos-based digital image encryption technology is very promising for real-time secure image.

The process for generating the binary stream is as follows:

- Generate a chaotic sequence using the sub-key k_1 as the initial conditions of the first chaotic system.

- Generate a binary stream from the above chaotic system x_i by using a threshold function F. The threshold function F is as given below:

$$F(x) = \begin{cases} 00000000 & \text{if } 0 \le x < \frac{1}{2^8} \\ 00000001 & \text{if } \frac{1}{2^8} \le x < \frac{2}{2^8} \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ 11111111 & \text{if } \frac{255}{2^8} \le x < 1 \end{cases}$$

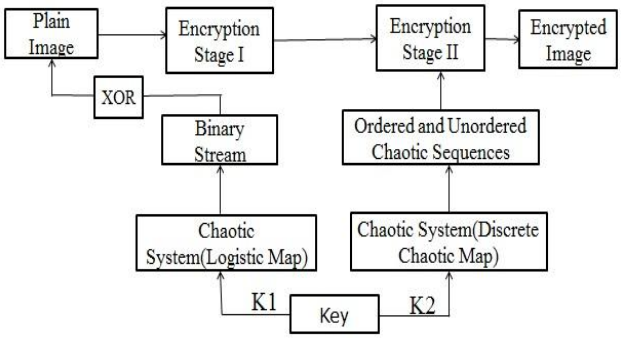


Fig.6. Encryption process with two stages

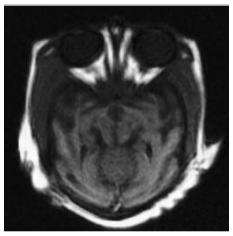
III. EXPERIMENTAL RESULTS

The method is applied on three images and its performance is shown in table 1 as follows.

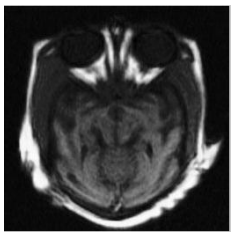
TABLE I. PERFORMANCE ANALYSIS

Image	L value	MSE	PSNR
Lena (512×512)	0	0	infinity
	1	0	infinity
	2	0	infinity
Baboon (298×298)	0	0	infinity
	1	0	infinity
	2	0	infinity
Brain (256×256)	0	0	infinity
	1	0.0041	71.94
	2	0.0898	58.59

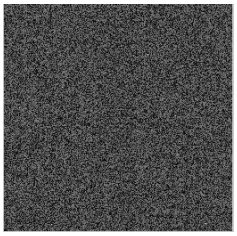
The output images obtained upon the application of proposed method on brain image for L value equals 0 is as follows.



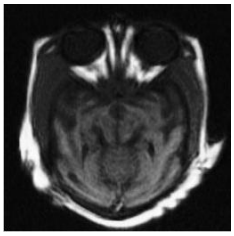
(a)



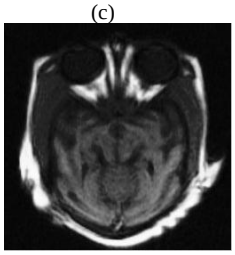
(b)



(c)



(d)



(e)

Fig.7. a) Input image b) Embedded image c) Encrypted image d) Decrypted image e) Output image

IV. CONCLUSION

The proposed method presents a reversible data hiding method accompanied with an encryption method so as to ensure the security of the host image which is in high demand as well as the security of the message or data hidden in the host image respectively. For reversible data hiding, an efficient extension of the histogram modification technique by considering the differences between adjacent pixels instead of simple pixel value is used. A binary tree structure is used to solve the problem of communicating pairs of peak points. Distribution of pixel differences is used to achieve large hiding capacity while keeping the distortion low. A histogram shifting technique is used to prevent overflow and underflow. The method ensures reversibility by showing higher rate of PSNR values. The encryption method in this, ensures reversibility of the host image in addition to security.

REFERENCES

[1] C.W. Honsinger, P.W. Jones, M. Rabbani, and J.C. Sto_el ,"Lossless recovery of an original image containing embedded data", US Pat.6,278,791, Aug 2001.

[2] Yasaman Zandi Mehran1, Mona Nafari,Alireza Nafari, and Nazanin Zandi Mehran, Histogram Shifting as a Data Hiding Technique:An Overview of Recent Developments,2009.

[3]] Z. Ni, Y.Q. Shi, N. Ansari, and W. Su, Reversible Data Hiding, In Proc. Of International Symposium on Circuits and Systems, Bangkok,Thailand, Vol.2, pp. 912-915, 25-28 May 2003

[4] W. L. Tai, C. M. Yeh, and C. C. Chang, Reversible data hiding based onhistogram modi_cation of pixel differencesIEEE Trans. Circuits Syst. VideoTechnol., vol. 19, no. 6, pp. 906910, Jun. 2009

[5] Hsiang-Cheh Huang , Wai-Chi Fang , Integrity Preservation and Privacy Protection for Medical Images with Histogram-Based Reversible Data Hiding,2011 IEEE/NIH Life Science Systems and Applications Workshop (LiSSA)

[6] Rajiv Ranjan ,Arup Kumar Pal , "Encryption of Image Using ChaoticMap," International Conference on Recent Trends in Engineering Technology (ICRTET 2012)

Reswin M R,Mtech Student,College of Engineering ,Chengannur
Gopakumar G, Mtech,Associate Professor, College of Engineering, Chengannur