ISSN: 2321-7782 (Online)
Volume 2, Issue 4, April 2014
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Paper / Case Study
Available online at: www.ijarcsms.com

Privacy Protection of Medical Data using Histogram Shifting based Reversible Data Hiding

Sumija.C[1]
PG Student, M.E (Embedded Systems)
CMS College of Engineering
Namakkal, Tamilnadu
India

A.Thillipan[2]
PG Student, M.E (Embedded Systems)
CMS College of Engineering
Namakkal, Tamilnadu
India

Manasy Mariet Thomas[3]
PG Student, M.E (Embedded Systems)
CMS College of Engineering
Namakkal, Tamilnadu
India

S.Rajesh[4]
Asst. Professor, Department of ECE
CMS College of Engineering
Namakkal, Tamilnadu
India

Abstract: ***The project presents privacy protection of medical images with information using histogram shifting based reversible data hiding. An Embedding module involves image encoding and histogram shifting based difference expansion. First of all, the patients' privacies need to be preserved. Therefore, embedding secret data into the medical images would be one of the useful methods for protecting the privacies. The secret data is first compressed using Lzw compression technique .so that more data can be embedded in a single image. Lzw is a "dictionary"-based compression algorithm. Lzw encodes data by referencing a dictionary. Thus, to encode a substring, only a single code number, corresponding to that substring's index in the dictionary, needs to be written to the output file.Next, because external data are hidden into the original image, some alterations are supposed to be induced. For securing data inside the image, encryption technique is used. The algorithm used for encryption is chaos. After data embedding, the output image should be as similar as its original counterpart, and medical doctors may lead to proper treatment by using the images with hidden data when necessary. Reversible data hiding is a newly developed branch in data hiding or watermarking researches. Reversibility means that data, including patients' private information and the diagnosis data, can be hidden into the medical image by some means developed by ourselves. Later on, the medical image containing data might be retrieved by medical doctors while necessary, and both the original image and the hidden data can be perfectly recovered with the algorithm corresponding to the embedding scheme. Finally the performance of an algorithm will be evaluated by mean square error, peak signal to noise ratio, entropy and correlation coefficient.***

I. INTRODUCTION

REVERSIBLE DATA HIDING (RDH)

RDH algorithm usually depends on some parameters (e.g., the integer *a* defined) and these parameters should be communicated to decoder. To this end, we may slightly modify the embedding and extraction procedures. Firstly, we divide host image into two parts to get *I*0 and *I* *I*0 contains a fixed number of pixels and *I* is the rest pixels.

Secondly, express the parameters in binary form to get a binary sequence and replace the LSBs of *I*0 by this sequence. Here, the original LSBs of *I*0 should be recorded and then embedded into host image as a part of hidden data. For example, in the modified version of Ni *et al.*'s method, we may reserve 8 pixels (i.e., _ *I*0 = 8) to embed the value of *a*. Finally, consider *I* as an "image" and embed data into it.

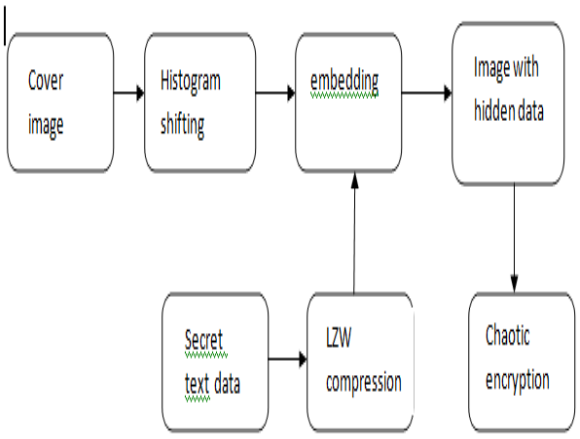
For decoder, it only needs to read LSBs of *I*0 to get the parameters, and then extract the hidden data from I. In particular, *I*0 can be restored by overwriting its LSBs by a certain part of extracted hidden data.

ADVANTAGES OF REVERSIBLE DATA HIDING

There are also a number of works on data hiding in the encrypted domain. The reversible data hiding in encrypted image is investigated in. The work on reversible data hiding focuses on the data embedding or extracting on the plain spatial domain.

This method by reserving room before encryption with a traditional RDH algorithm of it is easy for the data hiding to reversibly embed data in the encrypted image. The proposed method can achieve real reversibility; the data extraction and image recovery are free of any error. Thus the data hider can benefit from the extra space Emptied out in previous stage to make data hiding process effortless.

BLOCK DIAGRAM OF EMBEDDING MOULE



COVER IMAGE

In imaging science, image processing is any form of signal processing for which the input is an image, such as a photograph or video frame.

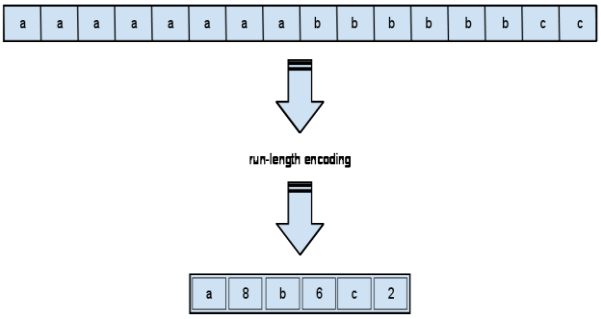
II. HISTOGRAM SHIFTING

HS-based algorithm is another important work of RDH, in which the peak of image histogram is utilized to embed data. In this method, each pixel value is modified at most by 1, and thus the visual quality of marked image is guaranteed. In Lee *et al.*'s proposed a method by using the histogram of difference image. This method outperforms Ni *et al.*'s by improving both EC and visual quality. The spatial correlation of natural images is exploited in Lee *et al.*'s method and thus a more appropriate histogram is obtained. In other words, compared with the ordinary one dimensional histogram, the difference-histogram is better for RDH since it is regular in shape and has a much higher peak point. In, Hong *et al.* proposed a new HS-based method by modifying the prediction-error histogram.

This method can well exploit the image redundancy and thus achieve a better performance compared with the previously introduced DE-based methods such as Recently, Wu and Huang proposed a novel HS-based method where the histogram bins used for expansion embedding are specifically selected such that the embedding distortion is minimized. The experimental results reported in demonstrated that this method is better than some state-of-the-art works including the most recently proposed integer-transform-based method.

III. ENCODING PROCESS

Here we use Runlength encoding for image compression.It is a very simple compression method used for sequential data. It is very useful in repetitive data. This technique replaces sequences of identical pixels, called runs by shorter symbols. The run length code for a gray scale image is represented by a sequence {Vi, Ri} where Vi is the intensity of pixel and Ri refers to the number of consecutive pixels with the intensity Vi as shown in the figure. If both Vi and Ri are represented by one byte, this span of 12 pixels is coded using eight bytes yielding a compression ration of 1:5.



IV. EMBEDDING PROCESS

The embedding procedure contains several steps. First, after dividing the host image into non-overlapping blocks, then the blocks are further divided into three parts to get $I1$, $I2$ and $I3$. Then, by using shifting and embedding functions, embed the hidden data into $I1$ and $I3$. Next, by using LSB replacement, embed the location map which records the underflow/overflow locations into $I1$. Notice that, before replacing LSBs, the original LSBs of $I1$ should be recorded into a LSB sequence. Finally, embed the LSB sequence into $I2$ using shifting and embedding functions. Here, the partition of three parts is to solve the underflow/ overflow problem by embedding the location map into the host image.

ADVANTAGES OF EMBEDDING

A lossless data hiding method based on histogram shifting embeds large-volume data into cover images. It also produces stego-images with high qualities by using a strategy of hierarchical block division. The bottleneck of data-hiding-rate increasing at the block size of 8×8 found in existing methods is broken by the proposed non-recursive algorithms.

The proposed recursive versions of the algorithms enhance the performance further both in the data hiding capacity and the PSNR value, which result from the proposed scheme of recursive looking-ahead estimation of the data hiding capacity.

V. LZW COMPRESSION

Here we use LZW compression for compressing message. LZW is a data compression method that takes advantage of this repetition. The original version of the method was created by Lempel and Ziv in 1978 (LZ78) and was further refined by Welch in 1984, hence the LZW acronym. Like any adaptive/dynamic compression method, the idea is to :

(1) Start with an initial model

(2) Read data piece by piece

(3) And update the model and encode the data as you go along.

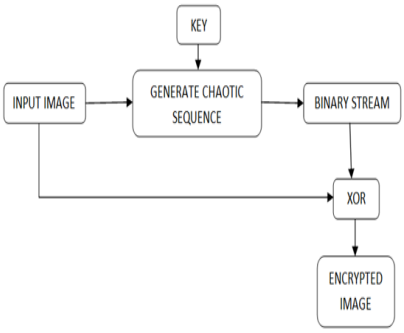
LZW is a "dictionary"-based compression algorithm. This means that instead of tabulating character counts and building trees (as for Huffman encoding), LZW encodes data by referencing a dictionary. Thus, to encode a substring, only a single code number, corresponding to that substring's index in the dictionary, needs to be written to the output file. Although LZW is often explained in the context of compressing text files, it can be used on any type of file. However, it generally performs best on files with repeated substrings, such as text files.

VI. CHAOTIC ENCRYPTION

The simplest way to encrypt an image or a video is perhaps to consider the 2-D and 3-D stream as a 1-D data stream, and then encrypt this 1-D stream with any available key , such a simple idea of encryption is called naive encryption. Although naive encryption is sufficient to protect digital images and videos in some civil applications, this issues have taken into consideration when advanced encryption algorithms are specially designed for sensitive digital images and videos, for their special features are very different from texts.

The recent research activities in the field of nonlinear dynamics and especially on systems with complex (chaotic) behaviours have forced many investigations on possible applications of such systems. Today, chaotic encryption is almost exclusively considered inside the nonlinear systems community.

Block diagram of image encryption using chaotic map



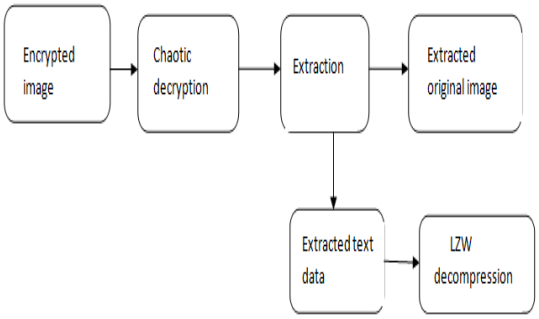
Algorithm

I. Chaotic sequence is generated using the sub-key k1 as the initial conditions of the first chaotic system.

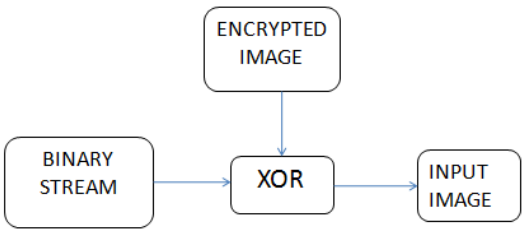
II. The chaotic sequence is transformed into a binary stream by a threshold function

III. Pixel values of the plain image A (i, j) is modified using the binary stream as a key stream and image A' (i, j) is created by bitwise-XOR operation

EXTRACTION MODULE



CHAOTIC DECRYPTION



The encrypted image is Xored with binary stream to get input image.

VII. DATA EXTRACTION

The data extraction procedure also contains several steps. First, the same as the data embedding, divide the marked image blocks into three parts to get $I1$, $I2$ and $I3$. Then, determine the location map by reading LSBs of $I1$. Next, according to the location map and by using shifting and embedding functions, determine the LSB sequence by extracting data from $I2$, and then replace the LSBs of $I1$ by the extracted LSB sequence. Finally, extract the embedded data from $I1$ and $I3$. Notice that, using shifting and embedding functions, the image restoration can be realized simultaneously with the data extraction.

Extraction can be done as a reversible process such that, from the technique, we can extract the hidden data & the image which is needed as a cover image, separately. Integer Wavelet Transform is done as a reversible process such that the transformed image can be retrieved back.

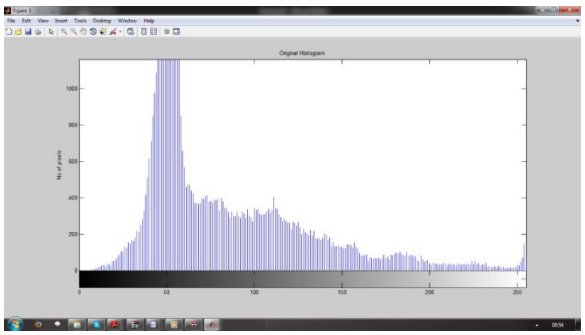
VIII. RESULT AND ANALYSIS

Our purpose is to hide medical data inside an image in order to protect data.

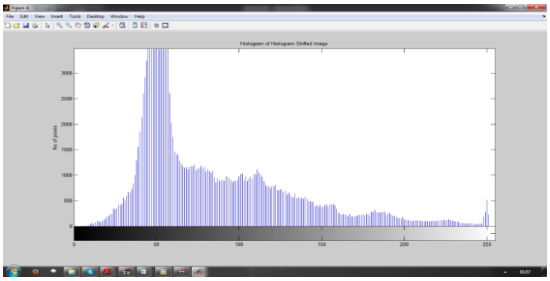
- First input image is taken.

- In the below figure RGB Colour image is converted to gray scale image

Below figure shows Histogram of input image

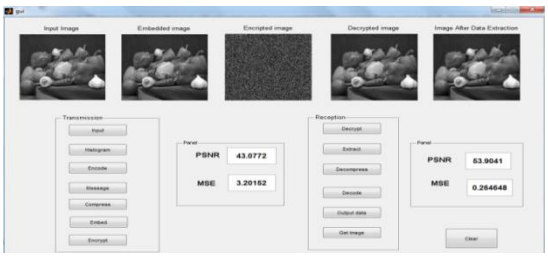


Below figure shows histogram shifted image



Above figure shows Image after histogram shifting.

- Next figure shows overall process in data hiding



- First input image is taken.
- Then histogram shifting, Run length encoding is performed to compress size of image
- Next compressed data is embedded to the histogram shifted encoded image.
- After that encryption is performed for security purpose.
- Decryption is performed to get original image.
- Next comes extraction process, data is extracted from the image.
- Decompression of data is done.
- Then decoding is performed, it is the reverse process of encoding..
- After decoding process, data is completely extracted from image.

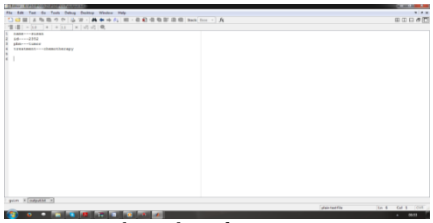


Figure shows data after extraction process

IX. CONCLUSION

This novel method can achieve real reversibility and the separate data extraction with greatly improvement on the quality of marked decrypted Images. After encrypting the entire data of an uncompressed image by a stream cipher system with the additional data can be embedded with the image by modifying a small proportion of the encrypted image.

References

1. G. Coatrieux, C. L. Guillou, J. M. Cauvin, and C. Roux, "Reversible watermarking for knowledge digest embedding and reliability control in medical images," IEEE Trans. Inf. Technol. Biomed., vol. 13, no. 2,pp. 158–165, Mar. 2009.
2. S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," IEEE Trans. Inf. Forens. Security, vol. 2, no. 3, pp. 321–330, Sep. 2007.
3. R. Li, O. C. Au, C. K. M. Yuk, S. Yip, and T. Chan, "Enhanced image trans-coding using reversible data hiding," in Proc. IEEE Int. Symp. Circuits Syst., May 2007, pp. 1273–1276.
4. R. Caldelli, F. Filippini, and R. Becarelli, "Reversible watermarking techniques: An overview and a classification," Eur. Assoc. Signal Process. J. Inf. Security, vol. 2010, no. 2, pp. 1–19, 2010.
5. J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding-new paradigm in digital watermarking," Eur. Assoc. Signal Process. J. Appl.Signal Process., vol. 2002, no. 2, pp. 185–196, Feb. 2002.
6. T. Kalker and F. M. J. Willems, "Capacity bounds and constructions for reversible data hiding," Security Watermarking Multimedia Contents V,vol. 5020, pp. 604–611, Jun. 2003.

