Traditional Reversible Data Hiding based Secret Communication

Sari R
PG Scholar
Marian Engineering College Trivandrum, India

Monisha Menon
Assistant professor
Marian Engineering College Trivandrum, India

Abstract

Reversible data embedding technique embeds secret message/data into a digital cover image in a reversible fashion. With its interesting feature ‘reversibility’ quality of recovered image after data extraction can maintain good enough. Existing reversible data embedding techniques may lead to some errors during data and cover image extraction process. This paper proposes a new method for reversible embedding by adapting reserving room before encryption scheme. Here histogram shifting method is used for self-reversible embedding and LSB replacement method for data embedding. Also reversible data hiding along with encryption technique provide multilevel security for hidden communication. Moreover the proposed algorithm can take advantages of some existing reversible data hiding techniques and worked with both grayscale and color images.

Keywords: Data embedding; Image encryption; Image recovery; Data extraction; Reversible data hiding (RDH)

I. INTRODUCTION

The transmission of multimedia data through internet is increased day by day due to the technological advancement. Safety of the content during transmission and personal privacy is a major issue in all applications. There exist two technologies to overcome this difficulty such as steganography and cryptography. Steganography is a form of data hiding intended for secret communication.

Data hiding is a process of hiding a message within another message, may be image audio or video file usually called cover. Among these, digital images are more popular because of its availability and ease of usage. Also depending on the size of image files, it can hide more information.

Usually data embedding process make changes in the cover image. During data extraction process, one can recover original cover reversibly, and then it is termed as Reversible Data hiding (RDH). Otherwise it is called irreversible data hiding. Some applications like privileged imagery need distortion free recovery of cover image; RDH is the only possible remedy for this case.

A. Reversible Data Hiding

Reversible Data Hiding, simply ‘RDH’, method which is also called distortion free embedding, embeds secret message into a digital cover image in a reversible fashion. The foremost requirement of RDH is quality of degradation of image after data embedding should be low. An important feature of reversible data embedding is its ‘reversibility’, means one can extract embedded message in order to recover original image losslessly.

The efficiency of a reversible data embedding algorithm can be calculated by the following

1) Payload capacity
2) Visual quality
3) Algorithm complexity

Though undetectable, embedding process will change original image content. Even a small alteration of pixel values may not be allowable in privileged imagery like medical and military imagery. In such cases every bit of information is valuable and any change will affect truthfulness of the image. With the reversibility feature RDH became a solution for this. Also the distinction between the embedded cover image and original cover image is almost undetectable from the human eyes, reversible embedding could be viewed as a Secret communication channel.

The safe keeping of multimedia information can be done with encryption algorithms. By joining both encryption and data hiding in a single step, it is possible to achieve maximal security and privacy of secret communication. The two main groups of technologies have been developed for this purpose include

– First one is based on content protection through encryption.
– Second one is based on protection on embedded data.

Nowadays there exist challenges to embed data in encrypted images. All previous works related to this adapted irreversible approach of data hiding. So a new idea can put forward to recover both the image as well as data reversibly and separable manner, which is explained by this paper.
II. RELATED WORKS

There are many reversible data hiding techniques have been come out recent years. In practical aspect, Fridrich et al. [1] developed a general method for RDH. By taking compressible features of cover image and compressing them losslessly, needed space can be saved for embedding secret data. Another method is based on difference expansion (DE) [2], in which the difference of each pixel pair is expanded, for embedding messages. One advanced technique for RDH is histogram shifting (HS) [3], in which space is created for data embedding by shifting the peaks of histogram of images. While applying RDH in encrypted images, which results in some promising applications. For instance, in [4], Hwang et al. advocated a reputation-based trust-management scheme enhanced with data coloring and software watermarking, in which data encryption and data hiding give chances for upholding the data owner’s privacy and data integrity.

In [5], Zhang dissenting the encrypted image into several blocks. By changing the 3 LSBs of the half of pixels in each block, needed space can be vacated for the to be embedded bit. The extraction process can be done by finding which part has been changed in one block. This can be done with the help of spatial correlation in decrypted image.

All the methods [2],[3] try to create space from the encrypted images directly. Since entropy of the encrypted image has been maximized, the existing techniques can achieve only small payloads or poor quality for larger payloads and also subject to error on data extraction or image recovery.

This paper proposes a noval method for Reversible data hiding by utilizing advantages of [3],[7] and using the concept Reserve Room before Encryption [6]. Also this algorithm is suited for both gray and color domain.

III. PROPOSED METHOD

The proposed algorithm used two branches of information technology, cryptography, and steganography. Cryptography for encrypting the image and steganography for the data hiding in encrypted image. Fig.1 shows the outline of the system.

The overall system is basically divided into two different modules sender and the receiver. Each module further divided into different phases. Transmitter side includes reserving room in image, encryption of image and data hiding. Here at the receiver side one can consider three different cases to achieve separability during extraction process as shown in fig1.

In the proposed system, the content owner first creates space or losslessly compresses the redundant information in the cover image and then encrypts it with an encryption key. Now the data hider embeds secret data into encrypted image without knowing the original image content. At the receiver side both the data and the image recovery can be done in separable manner.

Fig. 1: Architecture of the proposed system
While using color images as the cover medium, more space can be reserved for data embedding. The implementation steps are described as following.

**A. Reserve Space in Cover Image**

This step include two parts

1) Image partition
2) Self-reversible embedding

1) Image Partition

Goal of image partition is to construct a smoother area using a smoothening function, on which standard RDH algorithms [7] can achieve better performance. The original image can be a 8 bit grayscale or color image with size is M X N and pixels $C_{i,j}$ belongs to $[0,255]$, $1 \leq i \leq M, 1 \leq j \leq N$.

Initially, from the original image, the content owner extracts several overlapping blocks along the rows, whose number is determined by the size of secret message have to be embed denoted by s. In detail, every block consists of $m$ rows, where $m = \frac{s}{N}$ , and the number of blocks can be computed through $n = M - m + 1$.

An important point here is that each block is overlapped by previous and/or sub sequential blocks along the rows. For each block, define a function to measure its first-order smoothness (1).

$$ f = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \left| C_{i,j} - \frac{C_{i,j+1} + C_{i,j-1} + C_{i+1,j} + C_{i-1,j}}{4} \right| $$(1)

Higher $f$ relates to blocks with more complex textures to be A, and puts it to the front of the image concatenated by the rest part B with fewer textured areas.

2) Self-Reversible Embedding

The goal of self-reversible embedding is to embed the LSB-planes of A into B. For that first categorizes Partitioned image B into two sets: white pixels with its indices $i$ and $j$ satisfying $(i+j) \mod 2 = 0$ and black pixels whose indices meet $(i+j) \mod 2 = 1$. Then, each white pixel, $B_{i,j}$ is estimated by the interpolation value obtained with the four black pixels surrounding it as follows

$$ B_{i,j} = w_1 B_{i+1,j} + w_2 B_{i-1,j} + w_3 B_{i,j+1} + w_4 B_{i,j-1} $$

Then some data can be embedded into the estimating error sequence using ‘Histogram Shifting’ method. Also the same steps have to do for the black pixels and find $e_{i,j}$.

An important thing should be considered here is that in RRBE framework we first empty out room by reversibly embedding LSBs of partitioned image A into B with a traditional RDH method and applied encryption algorithm. So the positions of LSBs in the encrypted image can be used to embed secret data.

Also the parameters regarding LSB positions, numbers of LSB planes utilized etc must be informed to the data hider by appending them into LSBs of A during reversible embedding. It is effortless for the data hider to read this information before data hiding process.

**B. Image Encryption**

We can create encrypted image by performing the encryption on rearranged self-embedded image, denoted by X. Encryption of X can easily obtain using a stream cipher.

For example, a gray value encrypted pixels. $X_{i,j}$ ranging from 0 to 255 can be represented by 8 bits, $X_{i,j}(0), X_{i,j}(1), \ldots, X_{i,j}(7)$, such that

$$ X_{i,j}(k) = \left[ \frac{X_{i,j}}{2^k} \right] \mod 2; k = 0, 1 \ldots, 7 $$

Exclusive- or operation can be used for obtaining encrypted bits

$$ E_{i,j}(k) = X_{i,j}(k) \oplus r_{i,j}(k) $$

Where $r_{i,j}(k)$ is generated by a standard stream cipher determined by the encryption key.

**C. Data Hiding**

Data hider will not be provided with the original image. He can embed data to the encrypted image only. The embedding process can start with encrypted version of A. It is effortless for the data hider to read information in LSBs of first 8encrypted pixels regarding how many bit-planes and rows of pixels he can modify. After knowing it, the data hider simply adopts LSB replacement to substitute the available bit-planes with additional data.
**D. Data Extraction and Image recovery**

At the receiver side, there are three possibilities in real time practice, when the receiver is supplied with both the encryption and data hiding key

1) Data Hiding Key alone
2) Encryption Key alone
3) Data hiding and encryption Keys

3) Data Hiding Key alone

When receiver is supplied with data hiding key alone and then only the extraction of concealed information alone is possible, but not the decryption of cover or any small fragment or region of cover.

For retrieving concealed secret message receiver one should follow the data hiding algorithm in the reverse order.

a) Encryption Key alone

When the receiver has only data hiding key and encrypted image containing hidden data, he must be able to generate an image similar to the original image, but should not be able to read the hidden data. The receiver can decrypt the encrypted image containing hidden data by using the encryption key only.

To form the marked decrypted image, the content owner should do following steps.

- Step 1: With the encryption key, the content owner decrypts the image except the LSB-planes of A. The decrypted version of A’ containing the embedded data can be calculated by

\[
X_{i,j}(k) = E_{i,j}(k) \oplus r_{i,j}(k) \quad \text{And} \quad X_{i,j} = \sum_{k=0}^{7} X_{i,j}(k) \times 2^k
\]

- Step 2: Also extract the parameters that are embedded in encoding stage in order to get final decrypted image.

Both Data hiding and Encryption key

When the receiver has both keys, he first extracts the hidden data using data hiding key. This will recover the original value of distorted pixels due to data hiding. Then, the receiver decrypts the image. Thus, the hidden data is extracted exactly and the original image is recovered completely.

**IV. Simulation results**

The proposed RDH technique coded and tested in Mat lab R2012b with both gray and color images, each of size 512*512. In the implementation, here used histogram shifting method and LSB replacement method for self-reversible embedding and for concealing secret message. Also for simplicity only one max-min pair of histogram for concealing secret message is adopted.

Here if the receiver is supplied with encryption key alone, then he can retrieve only the watermarked image. Also PSNR of the directly decrypted images were obtained above greater than 48.13dB which is higher than existing methods and verifies the theoretical result.

Whenever receiver is supplied with data hiding key alone, then hidden secret message can be extracted without any error. Moreover if he is supplied with both encryption key and data hiding key, then both the operations can be done in separable manner. The recovered image is exactly same as the original cover medium.

“Fig. 2” shows the transformations of the original image ‘Gray and Color’ while passing through different phases of algorithm.
V. PERFORMANCE EVALUATION

The peak signal-to-noise ratio (PSNR) is the objective criteria to find out the quality of the images after decryption. PSNR is the ratio between a signal's maximum power and the power of the signal's noise. Each picture element (pixel) may get changed when an image is modified.

It is observed that the grayscale value of pixels between MAX and MIN will either be incremented or decremented by 1 during data hiding. In the worst case, the value of every pixel differs by a value of 1 from their original value. Thus, the Mean Square Error (MSE) of worst case is 1 and the lower bound of PSNR of the decrypted image containing hidden data is given by,

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}}$$

$$= 10 \log_{10} \frac{255^2}{1}$$

$$= 48.1358$$

The occurring lower bound of PSNR is much higher than that of existing reversible data hiding in encrypted image techniques.

VI. CONCLUSION

Here presented an advanced reversible data hiding technique in encrypted images using reserving room before encryption concept. Since this algorithm supported both gray and color images data embedding capacity is much higher than available algorithms. At the receiver side, data extraction and image recovery are performed in separable manner.

ACKNOWLEDGEMENT

We would like to extend our gratitude to the Management and Principal and Head of the Department (ECE) of Marian College of Engineering and Technology for the support and help in successfully completing this work.

REFERENCES