A COVERLESS IMAGE STEGANOGRAPHY

S.UMA MAHESWARI[1], RAMYA MJ[2], PRIYADHARSHINI U[2], PRIYADHARSHINI KS[2], PAVITHRA E[2]

[1] *Professor, Dept. of Electronics and Communication Engineering, Panimalar Engineering College, Tamil Nadu, India*

[2] *Student, Dept. of Electronics and Communication Engineering, Panimalar Engineering College, Tamil Nadu, India*

---***---

Abstract - *With the digitalization of information, a lot of multimedia data are under attack, information security has become a key issue of public concern. Image steganography, aiming at using cover images to convey secret information has become one of the most challenge and important subjects in the field of information security recently. Different from the traditional image steganography, coverless image steganography does not need to employ the designated cover image for embedding the secret data but directly transfers secret information through its own properties such as pixel brightness value, color, texture, edge, contour and high-level semantics. Therefore, it radically resist the detection of steganalysis tools and significantly improves the security of the image. Its basic idea is to analyze the attributes of the image and map them to the secret information according to certain rules based on the characteristics of the attributes.*

A new information hiding technology named coverless information hiding is proposed. It uses original natural images as stego images to represent secret information. The focus of coverless image steganography method is how to represent image features and establish a map relationship between image feature and the secret information. In this paper, we use three kinds of features which are Local Binary Pattern (LBP), the mean value of pixels and the variance value of pixels. Our project used the concept of RESERVING ROOM BEFORE ENCRYPTION, to achieve greater security than VACATING ROOM AFTER ENCRYPTION.

***Key Words*: Steganography, Coverless image steganography, Information hiding, Information security, Steganalysis, Stego images.**

1. INTRODUCTION

Reversible data hiding/cryptography (RDH) in images/texts is a technique, by which the original cover can be losslessly recovered after the embedded message is extracted. This important technique is widely used in medical imagery, military imagery and law forensics, where no distortion of the original cover is allowed. A more popular method is based on difference expansion (DE), in which the difference of each pixel group is expanded, multiplied by 2, and thus the least significant bits (LSBs) of the difference are all-zero and can be used for embedding messages. Another promising strategy for RDH is histogram shift (HS), in which space is saved for data embedding by shifting the bins of histogram of grey values. The state-of-art methods usually combined DE or HS to residuals of the image, e.g., the predicted errors, to achieve better performance.

In this framework, a content owner encrypts the original image using a standard cipher with an encryption key. After producing the encrypted image, the content owner hands over it to a data hider (e.g., a database manager) and the data hider can embed some auxiliary data into the encrypted image by losslessly vacating some room according to a data hiding key. Then a receiver, maybe the content owner himself or an authorized third party can extract the embedded data with the data hiding key and further recover the original image from the encrypted version according to the encryption key.

In the present paper, we propose a novel method for RDH in encrypted images/texts, for which we do not “vacate room after encryption” as done, but “reserve room before encryption”. In the proposed method, we first empty out room by embedding LSBs of some pixels into other pixels with a traditional RDH method and then encrypt the image, so the positions of these LSBs in the encrypted image can be used to embed data. Not only does the proposed method separate data extraction from image decryption but also achieves excellent performance. Reserving room prior to image encryption at content owner side, the RDH tasks in encrypted images/texts would be more natural and much easier which leads us to the novel framework, “reserving room before encryption (RRBE)”

2. EXISTING SYSTEM

In this framework, a content owner encrypts the original image using a standard cipher with an encryption key. After producing the encrypted image, the content owner hands over it to a data hider (e.g., a database manager) and the data hider can embed some auxiliary data into the encrypted image by losslessly vacating some room according to a data hiding key. Then a receiver, maybe the content owner himself or an authorized third party can extract the embedded data with the data hiding key and further recover the original image from the encrypted version according to the encryption key.

3. PROPOSED SYSTEM

In the present paper, we propose a novel method for RDH in encrypted images/texts, for which we do not "vacate room after encryption" as done, but "reserve room before encryption". In the proposed method, we first empty out room by embedding LSBs of some pixels into other pixels with a traditional RDH method and then encrypt the image, so the positions of these LSBs in the encrypted image can be used to embed data. Not only does the proposed method separate data extraction from image decryption but also achieves excellent performance. Reserving room prior to image encryption at content owner side, the RDH tasks in encrypted images/texts would be more natural and much easier which leads us to the novel framework, "reserving room before encryption (RRBE)".

4. SYSTEM ARCHITECTURE

The content owner first reserves enough space on original image and then converts the image into its encrypted version with the encryption key. Now, the data embed ding process in encrypted images/texts is inherently reversible for the data hider only needs to accommodate data into the spare space previous emptied out. The data extraction and image recovery are identical to that of Framework VRAE. Obviously, standard RDH algorithms are the ideal operator for reserving room before encryption and can be easily applied to Framework RRBE to achieve better performance compared with techniques from Framework VRAE. This is because in this new framework, we follow the customary idea that first losslessly compresses the redundant image content (e.g., using excellent RDH techniques) and then encrypts it with respect to protecting privacy.

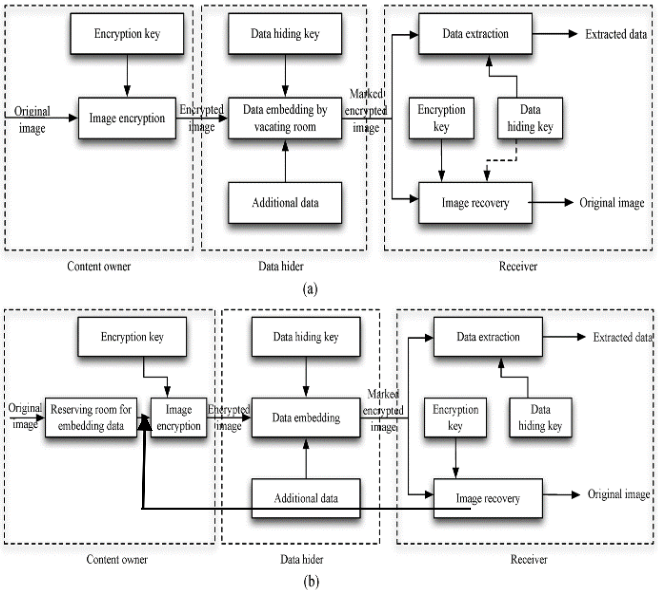


Fig -4.1: System Architecture

5. MODULE DESCRIPTION

Module 1: Vacating room after encryption – VRAE

In this framework, a content owner encrypts the original image using a standard cipher with an encryption key. After producing the encrypted image, the content owner hands over it to a data hider and the data hider can embed some auxiliary data into the encrypted image by losslessly vacating some room according to a data hiding key. Then a receiver, maybe the content owner himself or an authorized third party can extract the embedded data with the data hiding key and further recover the original image from the encrypted version according to the encryption key.

Module 2: Reserving room before encryption-RRBE

The content owner first reserves enough space on original image and then converts the image into its encrypted version with the encryption key. Now, the data embed ding process in encrypted images/texts is inherently reversible for the data hider only needs to accommodate data into the spare space previous emptied out. The data extraction and image recovery are identical to that of Framework VRAE. Obviously, standard RDH algorithms are the ideal operator for reserving room before encryption and can be easily applied to Framework RRBE to achieve better performance compared with techniques from Framework VRAE. This is because in this new framework, we follow the customary idea that first losslessly compresses the redundant image content (e.g., using excellent RDH techniques) and then encrypts it with respect to protecting privacy.

Module 3: Generation of Encrypted data

The operator here for reserving room before encryption is a standard RDH technique, so the goal of image partition is to construct a smoother area , on which standard RDH algorithms. The above discussion implicitly relies on the fact that only single LSB-plane A of is recorded. It is straightforward that the content owner can also embed two or more LSB-planes A of into B, which leads to half, or more than half, reduction in size of. However, the performance of, in terms of PSNR, after data embedding in the second stage decreases significantly with growing bit-planes exploited.

Module 4: Encryption

The same with other RDH algorithms, overflow/underflow problem occurs when natural boundary pixels change from 255 to 256 or from 0 to -1. To avoid it, we only embed data into estimating error with its corresponding pixel valued from 1 to 254. However, ambiguities still arise when non boundary pixels are changed from 1 to 0 or from 254 to 255 during the embedding process. These created boundary pixels in the embedding process are defined as pseudo-boundary pixels. Hence, a boundary map is introduced to tell whether boundary pixels in marked image are natural or pseudo in extracting process. It is a binary sequence with bit “0” for natural boundary pixel, bit “1” for pseudo-boundary pixel. Since estimating errors of marginal area of B cannot be calculated via (2), to make the best use of B we choose its marginal area shown in Fig. 2 to place the boundary map, and use B LSB replacement to embed it. The original LSBs of marginal area is assembled with messages, i.e., LSB-planes of, and reversibly embedded into. In most cases, even with a large embedding rate, the length of boundary map is very short; thus, the marginal area of B is enough to accommodate it.

Module 5: Self-Reversible Embedding

The goal of self-reversible embedding is to embed the LSB-planes of A into B by employing traditional RDH algorithms. For illustration, we simplify the method in to demonstrate the process of self-embedding. Note that this step does not rely on any specific RDH algorithm.

Module 6: Self-Irreversible Embedding

Extracting Data From Encrypted Images/texts:

To manage and update personal information of images/texts which are encrypted for protecting clients’ privacy, an inferior database manager may only get access to the data hiding key and have to manipulate data in encrypted domain. The order of data extraction before image decryption guarantees the feasibility of our work in this case.

Extracting Data From Decrypted Images/texts:

In Case 1, both embedding and extraction of the data are manipulated in encrypted domain. On the other hand, there is a different situation that the user wants to decrypt the image first and extracts the data from the decrypted image when it is needed. The following example is an application for such scenario.

6. HARDWARE AND SOFTWARE REQUIREMENTS

A. Hardware Requirements

- Processor :Core2Duo
- Hard Disk :80GB
- Memory : 1 GB
- CPU Rate : 2 GHz

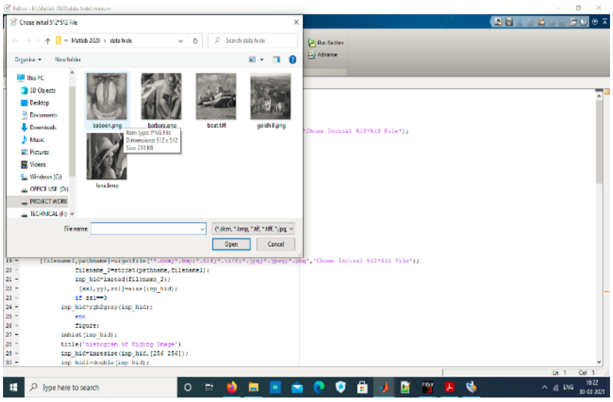
B.Software Requirements

- Operating System :WINDOWS XP
- Tool used : MATLAB
- Document Tool : Microsoft word
- Presentation Tool : Microsoft PowerPoint

C.Tool description

MATLAB is a high-level language and interactive environment for numerical computation, visualization, and programming. Using MATLAB, you can analyze data, develop algorithms, and create models and applications. The language, tools, and built-in math functions enable you to explore multiple approaches and reach a solution faster than with spreadsheets or traditional programming languages, such as C/C++ or Java.

7. RESULT



```
[filename1,pathname]=uigetfile({'*.dcm;*.bmp;*.tif;*.tiff;*.jpg;*.jpeg;*.png','Chose Initial 512*512 File'});
        filename_2=strcat(pathname,filename1);
        inp_hid=imread(filename_2);
         [xx1,yy1,zz1]=size(inp_hid);
        if zz1==3
    inp_hid=rgb2gray(inp_hid);
        end
        figure;
    imhist(inp_hid);
    title('histogram of Hiding Image')
    inp_hid=imresize(inp_hid,[256 256]);
    inp_hid1=double(inp_hid);
```

Fig -7.1: Selecting the original hiding image

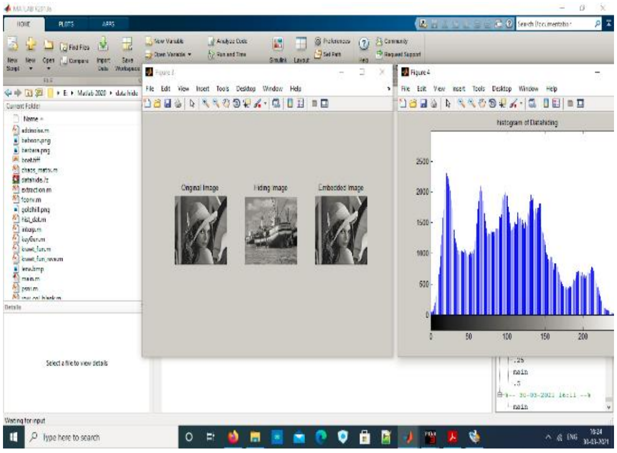


Fig -7.2: Histogram of data hiding

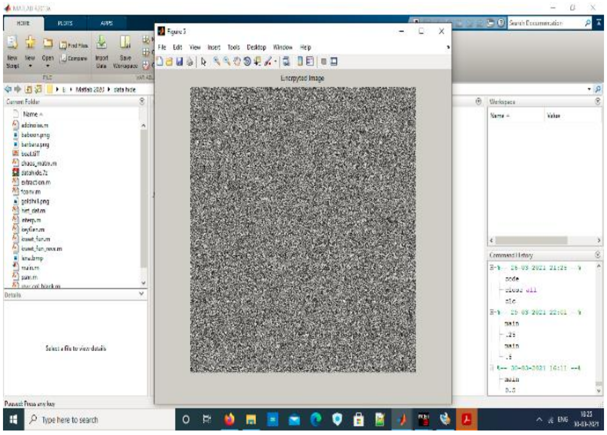


Fig -7.3: Encrypted Image

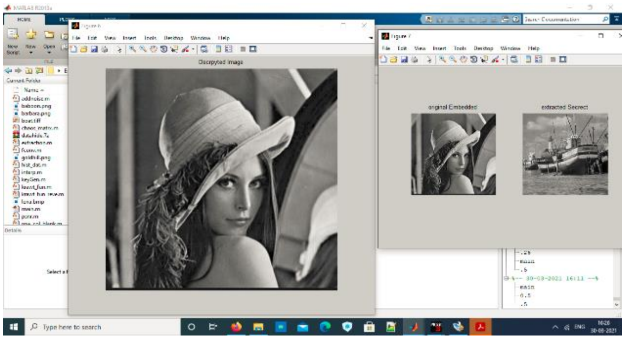


Fig -7.4: Decrypted Image

8. CONCLUSION

Reversible data hiding/cryptography in encrypted images/texts is a new topic drawing attention because of the privacy-preserving requirements from cloud data management. Previous methods implement RDH in encrypted images/texts by vacating room after encryption, as opposed to which we proposed by reserving room before encryption. Thus the data hider can benefit from the extra space emptied out in previous stage to make data hiding process effortless. The proposed method can take advantage of all traditional RDH techniques for plain images/texts and achieve excellent performance without loss of perfect secrecy. Furthermore, this novel method can achieve real reversibility, separate data extraction and greatly improvement on the quality of marked decrypted images/texts.

REFERENCES

[1] T. Kalker and F.M.Willems, "Capacity bounds and code constructions for reversible data- hiding," in Proc. 14th Int. Conf. Digital Signal Processing (DSP2002), 2002, pp. 71–76.

[2] W. Zhang, B. Chen, and N. Yu, "Capacity-approaching codes for Reversible data hiding/cryptography," in Proc 13th Information Hiding (IH'2011), LNCS 6958, 2011, pp. 255–269, Springer-Verlag.

[3] W. Zhang, B. Chen, and N. Yu, "Improving various Reversible data hiding/cryptography schemes via optimal codes for binary covers," IEEE Trans. Image Process., vol. 21, no. 6, pp. 2991–3003, Jun. 2012
.

[4] X. L. Li, B. Yang, and T. Y. Zeng, "Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection," IEEE Trans. Image Process., vol. 20, no. 12, pp. 3524–3533, Dec. 2011.