International Journal of Research in Advent Technology, Vol.3, No.2, February 2015
E-ISSN: 2321-9637

RDH in encrypted images by Reserving Room before Encryption

Nilesh Bhor, Pratik Chandgude, Kamlesh Patil, Aniket Wabale
Computer engineering,
Email: kamleshdpatil@gmail.com , pratik.chan@gmail.com

Abstract- There are also a number of works on data hiding in the encrypted domain. The RDH in encrypted image is investigated in. Most of the work on RDH focuses on the data embedding/extracting on the plain spatial domain. This method by reserving room before encryption with a traditional RDH algorithm, and thus it is very comfortable for the data hider to reversibly embed data in the encrypted image. The proposed method can give real reversibility, that is, data (information) extraction and image (pixels) recovery are free of any bug (error). Thus the data hider can benefit from the extra remaining space Emptied out in (all) previous stages to make data hiding process effortless. The proposed system can take advantage of all traditional RDH techniques for plain images and achieve excellent performance without loss of perfect secrecy.

Furthermore, the novel method (system) can achieve real reversibility, separate data extraction and greatly improvement on the quality of marked decrypted Images.

Index Terms- Image encryption, RDH Reversible data hiding, and Privacy protection.

1. INTRODUCTION

Reversible data hiding is a technique of data hiding by which original image can be recovered losslessly after the embedded message is extracted from the original image. This technique is widely used for military and medical applications.

All previous methods are vacating room from the encrypted images but all these methods are subject to some error rate on data extraction and image restoration.

In this paper, we propose a novel method for RDH in which we reserving room before encryption with a traditional RDH algorithm.

In this method, we first reserve space 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.

2. GENERATION OF ENCRYPTED IMAGE

To generate the encrypted image, this stage can be divided into three steps: image partition, self reversible embedding, image encryption.

2.1. *Image partition*

At the beginning, we divided the standard original image into two parts that is A and B. To achieve the better performance we construct the area B more smoother.

2.2. *Self reversible embedding*

To embed the LSB planes of A into B by using traditional RDH algorithm, the reversible embedding technique is used.Pixel in the image B are first categorized into two groups: white pixels with indices i and j satisfies $(i + j) \bmod 2 = 0$ and black pixels whose indices satisfies $(i + j) \bmod 2 = 1$.

2.3. *Image encryption*

we use Advanced Encryption Standard algorithm for encrypt the image. So first the image is converting into streams of data array and each data will be encrypted. The shares will be created based on the number of users. For example if 5 users are there means we create five shares. For each share the user can reveal the image but only after five shares he can view the full image.

3. DATA HIDING IN ENCRYPTED IMAGE

The data hider acquires the encrypted image, he can embed data into it but he does not get the access to the image. Data hider first locating the encrypted version of A. after getting how many bit-planes and pixels he can modify, the data hider substitute the available bit-planes with additional data M with LSB

replacement. At the end, the data hider sets the label to point out the end position of the embedding process

After generating the decrypted image the user can further extract the data and recover original image.

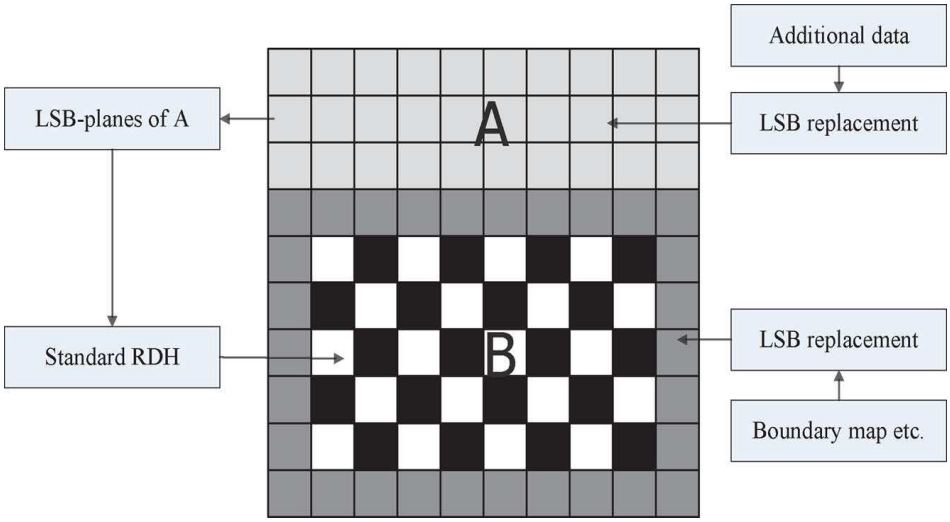


Fig.1. Illustration of image partition and embedding process

followed by M.

4. DATA EXTRACTION AND IMAGE RECOVERY

Data extraction and image decryption are completely independent process from each other. There are two cases for extracting data from the image.

4.1*: Extracting data from encrypted images:*
The database manager uses the data hiding key and decrypts the LSB plane of A and extract the additional data M when request comes for updating the information the database manager updates information using the LSB replacement and encrypt the information. As the entire information is get encrypted there are very less chances of the leakage of original content.

4.2. *Extracting data from decrypted images:*
According to the user requirements it decrypt the image first and extract the data from the decrypted image when it is needed.

5 . DATA EXTRACTION AND IMAGE RESTORATION:

This process is very similar to the traditional RDH methods.
Step1: According to data hiding key extract the data until the end label is reached.
Step2: Extract the pixels and boundary map from the LSB of marginal area of B and scan B.
Step3: If no black pixels participated in embedding process go to step 5.
Step4: Calculate the estimating the errors e of black pixels B.
Step 5: Calculate the estimating the errors e of white pixels B.
Step 6: Repeat step 2 to 5 and merge all extracted bits to form LSB planes of A. Now we have perfectly recovered B.
Step7: Substitute marked LSB planes of A with its extracted from B to get original cover image C.

6. IMPLEMENTATION ISSUES

This approach will be tested on available Standard images. The size of all images is 512*512*8.

6.1. *Boundary Map*
For distinguishing between natural and pseudo boundary pixels and its size is critical to practical

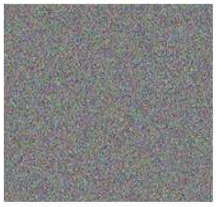
applicability of proposed approach Boundary map is used. In many cases, no any boundary map is needed.

6.2. *Choice of LSB-Plane Number*

The size of section A is analyzed not only by the length of to-be-embedded messages but also by the number of LSB-planes embedded reversibly in B When original image C is divided into A and B.

7. RESULTS IN IMAGES

(a) Original image



(b) encrypted image

(c) Decrypted image containing Messages

(d) recovery version

8. CONCLUSION

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

REFERENCES

[1] IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 8, NO. 3, MARCH 2013

[2] 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. 7176.

[3] 2. W. Zhang, B. Chen, and N. Yu, Capacity-approaching codes for reversible data hiding, in Proc 13th Information Hiding (IH2011), LNCS 6958, 2011, pp. 255269, Springer-Verlag.

[4] 3. J. Fridrich and M. Goljan, Lossless data embedding for all image formats, in Proc. SPIE Proc. Photonics West, Electronic Imaging, Security and Watermarking of Multimedia Contents, San Jose, CA, USA, Jan. 2002, vol. 4675, pp. 572583.

[5] 4. J. Tian, Reversible data embedding using a dierence expansion, IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890896, Aug. 2003.