International Journal of Technical Research and Applications e-ISSN: 2320-8163,
www.ijtra.com Volume 3, Issue 3 (May-June 2015), PP. 52-54

REVERSIBLE DATA HIDING IN ENCRYPTED IMAGES BY RESERVING ROOM BEFORE ENCRYPTION TECHNIQUE AND LSB MATCHING ALGORITHM

Harshali Sanglikar[1], Neha Jadhav[2], Pawankumar Thorat[3], Rubeena Khan[4]
Department of Computer engineering
Modern Education Society's college of engineering
Savitribai Phule Pune University, Pune
harshali101@gmail.com, neha.jadhav@gmail.com, thoratpawankumar@gmail.com
Rubeena.khan@mescoepune.org

Abstract:- **In recent years the topic of steganography has become very popular and a lot of research is being done in this field. Reversible data hiding is a method in which the image in which the encrypted data is hidden is losslessly recovered. In this paper we have proposed such a method that the previous work done limitations can be overcome. In this paper, we embed the encrypted data in the image by using LSB matching technique for reserving room, so that fast, optimal and lossless steganography is achieved. The proposed method provides total reversibility, that is, data extraction and image recovery.**

Index terms- **reversible data hiding, encryption, data hiding and extraction.**

I. INTRODUCTION

The basic concept of steganography is to hide the very presence of communication by embedding message into innocuous-looking cover objects.

Reversible data hiding is a method to hide (embed) additional message into some distortion free unacceptable cover media. It is needed in the fields such as military or medical images, with a reversible manner are use so that the original cover content can be perfectly recovered after extraction of the hidden message[1].

In[1][3][5], separable reversible data hiding technique a user or content owner encrypts the original carrier image then a data hider compress the image to create space to for accommodation of some additional data. However, in[4] some circumstances if the user (content owner) does not trust the service provider then he may encrypt it (secret data) when it is to be transmitted ted, channel provider without any knowledge of the cryptographic key may compress the encrypted data due to the limited channel resource[2].

Data hiding is referred to as a process to hide data or embed data, i.e, the data embedding process links two sets, a set of the embedded data and another set of the carrier media or cover media data. In most cases of data hiding, the cover media or carrier image becomes distorted due to data hiding and cannot be inverted back to the original image as it was before. That is, media has permanent distortion even after the embedded data have been removed. In some applications, such as medical diagnosis and law enforcement it is desired that the original cover media can be recovered efficiently with no loss.[7][8][9] The marking techniques satisfying this requirement are referred to as reversible, lossless, distortion free or invertible data hiding techniques. The separable means which is able to separate, in other words, we can separate the some things, activities using suitable criteria is as said. Separable reversible data hiding concept is the separation of activities i.e. extraction of original cover image and extraction of payload. Separable data hiding key is the separation that exists according to keys. Here at the receiver side, there are three different cases are encountered ie image recover, data extraction and data decryption.

There are several methods for data hiding in images available now, but most of them are not reversible in nature. In [1] paper method to achieve pure recovery of image and data is proposed. Thus here gives same importance for both image and data. In the Existing System, Reserving Room before Encryption technique is following. As losslessly reserving room in the encrypted images is relatively difficult and sometimes inefficient, but still we are so obsessed to find novel RDH techniques working directly for Encrypted Images. The method is of compressing the encrypted LSBs to reserve room for additional data by finding syndromes of a parity check matrix, and the side information used at the receiver side is also the spatial correlation of decrypted images. All the three methods try to vacate room from the encrypted images directly.

II. PROPOSED SYSTEM

In proposed method can achieve real reversibility, that is, data extraction and image recovery are free of any error. If we reverse the order of encryption and vacating room, i.e., reserving room prior to image encryption at content owner side, the RDH tasks in encrypted images would be more natural and much easier which leads us to the novel framework, "reserving room before encryption (RRBE)". Not only does the proposed method separate data extraction from image decryption but also achieves excellent performance in two different prospects. Real reversibility is realized, that is, data extraction and image recovery are free of any error. For given embedding rates, the PSNRs of decrypted image containing the embedded data are significantly improved; and for the acceptable PSNR, the range of embedding rates is greatly enlarged. The proposed method allows us to use any format of the image in which we want to store data i.e jpeg, png, bmp etc. The method allows us to use any format of data file i.e pdf,docx, etc.

A. Encrypted Image Generation

In this module, to construct the encrypted image, the first stage can be divided into two steps. Image Partition and Self Reversible Embedding followed by image encryption. At the beginning, image partition step divides original image into two parts and then, the LSBs of the least dominant channels in the image are reversibly embedded into the dominant colour channel with a standard RDH algorithm so that LSBs of can be

used for accommodating messages; at last, encrypt the rearranged image to generate its final version.

B. Data hiding in encrypted image

In this module, 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 lossless 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.

C. Data extraction and image recovery

In this module, Extracting Data from Encrypted Images to manage and update personal information of images 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. When the database manager gets the data hiding key, he can decrypt and extract the additional data by directly reading the decrypted version. When requesting for updating information of encrypted images, the database manager, then, updates information through LSB replacement and encrypts up dated information according to the data hiding key all over again. As the whole process is entirely operated on encrypted domain, it avoids the leakage of original content.

D. Data extraction and image restoration

In this module, after generating the marked decrypted image, the content owner can further extract the data and recover original image. Reversible hiding allows extraction of the original host signal and also the embedded message. There are two important requirements for reversible data hiding techniques: the embedding capacity should be large; and distortion should be low. These two requirements conflict with each other. In general, a higher embedding capacity results in a higher degree of distortion. An improved technique embeds the same capacity with lower distortion or vice versa. For the image restoration we have to follow the specified steps by which we can easily recover the original image. After the image encryption the image must be considered as an individual unit of work so we can fully concentrate on the watermarking technique and proceed further

III. AES ALGORITHM

The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) information. Encryption converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext. The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits. The AES algorithm is based on permutations and substitutions. Permutations are rearrangements of data, and substitutions replace one unit of data with another. AES performs permutations and substitutions using several different techniques.

In our system we are using 128 bit key and in AES this is represented by Nb = 4, which reflects the number of 32-bit words (number of columns) in the State. The length of the Cipher Key, K, is 128. The key length is represented by Nk = 4, 6, or 8, which reflects the number of 32-bit words (number of columns) in the Cipher Key. The number of rounds to be performed during the execution of the algorithm is dependent on the key size. The number of rounds is represented by Nr, where Nr = 10 when

Nk = 4, Nr = 12 when Nk = 6, and Nr = 14 when Nk = 8. For both its Cipher and Inverse Cipher, the AES

Algorithm uses a round function that is composed of four different byte-oriented transformations:

1) Substitution using a substitution table (S-box).
2) Shifting rows of the State array by different offsets
3) Mixing the data within each column of the State

Array

4) Adding a Round Key to the State.

IV. LSB MATCHING ALGORITHM

Least Significant Bit Embedding's (LSB) are a general stenographic technique that may be employed to embed data into a variety of digital media, but one of the most studied applications is using LSB embedding to hide one image inside another. In our system we are using LSB matching algorithm for embedding data in the image. For embedding data in the image we first find the dominant channel in the image. The dominant channel is basically the colour which is most present in the image. The other two channels that are present in the image are used for storing the data in the image. Before storing the data in the image we reserve room that is we store the values of the least dominant channels LSB's in the dominant channel by EXORing them. The space for storing the data in the image is now reserved. The data that is to be stored in the image is first compared with the LSBs of the image's least dominant channel, if the values are same then the bits are kept as it is and if there is a mismatch then the values are replaced. Thus the algorithm saves a lot of computation time and also is overcoming the abruptly replacing the LSB bits of the image.

V. FEATURES OF SYSTEM

A. Three Keys for more Data Security

Encrypted data is hidden in Encrypted Image with separate keys for Data Encryption, Image Encryption and Data Hiding. For decrypting of data receiver should have both Data Encryption and Data hiding key.

B. Protection for auto generated keys

To perform any operation the user has only 3 attempts. If user is fail to perform any of operation means user enter wrong 3 times then the system is goes to not responding state and one mail with receiver computer IP address is send to the admin.

C. *Allows any type of data or image file*

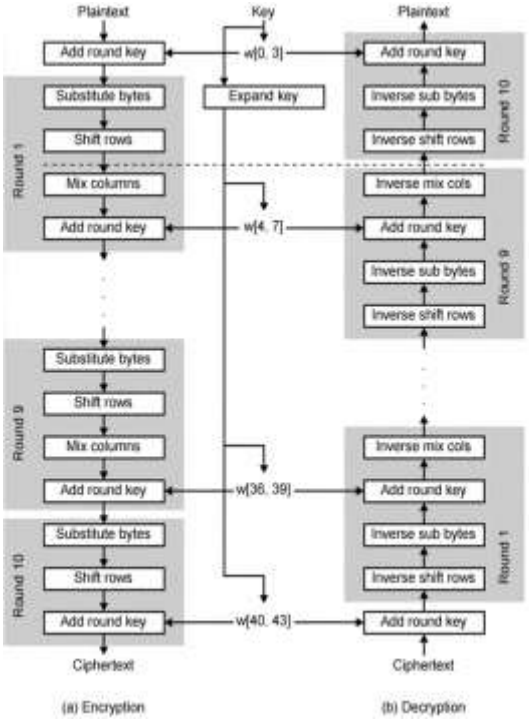
The system can work on any format of data file like .pdf, .docx, .rtf etc and any format of image file like .jpeg, .bmp, .png etc.

D. Allows large data files to be encrypted

The system allows very large size of data file to be encrypted easily as we are storing the image in the image file. If we want to store large size data in the image we would have to take a bigger image to store that much amount of data in it.

E. Faster computation time

The use of multithreading allows faster computation both while image encryption and decryption.



AES flow chart

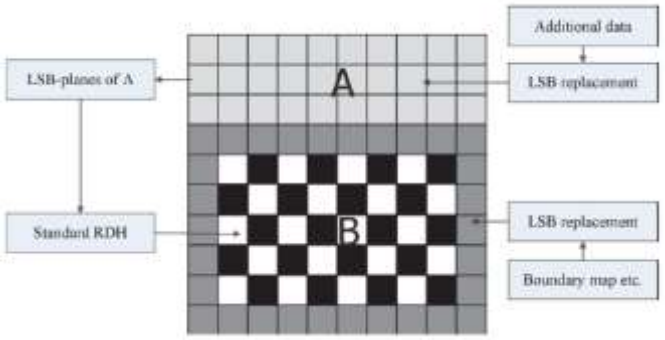


Illustration of image partitioning and embedding process

VI. CONCLUSION

Reversible data hiding in encrypted images is a new topic drawing attention because of the privacy-preserving requirements from cloud data management. Previous methods implement RDH in encrypted images 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 and achieve excellent performance without loss of perfect secrecy. Furthermore, by this novel method can achieve reversibility as well as separate data extraction and greatly improvement on the quality of marked decrypted images

REFERENCES

[1] Reversible Data Hiding in Encrypted Images by Reserving Room Before Encryption Kede Ma, Weiming Zhang, Xianfeng Zhao, Member, IEEE, Nenghai Yu, and Fenghua Li MARCH 2013

[2] W. Zhang, B. Chen, and N. Yu, "Capacity-approaching codes for reversible data hiding," 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 schemes via optimal codes for binary covers," IEEE Trans. Image Process., vol. 21, no. 6, pp. 2991–3003, Jun. 2012.

[4] 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. 572–583.

[5] J. Tian, "Reversible data embedding using a difference expansion," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896, Aug. 2003.

[6] Z. Ni, Y. Shi, N. Ansari, and S. Wei, "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar.2006.

[7] D.M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," IEEE Trans. Image Process., vol. 16, no. 3, pp. 721–730, Mar. 2007.

[8] 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.

[9] P. Tsai, Y. C. Hu, and H. L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," Signal Process., vol. 89, pp. 1129–1143, 2009.

[10] L. Luo et al., "Reversible imagewatermarking using interpolation technique," IEEE Trans. Inf. Forensics Security, vol. 5, no. 1, pp. 187–193, Mar. 2010.

[11] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y.-Q. Shi, "Reversible watermarking algorithm using sorting and prediction," IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 7, pp. 989–999, Jul. 2009.

[12] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, Handbook of Applied Cryptography. Boca Raton, FL, USA: CRC, 1996.

[13] K. Hwang and D. Li, "Trusted cloud computing with secure resources and data coloring," IEEE Internet Comput., vol. 14, no. 5, pp. 14–22, Sep./Oct. 2010.

[14] M. Johnson, P. Ishwar, V. M. Prabhakaran, D. Schonberg, and K. Ramchandran, "On compressing encrypted data," IEEE Trans. Signal Process., vol. 52, no. 10, pp. 2992–3006, Oct. 2004.