RESEARCH ARTICLE OPEN ACCESS

Data encryption using LSB matching algorithm and Reserving Room before Encryption

Harshali Sanglikar*, Pawamkumar Thorat**,Neha Jadhav***,R.AKhan****
*,**,****,*****(Department of Computer Science,Savitribai Phule Pune University, Pune)

ABSTRACT
Now a days, more and more attention is paid to reversible data hiding (RDH) in encrypted images, since it maintains the excellent property that the original cover can be losslessly recovered after embedded data is extracted while protecting the image content's confidentiality. Previously proposed methods embed data by reversibly vacating room from the encrypted images, which may cause some errors in data during data extraction and/or image restoration. In this paper, a novel method of reserving room before encryption with a traditional RDH algorithm, and thus it is easy for the data hider to reversibly embed data in the encrypted image is proposed. The proposed method can achieve real reversibility, that is, data extraction and image recovery are free of any error.
Keywords - *RDH, Prediction Error Expansion, Lossless Recovery*

I. INTRODUCTION

Here we are investigating the data hiding technique which is reversible in nature. Using the encrypted image as a cover data in which the data is embedded. Reversible data hiding is a technique to embed additional message into some distortion-unacceptable cover media, in fields such as military or medical images, with a reversible manner 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 firstly a content owner encrypts the original uncompressed image then a data hider compress the image to create space to accommodate some additional data. An effective and popular means for privacy protection, encryption converts the ordinary signal into incomprehensible data, so that the general signal processing typically takes place before encryption or after decryption. However, in[4] some circumstances that a content owner does not trust the service provider so when the secret data to be transmitted are encrypted, 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, i.e, the data hiding process links two sets of data, a set of the embedded data and another set of the cover media data. In most cases of data hiding, the cover media becomes distorted due to data hiding and cannot be inverted back to the original media. That is, cover media has permanent distortion even after the hidden 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. Separable reversible data hiding concept is the separation of activities i.e. extraction of original cover image and extraction of payload. In separable data hiding key the separation exists according to keys. Here at the receiver side, there are three different cases are encountered. The separation of extracting the data and getting the cover media come to be exists.

There are several methods for data hiding in images available now. But most of them are not reversible in nature. Here 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. Since losslessly reserving room from the encrypted images is relatively difficult and sometimes inefficient, still we are so obsessed to find novel RDH techniques working directly for Encrypted Images.[] The method in compressed 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 METHOD

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.

2.1 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 are reversibly embedded into 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.

2.1 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.

2.3 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.

2.4 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. ALGORITHM DESCRIPTION

3.1 LSB Matching for data hiding

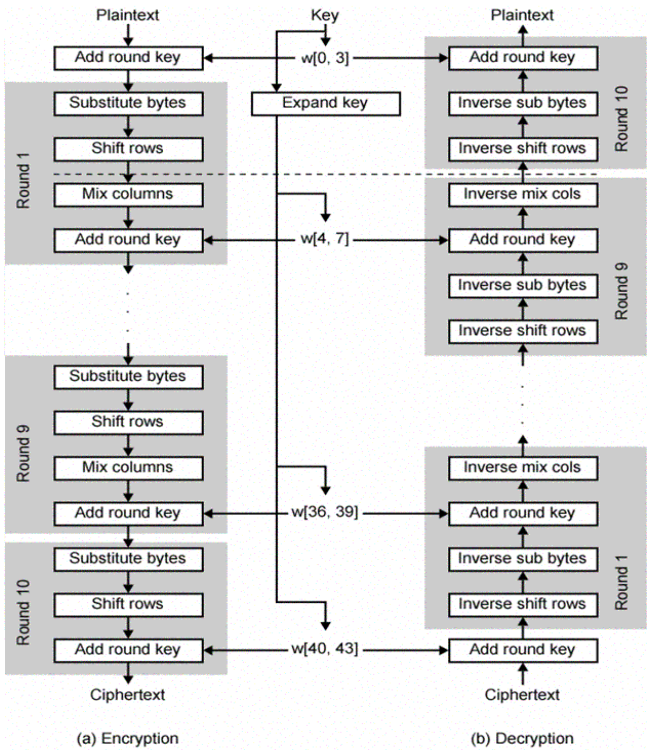
Least Significant Bit Embeddings (LSB) are a general steganographic technique that may be employed to embed data into a variety of digital media, but one of the most studied applications is using LSB embedding to hide one image inside another. My work focuses on LSB embedding with greyscale images, but the general principles extend to other applications of LSB embedding. LSB embeddings are remarkable for their simple design and alarming effectiveness. LSB is a

non-filtering algorithm in spatial domain. Least Significant Bit (LSB) approach is a simple, basic method for embedding information in an image in which the least significant bit of the colours (RGB) of the pixels in the image is replaced with a bit of the secret message. Using a 24-bit image, a bit of each of the colours, red, green, and blue is used for embedding, since each one is considered as a byte. In other words, three bits can be embedded in each pixel. Least Significant Bit (LSB) embedding is a simple strategy to implement steganography. The technique works by replacing some of the information in a given pixel with information from the data in the image. While it is possible to embed data into an image on any bit-plane, LSB embedding is performed on the least significant bit(s). This minimizes the variation in colors that the embedding creates.

For example, embedding into the least significant bit changes the color value by one. Embedding into the second bit-plane can change the color value by 2. If embedding is performed on the least significant two pixels, the result is that a color in the cover can be any of four colors after embedding. Steganography avoids introducing as much variation as possible, to minimize the likelihood of detection. In a LSB embedding, we always lose some information from the cover image. This is an effect of embedding directly into a pixel. To do this we must discard some of the cover's information and replace it with information from the data to hide. LSB algorithms have a choice about how they embed that data to hide. They can embed losslessly, preserving all information about the data, or the data may be generalized so that it takes up less space.

3.2 AES for data encryption

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.



The first paragraph under each heading or subheading should be flush left, and subsequent paragraphs should have a five-space indentation. A colon is inserted before an equation is presented, but there is no punctuation following the equation. All equations are numbered and referred to in the text solely by a number enclosed in a round bracket (i.e., (3) reads as "equation 3"). Ensure that any miscellaneous numbering system you use in your paper cannot be confused with a reference [4] or an equation (3) designation.

IV. 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.