ISSN:2320-0790

An International Journal of Advanced Computer Technology

Secure Reversible Data Hiding in Encrypted Images by Allocating Memory before Encryption via Security keys

Priya Jambhulkar[1], P.R.Pardhi[2]
CSE Department, RCOEM, Nagpur (MS), India

Abstract: ***Digital image and information embedding system have number of important multimedia applications. Recently, attention is paid to reversible data hiding (RDH) in encrypted images is more, since it maintains the excellent property that the original cover can be lossless recovered after embedded data is extracted while protecting the image content's confidentiality. RDH is a technique used to hide data inside image for high security in such a way that original data is not visible. All previous methods embed data by reversibly vacating room from the encrypted images, which may be subject to some errors on data extraction and/or image restoration. In this paper, we propose another method in which we simply encrypt an image without its header by using our own algorithm. And thus it is easy for the data hider to reversibly embed data in the encrypted image. The proposed method can achieve real reversibility, that is, data extraction and image recovery are free of any error.***

Keywords: Triple DES, Encryption, Decryption and Reversible Data hiding

1. Introduction

With the increasingly wide reach of the Internet, communications via Internet are getting more frequent. Due to large number of threats against communications security, information protection has become an significant issue. Information embedding and data hiding systems play a key role in addressing couple of major challenges that have arisen from the widespread distribution of multimedia content over digital communication networks. Data hiding is referred to as a process to hide data into cover media. It links two sets of data such as embedded set of data and another set of the cover media data. The relationship between these two sets of data characterizes different applications. In particular, these systems are enabling technologies for (1) enforcing and protecting copyrights (2) authenticating and detecting tampering of multimedia signals & images. This system has been widely used in military imagery, medical imagery, Satellite image viewer and law forensics, where no distortion of the original cover is acceptable.

In theoretical aspect, Kalker and Willems [1] established a rate-distortion model for RDH, through which they proved the rate-distortion bounds of RDH for memory-less covers and proposed a recursive code construction which, however, does not approach the bound. Zhang et al. [2], [3] improved the recursive code construction for binary covers and proved that this construction can achieve the rate-distortion bound as long as the compression algorithm reaches entropy, which establishes the equivalence between data compression and RDH for binary covers. In practical aspect, many RDH techniques have emerged in recent years. Fridrich et al. [4] constructed a general framework for RDH. By first extracting compressible features of original cover and then compressing them to lossless, spare space can be saved for embedding auxiliary data. A more popular method is based on difference expansion (DE) [5], in which the difference of each pixel group is expanded, e.g., 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) [6], in which space is saved for data embedding by shifting the bins of histogram of gray values. The state-of-art methods [7][11] usually combined DE or HS to residuals of the image, e.g., the predicted errors, to achieve better performance. With regard to providing confidentiality for images, encryption [12] is an effective and popular means as

it converts the original and meaningful content to incomprehensible one. In [13], Hwang et al. Advocated reputation-based trust-management scheme enhanced with data coloring (a way of embedding data into covers) and software watermarking, in which data encryption and coloring offer possibilities for upholding the content owner's privacy and data integrity.

In [16], Zhang divided the encrypted image into several blocks. By flipping 3 LSBs of the half of pixels in each block, room vacated for the embedded bit. The data extraction and image recovery proceed by finding which part has been flipped in one block. This process can be realized with the help of spatial correlation in decrypted image. Hong et al. [17] ameliorated Zhang's method at the decoder side by further exploiting the spatial correlation using a different estimation equation and side match technique to achieve much lower error rate.

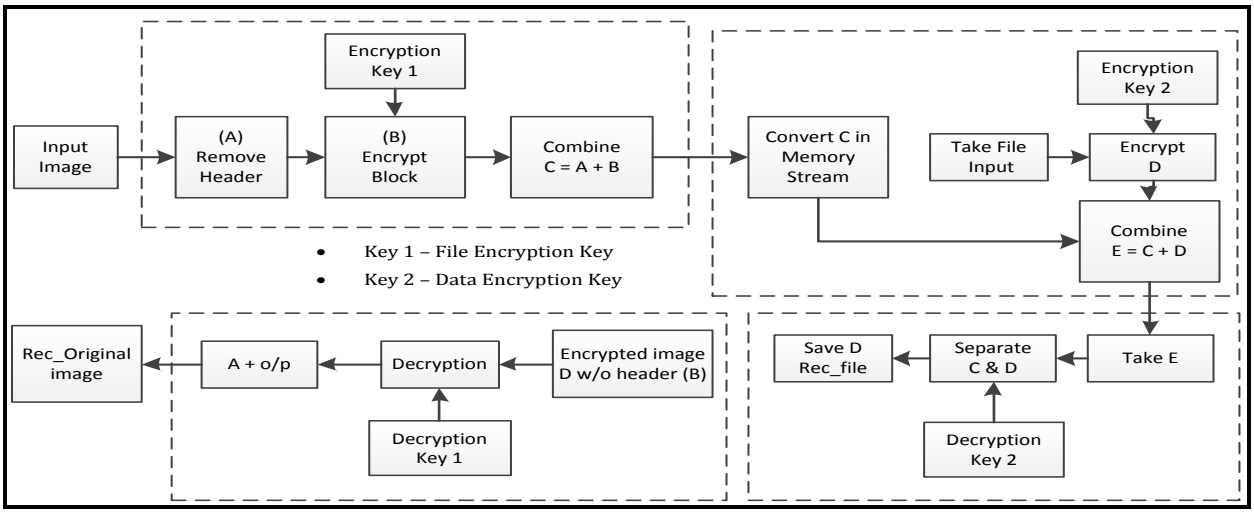

Figure 1: System Architecture for Secure Reversible Data Hiding in Encrypted image

To separate the data extraction from image decryption, Zhang [18] emptied out space for data embedding following the idea of compressing encrypted images [14], [15]. Compression of encrypted data can be formulated as source coding with side information at the decoder [14], in which the typical method is to generate the compressed data in lossless manner by exploiting the syndromes of parity-check matrix of channel codes. The method in [18] compressed the encrypted LSBs to vacate 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. However, since the entropy of encrypted images has been maximized, these techniques can only achieve small payloads [16], [17] or generate marked image with poor quality for large payload [18] and all of them are subject to some error rates on data extraction and/or image restoration. Although the methods in [16], [17] can eliminate errors by error correcting codes, the pure payloads will be further consumed. In the present paper, we propose a novel method for RDH in encrypted images, for which we do not "vacate room after encryption" as done in [16]–[18].

The rest of this paper is organized as follows. The problem definition is discussed in section 2. Section 3 explains Research methodology along with implantation detail and Section 4 describes how the system is work. Finally, Section 5 concludes and discusses future scope of this work.

2. Problem definition

Our aim is to develop a secure system to send the data over a network which consist of separate and reverse image encryption (like bmp image and txt file encryption), data embedding which prevents any third party access to the private data. In this method, with the help of symmetric key, we can achieve real reversibility i.e. data extraction and image recovery are free of any error. The System Architecture for

Secure Reversible Data Hiding in Encrypted image is as shown in figure 1.

Consider an input image which contains header part as A and data block as B. Then remove header A and encrypt B part by using encryption "key1". After encrypting combine header A and encrypted block B and store its result in C. This intermediate result is then converted into memory stream. Take another text file as input and encrypt the file by using encryption "key2" and save it in D. Further combine memory stream present in C and encrypted text file D and store combine result in E. Now E contain raw file which is in unreadable format.

On decryption side, take the raw file E which contains C and D. To separate image and file, first apply decryption on E using "key2" which will give result D i.e. text file is extracted. Now separate header form encrypted image and again apply decryption on E using "key1". The output of this process is the original image file without any distortion.

3. Research Methodology

Normally image is not visible after encryption and not even readable by any image viewing software. Changing image R, G, B component in order to store data will not affect image, however it would affect the quality of the image. If we try to change single bit in an encrypted image, we won't be able to decrypt the image. However, our main goal is to hide secret/confidential data or information in encrypted image and on other side that data should be recovered free from any error i.e. real reversibility is done. For encryption process, we have used triple DES to address the obvious flaws in DES. Triple DES prevent secrete data from attacks like theoretical attack, demonstrated exhaustive key search attacks and also from meet-in-the-middle attack. Triple DES algorithm uses DES algorithm applied three times in succession with three different key. In all, triple DES uses extended key which is combined key size of 168 bits (3 times 56), beyond the reach of the EFF DES Cracker. The following are the steps included in Encryption of data hiding process.

Step 1. Convert image into memory stream
Step 2. Separate first 54 bytes i.e. image header in A
Step 3. Separate remaining data bytes from image
Step 4. Use Triple DES algorithm and encrypt key
B => C
Step 5. Combine A and C results D
A + C => D
Step 6. Now D will be the encrypted image
Step 7. If we want to hide data inside D. Provision for data hiding is required because if we try to change single bit in encrypted image, we cannot decrypt the image and we also need to know the size of data to be encrypted.
Step 8. Take the file and make provision to add data without disturbing it.

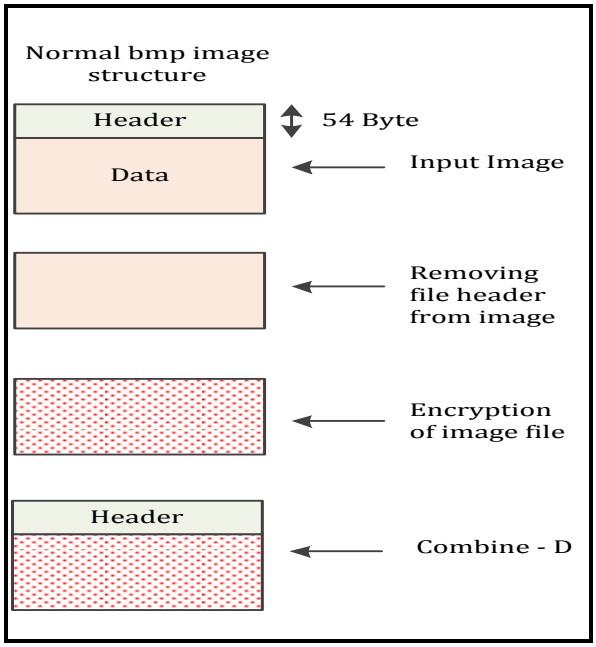

Figure 2: Encryption performed on an image file describing step by step process

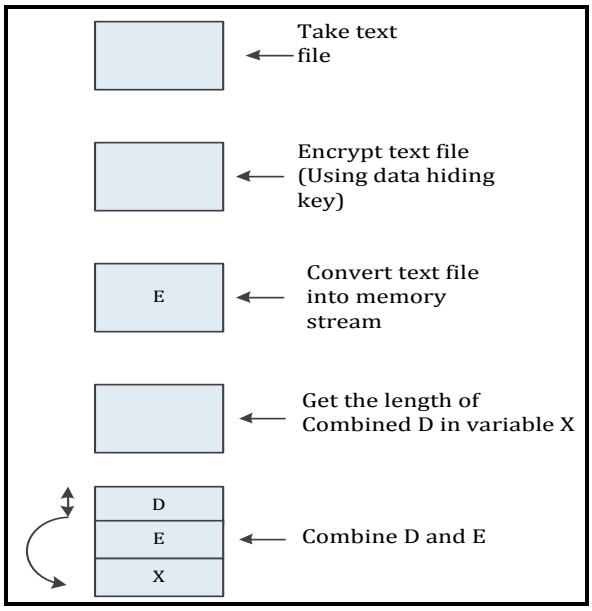

Figure 3. Hiding text file within encrypted image

4. Experimental result

The experiment is performed in three steps:-

1. Encryption
2. Data Hiding
3. Decryption & Recovery

The steps are as follows:

Encryption:
Encryption Process is the first step that contains the image to be encrypted as shown in figure 5. We select the image in which data is to be hidden and then Enter Encryption Key which is alphanumeric with special symbol. The output obtained is an Encrypted Image which is in unreadable format. This format is also not readable by even image viewing software.

Data Hiding:
In second step, we need a text file which is to be hidden in encrypted image. For data hiding process we use data hiding key which is also alphanumeric. The output of this process is unreadable raw file which encapsulate text file into encrypted image. Raw file is a file that has no significant with unknown name.

Decryption and Recovery
On other side, if we want to recover the data inside text file along with image, firstly we have to recover the data in specific folder by applying recovery process on raw file with the help of data hiding key. Now in order to extract original image, decrypt raw image using the symmetric key i.e. encryption key from the first step. The output of the process is exact original Image recover free from error as shown in figure 6.

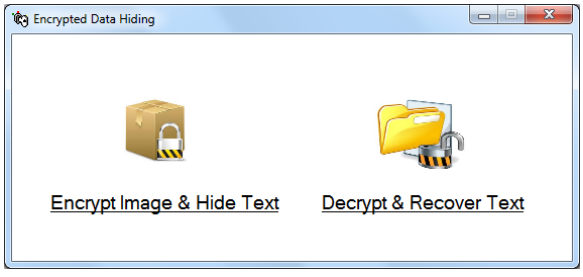

Figure 4: Application for encryption data hiding

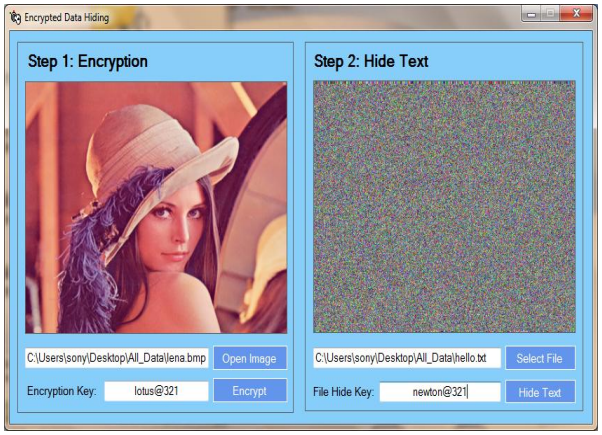

Figure 5: Encryption Process

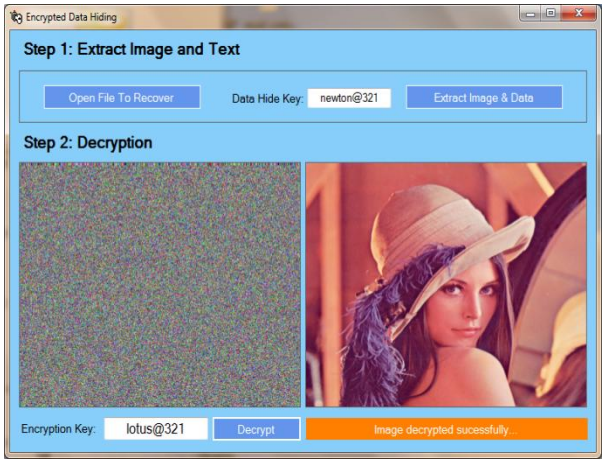

Figure 6: Decryption Process

5. Conclusion

It is a novel reversible data hiding scheme for encrypted image with a low computation complexity proposed, which consists of image encryption, data embedding and data extraction/ image-recovery phases. The embedded data can be correctly extracted while the original image can be perfectly recovered along with the data hidden inside the image. Without encryption and data-hiding key, it is impossible to extract the additional data and recover the original image by third party which is a safer way to send the message over a network.

Achieve excellent performance without loss of perfect secrecy. Furthermore, this new method can achieve real reversibility, separate data extraction and greatly improvement on the quality of marked decrypted images.

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] 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 image watermarking 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. 562 IEEE Transactions On Information Forensics And Security, Vol. 8, No. 3, March 2013

[15] W. Liu, W. Zeng, L. Dong, and Q. Yao, "Efficient compression of encrypted grayscale images," IEEE Trans. Image Process., vol. 19, no. 4, pp. 1097–1102, Apr. 2010.

[16] X. Zhang, "Reversible data hiding in encrypted images," IEEE Signal Process. Lett, vol. 18, no. 4, pp. 255–258, Apr. 2011.

[17] W. Hong, T. Chen, and H.Wu, "An improved reversible data hiding in encrypted images using side match," IEEE Signal Process. Lett, vol. 19, no. 4, pp. 199–202, Apr. 2012.

[18] X. Zhang, "Separable reversible data hiding in encrypted image," IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 826–832, Apr. 2012.

[19] Miscellaneous Gray Level Images [Online]. Available:http://decsai.ugr.es/cvg/dbimagenes/g512.php