Volume 3, Issue 8, August 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Reversible Data Embedding Using Asymmetric Cryptosystem

Dr.M.Gobi
Assistant Professor in Computer Science
Chikkanna Government Arts College
Tirupur, India

Mrs.R.Sridevi
Assistant Professor in Computer Science
PSG College of Arts & Science
Coimbatore, India

***Abstract*— This paper proposes a narrative reversible data embedding system within an image which is encrypted using a public key generated by ECC algorithm. Then the data which is to be hided can be embedded into the image by modifying a fraction of encrypted data. This encrypted image containing additional data will be decrypted using the private key, and the decrypted edition is similar to the original image. According to the data embedding key, with the aid of spatial correlation in natural image, the embedded data can be successfully extracted and the original image can be perfectly recovered.**

***Index Terms*— reversible data embedding, image encryption, image recovery, ECC**

I. Introduction

Reversible data embedding, in which the data can be reversed to the original cover media exactly, has attracted increasing interests from the data embedding community. Using this technique, the original cover content can be perfectly restored after extraction of the hidden message. A number of reversible data embedding systems have been proposed in recent years. Differences between two adjacent pixels are doubled to generate a new least significant bit (LSB) plane for accommodating additional data in difference expansion method [1]. A histogram shift mechanism can also be performed by the data hider, who utilizes the zero and peak points of the histogram of an image and slightly modifies the pixel gray values to embed data into the image [2]. Another method makes use of redundancy in a cover by performing lossless compression to create a spare space for data embedding [3]. In addition, various methods have been proposed into the typical reversible data embedding systems to improve the performance [4]–[6].

As is illustrious, encryption is a valuable and accepted means of privacy protection. In order to share a secret image with other person securely, a sender may encrypt the image before transmission. In some application scenarios, an intermediate hopes to append some additional message, such as the origin information, image notation or authentication data, within the encrypted image though he does not know the original image content. For example, when bank customers' images have been encrypted for protecting their privacy, a database administrator may aim to embed the personal information of customers into the corresponding encrypted images. It may be also hopeful that the original content can be recovered without any error after decryption and retrieval of additional message at receiver side, which means a reversible data embedding system for encrypted image, is desirable.

In some existing systems, a part of cover data is used to carry the additional message and the rest data are encrypted. For example [7], the intraprediction mode, motion vector difference and signs of DCT coefficients are encrypted, while a watermark is embedded into the amplitudes of DCT coefficients. In [8], the cover data in higher and lower bit planes of transform domain are respectively encrypted and watermarked. In [9], the message owner encrypts the signs of host DCT coefficients and each content user uses a different key to decrypt only a subset of the coefficients, so that a series of versions containing different fingerprints are generated for the users. In these joint systems, however, only a fractional encryption is involved, leading to a leakage of fractional information of the cover. Furthermore, the separation of original cover and embedded data from a watermarked version is not considered. In [10] and [11], each sample of a cover signal is encrypted by a public key mechanism and a homomorphism property of encryption is exploited to embed some additional data into the encrypted signal. But the data amount of encrypted signal is significantly expanded and the computation complexity is high. Also, the data embedding is not reversible. Some systems propose a data embedding scheme which encrypts the original image using an encryption key and the data hider, after encryption embed the additional data into it and send it through the network. The receiver then uses the same key to decrypt it to get the original image after which data is recovered from it [12]. This paper proposes a narrative reversible data embedding system for encrypted image, which is made up of image encryption, data embedding and data extraction/image recovery phases. The data of original cover are entirely encrypted, and the additional message is embedded by modifying a fraction of encrypted data. At receiver side, with the aid of spatial correlation in natural image, the embedded data are successfully extracted while the original image is perfectly recovered.

II. PROPOSED SYSTEM

An outline of the proposed system is given in Figure 1. Sender encrypts the original uncompressed image using a public key to produce an encrypted image, and then a data hider embeds additional data into the encrypted image using a data embedding key though he does not know the original content. With an encrypted image containing additional data, a receiver may firstly decrypt it using the private key, and the decrypted version is similar to the original image. According to the data embedding key, he can further extract the embedded data and recover the original image from the decrypted version.

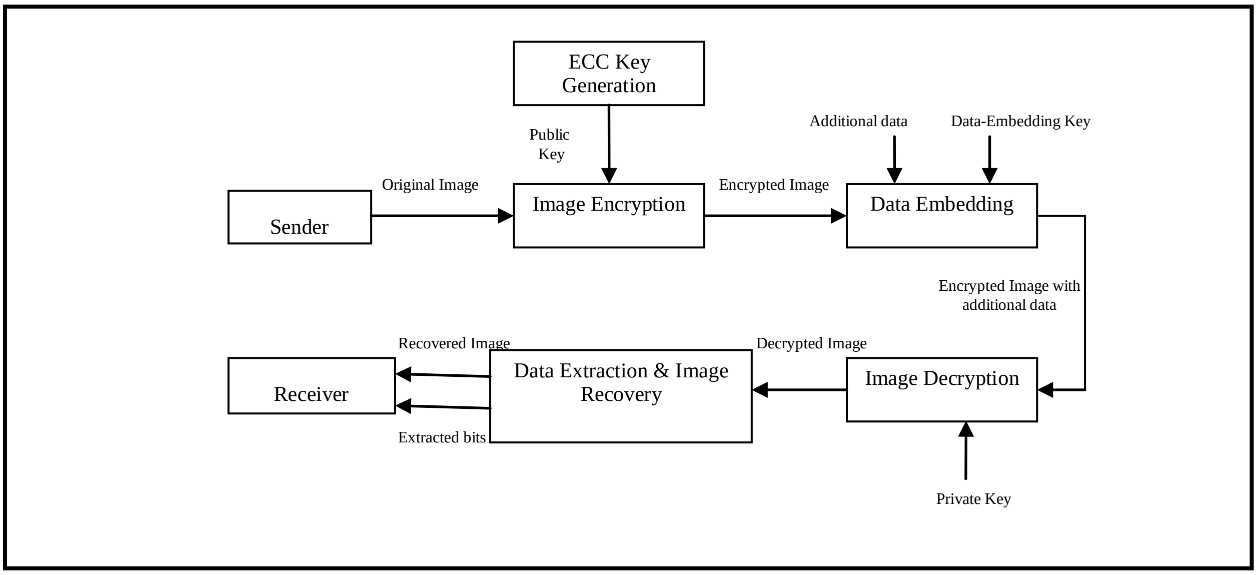

Figure 1: Reversible Data Embedding using Asymmetric Cryptosystem

A. ECC Key Generation

Key generation is an important part where we have to generate both public key and private key. Here, Elliptic curve cryptography algorithm is used to generate both of these keys. The sender will encrypt the message with receiver's public key and the receiver will decrypt it using his own private key. For this key generation, select a number'd' within the range of 'n', which is the random number representing the maximum limit. Using the following equation, the public key, Q can be generated.

$$Q = d * P \tag{1}$$

d = the random number that we have selected within the range of (1 to n-1).

P = is the point on the curve.

'Q' is the public key and'd' is the private key.

B. Image Encryption

Assume the original image is in uncompressed format and each pixel with gray value falling into [0, 255] is represented by 8 bits. Denote the bits of a pixel as $pb_{i,j,0}, pb_{i,j,1}, pb_{i,j,2,\ldots\ldots\ldots\ldots} pb_{i,j,7,}$ where (i,j) indicates the pixel position, and the gray value as $p_{i,j}$. Thus

$$pb_{i,j,k} = \left(\frac{p_{i,j}}{2^k} \right) \quad mod\ 2\ , \mathrm{k}= 0,1,2,\ldots,7 \tag{2}$$

and

$$p_{i,j} = \sum_{u=0}^{7} pb_{i,j,k} \,.\, 2^k \tag{3}$$

In encryption phase, the exclusive OR results of the original bits and key bits are calculated.

$$PB_{i,j,k} = pb_{i,j,k} \oplus PU_{i,j,k} \tag{4}$$

Where $PU_{i,j,k}$ are the public keys generated using Elliptic Curve Cryptography. Then, $PB_{i,j,k}$ are concatenated in order as the encrypted data. A secure asymmetric block cipher is used here to ensure that anyone without the key pair, such as a potential attacker or the data hider, cannot obtain any information about original content from the encrypted data.

C. Data Embedding

With the encrypted data, although a data hider does not know the original image content, he can embed additional message into the image by modifying a small fraction of encrypted data. Firstly, the data hider segments the encrypted image into a number of no overlapping blocks sized by s x s. In other words, the encrypted bits $PB_{i,j,k}$ satisfying $(m-1).s+1\leq i \leq m.s$, $(n-1).s+1\leq j\leq n.s$ and $0\leq k\leq 7$ (m and n are positive integers) are within a same block. Then, each block will be used to carry one additional bit.

For each block, pseudo randomly divide the s^2 pixels into two sets S_0 and S_1 according to a data embedding key. Here, the probability that a pixel belongs to S_0 or S_1 is 1/2. If the additional bit to be embedded is 0, flip the 3 least significant bits (LSB) of each encrypted pixel in S_0 ,

$$PB'_{i,j,k} = \overline{PB_{i,j,k}}\ , \qquad (i,j) \in S_0 \text{ and } k = 0,1,2. \tag{5}$$

If the additional bit is 1, flip the 3 encrypted LSB of pixels in S_1

$$PB'_{i,j,k} = PB_{i,j,k} \qquad (i,j) \in S_1 \text{ and } k = 0,1,2. \tag{6}$$

The other encrypted data are not changed

D. Data Extraction and Image Recovery

When having an encrypted image containing embedded data, a receiver firstly generates $PR_{i,j,k}$ according to the public key, and calculates the exclusive OR of the received data and $PR_{i,j,k}$ to decrypt the image. We denote the decrypted bits as $pb'_{i,j,k}$. Clearly, the original five most significant bits (MSB) are retrieved correctly. For a certain pixel, if the embedded bit in the block including the pixel is zero and the pixel belongs to S1, or the embedded bit is 1 and the pixel belongs to S0, the data embedding does not affect any encrypted bits of the pixel. So, the three decrypted LSB must be same as the original LSB, implying that the decrypted gray value of the pixel is correct. On the other hand, if the embedded bit in the pixel's block is 0 and the pixel belongs to S0, or the embedded bit is 1 and the pixel belongs to S_1, the decrypted LSB

$$\begin{aligned} pb'_{i,j,k} &= PR_{i,j,k} \oplus PB'_{i,j,k} \\ &= PR_{i,j,k} \oplus \overline{PB_{i,j,k}} \\ &= \underline{PR_{i,j,k}} \oplus \overline{pb_{i,j,k} \oplus PR_{i,j,k}} \\ &= pb_{i,j,k} \qquad k = 0,1,2. \end{aligned} \tag{7}$$

That means the three decrypted LSB must be different from the original LSB. In this case:

$$pb'_{i,j,k} + pb_{i,j,k} = 1, \qquad k=0,1,2. \tag{8}$$

So, the sum of decimal values of three decrypted LSB and three original LSB must be seven. The average energy of errors between the decrypted and original gray values is

$$EA=1/8.\sum_{u=0}^{7} [u-(7-u)]^2 =21 \tag{9}$$

As the probability of incorrect LSB decryption is 1/2, when reconstructing an image using the decrypted data, the value of PSNR in the decrypted image is approximately

$$PSNR = 10.\log 10\frac{255^2}{E_A/2} = 37.9 \text{ dB} \tag{10}$$

Then, the receiver will extract the embedded bits and recover the original content from the encrypted image. According to the data embedding key, he may segment the decrypted image into blocks and divide the pixels in each block into two sets in a same way. For each decrypted block, the receiver flips all the three LSB of pixels in S0 to form a new block, and flips all the three LSB of pixels in S1 to form another new block. We denote the two new blocks as H0 and H1. There must be that either H0 or H1 is the original block, and another one is more seriously interfered due to the LSB flip operation. For the two blocks sized by s x s, define a function to measure the fluctuation in them and denote the values of fluctuation function of H0 and H1 as f0 and f1, respectively. Because of spatial correlation in natural image, the fluctuation function of original block is generally lower than that of a seriously interfered version. So, the receiver can perform data extraction and image recovery by comparing f0 and f1. If f0 < f1, regard H0 as the original content of the block and let the extracted bit be 0. Otherwise, regard H1 as the original content of this block and extract a bit 1. Finally, concatenate the extracted bits to retrieve the additional message and collect the recovered blocks to form the original image [12].

III. CONCLUSION

In this paper, a narrative reversible data embedding system for encrypted image is proposed, which consists of image encryption, data embedding and data extraction/image recovery phases. The data of original image are entirely encrypted by using a public key. Although a data hider does not know the original content, he can embed additional data into the encrypted image by modifying a fraction of encrypted data. With an encrypted image containing embedded data, a receiver may firstly decrypt it using the private key, and the decrypted version is similar to the original image. According to the data embedding key, with the aid of spatial correlation in natural image, the embedded data can be correctly extracted while the original image can be perfectly recovered. Although someone with the knowledge of data embedding key can detect the presence of hidden data using LSB steganalytic methods, since he does not know the private key, it is still impossible to recover the original image. For ensuring the correct data extraction and the perfect image recovery, we may let the block side length be a big value, such as 32, or introduce error correction mechanism before data embedding to protect the additional data with a cost of payload reduction.

REFERENCES

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

[2] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "*Reversible data hiding*," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, 2006.

[3] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "*Lossless generalized LSB data embedding*," IEEE Trans. Image Process., vol. 14, no. 2, pp. 253–266, Feb. 2005.

[4] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "*Reversible image watermarking using interpolation technique*," IEEE Trans. Inf. Forensics Secure., vol. 5, no. 1, pp. 187–193, 2010.

[5] W. Hong, T. S. Chen, Y. P. Chang, and C. W. Shiu, "*A high capacity reversible data hiding scheme using orthogonal projection and prediction error modification*," Signal Process., vol. 90, pp. 2911–2922, 2010.

[6] C. C. Chang, C. C. Lin, and Y. H. Chen, "*Reversible data embedding scheme using differences between original and predicted pixel values*," Inform. Secure, vol. 2, no. 2, pp. 35–46, 2008.

[7] S. Lian, Z. Liu, Z. Ren, and H. Wang, "*Commutative encryption and watermarking in video compression*," IEEE Trans. Circuits Syst. Video Technol., vol. 17, no. 6, pp. 774–778, 2007.

[8] M. Cancellaro, F. Battisti, M. Carli, G. Boato, F. G. B. Natale, and A. Neri, "*A commutative digital image watermarking and encryption method in the tree structured haar transform domain*," Signal Process.: Image Commun., DOI 10.1016/j.image.2010.11.001, to be published.

[9] D. Kundur and K. Karthik, "*Video fingerprinting and encryption principles for digital rights management,*" Proc. IEEE, vol. 92, pp. 918–932, 2004.

[10] N. Memon and P. W. Wong, "*A buyer-seller watermarking protoco*l," IEEE Trans. Image Process., vol. 10, no. 4, pp. 643–649, Apr. 2001.

[11] M. Kuribayashi and H. Tanaka, "*Fingerprinting protocol for images based on additive homomorphic property*," IEEE Trans. Image Process., vol. 14, pp. 2129–2139, 2005.

[12] Xinpeng Zhang, "*Reversible Data Hiding in Encrypted Image*", IEEE Signal Processing Letters, vol.18, No.4, April 2011.