IOSR Journal of Engineering (IOSRJEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 09, Issue 5 (May. 2019), ||S (XI) || PP 37-41

"Efficient Data Hiding in Encrypted Images Using XOR-Permutation Method"

Reysh Chandrikapure[1], Mr. Pranjal Dhore[2] Nisha Balani[3]
Department. of Computer Science & Engineering, Jhulelal Institute of Technology, Nagpur, India

Abstract: This paper focuses on improvising the quality decrypted images by a classification and permutation method based on separable reversible data hiding in encrypted images. Security of pixel value and its location is the handled by classification permutation encryption in combination with the XOR-encryption. Bits are indiscriminately chosen based on the data-hiding key in the smooth set for embedding further data in the MSB – Most Significant Bit. This paper particularly shows the mid way implementation of the separable reversible data hiding method using XOR and Permutation in images.

Keywords: Separable-reversible data hiding, encrypted images, RDH-EI, classification permutation.

I. Introduction

Reversible data hiding (RDH), as the name suggests, is the method of data hiding into images and its reconstruction after the data has been extracted out of the image. Zhang, in 2011, proposed the Reversible data hiding in Encrypted images (RDH-EI) where extra data was hidden by flipping the 3 least significant bits of encrypted pixels. These encrypted pixels are a result of the bitwise XOR encryption. Parallel data extraction and image recovery in RDH-EI needs both encryption and data-hiding keys.

value of the original image for validation. The embedding formula had the original image and the marked image. The hash function is operated on the target image and the secret key to get watermarked image. Because of modulo-256 addition the over and/or underflow is prevented and the reversibility of data is achieved. Second reversible marking technique was developed in the transform domain by B. Macq and F. Deweyand. This is based on a lossless multi-resolution transform. It also uses modulo-256 addition.

Given RDH-EI is categorized into two parts; First, processing the original image which reserves space hiding the data and second, is embedding the data directly into the encrypted image. Zhang, in 2012, proposed a RDH- EI method where a doer encrypts the image by bitwise XOR operation with a key (encryption key) and a data hider that will constrict the least significant bits of the encrypted image to make some space for hiding the data. However, recovery of the original content without any error is complex is the data is greater than 004 bit per pixel.

II. Related Work

Chris W. Honsinger et.al invented Lossless recovery of original image holding embedded data which was carried outine the spatial domain. It uses modulo-256 addition where eight-bit greyscale images are considered to embed the hash J. Fridrichet.al told about a spatial domain technique that loss lessly compresses selected bit plane(s) to make some space for data embedding because the necessary book-keeping data are also embedded in the cover image as an overhead, the method is reversible.All the previous methods are focused on authentication and hence are able to hide lesser amount of data. M. Goljan et.al gave the first reversible marking technique that can work on larger amount data. This technique divides an image into non-overlapping blocks and then by using a function to categorize these blocks into the following three groups: Regular (R), Singular (S), and Unusable (U). It further performs a flipping operation, which can alter an R-block to an S-block and vice versa however U-block remains as it is even after the flipping operation. By assigning (binary) 1 to an R-block and (binary) 0 to an S-block, all R- and S-blocks are scanned in a selected chronological order, resulting in a biased sequence meaning that the binary numbers of 1 and 0 are imbalanced. This biased binary sequence is losslessly compressed to provide some space for data embedding and the compressed bit sequence is embedded into the image as an overhead for later reconstruction of the original image. In data embedding, the R-block and S-block are examined once again and the flipping operation is applied whenever essential to make the changed R-block and S-block sequence coincident with the data to be embedded followed by the overhead data mentioned above.

M.U.Celik et.al discussed a technique in which the host signal is quantized in the embedding phase and the residual is attained. Then the CALIC lossless image compression algorithm is accepted with the quantized values as side information to efficiently compress the quantization left over to produce high capacity for the payload data. The payload of this method ranges from 15 to 143 kb for a 512x512x8 gray scale image as the PSNR is 38dB. Here the PSNR is low.

Zhicheng Ni et.al gave a new reversible data embedding technique that makes use of the zero or the lowest point of the histogram and to some extent alters the pixel gray scale values to embed data. This technique can be applied virtually to all types of gray scale images which can embed a large quantity of data (5–80 kb for a 512x512x8). This method gives high quality for all natural images. The PSNR of the marked image vs. the original image is assured to be higher than 48 dB.

Using the zero or the minimum points of the histogram of an image, a reversible data hiding algorithm can recover the original image without any distortion from the marked image. It somewhat amends the pixel gray scale values to embed data into the image and it can embed extra data than many of the existing reversible data hiding algorithms. Experimentally, it is proved that the peak signal-to-noise ratio (PSNR) of the marked image produced by this method versus the original image is sure to be above 48 dB.

III. Current Implementation

Here, we would like to present the current implementation of our. As per the previous paper "SEPERABLE REVERSIBLE DATA HIDING USING XOR AND PERMUTATION ENCRYPTION IN IMAGES" there are 5 parts to this implementation out of which (A) image encryption and (B) data hiding are completed. (C) data extraction is in progress.

A. Image Encryption

We took an image *X* (original) with a size *MxN* pixels which is in uncompressed format; each pixel *xij* is represented by 8 bits with gray value falling into [0,255] and *1≤ i≤M; 1≤ j≤N*. Here, all the pixels in original image are initially classified into smooth and non-smooth pixels. In the encrypted image, classification permutation method is then premeditated for advancing the pixel-position privacy. Here, a binary matrix *T*

= *{tij |1≤ i ≤ M, 1≤ i ≤ N}* called the type-mark is used to represent the type of corresponding pixel in original image. Instead, *tij=0,* the corresponding pixel *xij* is smooth pixel in the original image; otherwise, the pixel *xij* is a non-smooth pixel. If the MSBs of all pixels in the *3×3* neighbourhood centred on this pixels *xij* are same for any non-boundary pixel *xij* in an original image *X*, we say the pixel *xij* is smooth; else it is considered as non smooth. It is,

=

$$t_{ij} = \begin{cases} 1 & , \quad i = 1, M \text{ or } j = 1, N \\ 1 & , \quad \text{MSBs of all pixels in } \Delta_{ij}^{X} \text{ are not same} \\ 0 & , \quad \text{MSBs of all pixels in } \Delta_{ij}^{X} \text{ are same} \end{cases} \quad (1)$$

Where, ^{x}ij is a 3×3 neighbourhood in the original image which is centred on the pixel *xij*. After pixel classification, the XOR-encrypted image is initially obtained by the XOR of the original bits along with the pseudo-random bits, generated according to the encryption key. The detailed XOR encryption method refers to Zhang's scheme. This step improves the privacy of the image content. Then the encrypted image is formed by scrambling the smooth and the non smooth pixels in the XOR-encrypted image in respective manner. The procedure of the proposed classification permutation method is described as given.

(1) According to the XOR-encrypted image and type mark, the smooth linear table *Ls* and non-smooth linear table *Ln* are generated by scanning order from top to bottom, left to right, respectively.

$$\begin{cases} L^s = \{l_m = x_{ij} | t_{ij} = 0, m| = 1,2, \dots, N_s\} \\ L^n = \{l_k = x_{ij} | t_{ij} = 1, k = 1,2, \dots, N_n\} \end{cases} \quad (2)$$

Where *Ns* and *Nn* depicts the number of smooth and non smooth pixels respectively, and *Ns+Nn=M×N.*

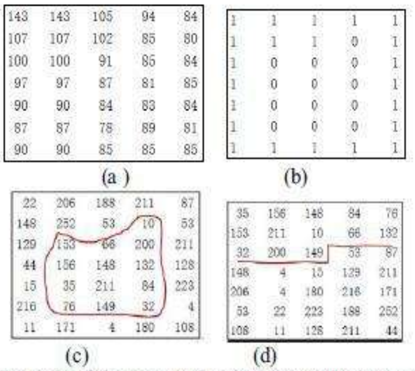
(2) Both linear tables *Ls* and *Ln* are randomly and respectively permuted based on the encryption key, given as *Ekey*(*Ls*) and *Ekey*(*Ln*), and it is connected to fabricate a linear table L.

(3) The encrypted image E is produced by scanning the linear table L into 2-dimensional with a size of $M \times N$. For understand the process of proposed encryption, Fig. 1 depicts an example of the proposed image encryption; Fig. 1 (a) is an

$$L = E_{key}(L^s) \,||E_{key}(L^n) \qquad (3)$$

Original image with a size of 5×7 pixels. According to (1), we can find the type-mark of Fig.1 (a), as given in Fig.1 (b), where the smooth and non-smooth pixels are counted as 13 and 22, respectively. The ratio of smooth pixels in the original image, α, is 13/35. Fig.1 (c) is the XOR-encrypted image, wherein the position of pixels in original image is the same. Comparison of Fig. 1(a) with Fig.1 (c) gives the value of pixels in the XOR-encrypted image differs from that of corresponding pixels in the original image. According to the type-mark in Fig.1 (b), it is known that the 13 pixels delineated by red line in Fig. 1(c) are the smooth pixels of the original image. Fig.1 (d) shows the encrypted image, where the first 13 pixels above the red line are the smooth pixels delineated by red line in Fig. 1(c). The back of the encrypted image shows non smooth pixels.

Note that the type-mark is also shared as part of the encryption key between the content-owner and receiver in the proposed encryption method. On the other hand, the data-hider does not need to know the type-mark including the ratio of smooth pixels. This makes the security of encrypted image to be improved and easy to find the smooth pixels in the encrypted image.



143	143	105	94	84
107	107	102	85	80
100	100	91	85	84
97	97	87	81	85
90	90	84	83	84
87	87	78	89	81
90	90	85	85	85

(a)

1	1	1	1	1
1	1	1	0	1
1	0	0	0	1
1	0	0	0	1
1	0	0	0	1
1	0	0	0	1
1	1	1	1	1

(b)

22	206	188	211	87
148	252	53	10	53
129	153	66	200	211
44	156	148	132	128
15	35	211	84	223
216	76	149	32	4
11	171	4	180	108

(c)

35	156	148	84	76
153	211	10	66	132
32	200	149	53	87
148	4	15	129	211
206	4	180	216	171
53	22	223	188	252
108	11	128	211	44

(d)

Fig.1 Example of the proposed encryption method (a) original image (b) type-mark (c) XOR-encrypted image and (d) encryption image

Additionally, we will be using a Master Base Record (MBR) table. An MBR is used for individual pixel value encryption i.e. we will be encrypting each selected pixel. This table is generated with respect to individual users; for each individual user his personal symmetric base encryption key is mentioned in the table and key will be generated as the image is encrypted. The table contains 256 values, one for each character.

B. Data Hiding

The data-hider, after receiving the encrypted image E embedded some additional data by modifying the MSB of small portion pixels of the encrypted image. Let D = {$dk|k$=1,2,…,Nd} be a binary additional data that is to be embedded in the encrypted image, where Nd indicates the number of bits in the additional data. Data-hider pseudo randomly picks Nd encrypted pixels by looking at the data-hiding key that is used to carry the additional data. For making the selected encrypted pixels smooth, one pseudorandom sequence with size of ⌊βMN⌋, which is denoted as R={$ri|i$=1,2,…, ⌊βMN⌋}, is initially created by the data-hiding key. Here ri is the real in interval [0,1], ⌊. ⌋ is the largest integer less than or equal to the given parameter. β is a parameter that is not more than the ratio of smooth pixels in the original image α. Finally the index sequence A is gained by sorting the real pseudorandom sequence R,

$$\mathrm{A} = \{\mathrm{a}_i | i = 1,2,\ldots,\mathrm{N_d}\}$$
$$such\ that\ r_{a_1} \le r_{a_2} \le \cdots \le r_{a_{N_d}} \qquad (4)$$

Evidently, αi is the integer in the given interval [1, ⌊βMN⌋] and the inequalities ai ≠ aj for ∀ i ≠ j. At the end, the *k*th (*k*=1,2,…,*Nd*) bit in the additional data *D*, i.e. *dk*, is embedded by modifying the MSB of the pixel eikjk in the encrypted image,

$$e_{i_k j_k} = 128 \times d_k + mod\left(e_{i_k j_k}, 128\right), \qquad (5)$$

Where, *mod* (,) indicates modulus after division, *ik* and *jk* are the row and column coordinates of the pixel and are computed with,

$$\begin{cases} i_k = \lfloor \mathrm{a}_k / N \rfloor + 1 \\ j_k = mod(\mathrm{a}_k, N) + 1 \end{cases}, k{=}1{,}2{,}\ldots{,}N_d \qquad (6)$$

A marked encrypted image is constructed when all the bits are embedded.

Here, the data encryption is performed using MD5 generated key. The MD5 message-digest algorithm is a widely used hash function for producing a 128-bit hash value (data hiding key in our case). A second level encryption with the user key will be applied on the previously encrypted data which will result as a double encrypted format resulting in a 2 layer encryption.

C. Image Decryption

As a part of decryption process data extraction steps will be to perform the two layer decryption (as we are using a two layer encryption) one with the MBR table value key and other with the user key.

D. Data Extraction

We have to reverse the hash function in this step in order to get the encrypted data. This is a user key decryption.

E. Image Recovery

Image recovered will the result of the above performed steps

IV. Results And Discussion

Encryption of the image along with the data hiding is done using XOR-encryption. In this method each pixel is embedded with 3 units of data (3 characters per pixel). Using Permutation along with the XOR method gives a complex and secure way of embedding data into encrypted images. A Master base record (MBR) consisting of 256 keys (ranging from 0-255) is used along with the user key for carrying out data hiding, resulting in unpredictable combination of keys thus improving the efficiency of the method.

V. Conclusion And Future Scope

A classification permutation based separable reversible data hiding in encrypted image is put forward in this paper. In the image encryption phase, this work gives a classification permutation encryption with a combination of the XOR-encryption to get better privacy of encrypted images. Additionally, it is possible for the data hider to get the smooth pixels in the encrypted image without the original content and the encryption key together with the type-mark. This results in improved quality of decrypted images and recovered ones too.

References

[1]. F. Huang, J. Huang and Y. Shi, “New framework for reversible data hiding in encrypted domain,” *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 12, pp. 2777-2789, Dec. 2016.

[2]. X. Zhang, “Reversible data hiding in encrypted image,” IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255-258, April. 2011.

[3]. 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, April. 2012.

[4]. X. Liao and C. Shu. “Reversible data hiding in encrypted images based on absolute mean difference of neighbouring pixels,” J. Vis. Commun. Image Represent., vol. 28, pp. 21-27, 2015.

[5]. C. Qian and X. Zhang. "Effective reversible data hiding in encrypted image with privacy protection for image content," J. Vis. Commun. Image Represent., vol. 31, pp. 154-164, 2015.
Vol.**6**(**9**), 04/2019, E-ISSN: **2347-2693**

[6]. K. Ma, W. Zhang, X. Zhao, et al., "Reversible data hiding in encrypted images by reserving room before encryption," IEEE Trans. Inf. Forensics Security, vol. 8, no. 3, pp. 553-562, Mar. 2013.

[7]. D. Xu and R. Wang, "Separable and error-free reversible data hiding in encrypted images," Signal Process. vol. 123, pp. 9-21, 2016.

[8]. T. Nguyen, C. Chang and W. Chang, "High capacity reversible data hiding scheme for encrypted images,"Signal Process.: Image Commun., vol. 44, pp. 84-91, 2016.

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

[10]. X. Wu andW. Sun, "High-capacity reversible data hiding in encrypted images by prediction error," Signal Process., vol. 104, pp. 387-400, 2014.

[11]. F. Chen, H. He, Y. Huo, "Self-embedding watermarking scheme against JPEG compression with superior imperceptibility", Multimed Tools Appl., DOI 10.1007/s 11042-016-3574-0, 2016.

[12]. Z. Ni, Y.Q. Shi, N. Ansari, and W.Su, Reversible data hiding, IEEE Transactions on Circuits and Systems for Video Technology, Vol 16, no.3, Mar 2006, pp 354-362.

[13]. Shilpa Sreekumar and Vincy Salam, Advanced Reversible Data Hiding with Encrypted Data, IJETT, Vol.13, no.7, Jul 2014, pp 310-313.

[14]. VinitAgham and TareekPattewar, A Survey on Separable Reversible Data Hiding Technique, IMACST, Vol.4, no.1, May 2013, pp 9-13.

[15]. M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, Reversible data hiding, International Conference on Image Processing, ISSN 1522-4880, Vol.2, Oct.2002, pp 157-160.

[16]. C. W. Honsinger, P. Jones, M. Rabbani, and J. C.Stoffel, "Lossless Recoveryof an Original Image Containing Embedded Data," U.S. Patent6 278 791 B1, Aug. 21, 2001.

[17]. B. Macq and F. Deweyand, "Trusted headers for medical images" DFG VIII-D II Watermarking Workshop, Erlangen, Germany,Oct. 1999.

[18]. J. Fridrich, M. Goljan, and R. Du, "Invertible authentication,"Proc.SPIESecurityWatermarking

[19]. International Journal of Computer Sciences and Engineering Vol.**6**(**9**), 04/2019, E-ISSN: **2347-2693**

[20]. Multimedia Contents, San Jose, CA, Jan.2001, pp. 197– 208.

[21]. M. Goljan, J. Fridrich, and R. Du, Distortion-free data embedding,Proc. 4th Inf. Hiding Workshop, Pittsburgh, PA, Apr. 2001, pp. 27–41.

[22]. A. R. Calderbank, I. Daubechies, W. Sweldens, and B. Yeo, Wavelettransforms that map integers to integers,Appl. Comput. HarmonicAnal., vol. 5, no. 3, pp. 332–369, 1998.

[redacted] Computer Science Engineering from Rashtrasant Tukadoji Maharaj Nagpur University in 2017 and is currently a Master of Technology scholar from Rashtrasant Tukadoji
Maharaj Nagpur University. His main research work focuses on Different Encryption Stratergies.