© 2014 IJIRT | Volume 1 Issue 7 | ISSN: 2349-6002

Privacy preserving requirements of Data management using Reserving Room Technique

M.Nirosha, D.Naveen Kumar
Department of CSE, SR Engineering College, Warangal, Telangana, India

Abstract- **Digital Image and information embedding systems have a number of important multimedia applications. These systems embed one signal, sometime called an "embedded signal" or "information" within another signal, called as "Host Signal". In recent times, more and more awareness is paid to reversible data hiding (RDH) in encrypted images. Reason being, it maintains the superlative property that the original cover can be lossless recovered after embedded data is extracted while shielding the image content's privacy. All earlier methods embed data by reversibly vacating room from the encrypted images. However, this may be subject to some slip-up on data extraction and/or image restoration. In this paper, we put forward a narrative method by reserving room before encryption with a conventional RDH algorithm. Hence, it is trouble-free for the data hider to reversibly embed data in the encrypted image. The projected technique can pull off real reversibility, that is, data extraction and image recovery are free of any error. We also develop a framework in which the performance of an information embedding method may be characterized based on its achievable rate-distortion-robustness trade-offs and discuss how previously proposed data hiding algorithms fit into this framework.**

Index Terms- **Image Encryption, Reversible data hiding, Image and data Recovery**

I. INTRODUCTION

Reversible data hiding (RDH) is a technique in image processing area for encryption, by which the original cover can be losslessly recovered after the embedded message is extracted. The RDH approach is widely used in medical science, defense field and forensic lab, where there is no degradation of the original content is allowed. Since more research RDH method in recently. In theoretical aspectrate-distortion model for RDH Kalker and Willems[1], through which they proved the rate-distortion bounds of RDH for memoryless covers and proposed a recursive code construction which, however, does not approach the bound. 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. [5] constructed a general framework for RDH. By first extracting compressible features of original cover and then compressing them lossless, spare space can be saved for embedding auxiliary data. A more popular method is based on difference expansion (DE) [6], 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) [7], in which space is saved for data embedding by shifting the bins of histogram of gray values. The state-of-art methods [8]–[12] 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 is an effective and popular means as it converts the original and meaningful content to incomprehensible one. In advocated a 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.

Obviously, the cloud service provider has no right to introduce permanent distortion during data coloring into encrypted data. Thus, a reversible data coloring technique based on encrypted data is preferred. Suppose a medical image database is stored in a data center, and a server in the data center can embed notations into an encrypted version of a medical image through a RDH technique. With the notations, the server can manage the image or verify its integrity without having the knowledge of the original content, and thus the patient's privacy is protected. On the other hand, a doctor, having the cryptographic key, can decrypt and restore the image in a reversible manner for the purpose of further diagnosing.

II. REVERSIBLE DATA HIDING (RDH)

Reversible data hiding (RDH) in images is a technique, by which the original cover can be lossless recovered after the embedded message is extracted. This important technique is widely used in medical imagery, military imagery and law forensics, where no distortion of the original cover is allowed. Since first introduced, RDH has attracted considerable research interest. In theoretical aspect, and 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 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 to construct a general framework for RDH. By first extracting compressible features of original cover and then compressing them lossless, spare space can be saved for embedding auxiliary data. A more popular method is based on difference expansion (DE), 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), in which space is saved for data embedding by shifting the bins of histogram of gray values. The state-of-art methods 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 is an effective and popular means as it converts the original and meaningful content to incomprehensible one. Although few RDH techniques in encrypted images have been published yet, there are some promising applications if RDH can be applied to encrypted images. Obviously, the cloud service provider has no right to introduce permanent distortion during data coloring into encrypted data. Thus, a reversible data coloring technique based on encrypted data is preferred. Suppose a medical image database is stored in a data center, and a server in the data center can embed notations into an encrypted version of a medical image through a RDH technique. With the notations, the server can manage the image or verify its integrity without having the knowledge of the original content, and thus the patient's privacy is protected. On the other hand, a doctor, having the cryptographic key, can decrypt and restore the image in a reversible manner for the purpose of further diagnosing. Some attempts on RDH in encrypted images have been made divided the encrypted image into several blocks. By flipping 3 LSBs of the half of pixels in each block, room can be 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. ameliorated 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. These two methods mentioned above rely on spatial correlation of original image to extract data. That is, the encrypted image should be decrypted first before data extraction

There are few technique by which we are vacating the room after encryption.

1. Fridich et al [4] constructed a general framework for RDH for vacating room in encrypted image. By first extracting compressible features of original image and then compressing them losslessly. In this way space can be created for embedding data.

2. Another method is based on difference expansion (DE) [3], for vacating room in encrypted image 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 the space created can be used for embedding data.

3. Another method is histogram shift (HS) [4], for vacating room in encrypted image in which space is saved for data embedding by shifting the bins of histogram of gray values. and the space created can be used for embedding data.

In all methods of [5]–[7], the encrypted 8-bit gray-scale images are generated by encrypting every bit-planes with a stream cipher. The method in [5] segments the encrypted image into a number of nonoverlapping blocks size x*x each block is used to carry one additional bit.

III. PROPOSED SCHEME

As we know lossless vacating rooms from the encrypted images is comparatively intricate and at times unproductive, why are we still so fanatical to discover novel RDH techniques running directly for encrypted images? Imagine 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 guide us to the novel framework, "reserving room before encryption (RRBE)".

As shown in Fig. 1(b), the content owner first reserves adequate space on original image. Then translate the image into its encrypted version with the encryption key. Now, the data embed-ding process in encrypted images is essentially reversible. Data hider only needs to have room for data into the spare space previous emptied out. The data extraction and image recovery are indistinguishable to that of Framework VRAE. Noticeably, standard RDH algorithms are the best operator for reserving room before encryption. This can be effortlessly applied to Framework RRBE to realize better performance compared with techniques from Framework VRAE. Reason is, in this new framework, we pursue the customary idea that first lossless compresses the unneeded image content (e.g., using excellent RDH techniques) and then encrypts it with respect to protecting privacy. In the proposed method (Fig 1(b)),

1. We first empty out room by embedding LSBs of some pixels into other pixels with a traditional RDH method.

2. Then encrypt the image, so the positions of these LSBs in the encrypted image can be used to embed data.

This proposed method does below:-

1. Separate data extraction from image decryption
2. Achieves excellent performance in two different prospects:

a. Real reversibility is realized, that is, data extraction and image recovery are free of any error.

b. 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 significantly enlarged.

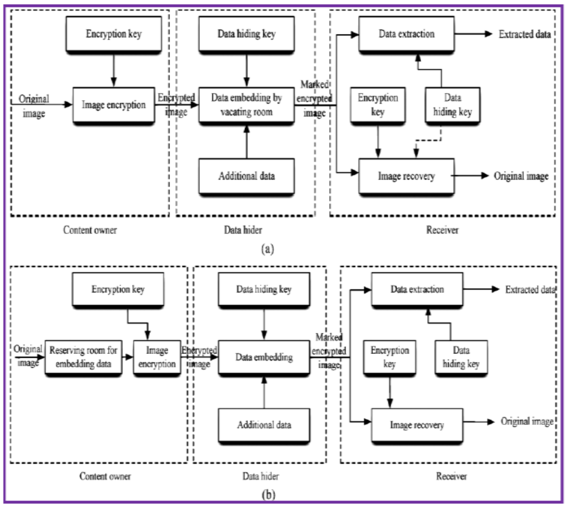

Fig.1. Framework: "vacating room after encryption (VRAE)" versus framework: "reserving room before encryption (RRBE)." (Dashed line in (a) states that the need of data hiding key in image recovery varies in different practical methods) (a) Framework VRAE. (b) Framework RRBE.

A. Image Encryption

Assuming original color image size is N1*N2 and each pixel of Red, green, blue value falling into [0,255] is represented by 8 bits. Denote each bits of a pixel represented as bj,k,0 ,bj,k,1 , . . .bj,k,7 where 1≤j≤N1 and 1≤k≤N2, and the rgb value as qj, k. Denote the other number of pixels as N(N=N1XN2).

$$\mathrm{Bj,k,a} = \left[\mathrm{qj,k,}\frac{\mathrm{a}}{2\mathrm{a}}\right] \mathrm{mod2, a} = 0{,}1, \ldots..7$$
(1)

$$\text{and } \mathrm{qj,k} =]2\mathrm{a}$$
(2)

$$\mathrm{Bj,k,a} = \mathrm{bj,k,a} + \mathrm{rj,k,a}$$
(3)

In encryption phase original bits and pseudo-random bits are calculated by exclusive-or. Where rj, k, a are determined by an encryption key using a standard stream cipher.

B. Data Embedding

In the data embedding, some parameters D, H, R are embedded into a small number of encrypted pixels, and the other encrypted pixels of LSB are compressed to creating a sparse space for accommodating the additional data. The detailed procedure is as follows. After encrypting the original color image content owner pseudo-randomly selects Nt encrypted pixels according to a data hiding key that will be used to carry the parameters (D, H, R) for data hiding. Here, Nt is a small positive integer. The other N-Nt encrypted pixels are pseudo-randomly permuted and divided into a number of groups using data hiding key, each group contains no of pixels which is denoted as H. Collect the D least significant bits of the H pixels in each group, which is denoted by B(g,1),B(g,2),.......B(g, D, H) where g is a group index within [1,(N-Nt)/H] and D is a positive integer less than 5. Here, S is a small positive integer

The content owner generates a M matrix which has two parts by (4).

$$\mathrm{M} = [\mathrm{ID.H} - \mathrm{R\ F}]$$
(4)

Where ID.H-R is an identity matrix ID.H-R = (D.H-R) X (D.H-R) and F = (D.H-R) XR which is derived from the data-hiding key. Then, the parameters D, H, and R embedded into the LSB of

Nt. For example if Nt=16 the values of D,H and R are represented as 2, 12 and 2 bits respectively, and Nt LSB encrypted pixels replaced by 16 bits.

In following, total bits made up of Nt and (N-Nt).R/H-Nt additional bits will be embedded into the pixel groups. For each group, calculate =

$$B'\ (g,1)\quad = F\ B'(g,1)$$

$$B'\big(g.D.\ddot{H} - R\big)\ B'\ (g.D.H)$$

(5)

Which is determined by modulo-2?

For data accommodation compress the bits of B (g, 1), B (g, 2),B (g, D. H) as (D.H-R) bits. In each group, the original LSB of selected encrypted pixels and the additional data to be embedded as [B' (g, D.H-R+1), B' (g, D.H-R+2).....B' (g. D. H)] then, replace the new [B'(g,1),B'(g,2)....B'(g, D. H)], with B(g,1),B(g,2),.......B(g, D. H) and put into their original positions by reversible manner. At the same time, the most significant bits (MSB) of encrypted pixels are kept unchanged. Since bits are embedded into each pixel-group, the total (N-Nt).R/H bits can be accommodated in all groups. Figure 3 shows the original input image and figure 4 shows the result of encrypted image containing embedded data.

C. Data Extraction and Image Recover

In this phase, there are three options at the receiver side;

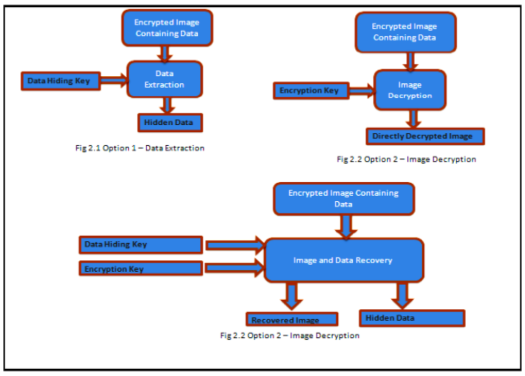

Fig.2. Three options in Receiver Side

These three options are shown in fig. 2.

a) If the receiver has only data hiding key, receiver can extract the data and does not know about the original content.

b) If the receiver has only encryption key, receiver can decrypt the image and does not know about the hidden data.

c) If the receiver has both encryption and data hiding key, receiver can extract the data and also recover the original content.

(a)In first option, with an encrypted image containing embedded data, receiver may first obtain the values of the parameters D, H and R from the LSB of the Nt selected encrypted pixels. Then, the receiver permutes and divides the other (N-Nt) pixels into (N-Nt)/R groups and extracts the R embedded bits from the D LSB-planes of each group. When having the total (N-Nt) R/H extracted bits, the receiver can divide them into Nt original LSB of selected encrypted pixels and (N-Nt) R/H-Nt additional bits.

Note that because of the pseudo-random pixel selection and permutation, any attacker without the data-hiding key cannot obtain the parameter values and the pixel-groups, therefore cannot extract the embedded data. Furthermore, if the receiver having the data-hiding key can successfully extract the embedded data, receiver cannot get any information about the original image content.

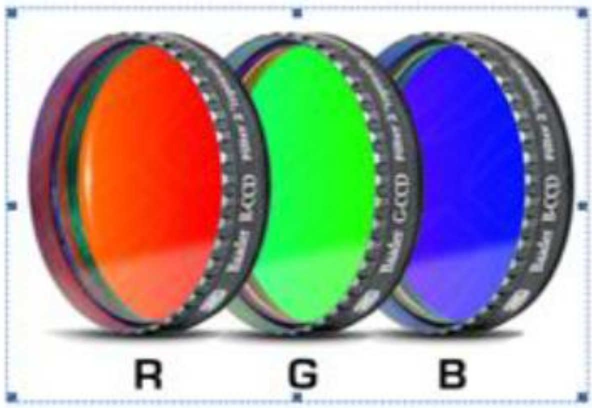

Fig. 3.original image

Fig. 4.Encrypted image containing data

(b) In second option, if the receiver has the encryption key but does not know the data-hiding key. Clearly, receiver cannot obtain the parameter values therefore cannot extract the embedded data. However, the original image content can be roughly recovered using encryption key. Denoting the bits of pixels in the encrypted image containing embedded data as B'j, k, 0, B'j, k, 1........ B'j, k, 7 the receiver can decrypt the data

$$b'_{j,k,a} = B'_{j,k,a} + r_{i,k,a} \quad \textbf{(6)}$$

The rgb values of decrypted pixels are

$$P'_{j,k} \ \sum_{a=0}^{7} [bj, k, a]\ 2^{a} \quad \textbf{(7)}$$

Fig.5 shows the result of directly decrypted image using decryption key.

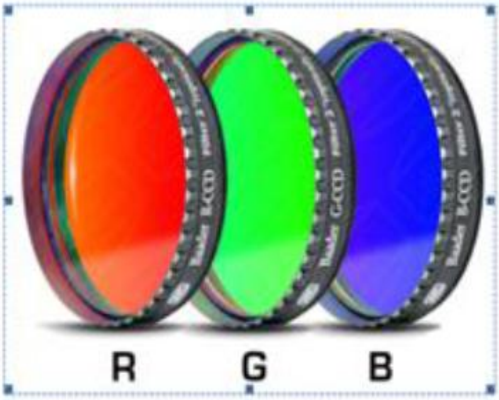

Fig.5.Directly decrypted image

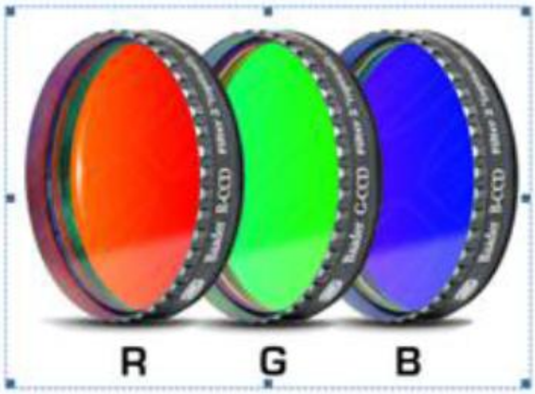

Fig.6.Decrypted image

The decrypted MSB must be same as the original MSB. Since the data-embedding operation does not alter any MSB of encrypted image. So, the content of decrypted image is similar to that of original image. If B (g.D.H-R+1) =B (g.D.H-R+2) =......=B (g. D.H-R) =0 there is

$$B'(g.x) = B(g.x). x = 1, 2, \ldots\ldots D.H - R \ldots\ldots (8)$$

The probability of this case is 1/2R and the original bits in D LSB-planes can be decrypted correctly. Since R is significantly less than D.H.

The distortion energy per each decrypted pixel is

$$D_E = 2^{-2D,} \sum_{\alpha=0}^{2D-H} (\alpha - \beta)\ 2 \ \ldots. \quad \textbf{(9)}$$

The average energy of distortion is

$$A_E = \frac{(2R-1)}{2R}, 2^{-2D,} \sum_{\alpha=0}^{2D-H} \sum_{\beta=0}^{2D-H} (\alpha - \beta)^2 \ \ldots. \ (1\ \textbf{0})$$

Here, the distortion in the selected pixels Nt is also ignored since their number is significantly less than the image size. So, the value of PSNR in the directly decrypted image is

$$PSNR = 10\log_{10}(A_E) \quad \ldots\ldots \quad \textbf{(11)}$$

In third option, if the receiver has both the data-hiding and the encryption keys, receiver may aim to extract the embedded data and recover the original content. According to the data-hiding key, the values of D, H and R, and the (N-Nt) R/H-Nt additional bits can be extracted from the encrypted image containing embedded data. By putting the Nt LSB into their original positions, the encrypted data of the Nt selected pixels are retrieved, and their original rgb values can be correctly decrypted using the encryption keys. In the following, we will recover the original rgb values of the other pixels. Fig.6 shows the result of decrypted image after extracting the hidden data which is similar to original image.

Table 1
Gives the theoretical values of PSNR with respect to D and R.

	R=1	R=2	R=3	R=4
D=1	56.0	54.2	51.7	51.4
D=2	49.2	47.1	44.7	44.3
D=3	40.9	39.1	38.5	38.2

Denoting the decrypted pixel group index as F_g and calculate the total difference between the decrypted and estimated rgb values in the group

$$D_i = \sum_{(j,k)\ \in F(g)} [t(j,k) - q^{\wedge}(j,k)] \quad \ldots\ldots \quad \textbf{(12)}$$

Where the estimated rgb values are generated from the neighbors in the directly decrypted image clearly, the estimated rgb values are only dependent on the MSB of neighbor pixels. Thus, let have 2R different Di corresponding to the 2R decrypted pixel-group Fg. Among the 2R decrypted pixel-group, there must be one that is just the original rgb values and possesses a low Di because of the spatial correlation in natural image. To keep a low computation complexity, let R be less than ten and use only the four neighboring pixels to calculate the estimated values.

IV. CONCLUSION

Reversible data hiding in encrypted images is a new topic which is attracting more interest because of the privacy-preserving requirements from cloud data management. Earlier methods implement RDH in encrypted images by vacating room after encryption, as opposed to which we proposed by reserving room before encryption. In new method data hider can benefit from the extra space emptied out in previous stage to make data hiding process effortless. The proposed method can take:-

1. Assistance of all traditional RDH techniques for plain images.

2. Achieve excellent performance without loss of perfect secrecy.

3. Furthermore, this novel method can achieve real reversibility, separate data extraction and greatly improvement on the quality of marked decrypted images.

V. REFERENCES

[1] Priya Kumar Jambhulkar, "Secure Reversible Data Hiding in Encrypted Images by Allocating Memory before Encryption via Security keys", International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 3, Issue 2, March 2014.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

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

[7] 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.

[8] 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.

[9] 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.

[10] 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.

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

[12] 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.

BIO DATA

Author 1

M.Nirosha currently pursuing her M.Tech, in SR Engineering College, Warangal, Telangana, India

Author 2

D.Naveen Kumar working as Asst.Prof Department of CSE, SR Engineering College, Warangal, Telangana, India