SECURE REVERSIBLE DATA HIDING IN ENCRYPTED IMAGE BY REVERSIBLE IMAGE TRANSFORMATION (RIT)

Dr.B.R.VIKRAM (M.E,Ph.D,LMISTE,MIEEE) 1

CHILUKA.SRIKANTH (MTECH) 2

1 PRINCIPAl, Vijay Rural Engineering College, Nizamabad, Telangana, 503003, INDIA

2 Department of ECE, Vijay Rural Engineering College, Nizamabad, Telangana, 503003, INDIA

vikramom@gmail.com[1] srikanth3693@gmail.com[2]

Abstract

With the popularity of outsourcing facts to the cloud, it's far vital to shield the privacy of facts and enable the cloud server to effortlessly control the data at the same time. Under such needs, reversible statistics hiding in encrypted photos (RDH-EI) attracts increasingly more researchers' attention. In this paper, we suggest a singular framework for RDH-EI primarily based on reversible photo transformation (RIT). Different from all previous encryption based totally frameworks, wherein the cipher-texts may additionally appeal to the notation of the curious cloud, RIT-primarily based framework allows the consumer to transform the content material of authentic image into the content of every other goal photo with the identical size. The converted picture, that seems like the goal photo, is used because the "encrypted image", and is outsourced to the cloud. Therefore, the cloud server can easily embed statistics into the "encrypted picture" through any RDH strategies for plaintext snap shots. And thus a patron-unfastened scheme for RDH-EI may be realized, that is, the records embedding manner carried out with the aid of the cloud server is inappropriate with the procedures of both encryption and decryption. Two RDH methods, such as conventional RDH scheme and unified embedding and scrambling scheme, are adopted to embed watermark in the encrypted image, which could satisfy unique desires on photograph first-class and big embedding capability respectively.

Index Terms—reversible data hiding, image encryption, reversible image transformation, privacy protection, out soured storage in cloud.

1. INTRODUCTION

The expanse of digital images has increased quickly on the Internet. Image security becomes gradually important for many applications, e.g., confidential transmission, video investigation, army and medical uses. For example, the requirement of fast and secure diagnosis is vital in the medical world. Nowadays, the transmission of images is a daily uses routine and it is necessary to find an effective way to transmit them over systems. To decrease the communication time, the data compression is necessary.

The protection of this multimedia data can be done with encryption or data hiding processes. Since few years, a difficult is to try to combine compression, encryption and data hiding in a single step. For example, some solutions were suggested in to association image encryption and compression. Two main sets of technologies have been developed for this purpose. The main based on contented security through encryption. There are some methods to encrypt binary images or gray level images. The next group bases the protection on data hiding, designed at secretly embedding a message into the data. Nowadays, a new task consists to embed data in encrypted images.

Previous work recommended inserting data in a converted image by using an irrevocable process of data hiding or data hiding, intended at secretly embedding a message into the data. Unique data is to apply revocable data hiding processes on converted images by desiring to remove the embedded data before the image decryption. Newest irreversible data hiding techniques have been suggested with large

size, but these approaches are not applicable on encrypted images. Data security basically means protection of data from illegal users or hackers and providing high security to check data medication. This area of data security has gained more attention over the recent period of time due to the huge increase in data transmission rate over the network.

In order to recover the security types in data transfers over the internet, many techniques have been developed like: Cryptography, Steganography. While Cryptography is a technique to secrete information by encrypting it to cipher texts and transmitting it to the intended receiver using an unknown key, Steganography provides further security by hiding the cipher text into a seemingly invisible image. On the other hand, cloud service for outsourced storage makes it challenging to protect the privacy of image contents. For instance, recently many private photos of Hollywood actress leaked from iCloud. Although RDH is helpful for managing the outsourced images, it cannot protect the image content. Encryption is the most popular technique for protecting privacy. So it is interesting to implement RDH in encrypted images (RDHEI), by which the cloud server can reversibly embed data into the image but cannot get any knowledge about the image contents.

Inspired by the needs of privacy protection, many methods have been presented to extend RDH methods to encryption domain. From the viewpoint of compression, these methods on RDH-EI belong to the next two frameworks: Framework I "vacating room after encryption (VRAE)" and Framework II "reserving room before encryption (RRBE)." In the framework 'VRAE," the cloud server inserts data by lossless vacating room from the encrypted images by using the idea of compressing encrypted images Compression of encrypted data can be communicated as source coding with side information at the decoder.

Usually the side information is the correlation of plaintexts that is exploited for decompression by the decoder. In divided the encrypted image into several blocks. By flipping 3 LSBs (least significant bits) of the half of pixels in each block, room can be vacated for the embedded bit. The data extraction and image retrieval proceed by finding which part has been reversed in one block. This process can be realized with the help of spatial association in the decrypted image. The cryptographer side by more misusing the spatial association using a dissimilar evaluation balance and side match system. For both methods in decrypting image and extracting data must be jointly executed. Recently proposed a novel RDH-EI method for joint decryption and extraction, in which the correlation of plaintexts is further exploited by distinguishing the encrypted and non-encrypted pixel blocks.

The rest of the paper is organized as follows. In Section II, we compare the RIT-based framework with previous frameworks and summarize the main contributions of the novel framework. A method of RIT is elaborated in Section III, and two kinds of RDH methods on transformed images are proposed in Section IV. The paper is concluded with a discussion in Section V.

2. LITERATURE SURVEY

[1] K. Hwang, D. Li: Cloud computing enables a new business model that supports ondemand, pay-for-use, and economies-of-scale IT services over the Internet. The Internet cloud works as a service factory built around virtualized data centers. 1 Cloud platforms are dynamically built through virtualization with provisioned hardware, software, networks, and datasets. The idea is to migrate desktop computing to a serviceoriented platform using virtual server clusters at data centers. However, a lack of trust between cloud users and providers has hindered the universal acceptance of clouds as outsourced computing services. To promote multi-tenancy, we must design the cloud ecosystem to be secure, trustworthy, and dependable.

[2] F. Bao, R. H. Deng, B. C. Ooi, et al: It is accepted that digital watermarking is quite relevant in medical imaging. However, due to the special nature of clinical practice, it is often required that watermarking not introduces irreversible distortions to medical images. The electronic clinical atlas has such a need of "lossless" watermarking. We present two tailored reversible watermarking schemes for the clinical atlas by exploiting its inherent characteristics.

We have implemented the schemes and our experimental results look very promising.

3. PROPOSED METHOD

In this section, we propose a method of RIT to encrypt spatial images, which is inspired by the technique of image transformation proposed by Lee et al. [26]. Lee et al.'s method can transform the original image to a freely-selected target image with the same size, yielding a secret-fragment-visible mosaic image defined in [25]. But the original image cannot be restored in a lossless way. It is not reversible, so it is not suitable for the scenario of RDH-EI. We will modify Lee et al.'s method to be reversible and obtain an encrypted image which looks like the target image.

For color images, we transform the color channel R, G, and B respectively in the same manner. So we just take gray images (one channel) as an example to describe the method. For an original image I, we randomly select a target image J having the same size with I from an image database.

Firstly, we divide the original image I and the target image J into N non-overlapping blocks respectively, and then pair the blocks of I and J as a sequence such that (B1, T1), . . . ,(BN , TN), where Bi is an original block of I and Ti is the corresponding target block of J, $1 \leq i \leq N$. We will transform Bi toward Ti and generate a Ti′ similar to Ti. After that, we replace each Ti with Ti′ in the target image J to get the transformed image J′. Finally we embed some accessorial information into J′ with an RDH method and generate the ultimate "encrypted image" E(I). This accessorial information is necessary for recovering I from J′. Before being embedded, these accessorial information will be compressed and encrypted with a key K shared with the receiver, so only a receiver having K can decrypt E(I).

The proposed transformation process consists of three steps: block pairing, block transformation and accessorial information embedding. We will mainly elaborate the first two steps in the subsections and the third step can be implemented by any traditional RDH method.

A. Block Pairing

To make the transformed image J′ look like target image J, we hope, after transformation, each transformed block will have close mean and standard deviation (SD) with the target block. So we first compute the mean and SD of each block of I and J respectively. Let a block B be a set of pixels such that B = {p1, p2, · · · ,pn}, and then the mean and SD of this block is calculated as follows.

$$u = \frac{1}{n}\sum_{i=1}^{n} p_i \quad (1)$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - u)^2} \quad (2)$$

When matching blocks between original image and target image, we hope two blocks with closest SDs to be a pair. In Lee et al.'s method, the blocks of original image and target image are sorted in ascending order according to their SDs respectively, and then each original block is paired up with a corresponding target tile in turn according to the order. To recover the original image from the transformed image, the positions of the original blocks should be recorded and embedded into the transformed image with an RDH method. If the image is divided into N blocks, N⌈log N⌉ bits are needed to record block indexes. Obviously, the smaller the block size is, the better the quality of transformed image will be, but which will result in a large N. Therefore, the amount of information used to record the index for each block may be so large that it will cause much distortion when embedding this information into the transformed image. In fact there may not exist enough redundant space to store this additional information. For instance, if we divide a 1024 × 1024 image into 4 × 4 blocks, 216 × 16 bits are needed to record the positions of blocks.

To compress the block indexes, we first classify the blocks according to their SD values before pairing them up. In fact, we found that the SD values of most blocks concentrate in a small range close to zero and the frequency quickly drops down with the increase of the SD value as displayed in Fig. 2, which is depicted from various sizes of 10000 images from the

BossBase image database [27]. Therefore, we divide the blocks into two classes with unequal proportions: class 0 for blocks with smaller SDs, and class 1 for blocks with larger SDs, and pair up the blocks belonging to the same class. By assigning the majority of blocks to the class 0, we can avoid the large deviation of SDs between a pair of blocks and efficiently compress the indexes at the same time.

In the present paper, we propose to divide both the original and target images into non-overlapping 4 × 4 blocks and calculate the SDs of each block. We first divide the blocks of original image I into 2 classes according to the quantile of SDs. Denote that the %α quantile of SDs by Nα. We assign the blocks with SDs ∈ [0,N_α] to "Class 0", and blocks with SDs ∈ (N_α], N100] to "Class 1". And then we will scan the blocks in the raster order, i.e., from left to right and from top to bottom, and assign a class label, 0 or 1, to each block.

Next, we label the blocks of target image based on the classes' volumes of original image. Assuming that the ith class in the original image includes n_i blocks for i = 0 or 1, we scan the target image in the raster order, and label the first n0 blocks with the smallest SDs as Class 0, and the rest n1 blocks as Class 1. As a result, each class in the target image includes the same number of blocks as the corresponding class in the original image. We pair the original block up with target block in the following manner. Scan the original image and target image in raster order respectively and pair the jth block of the class i in the original image up with the jth block of the class i in the target image for i = 0, 1 and j = 1, . . . , n_i. A simple example on the proposed block pairing method is shown in Fig. 3, in which the image only consists of 10 blocks. By setting α = 70, we assign 7 blocks with smallest SDs into class 0, and the rest 3 blocks into class 1 in the original image. In the target image, although the 8th and 9th block have the same SD value 5, the 8th block is assigned to class 0 but the 9th block is assigned to the class 1, because class 0 can only include 7 blocks as determined by the class 0 of the original image. After labeling the class indexes, we get a class index table (CIT) for original image and target image respectively, which will be helpful for understanding the procedure of block pairing.

According to the pairing rule, the first block of the original image is paired up with the forth block of the target image, because both of them is the first block of class 1 as shown in the CIT; the second block of original image is paired up with the ninth block of target image, because both of them is the second block of class 1, and so on. The pairing result is listed in Table I, which can be generated according to the CIT of original image and the CIT of the target image.

For each pair of blocks (B, T), as we will see in the next section, the original block B will be transformed to target block T by mean shifting and block rotation, yielding T ′. By replacing each T with T ′ in the target image, the sender will generate the transformed image. Note that both operations of mean shifting and block rotation will not change the SD value, so T ′ has the same SD as B. Therefore, the SDs in transformed image is only a permutation of those in original image. When classifying the blocks of transformed image according to %α quantile of SDs, the receiver can get a CIT that is same with the CIT of target image as shown in Fig. (b) and Fig. (c) in Fig. 3.

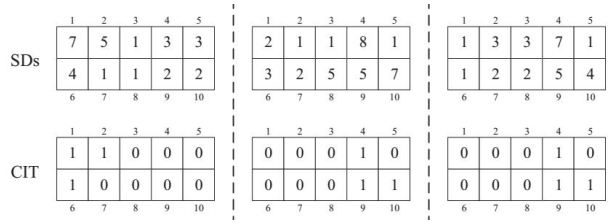
Therefore, to restore the original image from the transformed image, the receiver only needs to know the CIT of the original image. In fact, by CIT of original image and the CIT of transformed image (which is also the CIT of target image), the receiver can reconstruct Table I perfectly. Then according to the table he will know how to rearrange the transformed blocks to restore the original blocks. In the example of Fig. 3, the first block of the transformed image should be put back to position 3, and the second block should be put back to position 4 as indicated in Table I.

Note that CIT can be efficiently compressed because the ratio of 0 and 1 is bias. If the image is divided into N blocks, and these blocks are divided into two classes with %α quantile of SDs, we need N · H(α/100) bits to record S, where H is the binary entropy function. For instance, if we set α = 75 and divide a 1024 × 1024 image into 4 × 4 blocks, we only need 216 × H(0.75) ≈ 216 × 0.81 bits to record the positions of blocks, which is much less than 216 × 16 bits used by the method in [26]. The compressed

CIT will be encrypted and embedded into the transformed image as a part of accessorial information (AI).

B. Block Transformation

By the block pairing method described above, in each pair (B, T), the two blocks have close SD values. Therefore, when



(a) Original Image (b) Target Image (c) Transformed Image

Fig.1 An example of block pairing.

transforming B towards T , we only need a mean shifting transformation that is reversible. However, the transformation used in Lee et al.'s method [26] is not reversible because it changes the mean and SD at the same time.

Let the original block B = $\{p_1, p_2 \dots p_n\}$, and the corresponding target block T = $\{p_1', p_2' \dots p_n'\}$. With Eq. (1), we calculate the means of B and T and denote them by U_B and U_T respectively.

The transformed block T ′ = $\{p_1'', p_2'' \dots p_n''\}$ is generated by the mean shifting as follows.

$$p_i'' = p_i + u_T - u_B \quad (3)$$

Where ($U_T - U_B$) is the difference between the means of target block and original block. We want to shift each pixel value of original block by amplitude ($U_T - U_B$)) and thus the transformed block has the same mean with the corresponding target block. However, because the pixel value p″ i should be an integer, to keep the transformation reversible, we round the difference to be the closest integer as Eq. (4)

$$\Delta u = round(u_T - u_B) \quad (4)$$

and shift the pixel value by Δu, namely, each p″ i is gotten by

$$p_i'' = p_i + \Delta u \quad (5)$$

Note that the pixel value p″ i should be an integer between 0 and 255, so the transformation (5) may result in some overflow/underflow pixel values. To avoid such transformed blocks abstained by Eq. (5), we assume that the maximum overflow pixel value is OV_{max} for Δu ≥ 0 or the minimum underflow pixel value is UN_{min} for Δu < 0. If overflow/underflow occurs in some blocks, we eliminate them by modifying Δu

$$\Delta u = \begin{cases} \Delta u + 255 - OV_{max}, & if\ \Delta u \geq 0 \\ \Delta u - UN_{min}, & if \Delta u < 0 \end{cases} \quad (6)$$

We use the modified Δu to shift the pixels of block B, and thus all the pixels' values are controlled into the range of [0, 255]. However the range of Δu's value is still very large, which cannot be efficiently compressed. Thus we further modify Δu as in which the quantization step, λ, is an even parameter. Then it just needs to record Δu′ = 2|Δu|/λ, by which it has the advantage of not to record the sign of Δu. Because when Δu′ is an even number it means Δu ≥ 0 and when Δu′ is an odd number it means Δu < 0. Since when λ is large the amount of information recording Δu′ will be small but the offset between the modified Δu and the original Δu will be large, a tradeoff must made by choosing λ. We set λ = 8 in the following experiments.

$$\Delta u = \begin{cases} \lambda \times round\left(\frac{\Delta u}{\lambda}\right) & if\ \Delta u \geq 0 \\ \lambda \times floor\left(\frac{\Delta u}{\lambda}\right) + \frac{\lambda}{2} & if \Delta u < 0 \end{cases} \quad (7)$$

After shifting transformation and rotation, we get a new block T ′. With these new blocks, we replace the corresponding blocks in the target image and generate the transformed image J′. The parameters, Δu′ and rotation directions, will be compressed, encrypted and then embedded into the transformed image J′ as accessorial information (AI) to output the "encrypted image" E(I) called in this paper image.

Finally, to maintain the similarity between the transformed image and target image as much as possible, we further rotate the shifted block into one of the four directions 0o, 90o, 180o or 270o. The optimal direction is chosen for minimizing the root mean square error (RMSE) between the rotated block and the target block.

The transform and anti-transformation procedures of the proposed method are described in Algorithm 1 and Algorithm 2 respectively.

Algorithm 1 Procedure of Transformation

Input: An original image I and a secret key K.

Output: The encrypted image E(I).

1) Select a target image J having the same size as I from an image database.

2) Divide both I and J into several non-overlapping 4 ×4 blocks. Assuming that each image consists of N blocks, calculate the mean and SD of each block.

3) Classify the blocks with %α quantile of SDs and generate CITs for I and J respectively. Pair up blocks of I with blocks of J according the CITs as described in subsection III-A.

4) For each block pair (Bi, Ti) ($1 \leq i \leq N$), compute the mean difference Δui. Add Δui to each pixel of Bi and then rotate the block into the optimal direction θi (θi∈ {0o, 90o, 180o 270o}, which yields a transformed block T ′i .

5) In the target image J, replace each block Ti with the corresponding transformed block Ti′ for $1 \leq i \leq N$ and generate the transformed image J′.

6) Collect Δui's and θi's for all block pairs, and compress them together with the CIT of I. Encrypt the compressed sequence and the parameter α by a standard encryption scheme such as AES with the key K.

7) Take the encrypted sequence as accessorial information (AI), and embed AI into the transformed image J′ with an RDH method such as the one in [7], and output the encrypted image E(I)

Algorithm 2 Procedure of Anti-transformation

Input: The encrypted image E(I) and the key K.

Output: The original image I.

1) Extract AI and restore the transformed image J′ from E(I) with the RDH scheme in [7].

2) Decrypt AI by AES scheme with the key K, and then decompress the sequence to obtain CIT of I, Δui, θi ($1 \leq i \leq N$) and α.

3) Divide J′ into non-overlapping N blocks with size of 4 × 4. Calculate the SDs of blocks, and then generate the CIT of J′ according to the %α quantile of SDs.

4) According to the CITs of J′ and I, rearrange the blocks of J′ as described in Subsection III-A.

5) For each block Ti′ of J′ for $1 \leq i \leq N$, rotate Ti′ in the anti-direction of θi, and then subtract Δui from each pixel of Ti′, and finally output the original image I.

4. Experimental results

Fig2: Secrete image

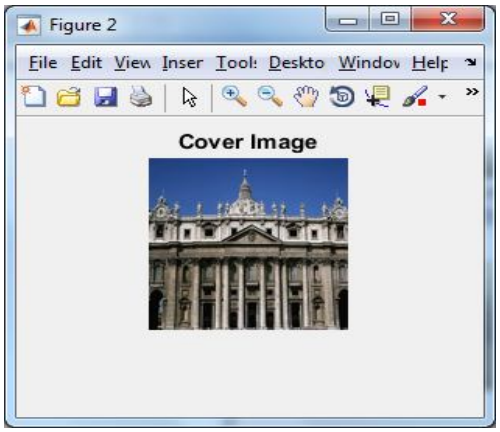


Fig 3: Cover image

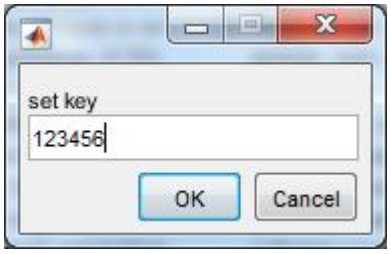


Fig 4: input key

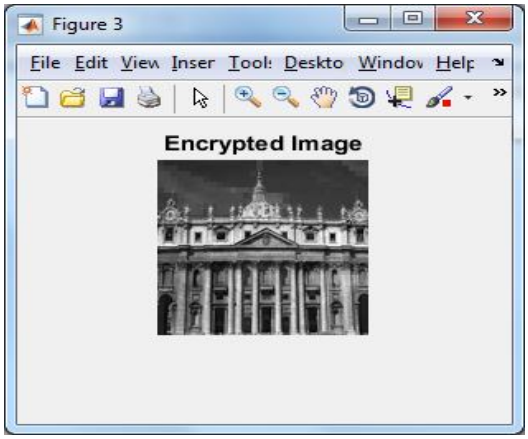


Fig 5: encrypted image

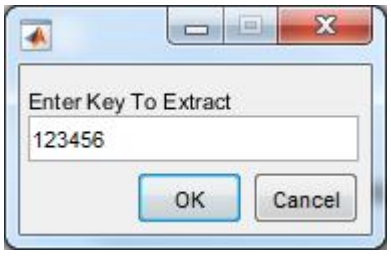


Fig 6: extracting key

Fig 7: decrypted image

5. CONCLUSION

In this paper we recommend a singular framework for reversible facts hiding in encrypted picture (RDH-EI) based totally on reversible photograph transformation (RIT). Different from preceding frameworks which encrypt a plaintext picture right into a cipher-text shape, RIT-based totally RDH-EI shifts the semantic of unique photo to the semantic of some other picture and for that reason guard the semantic of the original photograph. Because the encrypted photo has the shape of a plaintext picture, it'll avoid the notation of the curious cloud server and it's far free for the cloud sever to pick anybody of RDH strategies for plaintext images to embed watermark. We recognize an RIT based technique by using improving the photograph transformation approach in [26] to be reversible. By RIT, we can rework the authentic photograph to an arbitrary decided on target picture with the identical size, and restore the unique photograph from the encrypted photograph in a lossless way. Two RDH methods which include PEE-based totally RDH and UES are followed to embed watermark within the encrypted image to fulfill extraordinary wishes on image excellent and embedding capability.

References

REFERENCES [1] K. Hwang, D. Li, “Trusted cloud computing with secure resources and data coloring,” IEEE Internet Computing, vol. 14, no. 5, pp. 14-22, Sept.-Oct. 2010.

[2] F. Bao, R. H. Deng, B. C. Ooi, et al., "Tailored reversible watermarking schemes for authentication of electronic clinical atlas," IEEE Trans. on Information Technology in Biomedicine, vol. 9, no. 4, pp. 554-563, Dec. 2005.

[3] F. Willems, D. Maas, and T. Kalker, "Semantic lossless source coding," 42nd Annual Allerton Conference on Communication, Control and Computing, Monticello, Illinois, USA, pp. 1411-1418, 2004.

[4] W. Zhang, X. Hu, N. Yu, et al. "Recursive histogram modification: establishing equivalency between reversible data hiding and lossless data compression," IEEE Trans. on Image Processing, vol. 22, no. 7, pp. 2775-2785, Jul. 2013.

[5] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, "Reversible watermarking algorithm using sorting and prediction," IEEE Trans. on Circuits and Systems for Video Technology, vol.19, no.7, pp. 989-999, Jul. 2009.

[6] B.ou, X. Li, Y. Zhao, R. Ni, Y. Shi, "Pairwise prediction-error expansion for efficient reversible data hiding," IEEE Trans. on Image Processing, vol. 22, no.12, pp. 5010-5021, Dec. 2013.

[7] Ioan-CatalinDragoi, DinuColtuc, "Local-prediction-based difference expansion reversible watermarking," IEEE Trans. on Image Processing, vol. 23, no. 4, pp. 1779-1790, Apr. 2014.

[8] Z. Ni, Y. Shi, N. Ansari, and S. Wei, "Reversible data hiding," IEEE Trans. on Circuits and Systems for Video Technology, vol. 16, no. 3, pp. 354-362, Mar. 2006.

[9] J. Tian, "Reversible data embedding using a difference expansion," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13, no.8, pp. 890-896, Aug. 2003.

[10] X. Hu, W. Zhang, X. Li, N. Yu, "Minimum rate prediction and optimized histograms modification for reversible data hiding," IEEE Trans. on Information.