ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Reversible Data Hiding in Encrypted Image Using Histogram Modification

Mr.Vipin Wangikar
M.E. (ETC),
Department of Electronics & Telecommunication,
MSPM's S.S.I.E.M.S, Parbhani.

Prof.Sharad Jogdand, M.Tech (EDT)
Assistant Professor,
Department of Electronics & Telecommunication,
MSPM's S.S.I.E.M.S, Parbhani.

I. Abstract:

In the past two decades, reversible data hiding (RDH), also referred to as lossless o invertible data hiding, has gradually become a very active research area in the field of data hiding. This has been verified by more and more papers on increasingly wide-spread subjects in the field of RDH research that have been published these days. In this paper, the various RDH algorithms and researches have been classified into the following six categories: 1) RDH into image spatial domain; 2) RDH into image compressed domain (e.g., JPEG); 3) RDH suitable for image semi-fragile authentication; 4) RDH with image contrast enhancement; 5) RDH into encrypted images, which is expected to have wide application in the cloud computation; and 6) RDH into video and into audio. For each of these six categories, the history of technical developments, the current state of the arts, and the possible future researches are presented and discussed. It is expected that the RDH technology and its applications in the real word will continue to move ahead.

II. Introduction:

Data hiding has received much attention fromthe research community in the past more than two decades. By this technique, it can embed secret data into a cover medium, and later enable the intended user to extract the embedded data from the marked medium for various purposes. However, for most data hiding methods, the cover medium has been distorted during the data embedding operation and hence cannot be restored into its original form after data extraction. In some sensitive scenarios, such permanent distortion is strictly forbidden and the exact recovery of the original cover medium is required. To solved this issue, reversible data hiding (RDH) also called lossless or invertible data hiding, is proposed losslessly recover both the embedded data and the cover medium . That is, with the RDH, besides the embedded data, the cover medium can be exactly recovered from the marked data as well. The first RDH algorithm is the one proposed by Barton in a US patent in 1997. He proposes to embed the authentication information into a digital medium, and enable legitimate users to extract the embedded authentication information for verifying the authenticity of the received data.

II.1. Conventional techniques:

A. Digital Watermarking:

A watermarking system is usually divided into three distinct steps, embedding, attack, and detection. In embedding, an algorithm accepts the host and the data to be embedded, and produces a watermarked signal. Then the watermarked digital signal is transmitted or stored, usually transmitted to another person. Digital watermarking is the process of embedding information into digital multimedia content such that the information (which we call the watermark) can later be extracted or detected for a variety of purposes including copy prevention and control. Digital watermarking has become an active and important area of research, and development and commercialization of watermarking techniques is being deemed essential to help address some of the challenges faced by the rapid proliferation of digital content. A digital watermark can be visible or invisible. A visible watermark typically consists of a conspicuously visible message or a company logo indicating the ownership of the image. On the other hand, an invisibly watermarked image appears very similar to the original. The existence of an invisible watermark can only be determined using an appropriate watermark extraction or detection algorithm. In this chapter we restrict our attention to invisible watermarks. An invisible watermarking technique, in general, consists of an encoding process and a decoding process. Its main concern lies in how to reliably extract the embedded data from a possibly degraded marked medium. It considers the robustness as top priority, but rarely cares about the cover medium recovery.

B. Steganography:

Steganography deployed in spatial, transform, and compression domains of digital images. Transform domain techniques make changes in the frequency coefficients instead of manipulating the image pixels directly, thus distortion is kept at minimum level and that’s why they are preferred over spatial domain techniques. But when it comes to embedding capacity, spatial domain techniques give better results. However, there exists a trade-off between the image quality and the embedding capacity. Hiding more data results directly into more distortion of the image. So the steganography technique deployed is dependent on the type of application it is designed for. In recent years, some researchers have concentrated on embedding secret data into the compression codes of images. Such need arises keeping in mind the bandwidth requirements. Steganography can also be used misused like othertechnologies. For instance terrorists may use this technique fortheir secret secure communication or anti-virus systems canbe fooled if viruses are transmitted in this way. However, it is evident that steganography has numerous useful applications and will remain the point of attraction for researchers.

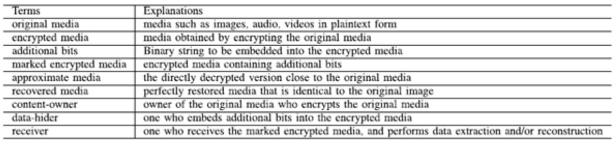
III. Proposed Method:

Compared with these two data hiding techniques, the specific property of RDH is the perfect recovery of both of the cover medium and the embedded secret data. In general, RDH is a fragile technique and it poses no robustness against possible attacks.

I. Reversible Data hiding in encrypted domain:

A. MOTIVATION:

As is well known, encryption is an effective and popular means of privacy protection. Recently, the research on signal processing over encrypted domain, primarily driven by the needs from Cloud computing platforms and various privacy preserving applications, has gained increasing attention. Combination of data hiding and encryption has also received some of the earliest attention. In some existing jointdata-hiding and encryption schemes, only a part of cover data are encrypted and the rest can be used to carry the additional message. In, the intra-prediction mode, motion vector difference and signs of DCT coefficients are encrypted, while a watermark is embedded into the amplitudes of DCT coefficients. In, the cover data in higher and lower bit-planes of transform domain are respectively encrypted and watermarked. In these joint schemes, however, since only partial encryption is involved, lead to leakage of partial information of the cover. Furthermore, the separation of original cover and embedded data from a watermarked version is not considered. Also, the data embedding is not reversible. Most of works on RDH discussed in previous sections are suitable for plaintext domain, namely, the additional bits are embedded into the original, un-encrypted multimedia data. Along with more and more attention on signal processing over encrypted domain, the investigation of embedding additional data in the encrypted domain in a reversible fashion is triggered. In order to securely store or share multimedia file with other person, a content owner may encrypt the media data before transmission. In some application scenarios, an inferior assistant or a channel administrator hopes to append some additional message, such as the origin information, image notation or authentication data, within the encrypted media though he does not know the original content. Taking medical images administration as an example, when medical images have been encrypted for protecting the patient privacy, a database administrator may aim to embed the personal information into the corresponding encrypted images. It may be also hopeful that the original content can be recovered exactly after decryption and data extraction at receiver side. That means RDH in encrypted domain (RDH-ED) is required. In this section, the state-of-the-art reversible embedding techniques of RDH-ED are reviewed.



Terms	Explanations
original media	media such as images, audio, videos in plaintext form
encrypted media	media obtained by encrypting the original media
additional bits	Binary string to be embedded into the encrypted media
marked encrypted media	encrypted media containing additional bits
approximate media	the directly decrypted version close to the original media
recovered media	perfectly restored media that is identical to the original image
content-owner	owner of the original media who encrypts the original media
data-hider	one who embeds additional bits into the encrypted media
receiver	one who receives the marked encrypted media, and performs data extraction and/or reconstruction

Table: Terms used in this paper

As an emerging technology, RDH-ED aims at embedding additional information into cipher-data without revealing theplaintext content and recovering the original plaintext contenterror-free at the receiver side. For the ease of discussion,explanations of some frequently used terms are listedin Table 1. The general framework of RDH-ED is sketched in Fig. Consider the three parties in the entire work_ow: content-owner, data-hider, and receiver, whose roles are described as follows.

- **Content-Owner:** Encrypt the original media to concealthe principal content with or without some preprocessing.An encryption key is chosen by the content-owner.
- **Data-Hider:** Embed the additional bits into the encrypted media. A data hiding key is used by the datahider for security.
- **Receiver:** Three options are available for receivers who hold different keys.

Option 1: decrypt the marked encrypted media to get approximate media.
Option 2: Extract the additionally embedded bits.
Option 3: generate the recovered media that is identical to the original. The existing RDH-ED methods can be classi_ed into two categories: ``vacating room before encryption (VRBE)" and ``vacating room after encryption (VRAE)". Here, we provide an overview in each category one by one.

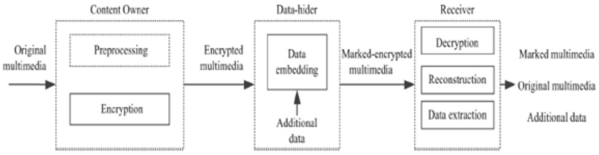


Fig1: General framework of RDH-ED

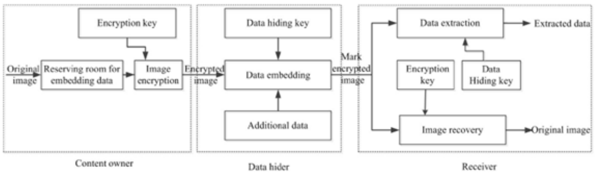


Fig2: VRBE framework.

A.VACATING ROOM BEFORE ENCRYPTION (VRBE):

VRBE framework creates embedding room in the plaintext domain, i.e., vacating embedding room before encryption. Thus, the content owner is expected to perform an extra preprocessing before encryption. Take image as example (similarly hereinafter), the sketch of VRBE framework is shown in Fig. In , the embedding room is created in digital images by embedding LSBs of certain pixels into other pixels using a traditional RDH method. The pre-processed image is then encrypted by the owner to generate an encrypted image. Thus, the positions of these vacated LSBs in the encrypted image can be used by the data-hider, and a large payload up to 0.5 bpp can be achieved. With a similar idea, another method based on a prediction technique is proposed. In this method, some pixels are estimated by the rest pixels before encryption and predicted errors are gained. Then, a special encryption scheme is designed to encrypt the predicted errors and a benchmark encryption algorithm (e.g. AES) can be applied to the rest pixels.

Instead of embedding data in encrypted images directly, additional data can be embedded by shifting the encrypted histogram of predicted errors. Further improvement is made by considering the patchlevel sparse representation. The widely used sparse coding technique has demonstrated that an image patch can be linearly represented by some atoms in an overcomplete dictionary. As the sparse coding is an approximation solution, the leading residual errors are encoded and self-embedded within the cover image. Furthermore, the learned dictionary is also embedded into the encrypted image. Thanks to the powerful representation of sparse coding, a large vacated room can be achieved, and thus the data hider can embed more secret messages in the encrypted image.

In, embedding room is vacated by combining Paillier homomorphic encryption and a traditional plaintext RDH technique, i.e., difference expansion. In this scheme, preprocessing is required. In other words, before image encryption, the image owner has to pre-processing original image to generate a processed image with a modified difference expansion method. Then, the processed image would be encrypted by Paillier homomorphic encryption and sent to the data-hider, who will embed one bit into each pair of adjacent encrypted pixels and generate the marked encrypted image containing additional bits. Based on the homomorphic property of Paillier encryption, the receiver compares all pairs of decrypted pixels to obtain the embedded bits, and also recovers the cover-image. VRBE framework might be impractical because it requires the content owner to perform an extra preprocessing before content encryption. In this sense, VRAE framework, what you will see in the following, is more close to practice.

C. VACATING ROOM AFTER ENCRYPTION (VRAE):

In VRAE methods, the original signal is encrypted directly bythe content owner, and the data-hider embeds the additionalbits by modifying some bits of the encrypted data.

Also takeimage as example, the sketch of VRAE framework is shownin Fig.Based on the domain in which the additional data canbe extracted, VRAE methods can be further grouped into three basic categories. These include data extraction in theplaintext domain, data extraction in the cipher domain and

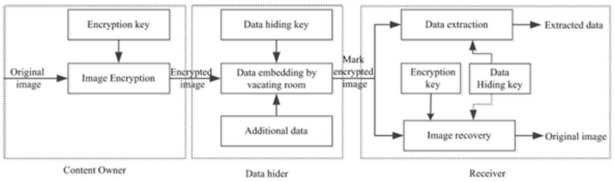


Fig3: VRAE framework.

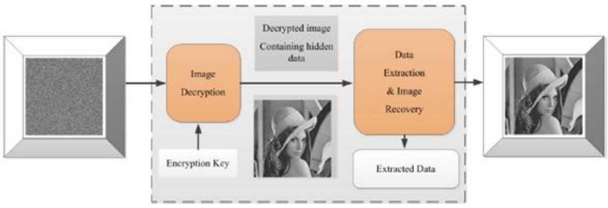


Image Decryption
Decrypted image Containing hidden data
Data Extraction & Image Recovery
Encryption Key
Extracted Data

Fig4: Sketch of data extraction and image recovery in plaintext domain.

Data extraction in both domains. In the first place, we introduce data extraction in the plaintext domain.

1. VRAE: DATA EXTRACTION IN THE PLAINTEXT DOMAIN:

In these methods, with an encrypted medium containing additional data, a receiver may first decrypt it according to the encryption key, and then extract the embedded data and recover the original image according to the data-hiding key. The sketch of data extraction and image recovery in plaintext domain is shown in Fig. The first method is proposed by Zhang for encrypted images , in which the data-hider divides the encrypted image into blocks and embeds one bit into each block by flipping three least significant bits (LSB) of half the pixels in the block. On the receiver side, the marked encrypted image is decrypted to an approximate image. The receiver flips the three LSBs of pixels to form a new block and uses a function to estimate the image-texture of each block.

Due to spatial correlation in natural images, original block is presumed to be much smoother than interfered block. Thus the embedded bits can be extracted and the original image can be recovered jointly. Embedding rate of this method depends on the block size. If an inappropriate block size is chosen, errors may occur during data extraction and image recovery. A similar method for JPEG images is proposed by Qian et al. , that hides data into the encrypted JPEG bit-stream and recovers the original bit-stream by analyzing the blocking artifacts caused by data hiding.
This method has been paid great attention . For example, in , the performance is improved by exploiting spatial correlation between neighboring blocks and using a side-match algorithm to achieve higher embedding payload with lower error rates in image recovery. The performance is further improved by introducing a flipping ratio and using unbalanced bit flipping. In, a more precise function is presented to estimate the image-texture of each image block and increase the correctness of data extraction/image recovery.

Furthermore, during data embedding in, the LSBs of fewer pixels are flipped instead of flipping the LSBs of half pixels in the encrypted image, which leads to the significant improvement for the visual quality of the approximate image. And a new adaptive judging function based on the distribution characteristic of image local contents is utilized to estimate the image-texture of each block in the procedures of data extraction and image recovery, which decrease the errors of extracted bits and recovered image to some extent. In , a two-class SVM classifier is adopted to distinguish encrypted and non-encrypted image blocks rather than the judging functions used . At the same time, this method gets a sharp rise in the embedding capacity since the data embedding is achieved through a public key modulation mechanism rather than LSBs flipping. In addition, some researchers have achieved this kind of RDH-ED based on public key cryptosystem and homomorphic encryption.

In, it is stated that a RDH scheme.for any encrypted signals is proposed and digital image is taken as an example for description. During image encryption, each pixel valuewas segmented into two parts, i.e., seven most significant bits (MSBs) and one LSB, and these twoparts were encrypted respectively. Then, two encrypted LSBsof each encrypted pixel pair were modified to reversibly embed one secret bit according to the properties of homomorphism.The receiver can easily extract the embedded bits andrecover the original image by judging the relationship of thetwo decrypted LSBs in each pixel pair. However, in the aspect of images, the inherent overflow cannot be avoided. These methods introduced above focus on the data extractingafter decryption. In other words, the additional datamust be extracted from the plaintext domain, so that theprincipal content is revealed before data extraction, and,if someone has the data-hiding key but not the encryptionkey, he cannot extract any information from the markedencrypted media which containing additional bits. Since theextraction of the embedded bits and the recovery of theoriginal media are often tied together, this category isalso called non-separable solution. Opposite to thiscategory, there is another type called separable solution,in which the data extraction can be separately carried outbefore image decryption, i.e., data extraction in the cipherdomain.

2. VRAE: DATA EXTRACTION IN THE CIPHER DOMAIN:

In these methods, with a marked encrypted medium, a legal receiver having the data-hiding key can extract the additional bits in the cipher domain directly, while a receiver having the encryption key can decrypt the received data to obtain an edition similar to the original one, i.e., the approximate medium. If the receiver has both the data-hiding and encryption keys, he can extract the additional bits and recover the original medium error-free. Sketch of data extraction in the cipher domain is shown in Fig.

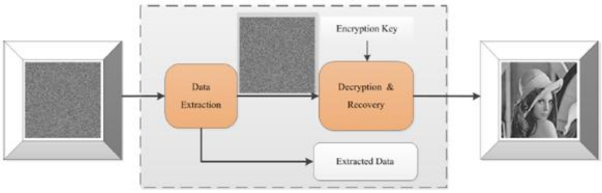


Fig5: Sketch of data extraction in the cipher domain

The idea was first proposed by , in which the owner encrypts the original image by Advanced Encryption Standard (AES), and the data-hider embeds one bit in each block containing n pixels, meaning that the embedding rate is 1 /n bpp. On the receiver side, data extraction and image recovery are realized by analyzing the local standard deviation after decryption of the marked encrypted image. Although this method provides a good embedding rate, two drawbacks are fatal on the recipient side. On the one hand, the attacker may break some information from the enciphered bits by statistical analysis. Since each block is encrypted independently using AES with an encryption key, redundancy in an image may result in the reduplicative encrypted blocks. On the other hand, if the receiver directly deciphers the marked encrypted image, quality of the decrypted image is rather poor, which is far from the human vision requirements.

To overcome the drawbacks in this solve the problem that additional data can only be extracted from the decrypted domain, Zhang further proposed an RDH-ED method for stream-enciphered images using the idea of compressing the encrypted bits to accommodate the additional bits [171]. The data-hider pseudo-randomly permutes and divides the encrypted image into groups with size of L. The P LSB-planes of each group are compressed with a matrix G sized (PL - S) *(PL) to generate corresponding vectors. Thus, S bits are available for data embedding. On the receiver side, a total of 8 - P most significant bits (MSB) of pixels are obtained by decryption. The receiver then estimates the P LSBs by the MSBs of neighboring pixels. By comparing the estimated bits with the vectors in the coset Ω corresponding to the extracted vectors, the receiver can recover the original bits of the P LSBs. Because the additional bits are embedded in LSBs of the encrypted images, they can be extracted directly before image decryption. This method ignites researchers' passion again. Qian et al. use histogram shifting to encipher the original image and then additional bits can be embedded into the encrypted image by using an n-nary data hiding scheme .In , the cover image is partitioned into non-overlapping blocks and encrypted by multi-granularity encryption. The additional data is then embedded into the blocks in a sorted order with respect to block smoothness by using a novel local histogram shifting. Both of them provide satisfactory embedding payload and nice image quality. However, as the original image is encrypted with pixel permutation and affine transformation, leakage of image histogram is inevitable under exhaustive attack.

Some other methods further vacate embedding room by encrypted bits compression. Zhang et al. extended the lossless compression based RDH approach to the encrypted domain. Half of the 4th LSBs of the encrypted image are losslessly compressed via low density parity check (LDPC) code to create space for data hiding. In, authors encode the selected bits taken from the stream-ciphered image using LDPC codes into syndrome bits to make spare room to accommodate the additional data. In, the least significant bits of pixels in encrypted image are losslessly compressed by the Hamming distance calculation between the LSB stream and auxiliary stream. In some application scenarios, a content owner encrypts the plaintext media, e.g., images, and asks a telecommunication operator or a channel provider to deliver the encrypted data to some users. Although the telecommunication operator/ channel provider does not know the plaintext content, he may hope to insert a visible watermark into the encrypted data and deliver the marked encrypted data to the users, so that the users who have not paid for data transmission can only obtain a visibly marked version, i.e., a quality degraded image, by decrypting the received data, while the users authorized both by the content owner and the telecommunication operator/channel provider can obtain the original image after data extraction and content recovery. That means the reversible visible watermark embedded in the encrypted domain is needed. In, a reversible visible watermarking scheme for encrypted images based on wet paper codes is proposed. In this scheme, the exclusive-or operation is employed for encryption, and a part of encrypted data corresponding to the black pixels of watermark image is altered to insert the visible watermark and carry some additional data used for content recovery. Although the watermark is inserted in encryption domain, it is invisible through the encrypted data and visible after direct decryption at receiver side, i.e., a direct decryption would result in a visibly marked image. If the data-hiding key is available, the receiver can employ the wet paper decoding to extract the embedded bits from the marked encrypted image and a joint decryption-extraction recovery operation may retrieve the original plaintext image error-free if both of the keys are available.

In addition, while most RDH-ED methods are based on stream cipher, Qian et al. presents an alternative method feasible for block-enciphered images. Before uploading data to a remote server, the content owner encrypts the original image with a block cipher algorithm using an encryption key. Then, the server embeds additional bits into the encrypted image with an embedding key to generate the marked encrypted image. On the recipient side, the additional bits can be extracted from the encrypted domain if the receiver has the embedding key. Compared with the existing block cipher based RDH-ED method, image security and quality can be improved. Besides, in, a universal reversible data embedding method applicable to any encrypted domain is described. It has stated that the coding redundancy of any encryptedsignal can be exploited by partitioning it into segments and using Golomb-Rice codewords (GRC) to entropy encode them. Then two bits can be embedded into each segment GRC code in a reversible manner. The experimental results shown in the article have demonstrated that, for every bit of the encrypted signal, an average embedding payload of 0.169 bit can be achieved. With the methods mentioned above, although the embedded data can be extracted in encrypted domain by using the data-hiding key, it is impossible to perform the data extraction in decrypted domain. Thus, there is one key problem of the works discussed so far, that the embedded data can only be extracted either before (shown in Fig. 14) or after (shown in Fig.) decryption. That means that a legal receiver who has the data hiding key but no decryption key he cannot extract the embedded bits from the cipher domain directly, or, on the other hand, a legal receiver who has the data hiding key and the decrypted media containing additional bits he cannot extract the embedded data. So, a kind of new RDH-ED framework in which the embedded bits can be extracted from both plaintext domain and cipher domain is desirable.

3. VRAE: DATA EXTRACTION IN BOTH DOMAINS:

In this framework, with the marked encrypted media containing additional bits, a legal receiver who knows the data-hiding key can extract the embedded data from the cipher domain directly. And a content user with the encryption key may decrypt the encrypted data to obtain a similar edition to the original one. If someone receives the approximate media and has the data-hiding key, he can also successfully extract the additional bits and perfecty recover the original image. The sketch of data extraction in both domains is shown in Fig.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

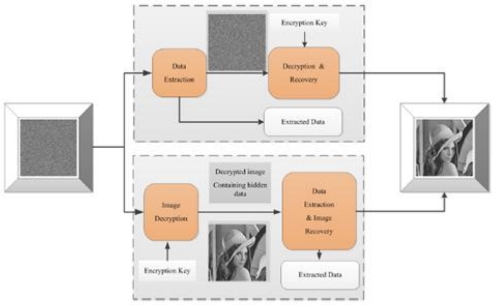


Fig 6: Sketch of data extraction in both domains

In, the first solution based on pseudorandom sequence modulation is provided. In this scheme, a part of data in LSB planes of encrypted image is replaced with the additional data and the rest data in LSB planes are modified by the pseudorandom sequences modulated by the replaced bits and embedded data. Then, the additional-data user with the data-hiding key can easily extract the additional data in encrypted domain. Since the data embedding operation affects only the LSB, a direct decryption may result in an image with principal original content. By finding the modulated sequences corresponding to the minimal fluctuation, the embedded data can be extracted from the decrypted image and the original content can also be recovered without any error when the embedding rate is not too high.

In some methods, RDH-ED is achieved by means of two neighboring pixels are masked by same pseudo-random bits. Then, the additional data are embedded into various bit planes with a reversible manner, and a parameter optimization method is used to ensure a good payload-distortion performance. Because the data space used for accommodating the additional data is not affected by the encryption operation, the data insertion/extraction can be performed in both the plain and encrypted domains, and the ways of data insertion/extraction in the two domains are same. In, a complete separable RDHEI method is proposed based on block division, RC4 encryption and block histogram modification. At first, the original image is divided into non-overlapping blocks. Then all the pixels within each block can be encrypted by RC4 with the same key. Thus each encrypted block keeps structure redundancy to carry additional bits and RDH is achieved by block histogram shifting. The embedded data can be extracted error-free both from the marked encrypted image (cipher domain) and directly decrypted image (plaintext domain). However, this method is not suitable for images containing saturated pixels. A similar idea is proposed in , where the image is divided into groups by cross division and all the pixels within each group are encrypted by RC4 with the same key. Thus the difference histogram is maintained after image encryption. Then additional bits can be reversibly embedded by using difference histogram shifting. In, the cover image is first encrypted by permutation in both block- and pixel-wise manners using a chaotic mapping. Then, the PVO embedding is adopted to reversibly embed additional bits into each permuted block. Since the pixel value order is unchanged in each block after PVO embedding, the embedded data can be exactly extracted using the inverse PVO whether the marked image is decrypted or not.

In, a stream cipher is utilized to encrypt sample pixels and a specific encryption mode is designed to encrypt interpolation-error of non sample pixels. Then, additional bits can be embedded into the interpolation-error by modified histogram shifting and difference expansion technique. Then, data extraction can be done either in the encrypted domain or in the decrypted domain. However, just as described in the experimental results of the paper, the leakage of image contour is inevitable. In another method, RDH-ED is achieved with the benefit of Homomorphic encryption [187]. This paper proposes a lossless, a reversible, and a combined data hiding schemes for cipher images encrypted by public key cryptosystems with probabilistic and homomorphic properties. In the lossless cheme, the cipher pixels are replaced with new values to embed the additional bits into several LSB-planes of cipher pixels by multi-layer wet paper coding.

Then, the embedded data can be directly extracted from the encrypted domain, and the data embedding operation does not affect the decryption of original plaintext image. In the reversible scheme, a preprocessing is employed to shrink the image histogram before image encryption, so that the modification on encrypted images for data embedding will not cause any pixel oversaturation in plaintext domain. Although a slight distortion is introduced, the embedded data can be extracted and the original image can be recovered from the directly decrypted image. Due to the compatibility between the lossless and reversible schemes, the data embedding operations in the two manners can be simultaneously performed in an encrypted image. With the combined technique, a receivermay extract a part of embedded data before decryption, and extract another part of embedded data and recover the original plaintext image after decryption.

II. HISTOGRAM MODIFICATION:

Histogram modification algorithm. Up-to-date histogram-based algorithms are also provided, including the use of prediction errors and adjacent pixel difference. Ni et al. proposed a reversible data hiding algorithm for grayscale images based on a histogram modification algorithm. For a given grayscale image, they first count the frequency of each pixel value and then generate a histogram. Thereafter two values, called the peak and the minimum, are obtained according to the frequency of each value. The peak value PV has the maximum frequency; while the minimum value MV has the minimum frequency. The minimum value MV can be called the zero value ZV if its frequency is equal to zero. If the frequency of MV is not equal to zero, all positions of the pixels with the value MV must be recorded previously. Histogram modification algorithm for lossless data hiding. The algorithm includes two procedures, data embedding and data extraction. The flow chart of the proposed algorithm is illustrated in Fig. 1. The embedding procedure takes an original image and the secret message as input. This procedure produces a marked image, the secret key (the peak/minimum values) and other essential information for data extraction. In comparison, the data extraction procedure takes a marked image, the secret key and residual information as input and can recover the original image after extracting the secret message correctly. In the following sections, we will discuss our proposed algorithm in detail.

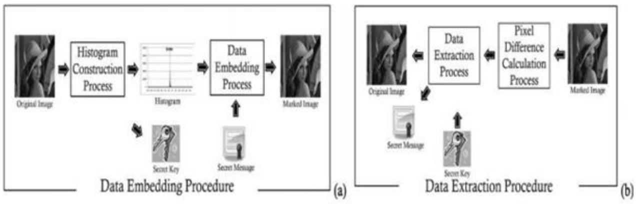


Fig7: The flow chart of proposed algorithm, including (a) the data embedding procedure and (b) the data extraction procedure.

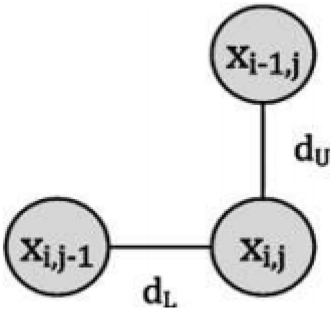


Fig8: The illustration for pixel difference calculation for each visiting pixel.

The Data Embedding Procedure:

This section illustrates the data embedding procedure in detail. This procedure starts by constructing the histogram for the input image. Thereafter, the data embedding process takes the secret message SM as input and then embeds it into the pixels located at the peak value in the constructed histogram.

Histogram Construction Process:

The histogram construction of our proposed algorithm is based on the difference between each visited pixel and its neighbours. Except for the first row and the first column, each pixel in the input image is visited in the raster order (i.e., from left to right and top to bottom).

Now, we can start to calculate the pixel difference for each visiting pixel For each visiting pixel , we calculate the pixel difference with its left and upper neighbouring pixels and respectively. The equation for calculating two pixel differences is shown in figure, where C , C , and C are the pixel values of the visiting pixel and its two neighbours. and are the calculated neighbouring pixel differences. Fig. 3 illustrates the spatial representation for the above three pixels.

$$\begin{cases} d_L = Cx_{i,j} - Cx_{i,j-1} \\ d_U = Cx_{i,j} - Cx_{i-1,j} \end{cases}$$

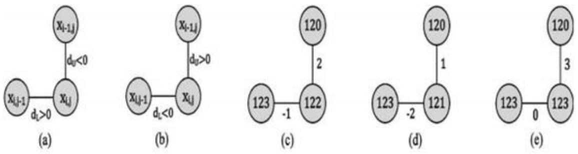


Fig.9 The pixels that are ignored during the histogram construction process.

After two pixel differences for each pixel are calculated, we start to construct the histogram. In our proposed algorithm, the pixel can be embedded into 1.5 bit while its two pixel differences are both equal to first peak value. One thing to be noted is that the pixels will be ignored for the histogram construction if their two pixels differences and have different signs, such as the case in Figs. (a) and (b). Such pixels may lead to extraction errors in the data extracting process. During the data embedding process, other than no modification, S will be produced to either increase or decrease at least one from C in the histogram based algorithms. Both pixel differences, and , are also either increased or decreased by one from the original difference after data embedding. Such modification makes one of two pixel differences approach the peak value and may lead to errors in message extraction. For example in Fig.(c), two pixel differences for the pixel with the value 122 are equal to 2 and −1. If the peak value is located at the value 0, this pixel cannot be used for data embedding and should shift away from the peak value.

Thus, we may subtract or add one from the original pixel value and the pixel difference should be modified (see Figs. (d) and (e)). However, this action will cause this pixel to be regarded as a pixel with a secret message embedded. In order to avoid the above situation, we ignore such pixels during the histogram construction process and use the abbreviation SNP to represent the above set of pixels.

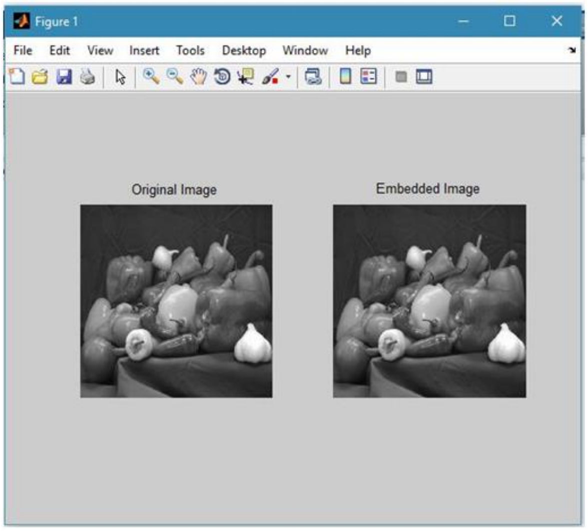
Data Embedding Process:

In the embedding process, similar to previous algorithms, we also embed the secret message into the pixels whose pixel difference is located at the peak value in the histogram. Due to the characteristic of our proposed algorithm, there are two different shifting directions for the peak value according to different situations. No modification occurs if SM is equal to 0; otherwise we ‘add’ one to the pixel value when SM is equal to 1. Therefore, the data embedded pixel difference for this pixel in the data extraction process should be equal to PV or PV + 1 because the other pixel difference must be larger than PV or PV + 1. On the contrary, for the situation if SM is equal to 0, we still take no action on the pixel; otherwise we ‘subtract’ one from the pixel value when SM is equal to 1. The data-embedded pixel difference for this pixel in the data extraction process should be equal to PV or PV−1 because the other pixel difference must be smaller than PV or PV − 1. The pixel can have three different statuses used for data embedding, including no modification, add one or subtract one from the original pixel value. After modification, both pixel differences become PV, PV + 1, and PV − 1. These modifications do not result in an extraction error. When both the pixel differences are equal to PV, PV+1, or PV−1 in the data extraction process, it is clear that this pixel must have the original pixel difference with the value PV. Therefore, users can integrate previous algorithms to increase the capacity efficiently for such pixels. Finally, such pixels should shift by a value of one toward the appropriate direction in order not to be confused with the pixels having the SM embedded.

The Data Extraction Procedure:

The first step for data extraction is to recalculate the pixel difference between each pixel and its neighbours. The same method used in the data embedding process is performed. Because the calculation of the pixel difference is based on the original pixel values of neighbouring pixels, we must recover each pixel value right away after deriving the secret message. Thus, the post-processed pixels can derive the original pixel values of the neighbouring pixels to calculate its pixel difference. Now, we can use the raster scan order to visit each pixel, extract the secret message and then recover the original image. Note that as in the embedding process, the pixel is ignored during data extraction if two pixel differences are with different signs. For each visiting pixel, we recalculate its two pixel differences. Thereafter, we can extract the secret message based on the pixel difference. When one of its two pixel differences is equal to the value PV, PV + 1, or PV − 1, this pixel must have the secret message embedded. The only thing we must do is to recover the original pixel value. If both pixel differences are larger than PV + 1, the original pixel value can be recovered by subtracting one from the marked pixel value; while the pixel value can be recovered by adding one to the marked pixel value when both pixel differences are smaller than PV−1.

RESULTS:



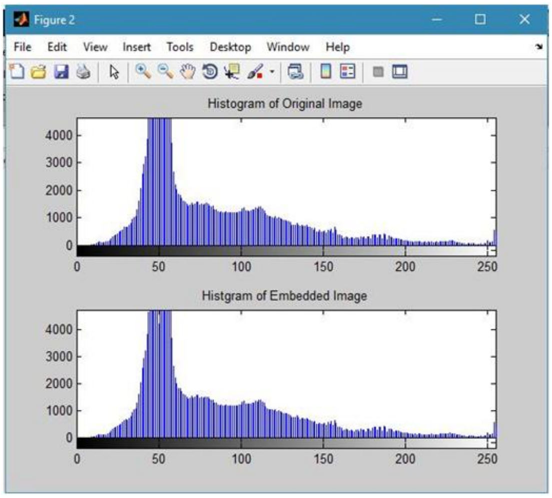
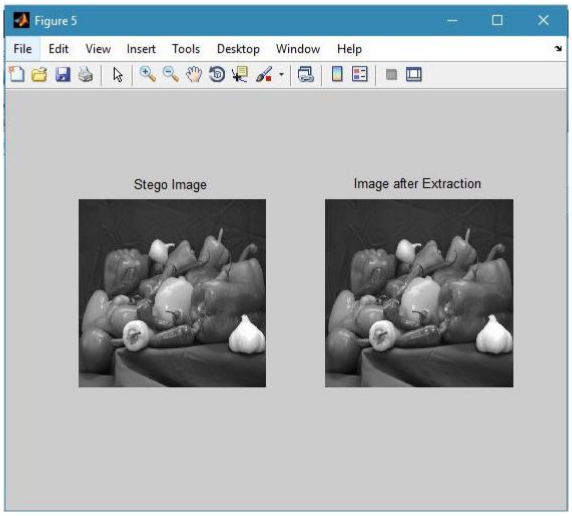
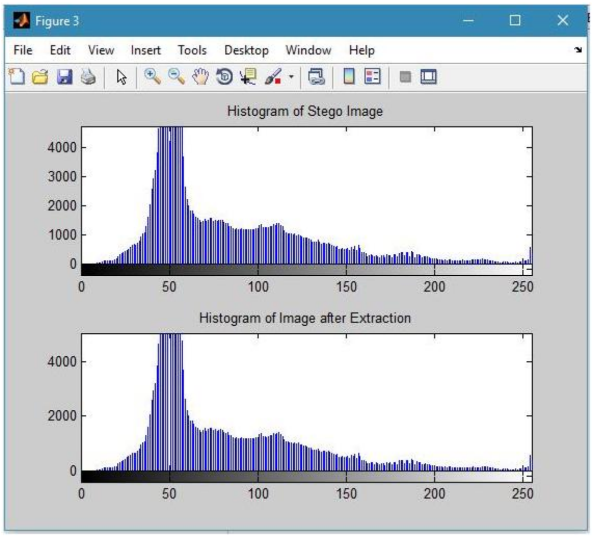


Figure 2
File Edit View Insert Tools Desktop Window Help
Histogram of Original Image
4000
3000
2000
1000
0
0
50
100
150
200
250
Histgram of Embedded Image
4000
3000
2000
1000
0
0
50
100
150
200
250





CONCLUSION:

With digital images used as the main media together with video and audio, this paper has addressed the following six subjects for the reversible data hiding (RDH).They are the RDH into digital images in the spatial domain, the RDH into digital images in the JPEG domain, the semi fragile RDH into digital images that have gone through somelossy compression, the image quality measure used for RDH which is different from the PSNR, the RDH into encrypted digital images and the RDH for video and audio. It is expected that the research on the RDH and the applications of the RDH will continue to move ahead in the future.

REFERENCES:

[1] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, Digital Water-marking and Steganography. San Mateo, CA, USA: Morgan Kaufmann,2007.

[2] J. Fridrich, Steganography in Digital Media: Principles, Algorithms, andApplications. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[3] Y. Q. Shi, Z. Ni, D. Zou, C. Liang, and G. Xuan, ``Lossless data hiding:Fundamentals, algorithms and applications," in Proc. IEEE Int. Symp.Circuits Syst., vol. 2.May 2004, pp. 33_36.

[4] Y. Q. Shi, ``Reversible data hiding," in Proc. Int. Workshop Digit.Watermarking, 2004, pp. 1_12.

[5] R. Caldelli, F. Filippini, and R. Becarelli, ``Reversible watermarkingtechniques: An overview and a classi_cation," EURASIP J. Inf. Secur.,vol. 2010, 2010, Art. no. 134546.

[6] J. M. Barton, ``Method and apparatus for embedding authentication informationwithin digital data," U.S. Patent 5 646 997, Jul. 8, 1997.

[7] C. W. Honsinger, P. W. Jones, M. Rabbani, and J. C. Stoffel,``Lossless recovery of an original image containing embedded data,"U.S. Patent 6 278 791, Aug. 21, 2001.

[8] F. Bao, R.-H.Deng, B.-C.Ooi, and Y. Yang, ``Tailored reversible watermarkingschemes for authentication of electronic clinical atlas," IEEETrans. Inf. Technol. Biomed., vol. 9, no. 4, pp. 554_563, Dec. 2005.

[9] G. Coatrieux, C. Le Guillou, J.-M.Cauvin, and C. Roux, ``Reversiblewatermarking for knowledge digest embedding and reliability controlin medical images," IEEE Trans. Inf. Technol. Biomed., vol. 13, no. 2,pp. 158_165, Mar. 2009.

[10] K. L. Chung, Y. H. Huang, P. C. Chang, and H. Y. M. Liao,``Reversible data hiding-based approach for intra-frame error concealmentin H.264/AVC," IEEE Trans. Circuits Syst. Video Technol., vol. 20,no. 11, pp. 1643_1647, Nov. 2010.

[11] D. Coltuc and I. Caciula, ``Stereo embedding by reversible watermarking:Further results," in Proc. Int. Symp. Signals, Circuits Syst., Jul. 2009,pp. 1_4.

[12] X. Tong et al., ``Stereo image coding with histogram-pair based reversibledata hiding," in Proc. Int. Workshop Digital-Forensics Watermarking,2014, pp. 201_214.

[13] X. Wang, C. Shao, X. Xu, and X. Niu, ``Reversible data-hiding schemefor 2-D vector maps based on difference expansion," IEEE Trans. Inf.Forensics Security, vol. 2, no. 3, pp. 311_320, Sep. 2007.

[14] F. Peng, Y.-Z.Lei, M. Long, and X.-M. Sun, ``A reversible watermarkingscheme for two-dimensional CAD engineering graphics based onimproved difference expansion," Comput.-Aided Design, vol. 43, no. 8,pp. 1018_1024, 2011.

[15] K. Hwang and D. Li, ``Trusted cloud computing with secure resourcesand data coloring,'' IEEE Internet Comput., vol. 14, no. 5, pp. 14_22,Sep. 2010.

[16] J. Fridrich, M. Goljan, and R. Du, ``Invertible authentication,'' Proc.SPIE, vol. 4314, pp. 197_208, Aug. 2001.

[17] M. Goljan, J. J. Fridrich, and R. Du, ``Distortion-free data embedding forimages,'' in Proc. 4th Inf. Hiding Workshop, 2001, pp. 27_41.

[18] J. Fridrich, M. Goljan, and R. Du, ``Lossless data embedding_New paradigm in digital watermarking,'' EURASIP J. Adv. Signal Process.,vol. 2002, no. 2, pp. 185_196, 2002.

[19] G. Xuan, J. Zhu, J. Chen, Y. Q. Shi, Z. Ni, and W. Su, ``Distortionlessdata hiding based on integer wavelet transform,'' Electron. Lett., vol. 38,no. 25, pp. 1646_1648, Dec. 2002.

[20] G. Xuan, J. Chen, J. Zhu, Y. Q. Shi, Z. Ni, and W. Su, ``Lossless datahiding based on integer wavelet transform,'' in Proc. IEEE Int. WorkshopMultimedia Signal Process., Dec. 2002, pp. 312_315.

[21] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, ``Reversible datahiding,'' in Proc. IEEE Int. Conf. Inf. Process., vol. 2. Sep. 2002, pp. 157_160.

[22] G. Xuan et al., ``High capacity lossless data hiding based on integerwavelet transform,'' in Proc. IEEE Int. Symp. Circuits Syst., vol. 2.May 2004, pp. 29_32.

[23] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, ``Losslessgeneralized-LSB data embedding,'' IEEE Trans. Image Process., vol. 14,no. 2, pp. 253_266, Feb. 2005.

[24] M. U. Celik, G. Sharma, and A. M. Tekalp, ``Lossless watermarking forimage authentication: A new framework and an implementation,'' IEEETrans. Image Process., vol. 15, no. 4, pp. 1042_1049, Apr. 2006.

[25] J. Tian, ``Wavelet-based reversible watermarking for authentication,''Proc. SPIE, vol. 4675, pp. 679_690, Apr. 2002.

[26] J. Tian, ``Reversible data embedding using a difference expansion,''IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890_896,Aug. 2003.