International Journal of Autonomous and Adaptive Communications Systems

ISSN online: 1754-8640 - ISSN print: 1754-8632
https://www.inderscience.com/ijaacs

Reversible data hiding algorithm in encrypted images using adaptive total variation and cross-cyclic shift

Mingfang Jiang

DOI: 10.1504/IJAACS.2023.10060344

Article History:

Received:	31 August 2023
Last revised:	21 September 2023
Accepted:	23 September 2023
Published online:	14 November 2023

Reversible data hiding algorithm in encrypted images using adaptive total variation and cross-cyclic shift

Mingfang Jiang

School of Computer Science,
Hunan First Normal University,
Changsha, 410205, China
Email: bingyuejiang@126.com

Abstract: To reduce prediction error and improve the embedding rate of secret messages, a new reversible data hiding algorithm for encrypted images (RDHEI) is designed using adaptive total variation and cross-cyclic shift called RDHEIAC. The adaptive total variation is first used to generate the prediction error image. Then, the bit-plane rearrangement based on Hilbert scanning is performed to make room for data hiding. Thirdly, an improved run-length encoding is applied to compress the error image. Subsequently, a cross-cyclic shift operation and a diffusion operation based on chaotic maps are used to produce encrypted images. Finally, secret messages are inserted into the encrypted image using bit substitution. Experimental results indicate that the proposed RDHEI scheme is privacy secure and has high embedding capacity and image fidelity. Moreover, information extraction and image restoration are separable. Compared with previous RDHEI algorithms, the proposed RDHEI algorithm has a 47.07% higher embedding rate.

Keywords: RDH; reversible data hiding; image encryption; adaptive total variation; run-length encoding; cross-cyclic shift; bit-plane rearrangement; Hilbert scanning; data hiding; bit substitution; Huffuman coding.

Reference to this paper should be made as follows: Jiang, M. (2023) 'Reversible data hiding algorithm in encrypted images using adaptive total variation and cross-cyclic shift', *Int. J. Autonomous and Adaptive Communications Systems*, Vol. 16, No. 6, pp.611–631.

Biographical notes: Mingfang Jiang received an MS in Computer Application from Hunan University, China, in 2011. She is currently an Associate Professor at Hunan First Normal University, China. Her research interests include information security, information management, and multimedia signal processing.

1 Introduction

Information hiding technology is the technology of hiding secret data into the carrier to achieve covert communication without causing a noticeable impact on the carrier data and without affecting the value of the carrier data (Kishk and Javidi, 2002). However, information hiding technology can cause certain changes to the carrier when embedding

information, which inevitably leads to a certain degree of information loss (Chen and Yan, 2023; Meng et al., 2023; Yamni et al., 2022). To overcome this disadvantage, reversible data hiding (RDH) technology has emerged (Huang et al., 2016; Wang et al., 2012). Unlike traditional information hiding techniques, RDH can not only accurately extract secret information, but also be used to verify and protect the originality and integrity of original data. In other words, RDH refers to the process of embedding additional data into the carrier data while maintaining the ability to completely recover the original carrier data without any destruction (Chang et al., 2021; Di et al., 2019). Therefore, RDH technology has important practical application value in some application scenarios that require high information integrity, such as medical imaging, military communication, and other fields (Huang et al., 2023; Kim et al., 2019; Zhang et al., 2023).

With the rapid development of cloud computing and big data, more and more enterprises and individuals are shifting their data storage and processing work to the cloud (Wang et al., 2012). Cloud computing provides users with more efficient, convenient, and economical data management and processing solutions. However, storing user data in the cloud poses issues of data security and privacy leakage. Therefore, for more sensitive data, stricter protection measures need to be taken. Encryption is an important method for protecting data confidentiality and user privacy (Behnia et al., 2020; Gai et al., 2017). In some public cloud storage applications, the content owner of multimedia data may not trust the information hiding party, which requires the content owner to encrypt the original data before uploading the data. Therefore, to facilitate the management of users' encrypted data, data managers want to hide information in the encrypted data, and reversible data hiding technology in encrypted images (RDHEI) has emerged (Arai and Imaizumi, 2022; Ge et al., 2023; Rai et al., 2023). Unlike traditional RDH technology, RDHEI technology can extract secret information without decryption. By applying RDHEI technology, users can extract confidential information without disclosing the plaintext. This technology can be used to protect user privacy data, such as personal identity information, medical records, financial records, etc. In multimedia data, digital images are one of the most commonly used information hiding carriers due to their large storage capacity, easy transmission, and easy acquisition. This paper focuses on secure communication and privacy protection of digital images and studies the RDH technology in encrypted images. In recent years, several RDH schemes have been proposed specifically for encrypted images. These schemes aim to provide a way to embed additional data into the encrypted image without decrypting it, thus preserving the confidentiality of the original image. Since the information entropy of encrypted images is approaching its maximum, it is more difficult to hide secret information in encrypted images. Current RDHEI schemes are mainly divided into two types: vacating room after encryption (VRAE) (Gao et al., 2022; Ge et al., 2023; Qian et al., 2014; Rai et al., 2023) and reserving room before encryption (RRBE) (Cao and Zhou, 2016; Dragoi and Coltuc, 2021; Ren et al., 2023). However, most RDHEI algorithms currently have shortcomings such as low security or low hidden capacity. To improve hiding capacity and enhance algorithm security, this paper proposes an RDHEI scheme that employs pixel prediction and cross-cyclic shift, The contributions of this paper are described as follows.

1 We propose a RDH algorithm in encrypted images using adaptive pixel prediction and chaotic image encryption. Adaptive pixel prediction based on total variation and adaptive run-length encoding achieve a high hiding capacity and chaotic image encryption ensures high security.

2 We design a pixel prediction method based on adaptive total variation. It has a very small prediction error due to the use of differential curvature and gradient norm during the pixel prediction.

3 Rearrangement of bit-planes based on chaotic map and Hilbert scanning are designed to compress the prediction error image and make room for data hiding.

4 A secure image encryption approach is developed. It employs a Logistic chaotic map to scramble the image and exploits cross-cyclic shift operation to diffuse the image.

5 A lot of experiments are conducted to validate that the proposed RDHEI algorithm outperforms existing RDHEI schemes in terms of embedding rate.

The rest of this paper is organised as follows. The related work is reported in Section 2. The proposed RDH algorithm in encrypted images is described in Section 3. Section 4 shows our experimental results and analysis. Finally, we conclude this paper in Section 5.

2 Related works

To meet users' management needs for massive encrypted image resources in cloud computing environments, researchers have developed many RDH algorithms in encrypted images which embed secret information into encrypted images.

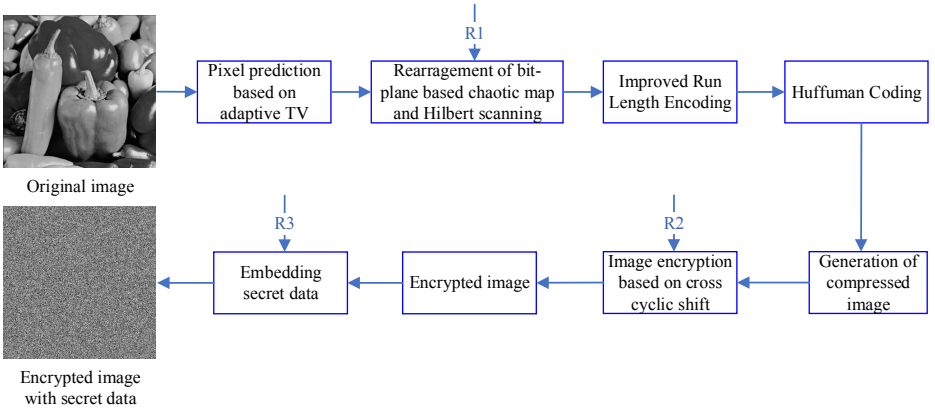
Puech et al. (2008) first proposed a new idea of embedding secret data in encrypted images. It hides secret messages by modifying only one bit in each block of n pixels. Zhang (2011) proposed a RDH scheme for encrypted images. In the scheme, the additional data is embedded into the encrypted image by modifying a small proportion of encrypted data. Users with the data-hiding key can extract the secret data and recover the original image with the aid of spatial correlation. Qian et al. (2014) proposed an RDH framework in an encrypted JPEG bitstream. It identifies usable bits suitable for data hiding and embeds a secret message encoded with error correction into the encrypted bitstream by slightly adjusting the JPEG stream. A receiver with encryption and hiding keys can perfectly extract the secret bits by analysing the blocking artefacts of the neighbouring blocks. Achuthshankar et al. (2015) presented a data hiding approach for encrypted images using the AS algorithm which extracts the hidden data after applying the decryption algorithm. Cao and Zhou (2016) developed an RDHEI algorithm using the rhombus prediction model and difference histogram shifting. The original image is encrypted by homomorphism encryption and the secret messages are embedded in encrypted images difference using histogram shifting. Liu et al. (2016) proposed an ROI-based RDH scheme for encrypted medical images. In the scheme, the least significant bits (LSB) of the encrypted image ROI and electronic patient record (EPR) are concatenated and embedded into the encrypted image. Chen and Chang (2019) exploit the spatial correlation in the original images to reserve room for hiding data before image

encryption. To achieve high embedding capacity, extended run-length coding and a block-based most significant bit (MSB) plane rearrangement mechanism vacate room for data embedding. Yao et al. (2019) designed a content- adaptive RDHEI scheme using the visual perceptual model. To achieve both good watermark transparency and satisfactory watermarked image quality, watermark embedding positions are adaptively selected according to visual perception of the original image before encryption. The traditional RDH algorithm is used to vacate room for data hiding before encryption, and additional data is embedded in encrypted images by bit substitution in chosen pixel positions. Wang et al. (Li et al., 2020) proposed an RDHEI scheme based on multikey encryption. In the scheme, the original image can be encrypted and embedded at the same time. Additional messages can be extracted after image decryption using spatial correlation. The original image can be losslessly recovered. Wu et al. (2020) employ parametric binary tree labelling to design an RDHEI scheme, which takes advantage of the spatial correlation in the whole original image to vacate room for data hiding. The encrypted pixels are labelled into two different categories using a parametric binary tree. The bit substitution is employed to embed secret messages into one of the two categories of encrypted pixels. To improve hiding capacity, Tang et al. (2021) proposed an RDHEI algorithm based on adaptive prediction error coding. The algorithm preserves the spatial correlation of the original image in the encrypted domain using a block-based encryption approach and exploits an adaptive prediction error coding to reserve room for data hiding. In Nguyen et al.'s scheme (Nguyen et al., 2022), the data hiding space is vacated for the embedding of secret data after image encryption by representing each chosen block with an absolute moment block truncation coding (AMBTC) code. The receiver can not only separately extract the secret data, but also recover the original image losslessly. Bhardwaj and Niranjan proposed a RDHEI algorithm based on hierarchical absolute moment block truncation coding (HAMBTC) (Bhardwaj and Niranjan, 2023). In the algorithm, both low mean tables and high mean tables are first encrypted, and then secret data is inserted into the quintuplet resolving underflow and overflow problems. An RDHEI method based on a 2D chaotic system and full bit-plane search (FBPS) is proposed in Ge et al. (2023). It uses pseudo-random sequences produced by chaotic systems for inter-block permutation and intra-block diffusion. This special encryption strategy can preserve the pixel correlation within a block and ensure high security. The FBPS technique is used to detect all smooth bit-planes. During data embedding, '0' and '1' are used to denote all smooth and rough bit-planes respectively. In Sui et al.'s RDHEI algorithm (Sui et al., 2023), to vacate room for data embedding, the MSB of each pixel was adaptively predicted and marked by Huffman coding. Additional data is embedded in the vacated space of the original image after the stream cipher. To meet the need for multiple data hiders, Yu et al. (2023) introduce pixel value order (PVO) and secret image sharing (SIS) to the RDHEI schemes, which ensures secure image communication by the combination of stream encryption based on chaotic systems with secret sharing reinforced by the Chinese remainder theorem (CRT). However, existing RDHEI schemes still suffer from low hiding capacity, which limits their applicability. Besides, such algorithms still lack a well-done mechanism to ensure their high security.

3 Proposed RDHEI algorithm

This proposed RDHEI algorithm includes three stages (illustrated in Figure 1): image encryption, data hiding, information extraction, and image restoration. In the image encryption stage, the data owner first calculates the predicted pixels through adaptive total variation to obtain the prediction error. Then, a Logistic chaotic map and Hilbert scanning are used to randomly rearrange the prediction error bit-planes to form a bit stream. Then, the improved run length encoding is used to compress the bitstream. All eight bit-planes are classified into compressed bit-planes and uncompressed bit-planes. The image can be reconstructed by sequentially storing auxiliary information and encoding results in the compressed bit-planes. Meanwhile, the uncompressed bit-planes remain unchanged. Finally, the compressed image pixels are encrypted using cross-cyclic shift and XOR-based diffusion.

Figure 1 The framework of the proposed RDHEIAC scheme (see online version for colours)



In the data hiding stage, the data hider embeds secret information in the vacated bit space of the compressed bit-planes with bit replacement.

In the stage of information extraction and image restoration, the receiver sequentially extracts the labelled bits of the bit-plane, compresses the size of the hidden space of the bit-plane, extracts secret information based on different keys, and finally recovers the original image through improved run length encoding and decoding operations.

3.1 Prediction error encoding based on adaptive total variation

3.1.1 Pixel prediction based on adaptive total variation

The total variation image denoising model was proposed by Rudin et al. (1992), and it has become one of the most successful methods in image denoising and image restoration. This model fully utilises the inherent regularity of natural images, which makes it easy to reflect the geometric regularity of real images from the de-noising images, such as the smoothness of boundaries. In the proposed RDHEI algorithm, we use differential curvature and gradient modulus to design an improved L_p norm-based image

prediction model. Let u be the predicted image and v be the input image (including noise). The image-denoising model based on L_p norm is defined as follows.

$$\min_u E_2(u) = \frac{1}{p}\iint_\Omega |\nabla u|^p \mathrm{d}x\mathrm{d}y + \frac{\lambda}{2}\iint_\Omega |u-v|^2 \mathrm{d}x\mathrm{d}y \tag{1}$$

where $p(x,y) = \begin{cases} 2, d(x,y) \le \tau \\ 1 + \dfrac{1}{1+\left(\left|\nabla \mathrm{D}\right| / k\right)^{1+d(x,y)}}, d(x,y) > \tau \end{cases}$, $d(x,y)$ is normalised differential curvature D, and τ is the threshold of the normalised differential curvature $d(x,y)$.

$$D = \left\| u_{\eta\eta} \right| - \left| u_{\xi\xi} \right\| \tag{2}$$

$$u_{\eta\eta} = \frac{u_x^2 u_{xx} + 2u_x u_y u_{xy} + u_y^2 u_{yy}}{u_x^2 + u_y^2} \tag{3}$$

$$u_{\xi\xi} = \frac{u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}}{u_x^2 + u_y^2} \tag{4}$$

The threshold k can be estimated by variance,

$$k = \sigma_{ij}^2 + \sigma_{pq}^2 \tag{5}$$

σ_{ij}^2 and σ_{pq}^2 are the variances of pixels (i, j) and their neighbouring pixels, respectively.

The differential curvature D can distinguish between smooth areas, edge areas, and isolated noise points in an image. In the edge zone, the D value is relatively high while in the smooth region, the D value is smaller. At isolated noise points, the D value is almost zero. Therefore, normalised differential curvature d can be used to distinguish noise points. The use of this prediction model can protect edges and smooth noise, resulting in predicted images with satisfactory visual quality. Figure 2 shows a prediction example of Peppers image using pixel prediction based on adaptive total variation, Figure 2(a) and 2(b) are the original image of Peppers and its predicted image respectively. The prediction error image as shown in Figure 2(c) is obtained by subtracting the original image and the predicted one. The predicted image has good visual quality with a PSNR of 34.4711 dB and an SSIM of 0.8995.

3.1.2 Rearrangement of bit-planes and adaptive run length encoding

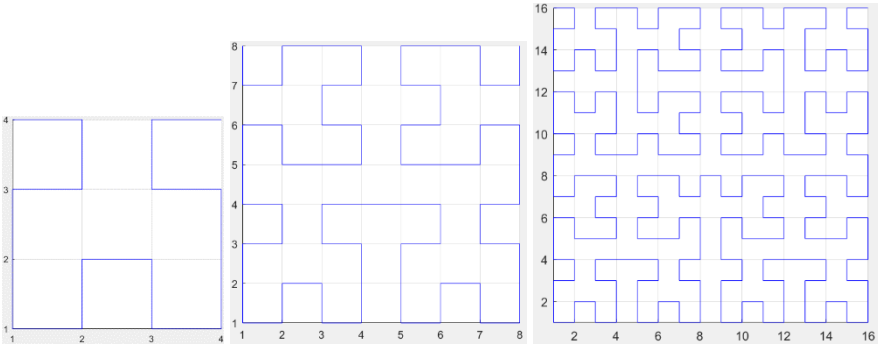
The Hilbert curve can be considered as a map from an N-dimensional (usually two-dimensional) space to a 1D space. C. Gostman and M. Lindenbaum proved that it is the best curve among all scanning curves to maintain spatial adjacency. The Hilbert curve has been widely used in applications such as image scanning and image compression. The proposed RDHEI algorithm utilises the excellent pixel adjacency preserving characteristic of Hilbert scanning to achieve a high compression ratio. Figure 3 shows the

process of iterating a second-order Hilbert curve into a higher-order curve through regular changes.

Figure 2 Prediction error image: (a) peppers; (b) predicted peppers and (c) prediction error image

(a) (b) (c)

Figure 3 The generation process of N-order (N = 2, 3, 4) Hilbert scanning curves (see online version for colours)



To fully utilise pixel correlation and improve the compression ratio of bit-planes, Hilbert scanning is used for scrambling, and improved run length encoding is used for bit encoding. The specific steps are as follows.

Step 1: Given the prediction error image I_d as input, Hilbert scanning is performed on the predicted error image I_d, and a 1D sequence I_h is obtained.

Step 2: The hash code K_1 with 256 bits is produced by applying the SHA 256 hash function.

Step 3: Given the user's secret key u_0 and x_0, we obtain the control parameters and initial values $(\dot{u}_t, \dot{x}_t), t = 0,1,2$ of the Logistic map, and these parameters and initial values are dependent on the plain image.

Step 4: Apply keystream $(\dot{u}_0, \dot{x}_0)$ to Logistic map and iterate the map $IT + M \times N$ times to obtain the sequence P_1. The first $IT = 1000$ values are discarded in the algorithm. The

bit number R_1 of circular shift can be generated from the sequence P_1 which are defined by equation (6).

$$R1(i) = mod\left(floor\left(P1(i)\times 10^{15}\right),8\right)+1, i = 0,1,\ldots,M\times N-1 \tag{6}$$

where $mod(x,y)$ returns the remainder of x divided by y, and $floor(x)$ returns the largest integer value that is equal to or less than x.

Step 5: Perform a random cyclic shift operation $CRShift\left(I_h(i),R1(i)\right)$ on each element in the sequence I_h and generate the scrambling sequence I_s. Where the operation $CRShift(x,y)$ denotes a bitwise right circular shift on the sequence x of y bits.

Step 6: Convert the scrambling sequence I_s into a 2D image I_2, and perform bit-plane decomposition on I_2, multiple bit-planes are generated.

Step 7: Convert the bit planes into a 1D bitstream using raster scanning, subsequently perform an improved run length encoding to the 1D bitstream and produce a compressed bitstream I_b.

Step 8: Divide the compressed bit stream I_b into multiple segments, and calculate the run length $A(i), i = 0,1,\ldots,n-1$ of each group (denoted as $PB_i, i = 0,1,\ldots,n-1$) based on the partition. Calculate the median N_d of the run length sequence $\boldsymbol{A}$. Each group is divided into the short-bit group Q_1 and the long-bit group Q_2 according to the absolute difference between the run length and the median N_d of each group.

$$PB_i \in \begin{cases} Q_1, & \text{if } \left|A(i)-N_d\right| \le 0 \\ Q_2, & \text{else} \end{cases} \tag{7}$$

Step 9: Calculate the frequency of each category PB_i, generate a Huffman code table, rescan the bitstream I_b, and encode all short and long bitstreams based on the Huffman code table.

Step 10: Repeat steps 7–9 until all bit planes are encoded.

Step 11: Replace the vacated bit space of all bit planes with 0 to obtain a compressed image I_c. Simultaneously auxiliary messages, including compressibility label F, the size of the information block B_s used for data hiding, and starting position B_p of the hidden blocks, are embedded into the first three bytes of the plane (excluding MSB and LSB bit planes).

3.2 Image encryption

To enhance the security of the encryption process, we design an image encryption method based on a cross-cyclic shift to scramble pixel positions.

Step 1: Apply $(\dot{u}_1,\dot{x}_1)$ to Logistic map and iterate the map $IT + M\times N$ times to obtain the sequence P_2. The pseudo sequence R_2 for permutation can be generated as follows.

$$R2(i) = mod\left(floor\left(P2(i)\times 10^{15}\right), M\times N\right), i = 0,1,\ldots, M\times N-1 \tag{8}$$

Step 2: Convert the compressed image I_c into a 1D sequence I_{c1}.

Step 3: Apply pixel cyclic shift operation. The operation can be described below.

$$\grave{I}_{c1}(i) = I_{c1}\left(i + mod\left(Rc(i) + R2(i-1), M\times N - i\right)\right) \tag{9}$$

$$R_c(i) = mod\left(floor\left(LM\left(\frac{i}{max(M,N)}\right)\times 10^{15}\right), M\right), i = 0,1,\ldots, M\times N-1 \tag{10}$$

where the parameter $R_c(i)$ is produced by the Logistic chaotic map, $LM(x)$ denotes the Logistic map.

Step 4: Apply $(\dot{u}_2, \dot{x}_2)$ to Logistic map and iterate the map $IT + M\times N$ times to obtain the sequence P_3. The pseudo sequence R_3 for diffusion can be produced as follows.

$$R3(i) = mod\left(floor\left(P3(i)\times 10^{15}\right), 256\right), i = 0,1,\ldots, M\times N-1 \tag{11}$$

Step 5: Conduct diffusion operation on the scrambled image $\grave{I}_{c1}$.

$$\begin{cases} \hat{I}_{c1}(0) = mod\left(R3(0) + \grave{I}_{c1}(0), 256\right) \\ \hat{I}_{c1}(i) = mod\left(R3(i) + \grave{I}_{c1}(i) + \grave{I}_{c1}(i-1), 256\right) \otimes \hat{I}_{c1}(i-1) \end{cases} \tag{12}$$

where $\otimes$ denotes exclusive operator.

Step 6: Convert 1D encrypted sequence $\grave{I}_{c1}$ to 2D sequence I_e, that is the encrypted image containing secret messages.

3.3 Data hiding

This proposed RDHEI algorithm embeds secret messages into the reserved bit space. The detailed process is described as follows.

Step 1: Auxiliary information, including compressibility label *F*, the size of the information block B_s used for data hiding, and starting position B_p of the hidden blocks are extracted from the bit planes of the encrypted image.

Step 2: Extract auxiliary information according to the compressibility label *F*. If $F = 1$, the start position and the bits number of the embedded secret information can be obtained according to the start position B_p of the hidden block and the size B_s of the information block.

Step 3: Apply the key group $(\dot{u}_3, \dot{x}_3)$ to Logistic map and iterate the map $IT + num$ times to obtain the sequence P_4. The pseudo sequence R_4 for encryption of the secret data can be produced as follows. The parameter num is the length of the secret data.

$$R4(i) = mod\left(floor\left(P_4(i) \times 10^{15}\right), 256\right), i = 0,1,\ldots, M \times N - 1 \tag{13}$$

Step 4: Finally, the secret messages after chaotic encryption are embedded into the vacated space of each compressed bit plane and generate an encrypted image containing secret data.

3.4 Data extraction and image restoration

There are 4 key streams used in the proposed RDHEI scheme. The key streams $(\dot{u}_t, \dot{x}_t), t = 0,1,2$ is used for image scrambling and diffusion, and the keystream $(\dot{u}_3, \dot{x}_3)$ is employed for encryption of the secret data during data hiding. The number of keys obtained by the receiver is crucial for successfully extracting secret information and restoring the original image. According to the different keys owned by users, the following three situations may occur.

1 If the receiver only receives the information hiding key $(\dot{u}_3, \dot{x}_3)$, then the extraction of secret information can be performed, and the extracted secret information will be completely correct. The compressibility labels *F* of all bit planes are first extracted to determine whether each bit plane is compressed. Then, the starting position for data hiding in each compressed bit plane is read and all encrypted secret messages are subsequently extracted. Finally, perform decryption operations on the extracted encrypted secret messages with hiding key $(\dot{u}_3, \dot{x}_3)$ to recover the original secret information. But if there is no encryption key $(\dot{u}_t, \dot{x}_t), t = 0,1,2$, it is impossible to restore the original image.

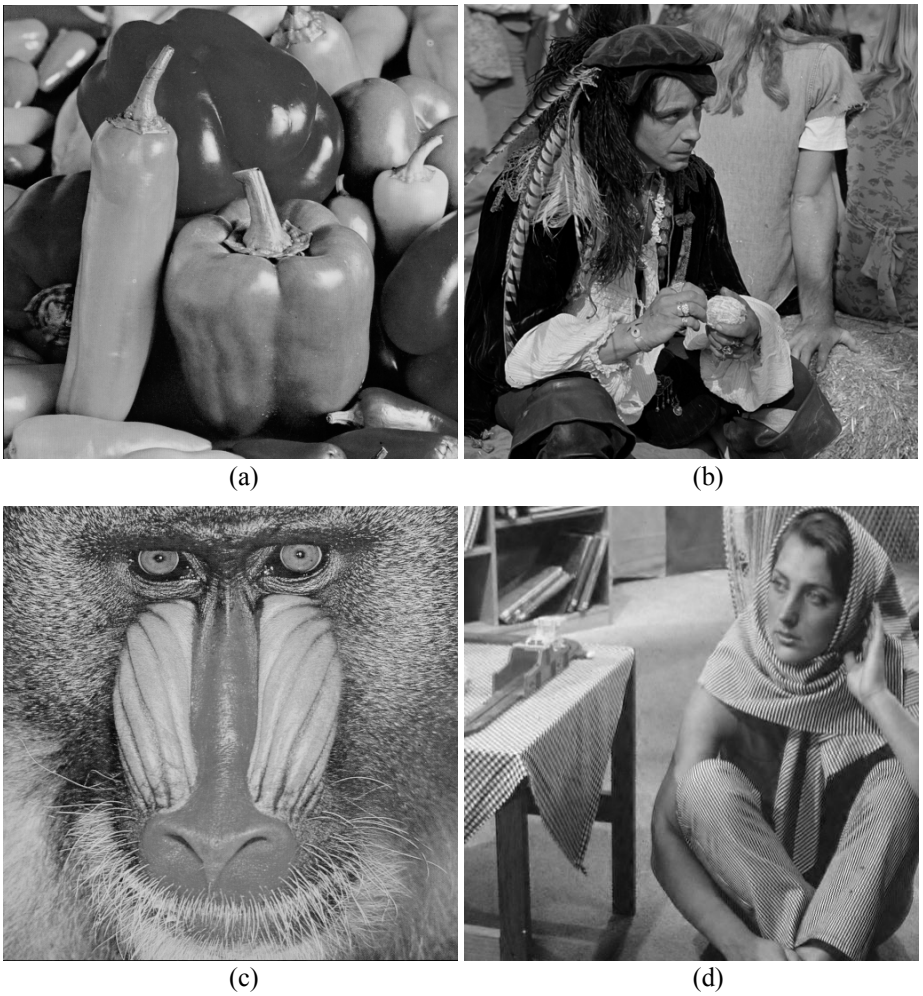
2 If the receiver only receives the encryption key $(\dot{u}_t, \dot{x}_t), t = 0,1,2$, then perfect original image reconstruction can be achieved. Firstly, extract the compression labels *F* and the size B_s of the vacated hiding space, and find the starting position of data embedding in each compressed bit plane. So, all secret information can be extracted using these auxiliary data. Then, generate pseudo-random number matrices R_1, R_2, R_3 with the encryption key $(\dot{u}_t, \dot{x}_t), t = 0,1,2.$ Perform reverse diffusion and descrambling operations on encrypted images to successfully decrypt the image. Finally, extract the compressed bitstream and decompress it to produce a prediction error image. Thus, the plaintext image is losslessly recovered. But if there is no key $(\dot{u}_3, \dot{x}_3)$, the extracted secret information is incorrect.

3 When the receiver has both encryption and hiding keys, He can accurately extract secret data and losslessly restore the original image.

4 Experiments and analysis.

In the experiments, the test images are greyscale images with a size of 512×512, which are sourced from the USC-SIPI Image Dataset (USC-SIPI, 1977). Some of these test images are shown in Figure 4. Taking the Peppers image as an example, the encryption and data-hiding process of the proposed ADHEIAC algorithm is illustrated in Figure 5. The encrypted image is shown in Figure 5(a), and the encrypted image containing secret data is shown in Figure 5(b). When the receiver only has the encryption key, the decrypted image is shown in Figure 5(c). If the receiver has both the encryption key and the hidden key, the original image can be obtained after secret data extraction, as shown in Figure 5(d).

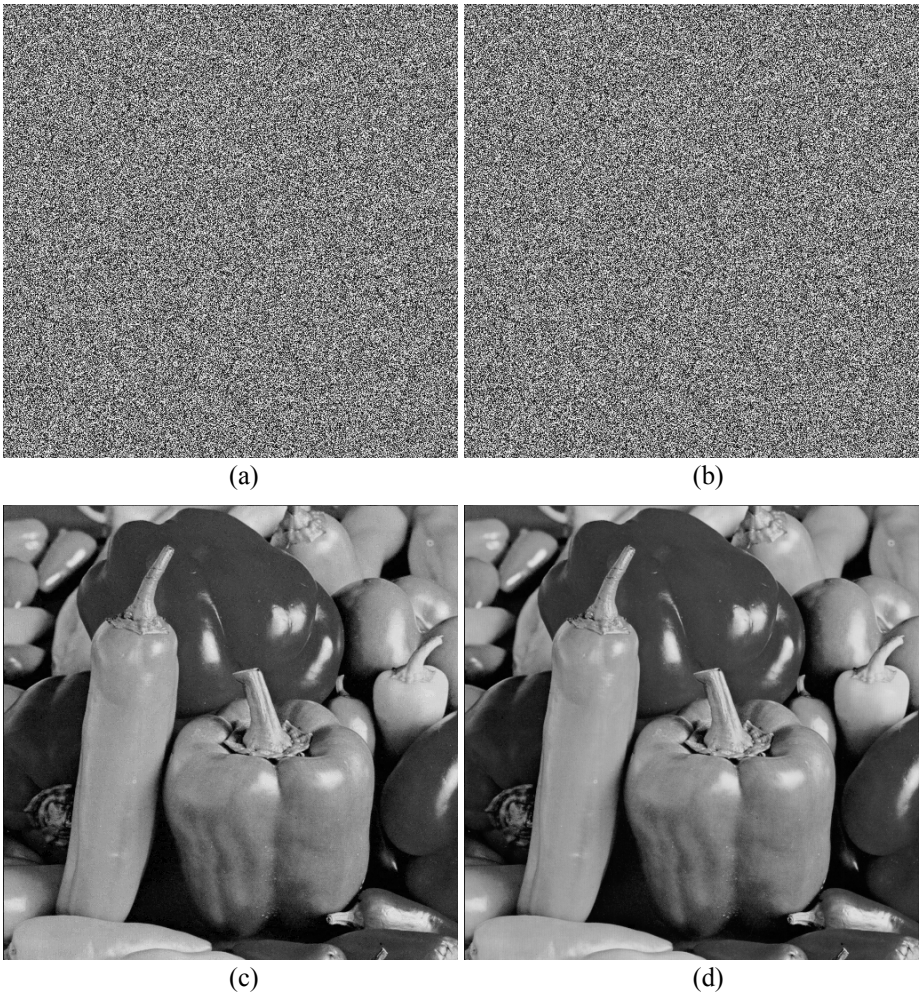
Figure 4 Some test images from the USC-SIPI image dataset: (a) peppers; (b) man; (c) baboon and (d) Barbara



(a) (b)

(c) (d)

Figure 5 An example of image encryption and data hiding: (a) encrypted image; (b) encrypted image containing secret data; (c) directly decrypted image and (d) recovered image



From Figure 5, one can find that the encrypted image shown in Figure 5(a) is It is a completely random, noise-like image due to the cyclic cross scrambling and the image diffusion based on chaos during the encryption phase. Authorised users with the correct key can correctly decrypt and lossless restore the original image.

To objectively evaluate the reversibility of the proposed algorithm, PSNR and SSIM are used to evaluate the quality of the restored image, and the results are listed in Table 1. From Table 1, it can be seen that the PSNR values of the restored images of all test images tend to infinity, and their SSIM values are all 1. This indicates that the proposed RDHEIAC algorithm can losslessly restore the original images for different types of images.

Table 1 PSNR and SSIM of restored images

Test images	*PSNR*	*SSIM*
Peppers	+∞	1
Man	+∞	1
Baboon	+∞	1
Barbara	+∞	1

4.1 Correlation analysis

In general, there is a high correlation between adjacent pixels in plaintext images. To achieve a good encryption effect, it is necessary to eliminate the correlation between pixels in ciphertext images. The correlation coefficient is defined as follows.

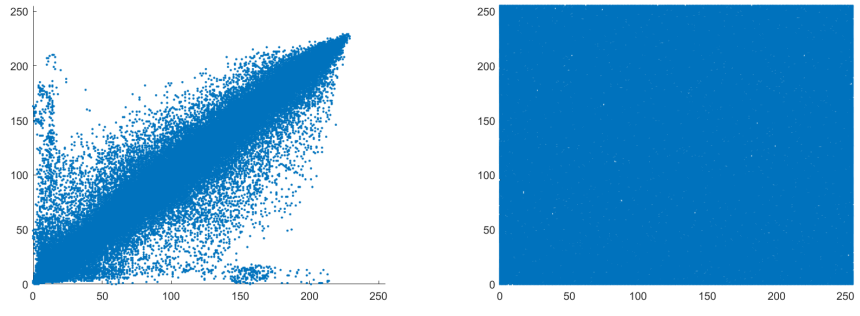
$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)D(y)}},\ r_{xy} \in [0,1] \tag{14}$$

$$cov(x,y) = E\{(x - E(x))(y - E(y))\} \tag{15}$$

where x and y are grey values of two adjacent pixels in the image, and E(x) and D(x) are the expectation and variance of variable x, respectively.

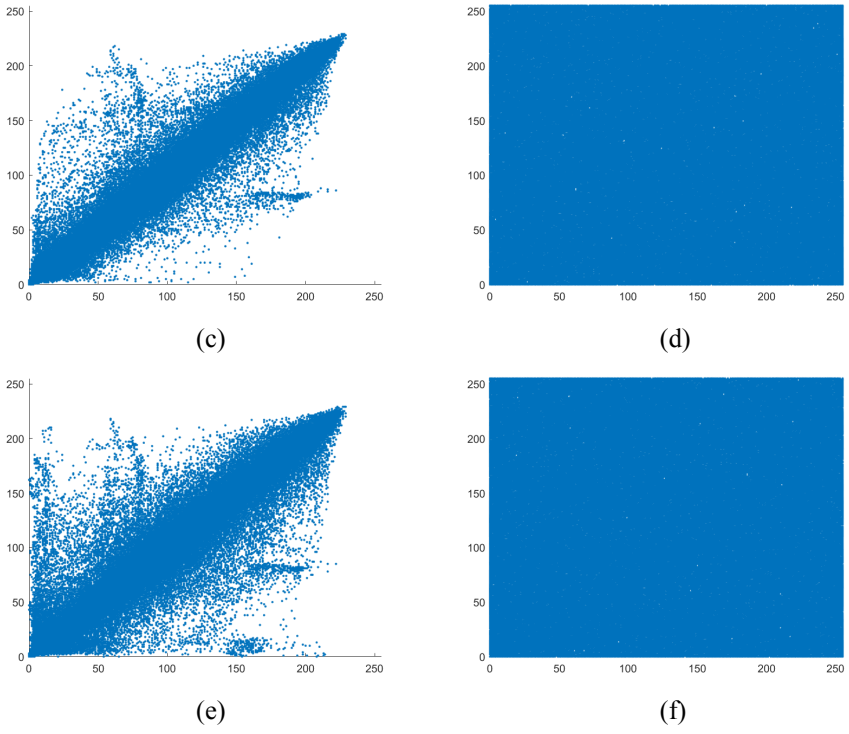
Figure 6 shows the correlation distribution of the Peppers greyscale image and its ciphertext image in different directions. Figure 6(a), (c), and (e) are correlation distributions of adjacent pixels along the horizontal, vertical, and diagonal directions in the original image. Similarly, Figure 6(b), (d), and (f) are correlation distributions of adjacent pixels along the horizontal, vertical, and diagonal directions in the encrypted image. It can be seen that after encryption, the points are fairly scattered within the correlation distribution plot, the encryption process effectively weakens the correlation between the adjacent pixels.

Figure 6 Correlation distributions of the adjacent pixels before and after encryption: (a) horizontal in original image; (b) horizontal in encrypted image; (c) vertical in original image; (d) vertical in encrypted image; (e) diagonal in original image and (f) diagonal in encrypted image (see online version for colours)



(a) (b)

Figure 6 Correlation distributions of the adjacent pixels before and after encryption: (a) horizontal in original image; (b) horizontal in encrypted image; (c) vertical in original image; (d) vertical in encrypted image; (e) diagonal in original image and (f) diagonal in encrypted image (see online version for colours) (continued)



Detailed correlation coefficients of adjacent pixels in original images and encrypted images are shown in Table 2. From Table 2, it can be found that the correlation coefficients of adjacent pixels of the encrypted image are very small. Thus, it is hard to detect correlations between the original image and its corresponding encrypted image, implying that the proposed scheme is secure against statistical attacks.

4.2 Histogram analysis

The histogram of an image reflects the distribution of all pixel values in the image. If the pixel values of the plaintext image are not changed, it will leak the pixel distribution information of the image. A good encryption scheme should generate ciphertext images with pixel values that are evenly distributed between 0 and 255. Thus, it is difficult to statistically analyse these ciphertext images. Figure 7 shows a comparison of histograms of four different test images before and after encryption.

Table 2 Correlation coefficients of the adjacent pixels before and after encryption

Images	*Directions*	*Plaintext images*	*Ciphertext images*
Peppers	Horizontal	0.9755	0.0002
	Vertical	0.9808	–0.0034
	Diagonal	0.9625	0.0016
Man	Horizontal	0.9623	0.0016
	Vertical	0.9699	0.0011
	Diagonal	0.9417	0.0020
Baboon	Horizontal	0.8653	–0.0031
	Vertical	0.7524	–0.0029
	Diagonal	0.7191	0.0008
Barbara	Horizontal	0.8913	–0.0026
	Vertical	0.9555	–0.0014
	Diagonal	0.8768	–0.0006

Figure 7 Histogram comparison of 4 test images before and after encryption: (a) peppers before encryption; (b) peppers after encryption; (c) man before encryption; (d) man after encryption; (e) baboon before encryption; (f) baboon after encryption; (g) Barbara before encryption and (h) Barbara after encryption (see online version for colours)

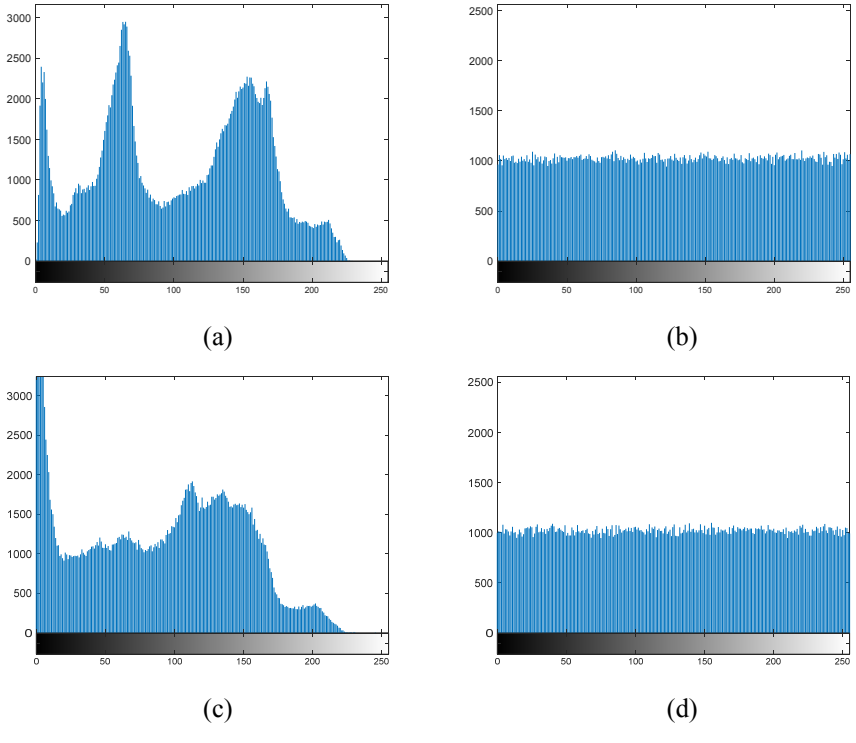
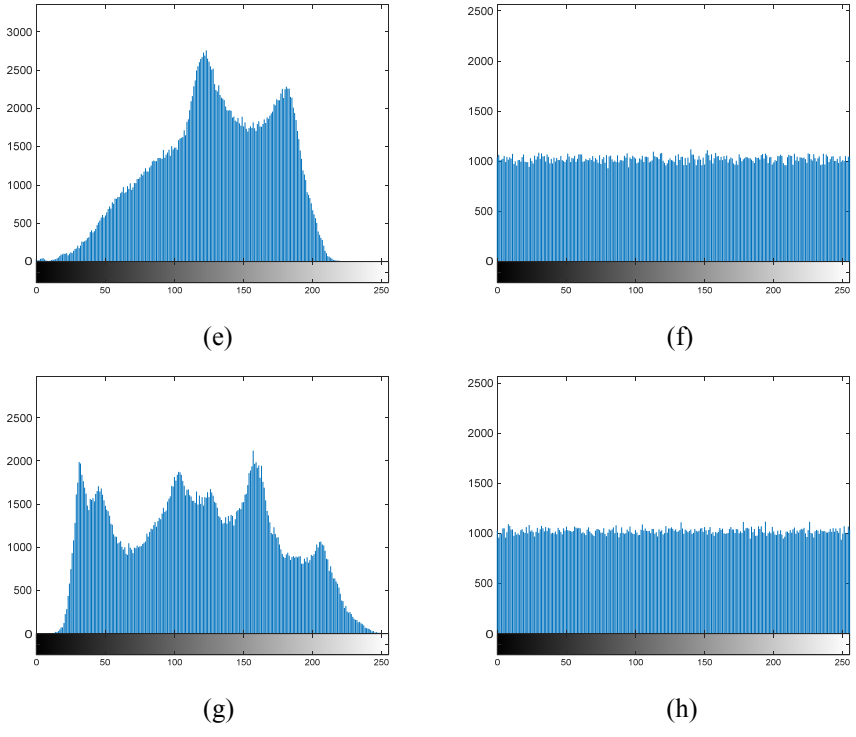


Figure 7 Histogram comparison of 4 test images before and after encryption: (a) peppers before encryption; (b) peppers after encryption; (c) man before encryption; (d) man after encryption; (e) baboon before encryption; (f) baboon after encryption; (g) Barbara before encryption and (h) Barbara after encryption (see online version for colours) (continued)



From Figure 7, it can be seen that the histogram distribution of plaintext images is relatively concentrated, while the histogram of ciphertext images is very evenly distributed between 0 and 255, indicating that our encryption algorithm has excellent diffusion characteristics and resistance to statistical analysis.

4.3 Differential attacks analysis

Generally, the attacker may notice the correlation between plaintext image and ciphertext image by the tiny change of plaintext image. In a good image encryption algorithm, a slight change in the plaintext image should cause some large changes in the ciphertext image. The number of changing pixel rates (NPCR) and the unified averaged changed intensity (UACI) are the two most common measures to evaluate the strength of image encryption algorithms. They are calculated as follows.

$$\begin{cases} NPCR = \dfrac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \\ D(i,j) = \begin{cases} 1, \; if \; C_1(i,j) \neq C_2(i,j) \\ 0, \; else \end{cases} \end{cases} \tag{16}$$

$$UACI = \frac{1}{M \times N}\left(\sum_{i,j}\frac{\left|C_1(i,j) - C_2(i,j)\right|}{255}\right) \times 100\% \tag{17}$$

where C_1 and C_2 are ciphertext images obtained from plaintext images that have a slight change in one pixel.

A secure algorithm must reach 99% of NPCR value and 33% of UACI value, indicating that the encryption algorithm is key sensitive to both plaintext and ciphertext images. Tests are performed on the USC-SIPI Image Dataset, and the corresponding results are presented in Table 3. It achieves values of the NPCR indicator between 99.5953% and 99.6323% and between 32.8216% and 34.7557% for the UACI indicator. This indicates that the proposed RDHEIAC algorithm has a good ability to resist differential attacks.

Table 3 NPCR and UACI results for various test images

Test images	*NPCR*	*UACI*
Peppers	99.6323	33.6709
Man	99.6017	34.7557
Baboon	99.6006	32.8216
Barbara	99.5953	33.1143

4.4 PSNR analysis of directly encrypted images

Information hiding usually involves slight modification of the carrier image, but at the same time, it requires data hiding to be imperceptible, meaning that changes to the carrier are noticeable. Transparency is subjectively perceived through human eye observation, and objectively, the visual quality of directly decrypted carrier images can be evaluated by the peak signal-to-noise ratio (PSNR). Directly decrypted images contain secret information, and directly decrypted images with high PSNR values imply good transparency. The comparison of PSNR results between this algorithm and three other related schemes is shown in Table 4. From Table 4, it can be seen that compared with existing RDHEI algorithms, our RDHEIAC algorithm has better transparency.

Table 4 PSNR comparison of directly decrypted images under different algorithms

Test images	*Gao et al. (2022)*	*Chen and Chang (2019)*	*Wu et al. (2020)*	*Our scheme*
Peppers	38.29	44.58	45.32	52.05
Man	38.97	45.52	45.76	47.75
Baboon	38.64	40.94	41.21	42.76
Barbara	38.42	46.92	45.39	45.81

4.5 Embedding rate

The embedding rate is an important indicator for evaluating the capacity of secret data embedding in RDH algorithms. It represents the average number of bits embedded per pixel, expressed in bit per pixel (bpp), and its definition is shown as follows.

$$ER = \frac{B_m}{M \times N} \tag{18}$$

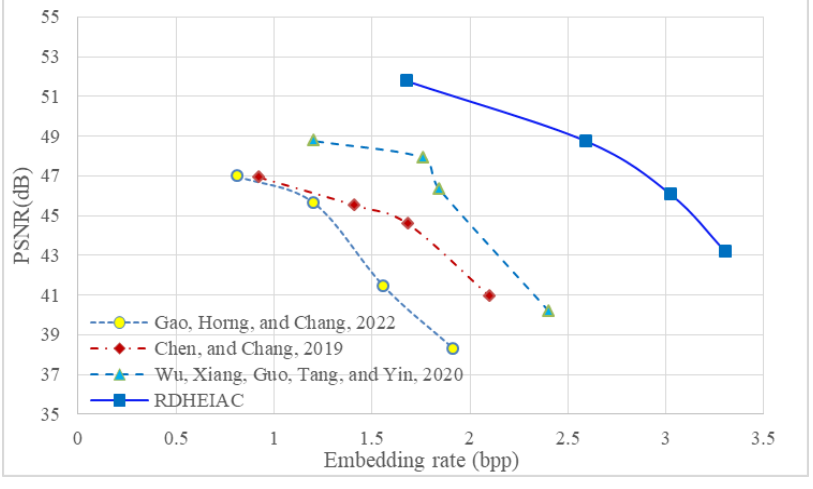
where B_m represents the maximum information capacity that the carrier image can embed, and $M \times N$ is the size of the carrier image. This evaluation indicator indicates that for the same carrier image, the larger the ER value, the greater the information capacity that can be embedded. The test results of different algorithms under different test images are shown in Table 5. From Table 5, it can be seen that the embedding rate of our RDHEIAC algorithm is higher than existing RDHEI algorithms, especially for Peppers and Barbara images, which has a significant improvement compared to existing schemes.

Table 5 Embedding rate of different algorithms

Test images	*Gao et al. (2022)*	*Chen and Chang (2019)*	*Wu et al. (2020)*	*Our algorithm*
Peppers	1.561	1.880	1.976	3.126
Man	1.617	1.683	1.923	2.891
Baboon	0.813	0.925	1.201	1.680
Barbara	1.202	1.409	1.944	3.304

In addition, Figure 8 shows the PSNR vs. embedding rate comparison of the proposed method with the state-of-the-art RDHEI methods for the Peppers image. It can be seen that the proposed RDHEIAC algorithm offers a 47.07% higher embedding rate under roughly the same PSNR value of decrypted images than existing RDHEI schemes.

Figure 8 PSNR vs. embedding rate of different algorithms (see online version for colours)



5 Conclusions

This paper proposes a novel RDHEI scheme using adaptive total variation and cross-cyclic shift. In the scheme, the adaptive total variation is used to produce the prediction error image. The rearrangement of bit planes and the improved Run-length encoding are employed to vacate room for hiding data. The cross-cyclic shift operation and chaos-based diffusion operation are exploited to achieve high security. The extraction of secret information and original image recovery is completely separable at the receiver side. The experimental results show that the proposed RDHEIAC algorithm has better visual quality of the decrypted image while ensuring a certain embedding rate. The future work directions include global optimisation of transparency, security, and hiding capacity, application expansion of RDHEI compatible with compressed images, and reversible visible watermarking in encrypted images.

Acknowledgement

This research was funded by the National Natural Science Foundation of China under Grant 61872408, the Natural Science Foundation of Changsha under Grant 2022199, and the Social Science Foundation of Hunan Province under Grant 19YBA098.

References

Achuthshankar, A., Arjun, K.P. and Sreenarayanan, N.M. (2015) 'Implementation of reversible data hiding in encrypted image using A-S algorithm', *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Greater Noida, Delhi, India, pp.1279–1284.

Arai, E. and Imaizumi, S. (2022) 'High-capacity reversible data hiding in encrypted images with flexible restoration', *Journal of Imaging*, Vol. 8, No. 7, pp.176, https://www.mdpi.com/2313-433X/8/7/176 (Accessed 2022).

Behnia, R., Ozmen, M.O. and Yavuz, A.A. (2020) 'Lattice-based public key searchable encryption from experimental perspectives', *IEEE Transactions on Dependable and Secure Computing*, Vol. 17, No. 6, pp.1269–1282.

Bhardwaj, R. and Niranjan, A. (2023) 'An improved dual image separable reversible data hiding algorithm for encrypted HAMBTC compressed images', *Multimedia Tools and Applications*, Vol. 82, No. 3, pp.3335–3362, https://doi.org/10.1007/s11042-022-13209-z (Accessed 2023).

Cao, L. and Zhou, H. (2016) 'A new reversible date-hiding algorithm for encrypted images', *Mathematical Problems in Engineering*, Vol. 2016, p.4313580, https://doi.org/10.1155/2016/4313580 (Accessed 2016).

Chang, C-C., Chang, J-F., Kao, W-J. and Horng, J-H. (2021) 'Two-layer reversible data hiding for VQ-compressed images based on de-clustering and indicator-free search-order coding', *Future Internet*, Vol. 13, No. 8, p.215.

Chen, K. and Chang, C-C. (2019) 'High-capacity reversible data hiding in encrypted images based on extended run-length coding and block-based MSB plane rearrangement', *Journal of Visual Communication and Image Representation*, Vol. 58, pp.334–344 https://www.sciencedirect.com/science/article/pii/S1047320318303493 (Accessed 2019).

Chen, T.H. and Yan, J.Y. (2023) 'Enhanced steganography for high dynamic range images with improved security and capacity', *Applied Sciences*, Vol. 13, No. 15, p.8865.

Di, F., Zhang, M., Huang, F., Liu, J. and Kong, Y. (2019) 'Reversible data hiding in JPEG images based on zero coefficients and distortion cost function', *Multimedia Tools and Applications*, Vol. 78, No. 24, pp.34541–34561 https://doi.org/10.1007/s11042-019-08109-8 (Accessed 2019).

Dragoi, I.C. and Coltuc, D. (2021) 'On the security of reversible data hiding in encrypted images by MSB prediction', *IEEE Transactions on Information Forensics and Security*, Vol. 16, pp.187–189.

Gai, K., Qiu, L., Chen, M., Zhao, H. and Qiu, M. (2017) 'SA-EAST: security-aware efficient data transmission for ITS in mobile heterogeneous cloud computing', *ACM Trans. Embed. Comput. Syst.*, Vol. 16, No. 2, Article 60, https://doi.org/10.1145/2979677 (Accessed 2017).

Gao, K., Horng, J-H. and Chang, C-C. (2022) 'High-capacity reversible data hiding in encrypted images based on adaptive block encoding', *Journal of Visual Communication and Image Representation*, Vol. 84, pp.103481, https://www.sciencedirect.com/science/article/pii/S1047320322000372 (Accessed 2022).

Ge, B., Ge, G., Xia, C. and Duan, X. (2023) 'High-capacity reversible data hiding in encrypted images based on 2D-HS chaotic system and full bit-plane searching', *Symmetry*, Vol. 15, No. 7, p.1423.

Huang, C-T., Weng, C-Y. and Shongwe, N.S. (2023) 'Capacity-raising reversible data hiding using empirical plus–minus one in dual images', *Mathematics*, Vol. 11, No. 8, p.1764.

Huang, F., Qu, X., Kim, H.J. and Huang, J. (2016) 'Reversible data hiding in JPEG images', *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 26, No. 9, pp.1610–1621.

Kim, S., Huang, F. and Kim, H.J. (2019) 'Reversible data hiding in JPEG images using quantized DC', *Entropy*, Vol. 21, No. 9, p.835.

Kishk, S. and Javidi, B. (2002) 'Information hiding technique with double phase encoding', *Applied Optics*, Vol. 41, No. 26, pp.5462–5470 https://opg.optica.org/ao/abstract.cfm?URI=ao-41-26-5462 (Accessed 2002).

Li, Z., Wang, Y., Wang, Z., Liu, Z., Zhang, J. and Li, M. (2020) 'Reversible information hiding algorithm based on multikey encryption', *Wireless Communications and Mobile Computing*, Vol. 2020, p.8847559, https://doi.org/10.1155/2020/8847559 (Accessed 2020)

Liu, Y., Qu, X. and Xin, G. (2016) 'A ROI-based reversible data hiding scheme in encrypted medical images', *Journal of Visual Communication and Image Representation*, Vol. 39, pp.51–57 https://www.sciencedirect.com/science/article/pii/S104732031630075X (Accessed 2016).

Meng, Y., Chen, X., Sun, X., Liu, Y. and Wei, G. (2023) 'A dual model watermarking framework for copyright protection in image processing networks', *Computers*, *Materials and Continua*, Vol. 75, No. 1, pp.831–844.

Nguyen, T-S., Chang, C-C. and Lin, C-C. (2022) 'High capacity reversible data hiding scheme based on AMBTC for encrypted images', *Journal of Internet Technology*, Vol. 23, No. 2, pp.255–266.

Puech, W., Chaumont, M. and Strauss, O. (2008) 'A reversible data hiding method for encrypted images', in Delp III, E.J.W., Wah, P., Dittmann, J. and Memon, N.D. (Eds.): *Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, SPIE, San Jose, California, USA, pp.1–9.

Qian, Z., Zhang, X. and Wang, S. (2014) 'Reversible data hiding in encrypted JPEG bitstream', *IEEE Transactions on Multimedia*, Vol. 16, No. 5, pp.1486–1491.

Rai, A.K., Om, H., Chand, S. and Lin, C-C. (2023) 'High-capacity reversible data hiding based on two-layer embedding scheme for encrypted image using blockchain', *Computers*, Vol. 12, No. 6, p.120.

Ren, F., Wu, Z., Xue, Y. and Hao, Y. (2023) 'Reversible data hiding in encrypted image based on bit-plane redundancy of prediction error', *Mathematics*, Vol. 11, No. 11, p.2537.

Rudin, L.I., Osher, S. and Fatemi, E. (1992) 'Nonlinear total variation based noise removal algorithms', *Physica D: Nonlinear Phenomena*, Vol. 60, No. 1, pp.259–268 https://www.sciencedirect.com/science/article/pii/016727899290242F (Accessed 1992).

Sui, L., Li, H., Liu, J., Xiao, Z. and Tian, A. (2023) 'Reversible data hiding in encrypted images based on hybrid prediction and Huffman coding', *Symmetry*, Vol. 15, No. 6, p.1222.

Tang, Z., Pang, M., Yu, C., Fan, G. and Zhang, X. (2021) 'Reversible data hiding for encrypted image based on adaptive prediction error coding', *IET Image Processing*, Vol. 15, No. 11, pp.2643–2655.

USC-SIPI (1977) *The USC-SIPI Image Database*, USC-SIPI.

Wang, C., Wang, Q., Ren, K., Cao, N. and Lou, W. (2012) 'Toward secure and dependable storage services in cloud computing', *IEEE Transactions on Services Computing*, Vol. 5, No. 2, pp.220–232.

Wang, K., Hu, Y. and Lu, Z.M. (2012) 'Reversible data hiding for block truncation coding compressed images based on prediction-error expansion', *Proceedings of 2012 Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Athens, Greece, pp.317–320.

Wu, Y., Xiang, Y., Guo, Y., Tang, J. and Yin, Z. (2020) 'An improved reversible data hiding in encrypted images using parametric binary tree labeling', *IEEE Transactions on Multimedia*, Vol. 22, No. 8, pp.1929–1938.

Yamni, M., Karmouni, H., Sayyouri, M. and Qjidaa, H. (2022) 'Efficient watermarking algorithm for digital audio/speech signal', *Digital Signal Processing*, Vol. 120, pp.103251, https://www.sciencedirect.com/science/article/pii/S1051200421002906 (Accessed 2022).

Yao, Y., Zhang, W., Wang, H., Zhou, H. and Yu, N. (2019) 'Content-adaptive reversible visible watermarking in encrypted images', *Signal Processing*, Vol. 164, pp.386–401, https://www.sciencedirect.com/science/article/pii/S0165168419302427 (Accessed 2019).

Yu, H., Zhang, J., Xiang, Z., Liu, B. and Feng, H. (2023) 'Lossless reversible data hiding in encrypted image for multiple data hiders based on pixel value order and secret sharing', *Sensors*, Vol. 23, No. 10, p.4865.

Zhang, M., Dong, J., Ren, N. and Guo, S. (2023) 'Lossless watermarking algorithm for geographic point cloud data based on vertical stability', *ISPRS International Journal of Geo-Information*, Vol. 12, No. 7, p.294.

Zhang, X. (2011) 'Reversible data hiding in encrypted image', *IEEE Signal Processing Letters*, Vol. 18, No. 4, pp.255–258.