Hindawi
Security and Communication Networks
Volume 2022, Article ID 6299469, 17 pages
https://doi.org/10.1155/2022/6299469

Research Article

Reversible Data Hiding in Encrypted Domain Based on the Error-Correction Redundancy of Encryption Process

Kong Yongjun, Zhang Minqing, Wang Zexi, Ke Yan, and Huang Siyuan

Key Laboratory of Network and Information Security under Chinese People Armed Police Force (PAP), Chinese People's Armed Police Force Engineering University, Xi'an 710086, China

Correspondence should be addressed to Kong Yongjun; fighting_kyj@163.com

Received 20 January 2022; Revised 20 April 2022; Accepted 22 April 2022; Published 12 May 2022

Academic Editor: Jinwei Wang

The existing reversible data hiding methods in encrypted domain separate image encryption from information embedding and do not make full use of the redundant space in the process of image encryption. In order to improve the performance of reversible data hiding by using the technical characteristics of image encryption, a reversible data hiding method based on McEliece error correction is proposed. Firstly, the segmentation position of bit plane is determined according to the embedding requirement and texture characteristic, and the image is divided into high and low significant bits. Secondly, because of the error-correcting characteristic of McEliece encryption, reversible data embedding can be realized while encrypting low significant bits. Then, the high significant bits are compressed to reserve space for the ciphertext extension of the low significant bits. Finally, the whole high significant bits information is stream-encrypted. As long as the image receiver has the decryption key, the image can be restored without distortion. By concealing the relationship between error correction and secret information mapping, the concealment of secret information transmission can be realized. In addition, due to different processing for different pixels, it can be efficiently transmitted with low computational complexity for applications that only need general images. The simulation results show that this scheme can not only realize the separable operation of information extraction and image recovery but also resist the noise attack to a certain extent. The maximum embedding rates of 10 standard images from USC-SIPI and 50 standard images from BOSS-BASE are 2.228 and 2.323 bpp, respectively.

1. Introduction

The rapid development of 5G technology has greatly improved communication capabilities, while also having a profound impact on digital information processing [1, 2]. With the help of high-speed communication technology, mobile cloud computing, which connects multiple mobile terminals together, has quietly become the mainstream trend for processing digital information [3]. Mobile cloud computing is different from the traditional way of storing information on each terminal and processing it one by one. Users can transfer terminal computing to the cloud computing center for centralized processing in real time, thus effectively reducing the energy consumption of mobile terminal storage and computing. While enjoying the convenience brought by mobile cloud computing, frequent privacy disclosure issues [4, 5] under this framework cannot be ignored. In order to deal with the threat of privacy disclosure, more and more cloud applications need encryption technology [6] to protect sensitive data in all aspects of transmission, storage, and computing. Combined with the actual needs of cloud space environment, how to make more effective use of ciphertext data has become a research hot spot in the field of network security in recent years [7–9].

Reversible Data Hiding in Encrypted Domain (RDH-ED) [10] is a technology that studies reversible embedding of secret information using encrypted data as the carrier. This technology can not only protect plaintext data by encryption but also realize application expansion of ciphertext data by using secret information embedded in ciphertext, which plays an important role. Taking encrypted images as an example, using RDH-ED, users with authority can not only

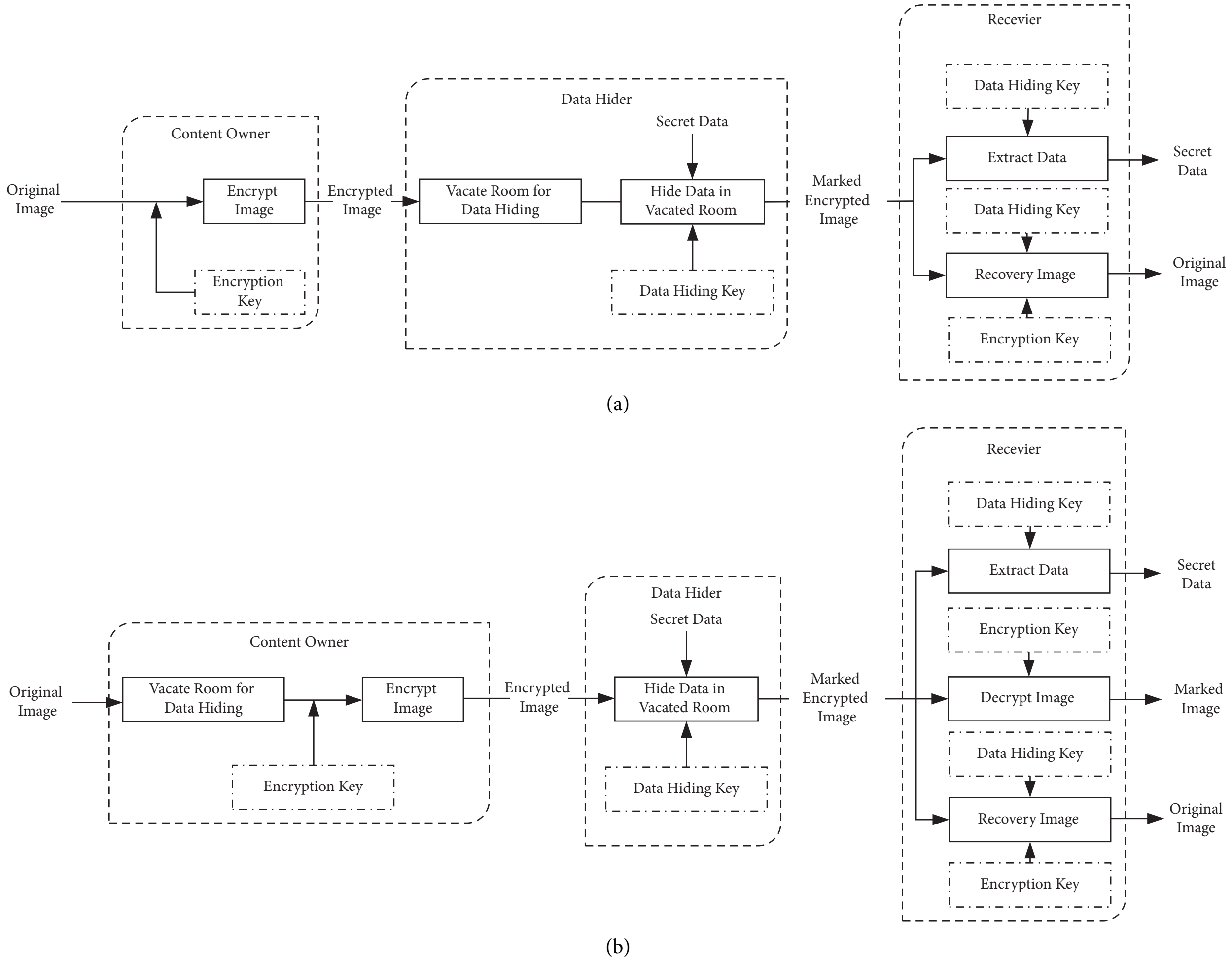


Figure 1: Two types of RDH-ED. (a) Vacating Room After Encryption (VRAE). (b) Vacating Room Before Encryption (VRBE).

ensure the accurate extraction of secret information but also ensure the original data recovery without distortion after extraction and decryption. To achieve the reversible embedding of secret information, RDH-ED schemes must vacate information embedding space. According to the process relationship between space vacation and encryption operation, the existing RDH-ED schemes are mainly divided into two categories: Vacating Room After Encryption (VRAE) and Vacating Room Before Encryption (VRBE) [11]. These two types of RDH-ED scheme frameworks are shown in Figure 1.

VRAE-type RDH-ED directly performs vacation operation on the redundant space of encrypted images to realize reversible embedding. The whole embedding and extraction operation does not affect the information protection by encryption, so it is suitable for third-party embedding operation [12–16]. Reference [12] uses Advanced Encryption Standard technology to encrypt the image and then divides the pixel block according to the size of the encrypted group. Combined with the pseudorandom generator for information replacement, the scheme realizes 1-bit information reversible embedding for each pixel block by changing texture standard deviation of pixel block. Due to the smoothness of the image texture, the receiver can modify decrypted image, so as to realize the lossless restoration of the original image and accurate extraction of secret information. As pixel block texture can be used to distinguish whether pixel block has changed or not, [13] introduced Support Vector Machine (SVM) to make the judgment of embedding more accurate. Meanwhile, public key adjustment mechanism was introduced in the process of information embedding to improve the performance. To solve inseparability of information extraction and image recovery in [12, 13], [14] uses wet paper coding technology to process encrypted data to achieve space vacation and inserts invisible watermarks into ciphertext state to achieve information embedding. Reference [15] uses the least significant bit (LSB) Hamming distance of ciphertext and preset auxiliary information to compress it, so as to vacate space in the encrypted image for information embedding. However, the schemes proposed in [14, 15] have strict requirements on the operation sequence of information extraction and image restoration. The image may be seriously distorted by direct decryption, which not only exposes the existence of the embedding process but also fails to extract the embedded data accurately. To achieve accurate extraction of embedded information before and after image decryption, [16] carries out key reuse in the same pixel block so that the encrypted pixel block can still retain the correlation of the original block. Then the Pixel Value Ordering (PVO) technique is applied to realize the reversible embedding of the encrypted pixel blocks. In this scheme, the accuracy of information extraction is not affected after decryption. In order to prevent the third party from obtaining valid information from the encrypted data, the correlation of the encrypted image will be severely damaged. Therefore, the space vacated for embedding of VRAE type is small.

In order to improve the embedding capacity, many VRBE-type RDH-ED schemes have been proposed to vacate larger embedding space through preprocessing operations before encryption [17–25]. Reference [17] proposed embedding the least significant bit (LSB) information of some specific pixels into the plaintext image by traditional RDH method before image encryption. Then, after encryption, the specific LSB information is directly replaced with secret message, so as to realize the reversible embedding. To further optimize the space vacation effect during the preprocessing operation, VRBE-type RDH-ED scheme based on the prediction mechanism was proposed in [18]. For some specific pixels used for information embedding, the scheme uses the information of surrounding pixels to obtain the prediction error before encryption and then modifies the prediction histogram after encryption to achieve reversible embedding. As the prediction mechanism improves the utilization of redundant space, the image distortion after direct decryption is significantly reduced in [18] compared with [17]. The carrier of RDH-ED is mainly transmitted in encrypted state, so there is no need to consider the image distortion caused by preprocessing on the basis of ensuring the reversible restoration after extraction. To improve the embedding capacity, [19] designed a sampling pixel method to represent images with less information during the preprocessing and then proposed an RDH-ED scheme with high capacity based on the Mirroring Ciphertext Groups (MCG) by using the homogeneity and probability of Paillier algorithm. In [20], patch-level sparse representation is introduced before the encryption for a superior performance. As the sparse coding is an approximation solution, the leading residual errors are encoded and self-embedded in the cover image together with the learned dictionary. Due to the powerful representation of sparse coding, a larger embedding space can be vacated. To solve the problem of the hierarchical data hiding, [21] carried out further research on homomorphic encrypted images. In this scheme, the preprocessed image is encrypted in Paillier cryptosystem and two embedding algorithms are applied on the encrypted image in succession. By conducting data embedding in two phases, the embedded data can be extracted, respectively, in encrypted domain and after decryption. Modifying the most significant bit (MSB) will cause large image distortion, but the strong correlation of MSB is beneficial to improve the embedding performance of Prediction Error Expansion algorithm. As the carrier characteristics of RDH-ED do not consider the image quality preprocessed before encryption, [22] proposed a high-capacity RDH-ED scheme based on MSB prediction mechanism. Since the preprocessing of MSB can vacate a larger embedding space, the selection of the available plane space is optimized in [23] to improve the embedding capacity. In [24], PBTL coding was introduced during the error prediction, and the embedding capacity was further improved by combining with coding technology. In [25], bit plane partition was introduced before the encryption. Combined with an efficient MSB prediction strategy, the embedding capacity was improved by embedding the LSB values into the other bit plane.

Since VRBE-type RDH-ED schemes make better use of the correlation in image information, they can obtain higher embedding capacity than VRAE-type ones. However, whether in VRBE or in VRAE, the embedding space is mainly obtained from the redundant space of the image information, so the embedded performance will be constrained by the image carriers. In addition, to achieve lossless recovery, most RDH-ED schemes must have extraction operations to correct distortions caused by embedding. In order to cover up the existence of embedding operation and directly obtain lossless images without extraction, this paper proposes a Vacating Room In Encryption (VRIE) scheme based on McEliece error-correction redundancy. Firstly, the segmentation position of bit plane is determined according to the embedding requirements and texture characteristics. Due to the subsequent error-correcting encryption of the low significant bit information, the ciphertext expansion will occur. In order to enable the encrypted image to be transmitted at its fixed size, the space compressed from the high significant bit information is used to place the extended ciphertext. Secondly, based on the error correction of McEliece encryption process, the secret information can be reversibly embedded while encrypting the low significant bit information. Next, while the high significant bit information is compressed by Huffman code, the extended part of the low significant bit ciphertext fills the space vacated by the compression of the highest significant bit. Finally, the high significant bit information after stream encryption and the low significant bit information after error-correction encryption are integrated to generate the encrypted image embedded with secret information.

2. Related Techniques

2.1. GOPPA Code. The GOPPA code is a rational fraction code defined by elements over the finite field F_{q^m} [26]. Because some of its subcodes can achieve the ideal performance of Shannon Channel Coding Theorem and can be quickly decoded, it plays an important role in error control systems and the construction of cryptography.

2.1.1. Construction Process. The construction process of GOPPA code is as follows:

(1)Suppose that $\mathbf{L} = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$ is an ordered subset of a finite field $GF^n(q^m)$, where $0 < n \leq q^m$, and, for any $i \neq j$, there is always $\alpha_i \neq \alpha_j$, $i, j \leq n$.

(2)Suppose that $G(x)$ is the polynomial of degree r over a finite field $GF(q^m)$, and $G(\alpha_i) \neq 0\,(i = 1, ..., r)$. The array of polynomial coefficients is $\{g_1, g_2, ..., g_r\}$.

(3) Suppose that $\mathbf{C} = \{\beta_1, \beta_2, ..., \beta_n\}$ is the code in n-dimensional linear space, and the rational fraction $R_C(x)$ corresponding to the code is

$$R_C(x) = \sum_{i=1}^{n} \frac{\beta_i}{x - \alpha_i}. \tag{1}$$

All codes $\mathbf{C}$, whose corresponding rational fraction satisfies formula (2), form set $\mathbf{L}$, and the GOPPA Code, generated by $G(x)$, is marked as $\Gamma(\mathbf{L}, G(x))$.

$$\{C: R_C(x) \equiv 0 \bmod G(x)\}. \tag{2}$$

According to the construction process, all the vectors of the GOPPA code can satisfy formula (3); that is, the GOPPA code $\Gamma(\mathbf{L}, G(x))$ is a linear code.

$$\forall \mathbf{a}, \mathbf{b} \in \Gamma(\mathbf{L}, G(x)) \Rightarrow \mathbf{a} + \mathbf{b} \in \Gamma(\mathbf{L}, G(x)). \tag{3}$$

2.1.2. Error-Correcting Decoding. From the definition of GOPPA code construction, it can be inferred that

$$\begin{aligned} R_C(x) &= \begin{bmatrix} \dfrac{1}{x - \alpha_1} & \cdots & \dfrac{1}{x - \alpha_n} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} \\ &= \mathbf{H}\mathbf{C}^T = 0 \mathrm{modG}(\mathrm{x}). \end{aligned} \tag{4}$$

Therefore, matrix $\mathbf{H}$ can be used as the check matrix of GOPPA code for decoding and error correction. In order to improve the speed of verification and decoding, matrix $\mathbf{H}$ needs to be deformed. By multiplying matrix $\mathbf{H}$ with an invertible matrix and extending it, we can get matrix $\widehat{\mathbf{H}}$ to correct the error when GOPPA code has a correctable error.

$$\widehat{\mathbf{H}} = \begin{bmatrix} \dfrac{1}{g(\alpha_1)} & \dfrac{1}{g(\alpha_2)} & \cdots & \dfrac{1}{g(\alpha_n)} \\ \vdots & \vdots & & \vdots \\ \dfrac{\alpha_1^{r-1}}{g(\alpha_1)} & \dfrac{\alpha_2^{r-1}}{g(\alpha_2)} & \cdots & \dfrac{\alpha_n^{r-1}}{g(\alpha_n)} \\ \dfrac{\alpha_1^{r}}{g(\alpha_1)} & \dfrac{\alpha_2^{r}}{g(\alpha_2)} & \cdots & \dfrac{\alpha_n^{r}}{g(\alpha_n)} \\ \vdots & \vdots & & \vdots \\ \dfrac{\alpha_1^{n-1}}{g(\alpha_1)} & \dfrac{\alpha_2^{n-1}}{g(\alpha_2)} & \cdots & \dfrac{\alpha_n^{n-1}}{g(\alpha_n)} \end{bmatrix}. \tag{5}$$

Suppose that $\mathbf{u}$ is the vector to be encoded, and the corresponding GOPPA code $\mathbf{c}$ without error can be obtained by the generation matrix $\mathbf{G}$.

$$\mathbf{c} = \mathbf{u}\mathbf{G}. \tag{6}$$

Under the GOPPA coding system, the receiving vector $\mathbf{R}$ can be expressed as

$$\mathbf{R} = \mathbf{c} + \mathbf{e} \in GF^n(q), \tag{7}$$

where $\mathbf{c}$ is the sending vector and $\mathbf{e}$ is the error vector.

According to formula (8), we can calculate the concomitant formula $s_i(\mathbf{R})$ of vector $\mathbf{R}$:

$$s_i(\mathbf{R}) = \sum\nolimits_{l=1}^{n} R_l g^{-1}(a_l) a_l^{i-1} \,(1 \leq i \leq r). \tag{8}$$

From the definition of GOPPA code construction, it can be deduced that

$$\mathbf{R} \in \Gamma(\mathbf{L}, G(x)) \Leftrightarrow s_i(\mathbf{R}) = 0\,(i = 1, ..., r). \tag{9}$$

If the concomitant formula $s_i(\mathbf{R})$ is not zero, according to (7), errors occur during transmission.

$$s_i(\mathbf{e}) = s_i(\mathbf{R})\,(i \leq r - 1). \tag{10}$$

The decoder operates as follows:

(1) Calculate the first r known concomitant formula sequences by the Berlekamp-Massey method [27], and the minimal polynomial of the concomitant formula $\{s_i(\mathbf{e}), i = 1, \ldots, n\}$ can be obtained.

(2) Calculate the remaining n-r unknown concomitant formula sequence $\{s_i(\mathbf{e}), i = \mathrm{r} + 1, \ldots, n\}$ by the minimal polynomial.

(3) Calculate the error vector $\mathbf{e}^T$:

$$\mathbf{e}^T = \widehat{\mathbf{H}}^{-1} (s_1(\mathbf{e}), s_n(\mathbf{e}))^T. \tag{11}$$

(4) Decode to get raw vector $\mathbf{u}$:

$$\mathbf{u} = (\mathbf{R} - \mathbf{e})\mathbf{G}^{-1}. \tag{12}$$

2.2. McEliece Encryption. In view of the NP-complete decoding problem of linear codes and the maturity of GOPPA decoding algorithm, McEliece proposed McEliece public key encryption scheme based on binary GOPPA codes [28].

2.2.1. Key Generation. Over the finite field F_{2^m}, an irreducible polynomial is selected randomly, whose Hamming distance is t. Then, the generating matrix G with GOPPA code parameters $[n, k, d]$ is constructed by this polynomial, where $n = 2^m$ and $d = 2 * t - 1$.

(1) Composition of a private key: an invertible matrix $\mathbf{S}$ of $k * k$ *dimension*, a permutation matrix $\mathbf{P}$ of $n * n$ dimension, and GOPPA check code $\mathbf{C}$ which can realize bit error correction.

(2) Calculation of the corresponding public key $\mathbf{G}_{\mathrm{pub}}$:

$$\mathbf{G}_{\text{pub}} = \mathbf{SGP}. \tag{13}$$

2.2.2. Encryption. To encrypt the message vector m of length k,

(1) generate a random error vector e of length n, and the error vector e must satisfy the condition that the Hamming weight $w(\mathbf{e})$ is less than t, where $w(\mathbf{x})$ represents the sum of different bits between vector x and the zero vector;

(2) generate the encrypted vector c according to the following formula:

$$\mathbf{c} = \mathbf{mG}_{\text{pub}} + \mathbf{e}. \tag{14}$$

2.2.3. Decryption. To decrypt the encrypted vector c of length n, we have the following:

(1) Eliminate the influence of permutation matrix P:

$$\mathbf{c}' = \mathbf{cP}^{-1} = \mathbf{mSG} + \mathbf{eP}^{-1}. \tag{15}$$

(2) Due to the equation $w(\mathbf{e}) = w(\mathbf{eP}^{-1})$, $\mathbf{m}'$ can be obtained by error correcting with the GOPPA check code $\mathbf{C}$ of the private key, where $\mathbf{m}' = \mathbf{mS}$.

(3) Calculate the message vector m:

$$\mathbf{m} = \mathbf{m}'\mathbf{S}^{-1}. \tag{16}$$

2.3. Lossless Compression

2.3.1. Run Length Coding (RLC). Run Length Coding (RLC) rules are as follows:

(1) Record the number of the same symbol from the beginning of the information string until it changes

(2) Represent the original information by the string variations and the length of each same-symbol string

On the decoding side, only need to restore the string according to the corresponding rules can realize the information decompression without loss. Assume that the information x = 1001110000 to be encoded, and the result of the Run Length Coding RLC$(\mathbf{x})$ is 1234.

2.3.2. Huffman Coding. Huffman encoding rules are as follows:

(1) Calculate the probability of different characters in the information string to be encoded, and sort the characters according to the probability from small to large

(2) Add the first two characters with the lowest probability and define them as new characters

(3) Reorder the probability of new characters with the remaining characters until all characters can be represented by a new character

A Huffman probability dictionary *Dic* can be finally generated according to the coding rules. For information x, the Huffman coding result is expressed as HC$(\mathbf{x})$. On the decoding side, receiver with the corresponding Huffman probability dictionary *Dic* can achieve lossless decompression.

3. Proposed Method

In order to cover up the existence of secret information transmission during the RDH-ED scheme, the secret information extraction and embedding can be selectively carried out on the basis of the lossless recovery of original information by encryption key. In this paper, the redundancy in the construction of encryption algorithm is used to realize information embedding, and a VRIE-type RDH-ED scheme based on error-correction redundancy in the encryption process is designed.

3.1. VRIE Framework. At the sending terminal, content owner uses encryption key K_{en} combined with specific encryption to achieve security encryption. During encryption, the embedding space is vacated by mapping the redundancy of ciphertext or encryption key. At the same time, the secret information protected by the hidden key K_{hide} can be selectively embedded in the way of controlling the redundancy during the encryption process, and finally an encrypted image that contains secret information (or not) is obtained.

At the receiving terminal, the original image can be decrypted without any distortion as long as key K_{en} is obtained. If the encrypted image carries additional secret information, authorized users can extract the secret information protected by the hidden key K_{hide} through the redundancy mapping during the decryption process. Then, the extracted information continues to be decrypted by the hidden key K_{hide}, so as to obtain the secret information accurately. The framework of the VRIE type is shown in Figure 2.

As can be seen in Figure 2, compared with the VRAE-type and the VRBE-type frameworks, the VRIE-type framework has the following advantages:

(1) The embedding and extraction operations are optional. No matter whether the transmission of secret information exists or not, for a third party, the receiver only uses the encryption key K_{en} for the image recovery. Therefore, this framework can realize the concealment of secret information transmission.

(2) Information extraction and image recovery can be separated. No matter whether secret information is extracted or not, the image obtained by direct decryption with encryption key K_{en} does not have any distortion compared with the original image.

(3) The embedding capacity is not restricted by image carriers. The embedding space mainly depends on the redundancy of the specific encryption process. Selecting an appropriate encryption algorithm can greatly improve the embedding capacity.

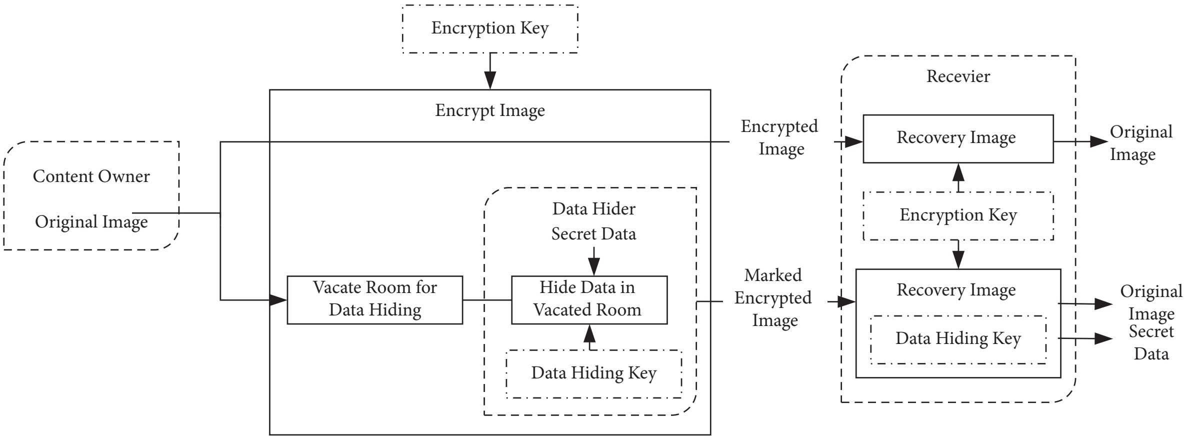


Figure 2: Vacating Room In Encryption (VRIE).

(4) Security has been improved. No operation is performed on the image before and after encryption, so the original image can be guaranteed without damaging the security of the encryption algorithm.

3.2. Reversible Data Hiding Based on Error-Correction Redundancy. Based on the error correction of GOPPA coding, McEliece encryption introduces random errors during the construction process to establish NP-difficult linear decoding problem to achieve information protection. For data encrypted with the same pair of public and private keys, the introduced random error can produce a large amount of ciphertext redundancy. At the same time, the receiver can eliminate the redundancy with the verification operation of GOPPA code to ensure the accurate recovery. On the basis of not destroying the security of the encryption system, we propose establishing the mapping relationship between the secret information and the random error and then controlling the random error to realize the reversible data hiding.

3.2.1. Error Mapping. Assuming that the McEliece encryption used for encryption is based on GOPPA code with parameter $[n, k, d]$, every time plaintext P with length k can be encrypted, the error vector $\mathbf{e}$ with length n can be corrected. The maximum number of error positions in the error vector $\mathbf{e}$ is d. The set of all available error vectors is E, and the number of elements $\text{Crad}(E)$ in the set can be expressed by the following formula:

$$\text{Crad}(E) = C_n^1 + C_n^2 + \ldots + C_n^d. \quad (17)$$

The information to be embedded is converted into binary representation, and each encryption can map $\lfloor \log_2(\text{Crad}(E) + 1) \rfloor$ bit information with one error vector. In order to make full use of the error-correction redundancy, the sending and receiving terminals can negotiate a bijective mapping relationship in advance. Up to $2^{\lfloor \log_2(\text{Crad}(E)+1) \rfloor}!$ kinds of mapping relationship are available for each McEliece encryption with specified parameters.

So each embedding operation can realize the embedding of 3-bit information, and a maximum of 4 kinds can be agreed.

Taking GOPPA code with parameters [1, 4, 7] as an example, there are $C_7^1 = 7$ error vectors that can be used for correction. Each encryption operation can embed $\lfloor \log_2(7 + 1) \rfloor = 3$ bits of additional information, and a maximum of $8! = 40320$ kinds of mapping relationship can be available for embedding. Figure 3 illustrates two different error mapping relationships under this parameter condition. $\mathbf{e}_i\,(i = 1, \ldots, 7)$ represents an error vector, respectively, and $\mathbf{e}_0$ indicates that no error occurs. $m_i\,(i = 1, \ldots, 8)$ represents the data to be embedded with a size of 3 bits.

3.2.2. Implementation Process. This scheme uses the error-correction redundancy of McEliece encryption to embed and extract information. For the image carrier with length a and width b, McEliece encryption based on parameters $[n, k, d]$ is selected to realize the secret information embedding. The specific implementation process is as follows:

(1) Send terminal

Step 1: Bit plane segmentation

The pixel information is converted into the eight-bit binary representation, with the bit plane order arranged from left to right. Then, according to the ciphertext expansion generated by embedding requirement and compressible redundancy space, the original pixel matrix is segmented into m-bit LSB $I_{(7-m,\ldots,8)}$ and (8-m)-bit MSB information $I_{(1,\ldots,8-m)}$.

Step 2: Compression of MSB information

Firstly, dimensionality reduction is carried out on the MSB information $I_{(1,\ldots,8-m)}$. Due to the strong correlation of the same bit plane information, a one-dimensional array $F(I_{(1,\ldots,8-m)})$ is arranged according to the order of the bit plane from small to large.

$$F\left(I_{(1,\ldots,8-m)}\right) = \{f_i | i = 1, \ldots, ab * (8 - m)\}. \quad (18)$$

Then, RLC and Huffman coding are performed on this array successively to obtain the compression coding $D_{I_{(1,\ldots,8-m)}}$ of the MSB information.

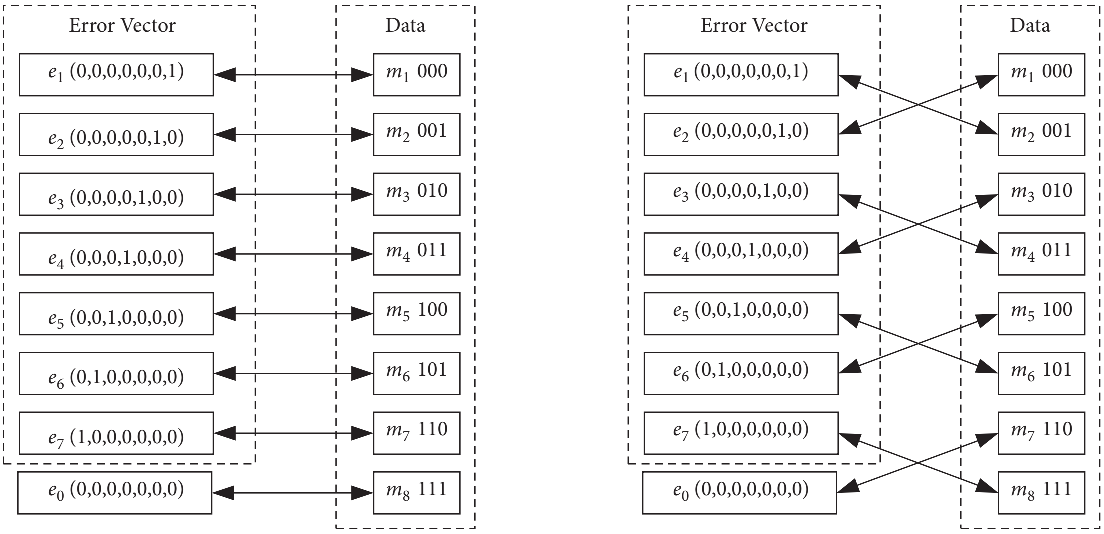


Figure 3: Two different mapping relationships between error vector and data to be embedded.

$$D_{I_{(1,\ldots,8-m)}} = \mathrm{HC}\Big(\mathrm{RLC}\Big(F\Big(I_{(1,\ldots,8-m)}\Big)\Big)\Big). \tag{19}$$

Step 3: McEliece encryption of LSB information with secret information embedding

Firstly, group the LSB information $I_{(7-m,\ldots,8)}$ in length of k, which can be divided into $\lfloor abm/k \rfloor$ groups of arrays.

Secondly, according to the construction parameters of McEliece encryption, the appropriate generation matrix is selected to calculate the corresponding public and private keys. Then, use the public key G_{pub} to encrypt each group $\{p_i | i = 1,\ldots,k\}$. In the process of each array encryption, the preset error mapping relationship can be used to control the correctable error and realize $\lfloor \log_2(\mathrm{Crad}(E)+1) \rfloor$ (introduced in 2.2.1) bit information embedding. During the entire encryption of the LSB information, up to $\lfloor abm/k \rfloor * \lfloor \log_2(\mathrm{Crad}(E)+1) \rfloor$. bits of additional information can be embedded. Additional information is generated by stream encrypting secret information with a protection key K_{hide}. The first *abm* bit ciphertext is saved as the low significant bit ciphertext C_L, and the remaining ciphertexts are filled into the MSB space as extended ciphertexts.

Step 4: Stream encryption of MSB information composed of compressed coding and extended ciphertexts

Firstly, the information in the extended ciphertext space S_1, which is generated from the McEliece encryption, is filled into the redundant space S_2 obtained after the MSB compression. At the same time, random information is added to the unfilled part of space S_2 to ensure that the image is transferred with the size of the original matrix. Formulas (20) and (21) represent the sizes of S_1 and S_2, respectively.

$$\begin{aligned} S_1 &= \lfloor \frac{abm}{k} \rfloor * n - abm, \\ \mathrm{S}_2 &= \mathrm{ab} * (8-\mathrm{m}) - \mathrm{D}_{I_{(1,\ldots,8-m)}}. \end{aligned} \tag{20}$$

Then, the pseudorandom seed K_{en} is used to generate sequences $(r_1,\ldots,r_{ab*(8-m)})$ to perform stream encryption on the filled MSB information.

Step 5: Generation of encrypted image

The encrypted information from different-significant-bit space is converted to the specified size and then merged into an encrypted image according to the size of original matrix.

(2) Receive terminal

Step 1: Bit plane segmentation

Using the negotiated parameter m, the encrypted pixel matrix is segmented into m-bit and (8-m)-bit MSB information. The original pixel is segmented into m-bit LSB and (8-m)-bit MSB information.

Step 2: Stream decryption of MSB information

The pseudorandom seed K_{en} is used to generate the same sequences $(r_1,\ldots,r_{ab*(8-m)})$. Then, stream decryption is performed on the MSB ciphertext to obtain the compressed most-effective-bit information and the extended ciphertext generated from the LSB encryption.

Step 3: McEliece decryption of LSB information with secret information extracting

Firstly, the extended ciphertext of McEliece encryption from the MSB space and the ciphertext from the LSB space are combined together and then grouped in units of n bits. For each group of ciphertexts, the invertible matrix **S**, permutation matrix **P**, and check code **C** in the McEliece private key are used for error correction and decryption. Thus, the LSB plaintext information in units of k bits is obtained. During decryption, specific errors introduced in the encryption can be checked. Therefore, the accurate extraction of additional information can be achieved by combining the mapping relationship. The authorized user with the protection key K_{hide} can stream-decrypt the additional information to get the secret information.

Step 4: Decompression of MSB information

Decompressing the compressed most-effective-bit information with RLC and Huffman coding, the MSB plaintext information can be obtained in the form of one-dimensional array.

Step 5: Generation of recovered image

The decrypted information from different-significant-bit space is converted to the specified size and then merged into a recovered image according to the size of original matrix.

3.2.3. Processing of Auxiliary Information. In order to ensure the lossless recovery of the encrypted image, the receive terminal must first get the necessary auxiliary parameters for the decryption process.

The auxiliary information consists of the following parts: auxiliary information A length L_1 ($\lceil \log_2 A \rceil$ bit), bit plane segmentation parameter m (3 bits), construction parameters of McEliece encryption $[n, k, d]$ ($\lceil \log_2 (nkd) \rceil$ bit), MSB compression length L_2 ($\log_2 |D_{I_{(1,\ldots,8-m)}}|$ bit), Huffman probability dictionary Dic length L_3 ($\lceil \log_2 (Dic) \rceil$), Huffman probability dictionary (Dic bit), and additional message M length L_4 ($\lceil \log_2 (M) \rceil$ bit). Among them, Huffman probability dictionary accounts for the largest proportion of auxiliary information, and other auxiliary parameters with variable lengths have little influence on the embedding scheme. As long as enough space is reserved for parameters, reversible data hiding can be realized.

If the image is transmitted without loss, all the auxiliary information can be transferred by substitution after the MSB compression. After the completion of Step 2, the first L_1 bits of compressed MSB information are saved as secret information M_c to be embedded and then replaced with auxiliary information. During the McEliece encryption of the LSB information, M_c is embedded first before the secret information protected by protection key K_{hide}.

After the stream decryption of the MSB with key K_{en}, the receive terminal directly extracts the first L_1 bits to obtain the auxiliary information A. With the auxiliary information A, the first $L_1 * n / \lfloor \log_2 (\text{Crad}(E) + 1) \rfloor$ bits LSB information can be used to extract M_c under McEliece decryption. The extracted information M_c is used to replace the first L_1 bit MSB information, and the original MSB information can be obtained after decompression.

3.3. Contributions. In order to solve the problem that the embedding performance of VRAE and VRBE is restricted by the image texture, we analyze the encryption process and propose a VRIE-type framework. Furthermore, we use error vectors of the McEliece encryption to establish the mapping relationship. Thus, we can achieve reversible data hiding without texture restriction of original image.

The main contributions of the proposed scheme in this paper are as follows:

(1) Under the specified encryption parameter setting, the embedding capacity of the proposed scheme is related to the total number of error vectors that can be corrected, which completely breaks through the information entropy constraint of image texture.

(2) This scheme realizes RDH by establishing the mapping relationship between error vector and data to be embedded. Therefore, by establishing access control over the mapping relationship, we can realize the separable extraction of additional information, so as to realize the concealment of information embedding behavior.

(3) In this scheme, bit plane segmentation is carried out first, and then different encryption operations are carried out, respectively, so as to ensure that images can be transmitted in a fixed size. While McEliece encryption is performed on LSB information, embedding is realized by controlling the encryption process. The MSB information is compressed first and then the stream encryption is performed. The space vacated by compression can be used for ciphertext expansion generated by the LSB encryption.

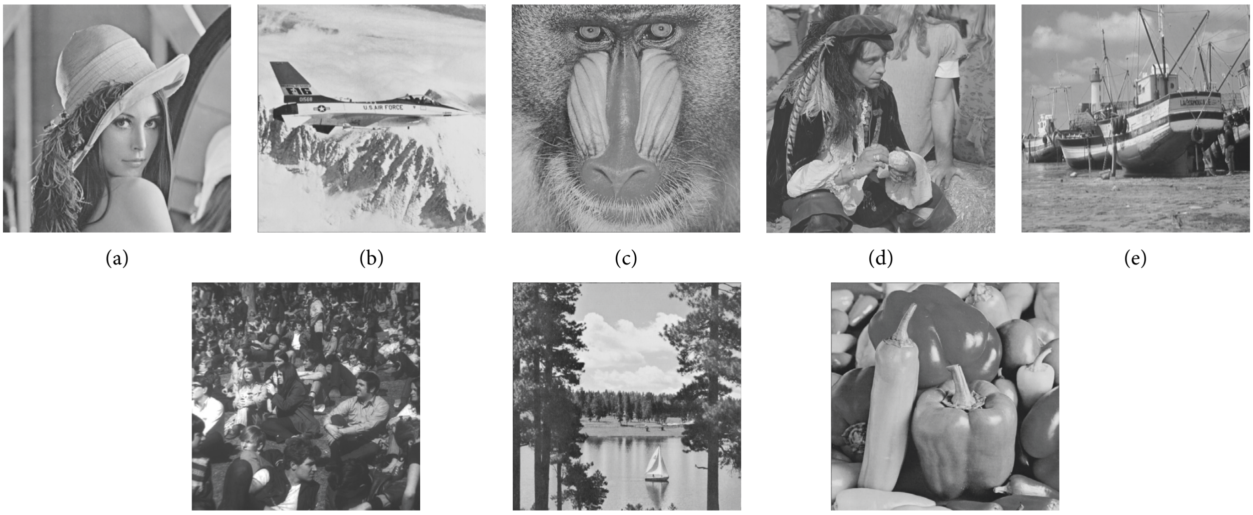
(4) This scheme supports encryption key management for users with different permissions and can realize quick decryption of general image and lossless recovery of complete image.

On the basis of breaking through the texture restriction to improve the embedding capacity, the proposed scheme can meet the authorization management of multiuser content access in the cloud environment by key distribution. Receive terminal can decide to obtain general image or lossless image according to the permissions. At the same time, by virtue of the separability of extraction and lossless recovery, this scheme can be disguised as only encryption and decryption of the original information for the third party. In this way, authentication information can be embedded without being detected, which is of great significance to the authentication of encrypted information.

4. Experimental Results

In this paper, MATLAB 2021b software is used to carry out the simulation experiment of the test images selected from the USC-SIPI and BOSS-BASE image library. The images used for performance comparison of each scheme are shown in Figure 4.

During the simulation in this section, the McEliece structure parameters $[n, k, d]$ are set to $[1, 4, 7]$, and the



(a) (b) (c) (d) (e)

(f) (g) (h)

Figure 4: Test images. (a) Lena. (b) Plane. (c) Baboon. (d) Man. (e) Boat. (f) Crowd. (g) Lake. (h) Peppers.

generating matrix G is set to $\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$. K_h and K_{en} are used as pseudorandom seeds to generate pseudorandom numbers in logistic chaotic system with parameter $\mu = 3.822$. The iterative formula of the system is

$$y_{n+1} = \mu y_n \left(1 - y_n\right) \left(y_n \in (0, 1)\right). \qquad (21)$$

4.1. Analysis of Image Quality. We use existing RDH-ED schemes [15–19] to embed additional data with the Lena image and decrypt the image directly without extraction. The trend curves of different RDH-ED methods between the embedding rate (ER) and the Peak Signal-to-Noise Ratio (PSNR) are shown in Figure 5.

By changing the order of embedding and encrypting operations, VBRE-type RDH-ED schemes can not only expand the application requirements of RDH-ED but also eliminate the influence of encryption and decryption operations on embedding. With the help of the high-fidelity RDH method in the spatial domain [17, 18], the general image can be obtained by direct decryption without affecting the lossless recovery after extraction. As the carriers of RDH-ED scheme are generally transmitted in encrypted form, some applications have stricter requirements on the embedding capacity than image quality after direct decryption. According to the characteristics of encrypted domain, [19] applied MCG technology to design large-capacity RDH-ED scheme. As can be seen from Figure 5, the image quality after direct decryption by this method is not high under the same embedding capacity.

Further analysis of Figure 5 leads to the conclusion that different parameters (as shown in [15]) under the same RDH-ED method have an impact on the image quality after direct decryption, but the impact is much smaller compared with the performance of different schemes.

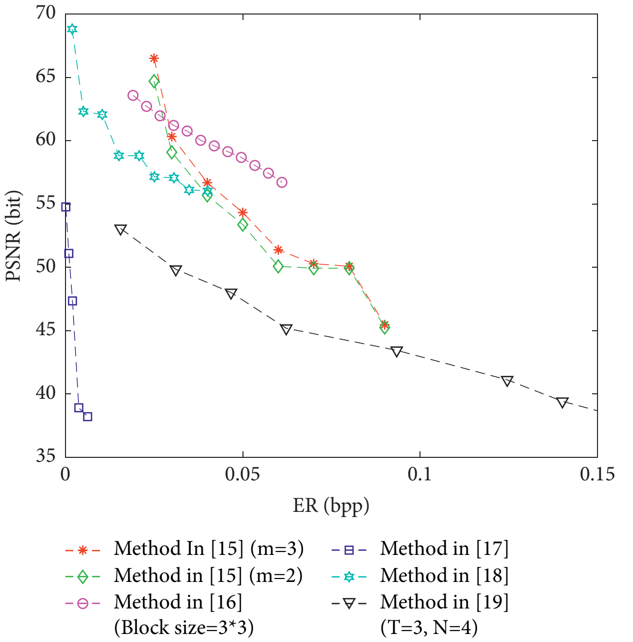
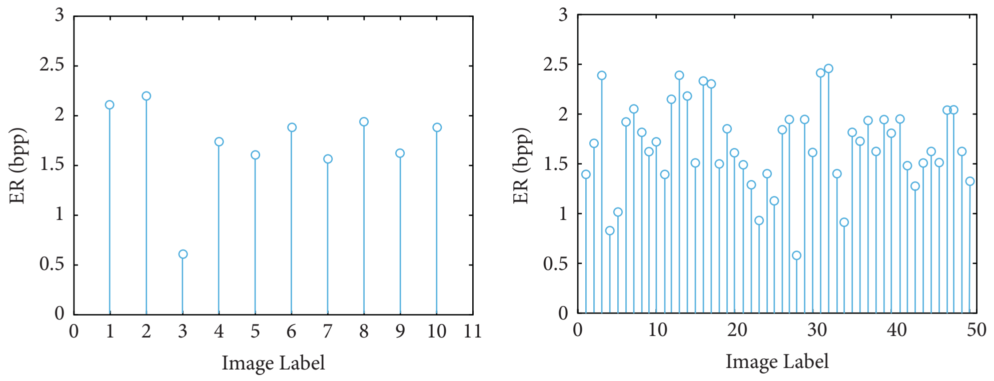


Figure 5: Directly decrypted image's ER-PSNR comparison of different RDH-ED methods.

Through the simulation experiments of 10 standard images from USC-SIPI and 50 standard images from BOSSBASE, we obtained the performance results as shown in Figure 6 and Table 1. The maximum embedding rates of the two databases were 2.228 and 2.323 bpp, respectively. The VRIE-type RDH-ED uses the redundancy of encryption process for embedding, so the embedding capacity is less restricted by pixel information. Therefore, the maximum embedding capacity is greatly improved compared with



(a)

(b)

Figure 6: ER results from different image libraries. (a) USC-SIPI. (b) BOSS-BASE.

Table 1: Experimental results from USC-SIPI and BOSS-BASE.

Database	Indicators	Best case	Worst case	Average
USC-SIPI	ER (bpp)	2.228	0.607	1.716
	PNSR	∞	∞	∞
BOSS-BASE	ER (bpp)	2.323	0.588	1.684
	PNSR	∞	∞	∞

Table 2: Huffman probability dictionary with condition m set to 4.

Probable sequence	Character	Huffman code
1	1	1
2	2	[0, 0, 0]
3	3	[0, 1, 1]
...	...	...
356	537	[0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0]
357	735	[0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1]

[15–19]. In addition, the proposed scheme can selectively extract information embedding with the decryption. Even without the permission to extract secret information, the lossless images also can be obtained by direct decryption. Therefore, the proposed scheme can selectively disclose the protection key K_{hide} according to the permissions. In this way, the proposed scheme can meet the requirements of high quality image transmission and realize the covert transmission of secret information.

4.2. Analysis of Embedding Capacity. In our scheme, after encrypting every n bit original pixel information by McEliece encryption, reversible embedding of $\lfloor \log_2(\mathrm{Crad}(E)+1) \rfloor$ bit additional information can be realized, and (d-n) bit ciphertext extension will be generated. For communication channel without considering extension, the embedding capacity based on error-correction redundancy is not affected by the original information. For the image carrier with fixed size, we need to carry out bit plane segmentation and then compress the MSB information to solve the problem of LSB ciphertext expansion. Auxiliary information is embedded to achieve parameter negotiation, so it will occupy a certain embedding space. In the composition of auxiliary information, Huffman probability dictionary *Dic* has great influence on embedding performance. For the Lena image, when the bit plane parameter m is set to 4, the probability dictionary *Dic* is shown in Table 2.

To achieve lossless decompression, we need L_3 bit space reserved to represent the *Dic*.

$$L_3 = m_{ax}(S_e) * \left(\lceil \log_2(m_{ax}(L_a)) \rceil + m_{ax}|C_o| \right). \tag{22}$$

In the above equation, $m_{ax}(S_e)$ represents the maximum of the probability sequence, $\lceil \log_2(m_{ax}(L_a)) \rceil$ represents the maximum binary encoding of a single character, and $m_{ax}|C_o|$ represents the maximum Huffman encoding of a single character.

For Huffman probability dictionary shown in Table 1, $L_3 = 357 * (\lceil \log_2(735) \rceil + 18) = 9996$ bit space should be reserved for representation.

When McEliece parameters are determined, bit plane segmentation parameters will influence the embedding performance to some extent. Figure 7 illustrates the trend curves of embedding rate corresponding to different bit plane segmentation parameters. As can be seen from Figure 7, for images with complex textures (such as Baboon), bit plane segmentation has little influence on the maximum embedding rate. However, bit plane segmentation has a great influence on the maximum embedding rate for the images with natural texture. When the bit plane segmentation parameter m is set to 3, 4, or 5, the embedding rate can reach the peak.

With further analysis of Table 3, we can conclude that the error-correction redundancy space is fixed as long as parameter m is determined. Therefore, if fixed-size images are used as the carrier for data hiding, the embedding capacity is mainly affected by auxiliary information space and compressed space. When m is small, the error-correction redundancy space of LSB information is limited, and the compressed space which is reserved for ciphertext extension cannot be fully utilized. With the increase of m, the error-correction redundancy space increases linearly, while the compressed space decreases correspondingly. When the error-correction redundancy space is close to the compressed space (m is set to 3, 4, or 5), the scheme can obtain the highest embedding capacity. When m is large, the compressed space

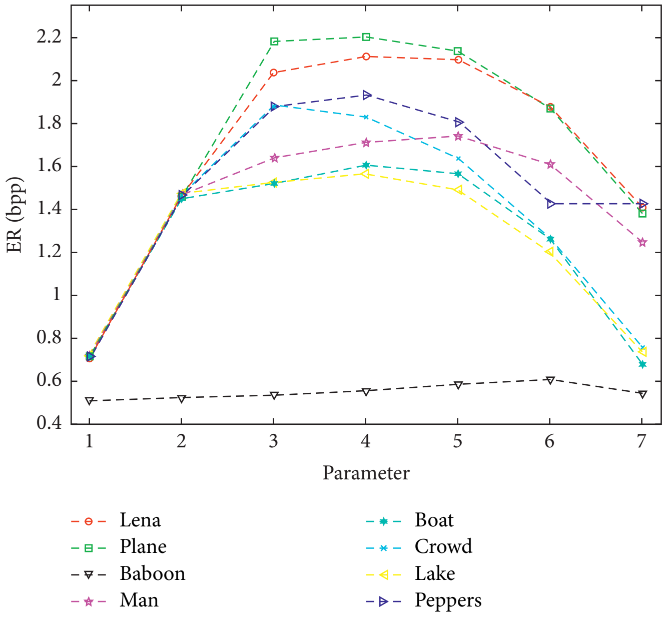


Figure 7: Bit plane segmentation parameter (m)-embedding rate.

Table 3: The respective space size under different bit plane segmentation parameter m (bit).

		$m = 1$	$m = 2$	$m = 3$	$m = 4$	$m = 5$	$m = 6$	$m = 7$
Error-correction redundancy space		196608	393216	589824	786432	983040	1179648	1376256
Lena	Auxiliary information space	10839	10869	10627	10153	10173	9179	8507
	Compressed space	521546	531933	544662	563233	559971	501377	377129
	Embedding space	185769	382347	534035	553080	549798	492198	368622
Plane	Auxiliary information space	9359	7815	7894	8755	8526	8696	8696
	Compressed space	552929	566128	579691	586003	568903	498617	369975
	Embedding space	187249	385401	571797	577248	560377	489921	361279
Baboon	Auxiliary information space	6208	6181	6395	6395	6609	6609	6609
	Compressed space	140093	143430	146819	152153	159876	165672	148889
	Embedding space	133885	137249	140424	145758	153267	159063	142280
Man	Auxiliary information space	8164	8105	8011	7783	7713	7231	7020
	Compressed space	413368	425258	438108	455545	464348	429043	333673
	Embedding space	188444	385111	430097	447762	456635	421812	326653
Lake	Auxiliary information space	376341	388286	405961	428663	418049	337259	183908
	Compressed space	8310	7728	7728	7641	7098	6775	6072
	Embedding space	188298	380558	398233	421022	410951	330484	177836
Crowd	Auxiliary information space	482912	494800	501485	486814	435136	337299	205128
	Compressed space	6944	7000	6912	6912	6552	6650	6693
	Embedding space	189664	386216	494573	479902	428584	330649	198435
Lake	Auxiliary information space	384884	395122	407367	418387	397690	322751	199363
	Compressed space	7830	7890	7656	7336	7074	6864	6325
	Embedding space	188778	387232	399711	411051	390616	315887	193038
Peppers	Auxiliary information space	467115	480582	502146	516006	481975	381862	381862
	Compressed space	9512	9512	9512	9184	8856	8175	8175
	Embedding space	187096	383704	492634	506822	473119	373687	373687

cannot accommodate the ciphertext expansion generated by the McEliece encryption of the LSB information. The reversible embedding can only be realized in the limited error-correction redundancy space which is smaller than the compressed space. In this case, the embedding performance is greatly affected by the correlation of the image information. In addition, for images with complex textures (such as Baboon), the compressed space under any bit plane segmentation is smaller than the error-correction redundancy space, so the embedding space is only related to the compressed space. In order to ensure the reversible recovery, sufficient space should be reserved for embedding auxiliary information in advance. Huffman probability dictionary *Dic* occupies a large proportion in the auxiliary information. With the increase of m,

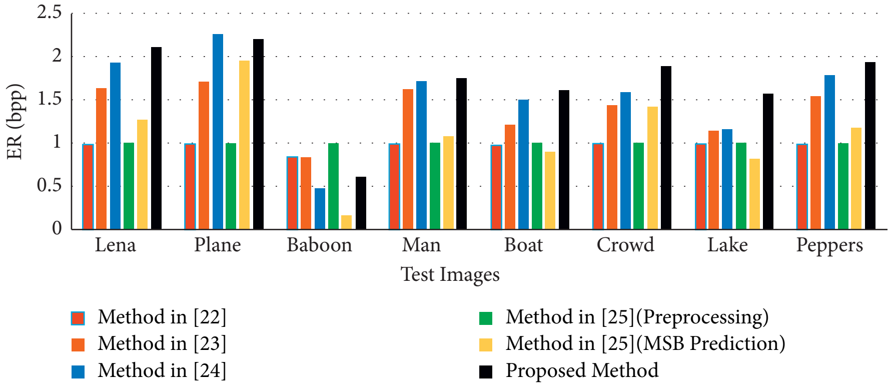


Figure 8: Comparison of embedding rate (bpp).

the MSB information that needs to be compressed decreases, and the space occupied by auxiliary information also decreases accordingly. Compared with embedding space, auxiliary information space occupies less proportion. For the Baboon image with the least embedding capacity, the proportion of auxiliary information space is always less than 5%, while the embedding space can reach more than 130000 bits, which has practical application value.

By comparing the embedding rate in the existing large-capacity RDH-ED schemes [22–25], we analyze the embedding performance of the proposed scheme. As is shown in Figure 8, the embedding space of [22, 23] is reserved before the encryption, so it is less affected by the image information and more suitable for complex images (such as Baboon). Reference [24] reserved embedding space by PBTL coding with high-correlation textures, which is more suitable for smooth image (such as Plane). In [25], bit plane segmentation is also carried out for information, and information embedding is divided into two stages: self-embedding in the preprocessing process and embedding with MSB prediction. Each single embedding stage is restricted by the image texture, so [25] cannot obtain an excellent embedding performance. In this scheme, error-correction redundancy of McEliece encryption is used to realize information embedding. Embedding performance has no relation with LSB information. But the compression of MSB information needs to reserve enough space for the placement of ciphertext expansion. For a specific image, when the compressed space of MSB information is close to the error-correction redundancy space of LSB information with an appropriate segmentation parameter, better embedding performance can be achieved. For images with general texture (such as Lena), this scheme improves the embedding rate by 0.082 bpp compared with [22].

4.3. Complexity Analysis. In order to keep the size of the image embedded with secret data consistent with the original image, this scheme firstly performs bit plane segmentation and then compresses the MSB information to solve the ciphertext expansion caused by the LSB information. Since different encryption operations are performed on different bit plane data, their complexity needs to be analyzed, respectively:

(1) After segmentation, it is assumed that the MSB data to be encrypted is of N_1 bits. Since the encryption and decryption of the MSB data are XOR operations on each bit information, the computational complexity of MSB data encryption and decryption is $O(N_1)$.

(2) After segmentation, it is assumed that the MSB data to be encrypted is of N_2 bits. Since the McEliece encryption and decryption are implemented on the LSB data, the computational complexity of the encryption mainly lies in the quasi-cyclic bivalent matrix multiplied by a one-dimensional vector, and the computational complexity of the bivalent matrix convolution is $O(N_2\log(N_2))$. The complexity of decryption is mainly determined by decoding complexity. For the decoding of quasi-bivalent GOPPA codes, the computational complexity is close to $O(N_2\log(N_2))$.

The MSB data has low computational complexity but contains more image information. According to this feature, the decryption end can use key distribution to control the access of the image content according to the actual application requirements. Figure 9 shows the general image obtained after decryption of MSB data only when parameter m is set to 4, 5, or 6. As can be seen from Figure 9, these images can still convey the main information of the image content. For applications with general images that can meet the requirements, this receiver can decrypt the MSB data only. Thus, only computation with complexity $O(N_1)$ needs to be carried out, so the communication efficiency is improved.

4.4. Security Analysis. In order to analyze other features of the proposed scheme with high embedding capacity, this section conducts information embedding and extraction with the Lena image under the condition that parameter m is set to 4. As illustrated in Figure 10, since the proposed scheme uses error-correction redundancy to realize reliable embedding, image carrier has only three stages: (1) original image, (2) encrypted image (embedded with secret information or not), and (3) decrypted image. Compared with the existing RDH-ED schemes, the proposed scheme has neither the difference between the ciphertexts containing the

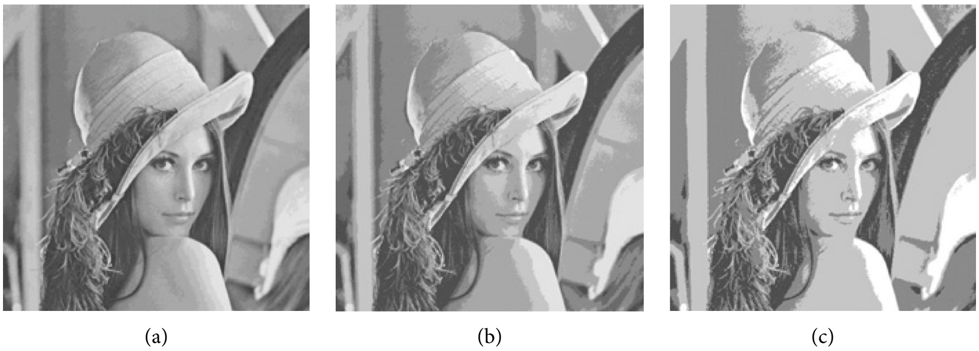


Figure 9: Direct decryption of MSB data under different values of parameter m. (a) $m = 4$. (b) $m = 5$. (c) $m = 6$.

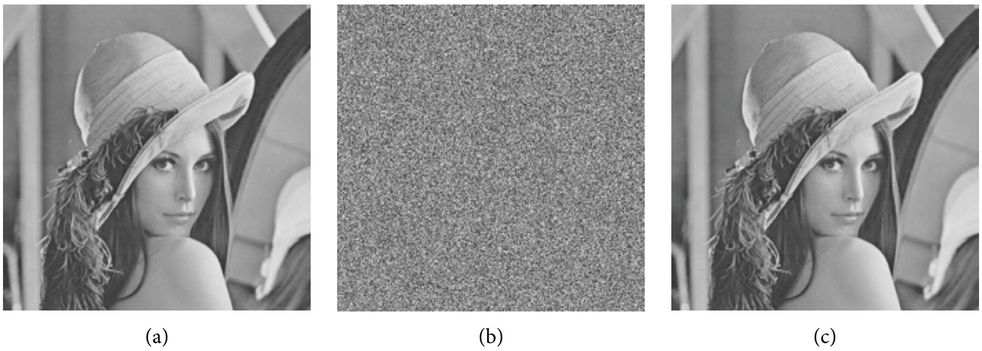


Figure 10: The Lena Image in different stages. (a) Original image. (b) Encrypted image. (c) Decrypted image.

embedded message or not of type VRAE nor the difference between the plaintexts containing the embedded message or not of type VRBE. For the third party, only image encryption and lossless decryption are carried out by the sender and the receiver. Therefore, the existence of secret message transmission can be concealed.

Because the error vector corresponding to secret information is introduced during the McEliece encryption, the same group of plaintext data can be encrypted to generate $C_n^d + \ldots + C_n^1 + 1$ groups of ciphertext data with the same key. Therefore, this scheme has the characteristics of probabilistic encryption. Probabilistic encryption can effectively resist the third party from comparing different ciphertexts of the chosen plaintext to deduce the encrypt key, so the scheme has Chosen Ciphertext Attack (CCA) security. MSB information has a great influence on image quality. This scheme destroys the correlation by the chaos of logistic encryption, so as to effectively protect the compressed data of MSB information and the extended data of LSB information. As can be seen from Figures 11 and 12, the statistical characteristics of the image are seriously damaged, so the proposed scheme can effectively protect the original image information.

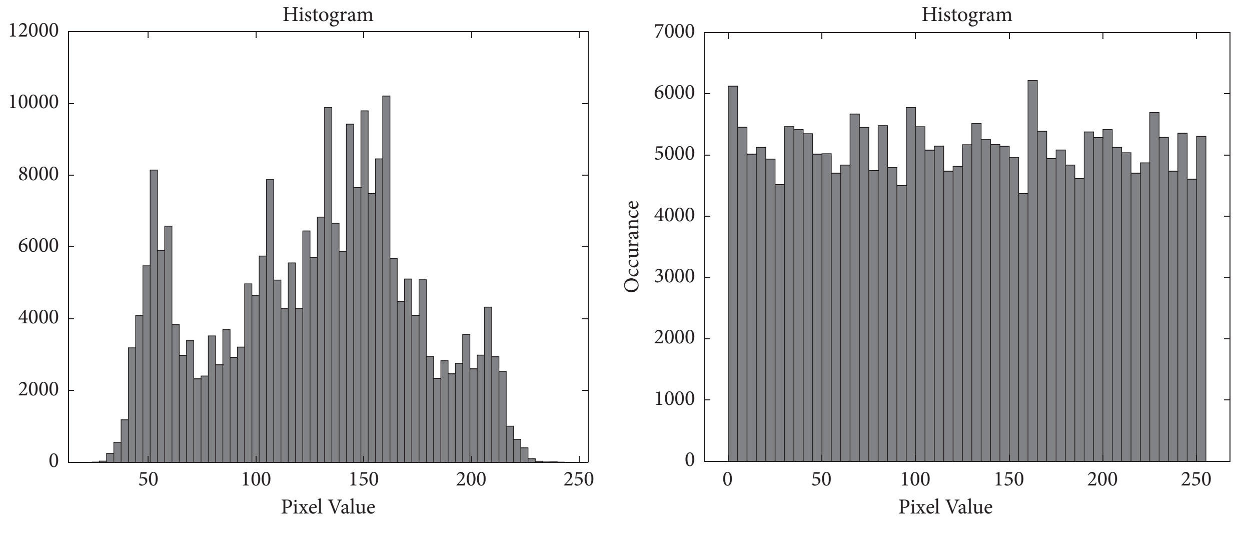
When parameter m is set to 4, our scheme can embed additional information up to 553080 bits. Under the condition of full embedding, the decrypted image is compared with the original image and the extracted information is compared with the embedded information. The results of their bit error are shown in Figures 13 and 14. We conclude that the scheme can be used to restore the original image reversibly and extract the secret information accurately in lossless channel.

In the process of encrypted image transmission, interference will occur due to channel instability. Due to the error correction of McEliece encryption, it can resist certain noise interference in the decryption process, so as to improve the accuracy of image recovery. To verify the robustness of the proposed scheme, White Gaussian Noise (WGN) is applied to the encrypted image. During the experiment, Signal-to-Noise Ratio (SNR) is used to modulate the WGN's intensity. Formula (23) shows the calculation process of SNR.

$$\begin{aligned} \mathrm{SNR} &= \frac{\mathrm{Power}\left(S_{\mathrm{o}}\right)}{\mathrm{Power}\left(S_{\mathrm{n}}\right)}, \\ \mathrm{Power}(S) &= \sum\left(\frac{|S|^2}{\mathrm{Length}(S)}\right). \end{aligned} \tag{23}$$

S, S_{o}, and S_{n} represent the signal intensity, the original signal intensity, and the noise signal intensity, respectively, and Length (S) is the length function of the signal. Figure 15 shows the White Gaussian Noise with the intensity of 5 dB according to the original information of the Lena image.

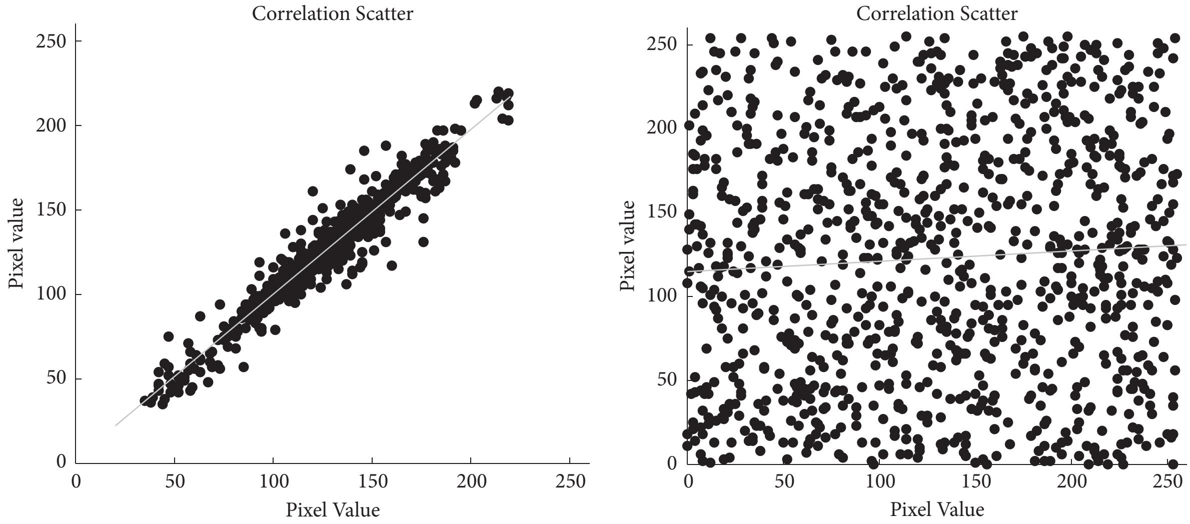
During the experiment, WGN between 0.5 and 5 dB is applied to the encrypted image with embedded information. After comparing the decrypted image disturbed by WGN with the original image, we get the bit error rate (BER) of the image as shown in the following formula:



(a)

(b)

Figure 11: Pixel histogram of original image and encrypted image. (a) Original image. (b) Encrypted image.



(a)

(b)

Figure 12: Correlation scatter of original image and encrypted image. (a) Original image. (b) Encrypted image.

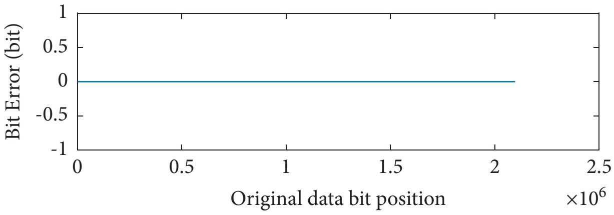


Figure 13: Bit error of recovery image.

$$\mathrm{BER}=\frac{n_{\mathrm{error}}}{n_{\mathrm{correct+}}n_{\mathrm{error}}}. \tag{24}$$

n_{correct} and n_{error}, respectively, represent the correct and error bits obtained after decryption compared with the original image.

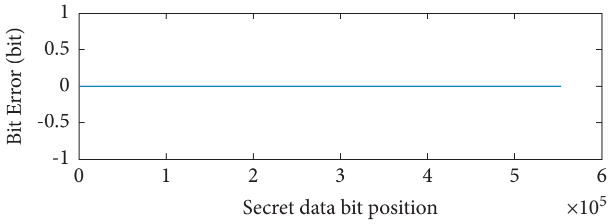


Figure 14: Bit error of extracted data.

As shown in Figure 16, the error rate of image restoration decreases linearly with the increase of WGN. Due to the error correction of McEliece encryption, the proposed scheme is robust. For an encrypted group with length n, when the interference caused by the embedding and WGN

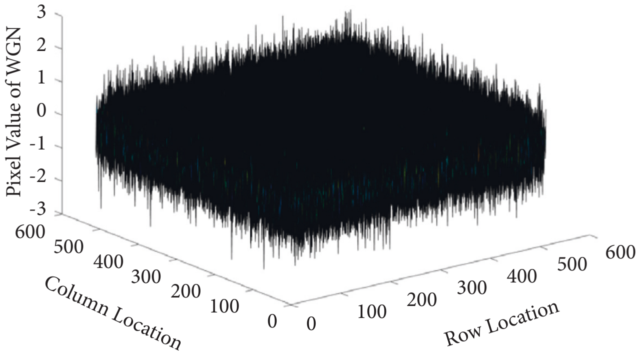


Figure 15: White Gaussian Noise with the intensity of 5 dB.

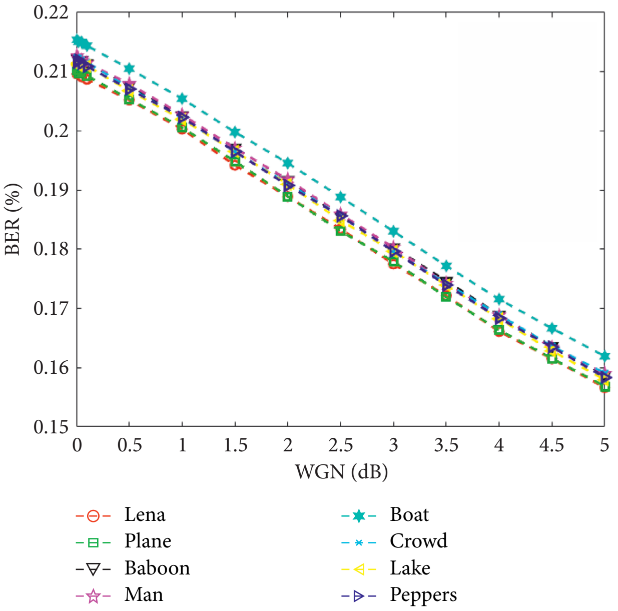


Figure 16: Bit error rate of image restoration under different WGN intensity.

can be handled by the error-correction capability, that is, when the total number of changing bit positions is less than d, the group with length k can recover correctly. As can be seen from Figure 15, the change of most pixel information caused by 5 dB WGN interference is less than 2. Therefore, the interference of WGN alone will not affect the recovery of original information seriously. If the coding position changed by the error vector is the same as that changed by the WGN, their influence on the image can be eliminated. Only when these two changed encoding positions are different do they lead to the accumulation of error positions, and the error bits will be produced. Further analysis of Figure 16 showed that the intensity of WGN, not the texture of original information, has a great influence on the accuracy of information recovery. When the intensity of WGN is set to 5 dB, the BERs of image restoration in the Lena, Plane, Baboon, Man, Boat, Crowd, Lake, and Peppers images are 15.66%, 15.68%, 15.87%, 15.90%, 16.19%, 15.90%, 15.79%, and 15.83%, respectively. In addition, bit error mainly occurs in low significant bits, and even if a bit error occurs, it has little impact on image quality. Therefore, the robustness of the proposed scheme can meet the requirements of most applications.

5. Conclusion

This paper proposes an RDH-ED scheme based on the error-correction redundancy of encryption process. In this scheme, McEliece encryption is implemented for LSB information, and secret information can be embedded reversibly while encrypting. Huffman and Run Length Coding is used to compress the MSB information to reserve the space for the ciphertext expansion placement, and then logistic chaotic sequence generated by the pseudorandom seed K_{en} is implemented to encrypt the MSB information. Since the embedding performance is not constrained by image carrier, the proposed scheme has better embedding capacity compared with existing schemes. For secret information, it is encrypted to become additional information with the protection key K_{hide} first and then embedded according to the error mapping relationship during encryption process. The extraction operation of secret information does not affect the accurate recovery, so the extraction operation is optional. Access of secret information can be controlled by key distribution. Thus, this scheme can conceal the transmission of secret information on the basis of realizing image lossless recovery. At the same time, after being attacked by noise, most of the information can be recovered correctly with the error correction of decryption, so the proposed RDH scheme is robust. In the future, the construction process of existing encryption algorithms can be further studied and the RDH-ED schemes can be designed according to the redundancy of the specific encryption process. Thus, the application scenarios of existing mature encryption algorithms can be further expanded.

6. Future Scope

VRIE-type RDH-ED has the advantages that previous RDH-ED schemes did not have, such as the unconstrained embedding capacity by the image texture, the lossless recovery without extraction, and the undamaged security of the encryption scheme. This paper proposed a specific scheme based on the error-correction redundancy during the process of McElieceencryption. However, there are still the following problems in the specific implementation process:

(1) This proposed method uses the process redundancy of McEliece encryption to embed additional information, which has the problems of large computational complexity and ciphertext expansion inherent in public key encryption.

(2) The robustness of this proposed method can only resist the small noise interference caused by channel transmission. If it faces serious attacks such as shear and rotation, the disaster tolerance ability is not enough.

(3) Whether the information is extracted or not does not affect the lossless recovery of the image, so it will not

be found by the third party, as well as realizing the concealment of information embedding behavior, which is of great significance for the forensics and authentication of encrypted data. However, if information extraction is needed, the image decryption key must be obtained first. Therefore, in the proposed scheme, the authority of information extraction must be higher than that of the image recovery, which will have certain application limitations and is not applicable to the applications that need the blind extraction feature of the third party in cloud space.

In the future, for specific application requirements, the process redundancy of encryption schemes such as stream encryption with low computational complexity and secret sharing encryption with strong disaster recovery ability can be studied to deal with the above problems. In addition, it is also meaningful to do further research on the extraction methods. On the basis that the existence of extraction operation does not affect the lossless recovery, how to design an embedding scheme that can be extracted blindly is worth considering.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants nos. 61872384 and U1603261 and National Key R&D Program of China under Grant no. 2017YFB0802000.

References

[1] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, “Five disruptive technology directions for 5G,” *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, 2014.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “A survey on mobile edge computing: the communication perspective,” *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: a survey on enabling technologies, protocols, and applications,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[4] K. Yan, G. Luo, X. Zheng, L. Tian, and A. Maradapu Vera Venkata Sai, “A comprehensive location-privacy-awareness task selection mechanism in mobile crowd-sensing,” *IEEE Access*, vol. 7, pp. 77541–77554, 2019.

[5] D. Wu, S. Si, S. Wu, and R. Wang, “Dynamic trust relationships aware data privacy protection in mobile crowd-sensing,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2958–2970, 2018.

[6] J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos, “Security and privacy for cloud-based IoT: challenges,” *IEEE Communications Magazine*, vol. 55, no. 1, pp. 26–33, 2017.

[7] X. Ge, J. Yu, and H. Zhang, “Towards achieving keyword search over dynamic encrypted cloud data with symmetric-key based verification,” *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 1, pp. 490–504, 2019.

[8] X. Liang, Z. Yan, X. Chen, L. T. Yang, W. Lou, and Y. T. Hou, “Game theoretical analysis on encrypted cloud data deduplication,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 10, pp. 5778–5789, 2019.

[9] J. Sun, H. Xiong, H. Zhang, and L. Peng, “Mobile access and flexible search over encrypted cloud data in heterogeneous systems,” *Information Sciences*, vol. 507, pp. 1–15, 2020.

[10] Y. Ke, M. Zhang, and J. Liu, “Overview on reversible data hiding in encrypted domain,” *Journal of Computer Applications*, vol. 36, no. 11, pp. 3067–3076, 2016.

[11] Y.-Q. Shi, X. Li, X. Zhang, H.-T. Wu, and B. Ma, “Reversible data hiding: advances in the past two decades,” *IEEE Access*, vol. 4, pp. 3210–3237, 2016.

[12] W. Puech, M. Chaumont, and O. Strauss, “A reversible data hiding method for encrypted images,” in *Proceedings of Security, Forensics, Steganography, and Watermarking of Multimedia Contents*, p. 68191E, X.San Jose, United States, SPIE, 2008.

[13] J. Zhou, W. Sun, L. Dong, X. Liu, O. C. Au, and Y. Y. Tang, “Secure reversible image data hiding over encrypted domain via key modulation,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 3, pp. 441–452, 2016.

[14] X. Zhang, Z. Wang, and J. Yu, “Reversible visible watermark embedded in encrypted domain,” in *Proceedings of the 2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pp. 826–830, IEEE, Chengdu, China, 2015.

[15] S. Zheng, D. Li, D. Hu, D. Ye, L. Wang, and J. Wang, “Lossless data hiding algorithm for encrypted images with high capacity,” *Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13765–13778, 2016.

[16] B. Ou, X. Li, and W. Zhang, “PVO-based reversible data hiding for encrypted images,” in *Proceedings of the 2015 2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pp. 831–835, IEEE, 2015.

[17] K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, “Reversible data hiding in encrypted images by reserving room before encryption,” *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 3, pp. 553–562, 2013.

[18] W. Zhang, K. Ma, and N. Yu, “Reversibility improved data hiding in encrypted images,” *Signal Processing*, vol. 94, pp. 118–127, 2014.

[19] S. Xiang and X. Luo, “Reversible data hiding in homomorphic encrypted domain by mirroring ciphertext group,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 11, pp. 3099–3110, 2018.

[20] X. Cao, L. Du, X. Wei, D. Meng, and X. Guo, “High capacity reversible data hiding in encrypted images by patch-level sparse representation,” *IEEE Transactions on Cybernetics*, vol. 46, no. 5, pp. 1132–1143, 2015.

[21] H.-T. Wu, Y.-m. Cheung, Z. Yang, and S. Tang, “A high-capacity reversible data hiding method for homomorphic encrypted images,” *Journal of Visual Communication and Image Representation*, vol. 62, pp. 87–96, 2019.

[22] P. Puteaux and W. Puech, “An efficient MSB prediction-based method for high-capacity reversible data hiding in encrypted

images," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 7, pp. 1670–1681, 2018.

[23] P. Puteaux and W. Puech, "EPE-based huge-capacity reversible data hiding in encrypted images," in *Proceedings of the 2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, IEEE, 2018.

[24] S. Yi and Y. Zhou, "Separable and reversible data hiding in encrypted images using parametric binary tree labeling," *IEEE Transactions on Multimedia*, vol. 21, no. 1, pp. 51–64, 2019.

[25] H.-T. Wu, Z. Yang, Y.-M. Cheung, L. Xu, and S. Tang, "High-capacity reversible data hiding in encrypted images by bit plane partition and MSB prediction," *IEEE Access*, vol. 7, pp. 62361–62371, 2019.

[26] E. Berlekamp, "Goppa codes," *IEEE Transactions on Information Theory*, vol. 19, no. 5, pp. 590–592, 1973.

[27] J. L. Massey, "Theory and practice of error control codes," *Proceedings of the IEEE*, vol. 74, no. 9, pp. 1293-1294, 1986.

[28] E. Berlekamp, R. Mceliece, and H. Van Tilborg, "On the inherent intractability of certain coding problems (Corresp.)," *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 384–386, 1978.