Hindawi
Security and Communication Networks
Volume 2022, Article ID 4671799, 14 pages
https://doi.org/10.1155/2022/4671799

Research Article

Noise Modulation-Based Reversible Data Hiding with McEliece Encryption

Zexi Wang[1,2], **Minqing Zhang**[1,2], **Yongjun Kong**[1,2], **Yan Ke**[1,2] and **Fuqiang Di**[1,2]

[1]*College of Cryptography Engineering, Engineering University of PAP, Xian 710086, China*
[2]*Key Laboratory of PAP for Cryptology and Information Security, Xian 710086, China*

Correspondence should be addressed to Minqing Zhang; api_zmq@126.com

Received 22 June 2022; Revised 17 September 2022; Accepted 11 October 2022; Published 30 October 2022

Academic Editor: Xuehu Yan

McEliece cryptosystem is expected to be the next generation of the cryptographic algorithm due to its ability to resist quantum computing attacks. Few research studies have combined it with reversible data hiding in the encrypted domain (RDH-ED). In this article, we analysed and proved that there is a redundancy in the McEliece encryption process that is suitable for embedding. Then, a noise modulation-based scheme is proposed, called NM-RDHED, which is suitable for any signal and not only for images. The content owner scrambles the original image and then encrypts it with the receiver's public key. The data hider generates a load noise by modulating additional data. After that, the load noise is added to the encrypted image, which achieves the data embedding. The reconstructed image is without any distortion after the direct decryption of the marked image, and the extracted data are no errors. The experimental results demonstrate our scheme has a higher embedding rate and more security, which is superior to existing schemes.

1. Introduction

Information hiding and cryptography are both important technologies to protect user privacy and have been inseparable from people's life. Reversible data hiding in the encrypted domain (RDH-ED) [1–3] as their cross-research hot spot has the characteristics of both privacy protection and secret data transmission; it is not only to embed additional data but also to reconstruct the original carrier without loss. Particularly, it has been applied in areas such as telemedicine, judicial forensics, and the military. In the past decades of development, researchers have been working to improve embedding capacity (EC) and enhance the security of RDH-ED, and have also achieved significant results.

1.1. In terms of Improving Embedding Capacity. Researchers have proposed two basic frameworks: vacating room after encryption (VRAE) and vacating room before encryption (VRBE). The main methods of the former are replacement or flipping of the least significant bits (LSBs), such as the first RDH-ED scheme based on VRAE proposed by Puech et al. [4], which encrypts an image with advanced encryption standard (AES) and embeds 1-bit additional data into a sub-block of the image containing 16 pixels. The receiver extracts the embedded data based on the local standard deviation of the image with the recovery of the original image. Subsequently, Zhang [5] proposed a scheme based on stream encryption, partitioning the encrypted image into nonoverlapping sub-blocks, and vacating room to embed 1-bit additional data by flipping the 3 LSBs of sub-block pixels; the EC is affected by the sub-block size; and the quality of the recovered image and the EC are mutually constrained. Hong et al. [6] improved the scheme [5] with the side match method, which increases the EC and reduces the bit error rate for extracting additional data. In addition, the schemes based on compressing the least significant bits [7, 8], re-encoding [9, 10], and pixel value ordering (PVO) [11] are presented successively. Furthermore, adaptive embedding, multi-layer embedding, and hierarchical embedding strategies [12–14] are effective to improve the EC.

Since the weak correlation of encrypted images, it is difficult to generate a large redundancy room so that the EC is limited. To address the issue, Ma et al. [15] proposed a new embedding framework of VRBE; that is, the original image is fully compressed before encryption to reserve more space for embedding. In Reference [15], the encrypted image is divided into two sets, the LSBs of one set are embedded into the other to generate redundancy space, then, the image is encrypted, and the data hider can directly replace the LSBs with additional data to achieve embedding, which improves the EC. Later, more and more methods that are used to vacate the room before encryption was presented, such as the most significant bit (MSB) prediction [16], bit plane rearrangement [17], parametric binary tree labeling (PBTL) [18], and compressed coding, like sparse coding [19] and entropy coding [20]. Most of the schemes rely on image correlation and usually can obtain high EC for smooth images, while it is smaller for images with complex textures. It is worth noting that if the data hider wants to embed additional data into the encrypted image, the image must be preprocessed before encryption. However, to protect the image privacy, the content owner can only complete this process, which exposes the purpose of hiding and is not practical.

Given the problems existing in the two embedding frameworks of VRAE and VRBE, a new embedding framework for vacating redundancy in encryption (VRIE) was proposed by Ke et al. [21]. They explored the redundancy in the process of public-key encryption and proposed an RDH-ED scheme based on LWE, by quantizing the encrypted domain room of LWE encryption and re-encoding its ciphertext to load it with additional bits. After that, they encapsulated the difference expansion method with fully homomorphic encryption (FHE) to further enhance security [22]. Recently, Kong et al. [23] have declared their scheme based on McEliece encryption, but it does not reach the security level required. Rather, it takes advantage of its error correction capability to increase the robustness of the scheme.

1.2. In terms of Enhanced Security. RDH-ED mainly utilizes stream cipher [5–8, 10, 12, 15] and block cipher [4, 24] in the early. The distribution of keys is difficult in a symmetric cryptosystem, and the number of keys is large, thus costly to manage. Public key encryption was introduced into RDH-ED, and the first scheme based on Paillier encryption was proposed by Chen et al. [25], which divides a pixel into two parts and encrypts them separately, and the data hider uses the homomorphic property to embed 1-bit data into the two LSBs of the encrypted pixels pair, and the decrypted image can still maintain the relevance of the embedded data, but the embedding rate (ER) is only 0.25 bit per pixel (bpp). Later, Zhang et al. [26] proposed a lossless and reversible method according to the probabilistic and homomorphic properties of Paillier. Wu et al. [27] developed a hierarchical embedding algorithm with Paillier encryption, which has a higher EC. Subsequently, several excellent schemes are designed [28, 29]. However, another issue of encrypted data expansion is raised by public key encryption. Wu et al. [30] and Chen et al. [31] adopted secret sharing as a lightweight encryption method for RDH-ED to reduce data expansion, enhance the privacy of images, and meet the needs of multiple users. The shares are changed because of the embedding, and it must be required that the shares can recover lossless after extracting data, including schemes [32, 33]. There is some auxiliary information to achieve the reversibility for most schemes, which may be self-embedded in the encrypted image or may be transmitted additionally; maybe, it is a security hole. Therefore, Yu et al. [34] proposed a more secure scheme without additional information transmission.

As we all know, Rivest Cipher 4 (RC4) was declared to be broken in 2013 [35]. Furthermore, the security of most public-key cryptographic algorithms is based on the difficulties of integer factorization or the discrete log problem, as well as on elliptic curves. However, the discovery of Shor's algorithm and Grover's search algorithm may reduce the difficulty of integer factorization or shorten the search time of keys, which will have a huge impact on the security of public keys and even symmetric ciphers [36]. It will affect the RDH-ED because its security depends in part on the cryptographic algorithm, which means that more secure encryption algorithms are considered to design the RDH-ED scheme. McEliece encryption is one of the shortlisted algorithms for postquantum cryptography according to NIST [37], which can resist quantum computing attacks and is expected to be a new generation of cryptographic algorithms. To the best of our knowledge, there has been little research work to combine McEliece with RDH-ED.

In this work, we focus on McEliece encryption to analyse the redundancy for embedding in the encryption process and proposed a noise modulation-based RDH-ED scheme (NM-RDHED), which is suitable for any encrypted signal. Compared with the state of the art, it has more security that can resist quantum computing attacks, and a higher embedding rate due to it is not affected by carrier redundancy. The experimental results verify the excellent performance of our scheme. The main contributions are summarized as follows:

(1) McEliece cryptosystem as one of the postquantum cryptographies is introduced into RDH-ED so that the carriers and additional data can be better protected.

(2) We proved that there is a redundancy in the McEliece encryption process that is suitable for embedding. According to the error correction characteristics of the coding base cipher and the randomness of the noise, the random noise added to the ciphertext can be regarded as embedded redundancy. We divide the noise into various subnoises and simplify it into two cases depending on whether the Hamming weight is zero or not. It concludes that there are two forms of redundancy in the McEliece encryption process.

(3) A noise modulation-based embedding method is proposed, and it modulates the additional data into a load noise. We calculate the number of subnoises with different Hamming weights by probabilistic estimation, define a modulation principle to make

full use of the redundant room, and then build modulation tables. According to the table, the additional data can be modulated into a load noise, which achieves the embedding.

(4) An NM-RDHED scheme is proposed. It has a higher embedding rate and the reconstructed image is with no distortion after the direct decryption of a marked image, because the operation of data hiding does not affect the procedure of encryption. Meanwhile, no extra steps are required for decryption, so it has strong concealment.

The rest of this article is organized as follows: in Section 2, we introduce McEliece cryptosystem before analysing and proving the redundancy for embedding. Then, Section 3 details the proposed noise modulation-based RDH-ED scheme. Section 4 provides the experimental results, analysis, and comparisons. Finally, Section 5 draws a conclusion.

2. Methodology

2.1. McEliece Cryptosystem. The McEliece cryptosystem [38] is a type of code-based public key cryptosystem that uses binary Goppa error-correcting code [39], which security is based on the NP-hard problem of finding a code word with minimal Hamming distance to a given word. It has several advantages, which can resist cryptanalysis in some quantum computer settings.

2.1.1. Goppa Code and Setting. We will briefly describe how to construct a binary $[n, k, d]$ Goppa code $\Gamma(L, g(x))$ over the finite field $\mathrm{GF}_{2^m} = \mathrm{GF}_2[x]/k(x)$, which satisfies $m \geq 3, mt + 1 \leq n \leq 2^m, 2 \leq t \leq (2^m - 1)/m$, and $k(x)$ is an m-degree irreducible polynomial, where t is the maximum error-correcting capacity. Firstly, select n distinct elements from GF_{2^m} to form a finite subset $L = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$. Then, choose a t-degree irreducible polynomial $g(x) \in GF_{2^m}$, which satisfies $g(\alpha_i) \neq 0$ for all $\alpha_i \in L$. Finally, compute all code words c_i, which satisfy the polynomial $g(x)$ and divide the sum function:

$$\Gamma = \left\{ c \in \mathrm{GF}2^n | \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \mathrm{mod}\, g(x) \equiv 0 \right\}. \tag{1}$$

To set up a McEliece Cryptosystem, suppose a binary Goppa code, which has parameters $[n = 2^m, k \geq n - mt, d \geq 2t + 1]$, and its generated matrix and parity check matrix are denoted by $\mathbf{G}_{k\times n}$ and $\mathbf{H}_{(n-k)\times n}$, respectively.

2.1.2. Key Generate. Generating a public and private key is detailed as follows: firstly, randomly choose an invertible matrix $\mathbf{S}_{k\times k}$ and a permutation matrix $\mathbf{P}_{n\times n}$. Then, compute $\mathbf{G}' = \mathbf{S} \cdot \mathbf{G} \cdot \mathbf{P}$, where $\mathbf{P}$ has exactly one "1" in every row and column, with all other entries being zero. Finally, the public key is $Pk = \{\mathbf{G}', t\}$ and the private key is $Sk = \{g(x), \mathbf{G}, \mathbf{S}, \mathbf{P}\}$.

2.1.3. Encryption. To encrypt a k-length binary sequence message $\mathbf{M}$, use the public key G' dot it and add random noise $\mathbf{E}$ to disguise the ciphertext:

$$\mathbf{C} = \mathbf{M} \cdot \mathbf{G}' + \mathbf{E}, \tag{2}$$

where both the encrypted message sequence $\mathbf{C}$ and $\mathbf{E}$ are the length of n and the Hamming weight $wt(\mathbf{E}) = t$.

2.1.4. Decryption. The receiver first uses the matrix $\mathbf{P}^{-1}$ to eliminate the influence of permutation. Then, according to Patterson's decoding algorithm, he can use the parity check matrix $\mathbf{H}$ to correct the error $\mathbf{E}'$ to decode $\mathbf{C}'$ and obtains the message $\mathbf{M}' = \mathbf{M} \cdot \mathbf{S}$. Finally, recover the original message $\mathbf{M}$ by eliminating $\mathbf{S}$ so that

$$\begin{aligned} \mathbf{C}' &= (\mathbf{M} \cdot \mathbf{S} \cdot \mathbf{G} \cdot \mathbf{P} + \mathrm{E}) \cdot \mathbf{P}^{-1}, \\ &= (\mathbf{M} \cdot \mathbf{S}) \cdot \mathbf{G} \cdot \left(\mathbf{P} \cdot \mathbf{P}^{-1}\right) + \mathrm{E} \cdot \mathbf{P}^{-1}, \\ &= (\mathbf{M} \cdot \mathbf{S}) \cdot \mathbf{G} + \mathrm{E}', \end{aligned} \tag{3}$$

where the noise $\mathbf{E}'$ satisfies $wt(\mathbf{E}') = wt(\mathbf{E})$.

$$\mathbf{M} = \mathbf{M} \cdot \mathbf{S} \cdot \mathbf{S}^{-1}. \tag{4}$$

2.2. Redundancy Analysis for Embedding. In the process of McEliece encryption, we find that there is a step called disturbance that requires adding random noise to the ciphertext. Because the random noise can be completely corrected in the decryption process, the additional data can be embedded into the ciphertext through it and can be extracted without errors. Besides, the randomness of the noise allows us to generate a load noise that contains additional data to replace the random noise. Therefore, the random noise can be regarded as redundant space for embedding. Here, we will analyse the redundancy of the random noise and demonstrate the feasibility of loading additional data without reducing the security of the encryption algorithm.

The random noise is a binary error pattern in coding schemes, which uses "1" to indicate where an error has occurred in a code word and "0" to indicate where no error has occurred. Specifically, the random noise is a sparse vector that consists of many "0" and a small number of "1" under the security encryption parameters. The random noise produced by a pseudo-random sequence generator (PRSG) obeys a uniform distribution. To generate a load noise that has the same statistical character as the random noise, we regarded a binary random noise of n bits with a Hamming weight of at most t as a discrete memoryless source E and its sample space is {0, 1}. Next, we use L elements as a group to make up a new random variable that has which is equal to a new source containing 2^L symbols and called L-degree extended source of E. Therefore, the load noise can be divided into many subnoises and building a special mapping relation between the additional data with them is easier. To simplify, these subnoises are classified into two cases: one where the Hamming weight is zero, and the other where the

Hamming weight is not zero. It concludes that there are two forms of embedding redundancy.

There are $\binom{x}{r}$ possibilities for a subnoise of x bits with a Hamming weight of r. More generally, the possibilities of vectors with different Hamming weights satisfy $1 < \binom{x}{1} = \binom{x}{x-1} < \binom{x}{2} = \binom{x}{x-2} \cdots < \binom{x}{\lfloor r/2 \rfloor} = \binom{x}{\lceil r/2 \rceil}$. Considering a sequence of x bits represents possibilities at most, if only the subnoises with Hamming weights of 0 and 1 are used to load additional data, there are $1 + 2^x - 1$ possibilities, so the length of the subnoise is at least $2^x - 1$. Then, we denote the probability of subnoises with Hamming weight y as $\Pr(\mathbf{e}_y) = \binom{2^x - 1}{y} (t/n)^y (1 - t/n)^{2^x - 1 - y}$, with $\sum_{y=0}^{2^x-1} \Pr(\mathbf{e}_y) = 1, x > 1, 0 \le y \le 2^x - 1$, where $\mathbf{e}$ represents the subnoise. The sum of the number of subnoises is $\lfloor n/2^x - 1 \rfloor$, and their Hamming weights are less than or equal to t.

$$\begin{cases} N_0 + N_1 + \cdots + N_y + \cdots + N_{2^x-1} = \left\lfloor \dfrac{n}{(2^x - 1)} \right\rfloor, \\ N_1 + 2 * N_2 + \cdots + y * N_y + \cdots + (2x - 1) * N_{2^x-1} \le t, \end{cases} \tag{5}$$

where N_y is the number of subnoises with a Hamming weight of y.

The subnoise of length $2^x - 1$ bits has at most 2^{2^x-1} possibilities. In this case, the mapping space of the subnoise is larger than that of x bits. However, the number of subnoises is calculated by (5) before we know $N_0 \gg N_1 > N_2 > N_3 \gg \cdots \gg N_{2^x-1}$. Besides, since the number of the subnoise with Hamming weight of 3 is less than 1 but not 0, we decide with a 50% probability whether to use it. If used, subtract 1 from N_3 and add 1 to both N_1 and N_2, but it carries no additional data. Therefore, only the subnoises with Hamming weights of 0, 1, and 2 are used to carry the additional data, and the actual probabilities of the subnoises are approximated by their frequency:

$$\Pr(\mathbf{e}_y) \approx Pr'(\mathbf{e}_y) = \frac{N_y}{\lfloor n/2^x - 1 \rfloor}, \text{ and } \sum_{y=1}^{3} Pr'(\mathbf{e}_y) = 1. \tag{6}$$

The number of subnoises with different Hamming weights is calculated by equation (6) and listed in Table 1 in different settings. Note that not all subnoises with Hamming weight of 0 are used to carry the additional data.

In general, the additional data to be embedded is encrypted and it obeys a uniform distribution. However, there are certain statistical characteristics in the local scope of encrypted data, and we have verified them through many experiments. First, we generate a random sequence by PRSG as the encrypted data and split it into a large number of groups of length v bits. Then, each group of the encrypted data is divided into several code words consisting of x bits, and the code word has 2^x possibilities. Finally, the code words in each group of encrypted data are counted. We found that out of 100,000 tests, there are always certain code words that account for a higher percentage. Furthermore, considering that the number of the subnoise with Hamming weight of 0 is also the most, the code word with the highest percentage should be modulated into the subnoise as much as possible. Therefore, we define a modulation principle to make full use of the redundant room as follows.

Definition 1. The process of mapping code words consisting of x bits into a subnoise of length of $2^x - 1$ bits is called noise modulation. Meanwhile, the ratio of the length of the additional data to a subnoise as a modulation rate (MR) is

$$\text{MR} = \frac{\text{len (additional data)}}{\text{len (sub noise)}} = \frac{x}{2^x - 1}. \tag{7}$$

When the greater the MR, the more embedded the additional data is, so that it can be used to indicate the efficiency of embedding. Note that the MR is maximum when $x = 2$; thus, we mainly discuss the modulation method under this case.

Definition 2. A code word with a higher percentage in a group of data is supposed to be modulated into a subnoise with a larger number, which we adopt as a modulation principle.

Finally, build a one-to-one mapping relationship between subnoises and additional data, the subnoises with Hamming weights of 0 and 1 are grouped into ST_1, and Hamming weights of 0 and 2 are grouped into ST_2. There are T_1 and T_2 kinds of mapping relationships, respectively, and are $T_1 \times T_2$ kinds in total:

$$T_y = \binom{2^x}{1} \cdot \binom{\binom{2^x - 1}{y}}{2^x - 1} \cdot (2^x - 1)!, \quad y = 1, 2, \tag{8}$$

where y represents the Hamming weights of the subnoise.

3. Proposed Scheme

In this section, we propose a noise modulation-based reversible data hiding scheme called NM-RDHED, which uses images as a case of signals. There are some main symbols and information listed in Table 2.

The proposed NM-RDHED scheme embeds and extracts additional data in the encryption and decryption process, and it does not affect these processes so it has strong security and concealment. Take the image as a case of signal to introduce the scheme and provide a structure of our scheme in Figure 1. The content owner provides an original image, scrambles it with security parameters, and encrypts it with a public key of the receiver. Then, the additional data are modulated into a load noise by building a mapping. Finally, add the load noise to the encrypted image to obtain a marked image. During the decryption process, a receiver who has a private key can directly decrypt the marked image and

Table 1: The number of subnoises with different Hamming weights in different settings.

	$t = 53$	$t = 71$	$t = 97$	$t = 125$	$t = 157$
$m = 10$	[292, 46, 2, 1]	[275, 62, 3, 1]	[253, 80, 7, 1]	—	—
$m = 11$	[632, 48, 1, 1]	[614, 66, 1, 1]	[590, 88, 3, 1]	[565, 110, 6, 1]	[537, 134, 10, 1]
$m = 12$	[1315, 48, 1, 1]	[1297, 66, 1, 1]	[1271, 92, 1, 1]	[1244, 118, 2, 1]	[1244, 146, 4, 1]

Note. $[N_0, N_1, N_2, N_3]$ represents the number of subnoises with Hamming weights of 0, 1, 2, and 3.

Table 2: Notions.

Symbols	Information
I	Original image
I_s	Scrambled image
I_e	Encrypted image after McEliece encryption
I_m	Marked image with additional data
E	Random noise
E_d	Load noise that contains additional data
SI	Side information
Sk	The private key for the original image
Pk	The public key for the original image
Kd	Data hiding key
M	An encryption parameter
K	Length of plaintext
N	Length of ciphertext
T	Maximum error-correcting capacity
v	Grouping length of additional data

correct the noise to recover the original image, and who has a data hiding key can extract the embedded data from the noise. Note that the scrambling parameters, the private key, and the data hiding key are transformed through the secure channel or the public channel based on the Diffie–Hellman key exchange protocol.

3.1. Image Encryption

Step 1: to remove the correlations of the original image, scrambling is necessary. First, we transform all pixels of the grey-scale image $\mathbf{I}$ sized $M \times N$ to binary sequence and then scramble in the pixel level, which dislocates the position of all elements with Guan et al. [40]. Then, we segment the image into eight bit planes and scramble within each bit plane by Li et al. [41]. Finally, we denote $\mathbf{I}_s$ as the scrambled image.

$$p = \sum_{h=0}^{7} 2^h \cdot p_h = [p_0, p_1, p_2, p_3, p_4, p_5, p_6, p_7], \tag{9}$$

$$p_s = \text{Josh}(p, \text{start}, \text{step}) = [p_7, p_5, p_4, p_0, p_6, p_2, p_3, p_1], \tag{10}$$

where the function of Josephus is described by Josh$(*)$, whose input p is an original pixel, the start is an initial index and step is a step length, and the output p_s is a dislocated pixel; an example is given by equation (10).

$$\begin{bmatrix} i' \\ j' \end{bmatrix} = \begin{bmatrix} 1 & b \\ a & a \cdot b \end{bmatrix} \begin{bmatrix} i \\ j \end{bmatrix}, \tag{11}$$

where i and j are the current index of bits in planes, and i' and j' are the new index of bits, and a, b are the parameters of Arnold.

Step 2: supposing the McEliece cryptosystem has parameters $[n, k, t]$, public key $Pk = \{G', t\}$, and private key $Sk = \{g(x), \mathbf{G}, \mathbf{S}, \mathbf{P}\}$. The scrambled image is segmented into eight bit planes and reshaped into binary sequences in order of left to right and top to bottom. Next, these sequences are divided into different groups of the same length k and denoted as $\mathbf{I}_v^{[i][j]}$, with $1 \le i \le 8, 1 \le j \le \lfloor (8 \times M \times N)/k \rfloor$. The content owner encrypts each group of sequences using a public key of the receiver:

$$\mathbf{I}_{ev}^{[i][j]} = \mathbf{I}_v^{[i][j]} \cdot \mathbf{G}', \tag{12}$$

where $\mathbf{I}_{ev}^{[i][j]}$ is a group of ciphertext sequences that is expanded from k bits to n bits, and $[i][j]$ is the j-th group of sequences at the i-th bit plane.

3.2. Data Embedding

Step 1: Generate a data hiding key Kd with a hyper chaotic system [42, 43], which can provide a pseudo-random sequence of sufficient length. Next, encrypt additional data with Kd.

Step 2: The encrypted additional data are split into numerous groups of length v bits, and each group of data is divided into several code words of length x bits. Then, count the code words in each group of data, and construct a modulation table that contains the relationship between the encrypted data and the subnoise according to the modulation principle and Table 1. Note that the modulation table has $T_1 \times T_2$ possibilities, of which modulation table id used depending on Kd. Table 3 provides an example of the modulation table.

Step 3: Modulate code words of additional data into many subnoises based on the modulation table generated in Step 2. After that, all the subnoises are used to make up the load noise $\mathbf{E}_d$. We select w bits from Kd at each time and transform them to decimal digits as indexes of the load noise. If the current index duplicates the previous one, it will be skipped and the next is checked until all subnoises are filled. Finally, the parts unfilled will be filled by the subnoise with Hamming weight of zero, and they do not carry additional data:

$$\text{index}^{[i]} = \sum_{i=0}^{w-1} 2^i \cdot Kd^{[i]}, w \le \left\lfloor \log_2 \left(\left\lfloor \frac{n}{2^x - 1} \right\rfloor \right) \right\rfloor, \tag{13}$$

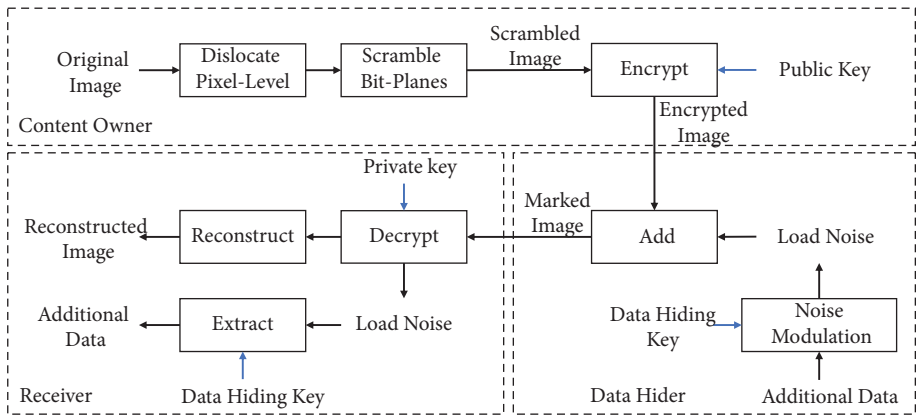


Figure 1: Structure of the NM-RDHED scheme.

Table 3: An example of the modulation table.

Code words	Percent (%)	Subnoises	
		$wt(e) = 0, 1$	$wt(e) = 0, 2$
[0, 0]	12.5	[0, 1, 0]	[1, 1, 0]
[0, 1]	25.0	[1, 0, 0]	[0, 1, 1]
[1, 0]	50.0	[0, 0, 0]	[0, 0, 0]
[1, 1]	12.5	[0, 0, 1]	[1, 0, 1]

where the symbol of $\lfloor * \rfloor$ represents the operation of round down.

Step 4: The load noise containing additional data is added to the ciphertext by equation (14) so that a marked ciphertext is obtained. Repeat Steps 2 to 4, and then, all marked ciphertexts make up the marked image $\mathbf{I}_m$, which still has eight bit planes, but is larger than the original image:

$$\mathbf{I}_{mv}^{[i][j]} = \mathbf{I}_{ev}^{[i][j]} + \mathbf{E}_d^{[i][j]}, \tag{14}$$

where the symbol "+" represents XOR, the size of the marked image is $M\prime \times N\prime = \lceil (n \times M \times N)/k \rceil$, and the symbol of $\lceil * \rceil$ represents the operation of round up.

Side Information. The code words with the highest percentage in each group of additional data need to be recorded as side information side information (SI), which ensures that the unique modulation table can be identified when extracting the data. The side information is regarded as additional data and is embedded into the ciphertext. Note that the side information new generated whose size is smaller is filled into the marked image, because some random pixels need to be filled when the marked ciphertext sequences are converted into an image.

3.3. Data Extraction and Image Reconstruction. The receiver decrypts the marked image with *Sk* to reconstruct the original image. Meanwhile, the load noise can be corrected during the decryption so that the additional data are extracted with the *Kd* extracted. There are three possible outcomes: the first is that the receiver has only the *Kd* and he cannot get any information. The second scenario is that the receiver has only the *Sk* and he can only reconstruct the original image. The last case is that the receiver has both keys, and he can not only extract the additional data but also reconstruct the original image.

3.3.1. Image Reconstruction. The receiver segments the marked image into a stack of eight bit planes and reshapes each bit plane into some sequences of n bits. Then, decrypt marked ciphertext $\mathbf{I}_{mv}^{[i][j]}$ and correct load noise $\mathbf{E}_d^{[i][j]}$ group by group, $1 \le i \le 8, 1 \le j \le (8 \times M' \times N')/n$. Finally, calculate $\mathbf{I}_v^{[i][j]}$ by using matrix $\mathbf{S}^{-1}$ of the private key and then inverse scrambling of images in bit plane and pixel level. The reconstructed image has no distortion compared to the original image:

$$\begin{aligned} \mathbf{I}'^{[i][j]}_{ev} &= \mathbf{I}_{mv}^{[i][j]} \cdot \mathbf{P}^{-1}, \\ \mathbf{E}_d^{[i][j]} &= \text{Correct}\left(I'^{[\mathbf{i}][\mathbf{j}]}_{\mathbf{ev}}, \mathbf{H}_{(n-k)\times n}\right), \end{aligned} \tag{15}$$

where the function of Correct$(*)$ is Patterson's decoding algorithm, and $\mathbf{G} \cdot \mathbf{H}^{\mathrm{T}} = 0$.

3.3.2. Data Extraction. Divide each load noise $\mathbf{E}_d^{[i][j]}$ into some subnoises consisting of $2^x - 1$ bits, and create indexes for them, by using the *Kd* as the indexes to identify which subnoises carry additional data and extract them sequentially. Next, extract the first group of SI from the marked ciphertext, and the unique modulation table that is used to modulate the load noise in each group is determined by *Kd* and SI. Finally, recover the additional data according to the modulation table.

3.4. Example. Figure 2 provides an example that can help readers better understand the NM-RDHED scheme, where the encryption parameters are $[m = 10, n = 1024, t = 71, k = 314]$ and embedding parameters are $[v = 16, x = 2]$. The image scrambling consists of two phases. First, all pixels are transformed into binary sequences, and then, dislocate each element of the sequence, such as the pixel of 164; its binary sequence is "10100100," and the dislocated sequence is "01010001" after Josephus scrambling. Therefore, the original pixel [164, 167, 170,

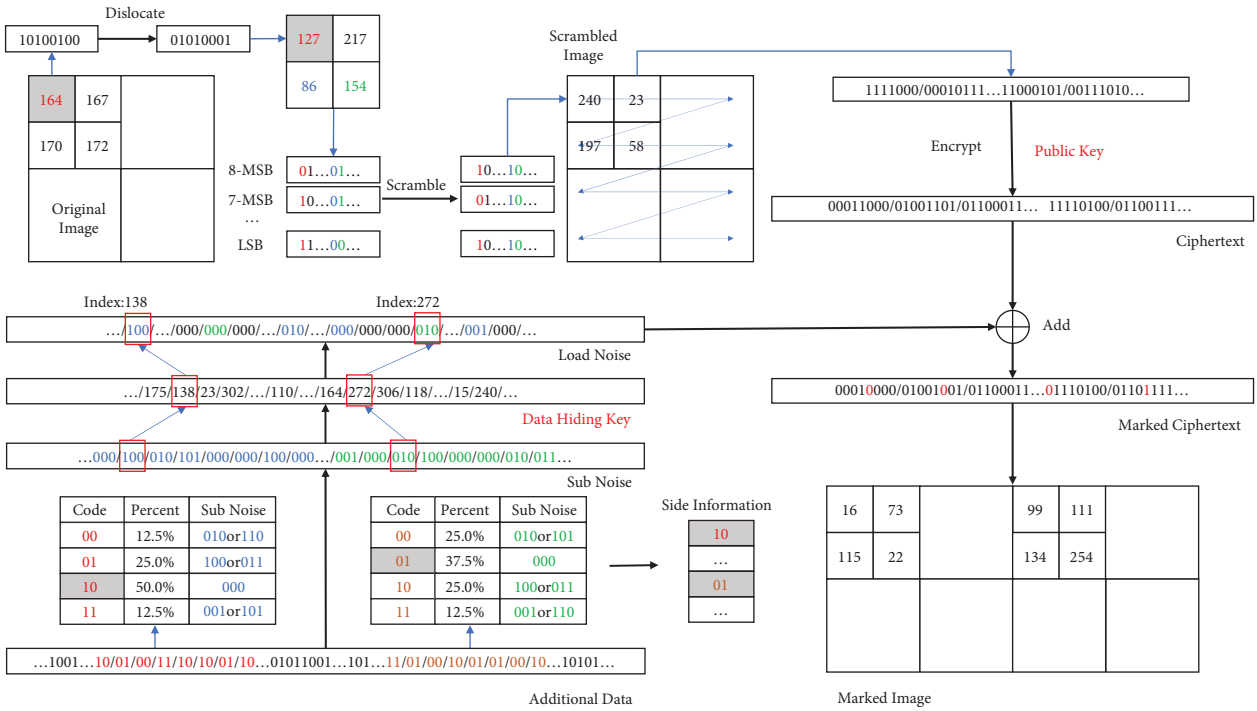


Figure 2: An example of the NM-RDHED scheme.

172] is scrambled to [127, 217, 86, 54]. Secondly, each bit plane is scrambled by the Arnold algorithm and obtained a scrambled image, and the 8-MSBs of the dislocated pixels make up a new sequence "01…01…" that is scrambled to "10…10…"; thus, the scrambled pixels are [240, 23, 197, 58]. Then, all scrambled pixels are transformed into many groups of binary sequences that consist of 314 bits "1111100000011111…1100010100111010…," and they are encrypted by the public key of the receiver. The ciphertext sequence "0001100001001101011000011…1111010001100111…" is obtained, which extends to 1024 bits. A group of encrypted additional data "1001001110100110" that contains 16 bits, where each 2-bit is a code word, and count them; we find that the code word '10' takes up 50%, both '00' and '11' take up 12.5%, and the code word '01' takes up 25%. According to the modulation principle, there are $24 \times 72 = 1728$ kinds of modulation tables, and only one is adopted that is determined by the data hiding key.

Here, taking the left modulation table as an instance, the code word "00" can be modulated into "010" or "110" based on the number of subnoises provided in Table 1. However, the code word "10" only can be modulated into "000" due to it having the highest percentage. After that, the subnoise "100" is filled into the index of 138 of the load noise, where the index is formed by the data hiding key. Until all subnoises are filled, the unfilled parts are filled with "000." Finally, the load noise is added to the ciphertext to get a marked ciphertext. The process of data extraction is the opposite of embedding.

4. Experimental Results

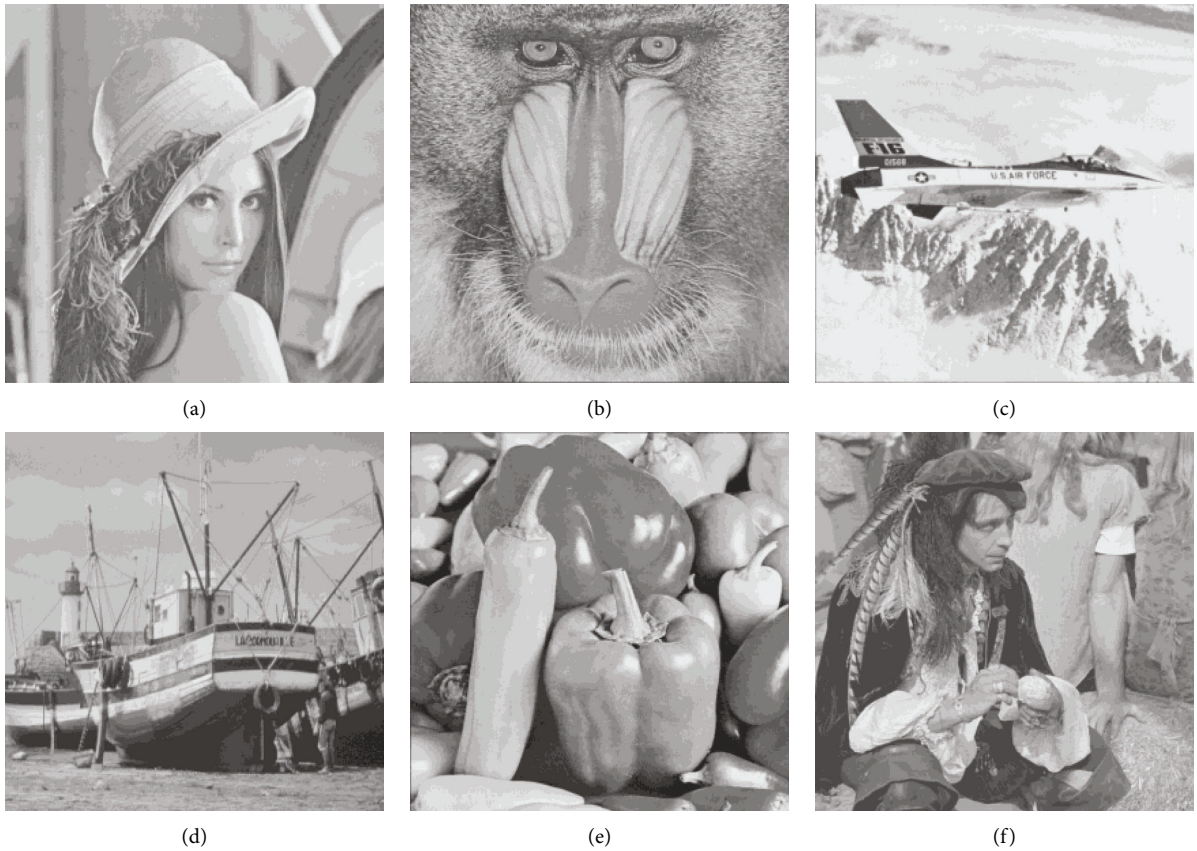
In this section, we use six different features of grey-scale images with a size of 512×512 as a case of signal to experiment, as shown in Figure 3. Furthermore, 100 images are randomly selected from the BOSS Base library and converted into binary sequences as a universal signal. The results are elaborated to demonstrate the performance of the proposed scheme. The simulation program is run on a computer with eight cores and a 2.30 GHz CPU, 32 GB of RAM, and a Windows 10 operating system with MATLAB 2021b.

4.1. Embedding Rate. In our scheme, additional bits embed into a group of load noise at each time, and the groups are independent of each other. The load noise is of the same length as the encrypted data, which is considered a cover. What's more, the side information also affects the actual embedding rate. It converts the number of noise bits into the number of pixel blocks and uses bit per pixel (bpp) as the unit. Define the embedding rate (ER) and effective embedding rate (EER) as follows:

$$\begin{aligned} \mathrm{ER} &= 8 \cdot \frac{\text{Embedded bits}}{\text{Noise bits}}, \\ \mathrm{EER} &= 8 \cdot \frac{\text{Embedded bits} - \text{Side information bits}}{\text{Noise bits}}. \end{aligned} \tag{16}$$

There are two primary factors affecting the ER of the NM-RDHED scheme, which is not constrained by the image content. Thus, we generate 100,000 random sequences by PRSG as the encrypted additional data to evaluate ER.

4.1.1. The Factor of Encryption Parameters. When m is fixed, the larger t is, the higher the ER, because there are more bits "1" in the noise to load the additional bits. Due to the length



(a) (b) (c)

(d) (e) (f)

Figure 3: Six different features grey-scale test images: (a) Lena, (b) Baboon, (c) Plane, (d) Boat, (e) Peppers, and (f) Man.

Table 4: Embedding rate of the proposed scheme in different settings.

ER (bpp)	$m = 10$			$m = 11$			$m = 12$		
	$t = 53$	$t = 71$	$t = 97$	$t = 53$	$t = 71$	$t = 97$	$t = 53$	$t = 71$	$t = 97$
Best	1.88	2.50	3.28	0.94	1.23	1.65	0.49	0.62	0.81
Worst	1.09	1.52	2.13	0.57	0.78	1.11	0.29	0.40	0.55
Average	1.37	1.87	2.60	0.70	0.95	1.30	0.35	0.48	0.66

n of noise being determined by the parameters m, fixing t, and increasing m, the ER decreases. Table 4 shows the embedding rate of the proposed scheme in different settings, and the average ER reaches 2.60 bpp when $m = 10$, $t = 97$. To show the embedded rate more visually and comprehensively, Figure 4 shows the trend of ER, which is linearly and positively correlated with t. When $m = 12$, $t = 340$, the ER still reaches 2.34 bpp. To illustrate the embedding performance, the ER on 100 randomly selected images from BOSS Base is shown in Figure 5.

4.1.2. The Factor of Grouping Length. On the one hand, the shorter the grouping length v, the higher the statistical correlation of code words, and the higher the percentage of certain code words. It means there are more subnoises with a Hamming weight of 0 that can be used to load additional data according to the modulation principle. Figure 6(a) provides the percentages of the largest percentage in the 100,000 tests after counting the largest percentage of code words in different-length encrypted data. We found that the largest percentage is 37% when the grouping length is 16 bits, which is over 50%, and the largest percentage is 50% whose percentage is over 30%. We conclude that the shorter the grouping length, the higher the ER. On the other hand, the code words with the highest percentage are recorded as the side information in the embedding process. The amount of side information depends on v. Consider that the EER is constrained by v. Figure 6(b) illustrates the influence of grouping length on the ER. The longer the grouping length is, the smaller the amount of side information is, the smaller the effect on the EER is, and the EER is closer to the ER. However, with the grouping length increasing, the ER decreased.

Table 5 presents a comparison of embedding rates with different schemes. Both schemes [16, 34] are based on stream cipher, which embeds data by using MSB replacement. The former sufficiently uses the image redundancy recursively and gets a higher ER, an average of 1.71 bpp. The latter does not

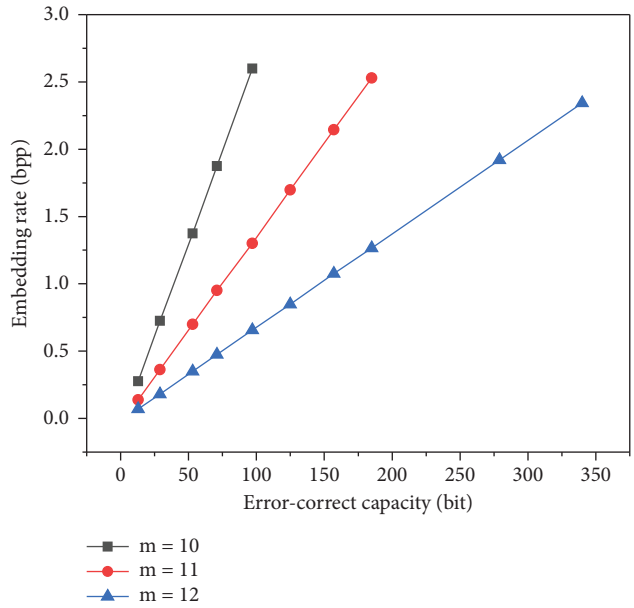


Figure 4: Average embedding rate in different encryption parameters.

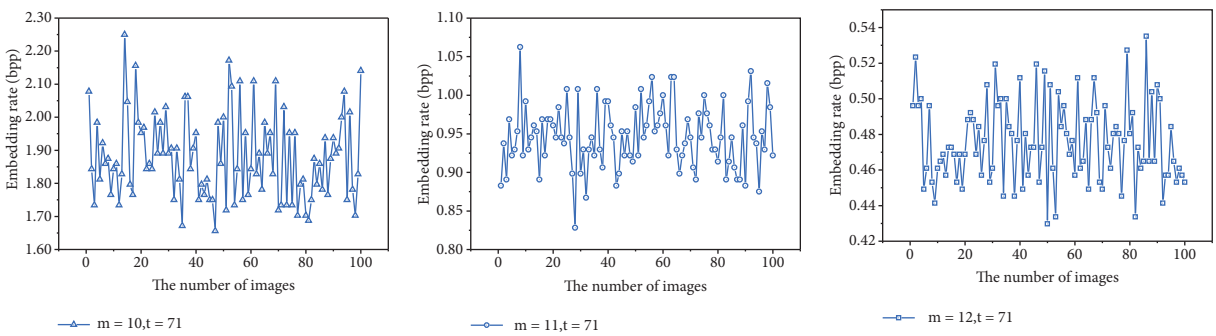


Figure 5: Embedding rate of NM-RDHED on 100 randomly selected images from BOSS Base.

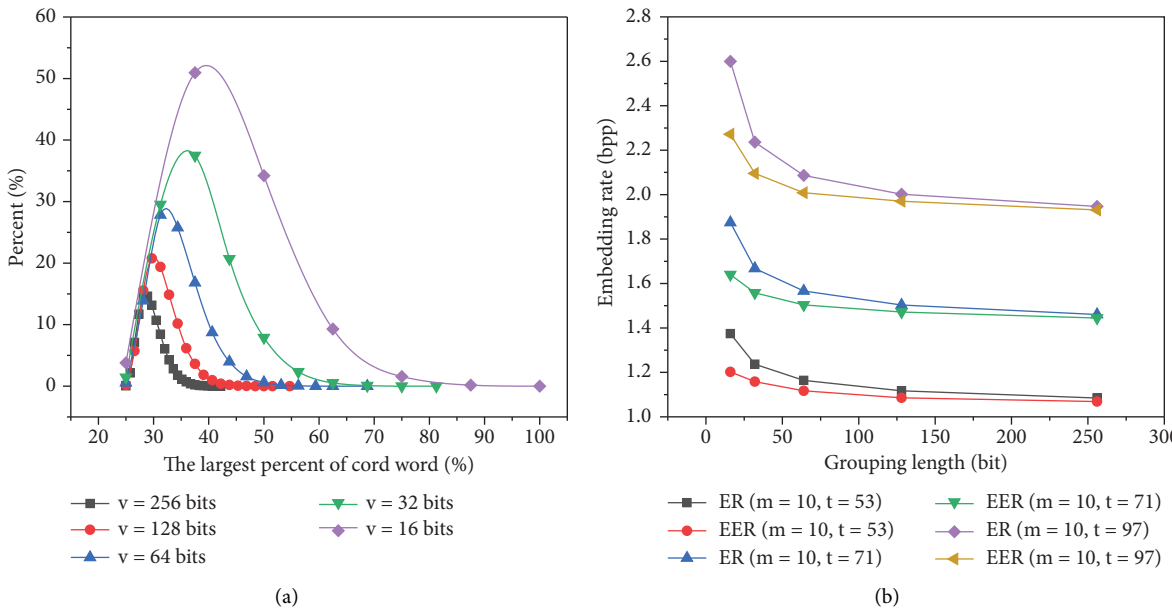


Figure 6: Average embedding effect of different grouping lengths: (a) Percentages of the largest percentage of the code word in 100,000 tests in different grouping lengths of encrypted data and (b) average ER and EER in different grouping lengths.

Table 5: Comparison with other schemes in aspects of the ER and PSNR.

Schemes	Encryption methods	ER (bpp)							PSNR
		Lena	Baboon	Plane	Boat	Peppers	Man	Average	
[16]	Stream cipher	1.70	0.87	0.96	1.50	1.66	1.45	1.71	$+\infty$
[34]	Stream cipher	0.25	0.24	0.25	0.25	0.25	0.25	0.24	≥35
[25]	Paillier	0.50	0.50	0.50	0.50	0.50	0.50	0.50	≥40
[26]	Paillier	0.36	0.27	0.31	0.24	0.29	0.28	0.30	≥35
[27]	Paillier	0.56	0.30	0.73	0.43	0.43	0.41	0.56	≥35
[22]	FHE	0.42	0.26	0.44	0.37	0.42	0.40	0.40	≥40
[32]	Secret sharing	2.91	1.25	3.24	2.78	2.57	2.19	2.55	$+\infty$
[33]	Secret sharing	0.33	0.16	0.38	0.21	0.31	0.28	0.32	≥45
[23]	McEliece	2.11	0.61	2.18	1.61	1.94	1.74	1.70	$+\infty$
Proposed	McEliece	2.93	2.13	2.85	2.41	2.62	2.21	2.53	$+\infty$

consider the redundancy of nature images so that ER is smaller and more stable. Besides, the public key encryption-based schemes [25–27] are aiming to embed additional data in encrypted images directly, which is achieved by homomorphic addition. Therefore, the embedding rate is lower and is constrained by the Paillier encryption. Moreover, fully homomorphic encryption encapsulated a difference expansion scheme [22], as expected, which ER is not higher because of the principle of DE, as well as the scheme [33]. For schemes [23, 32], even if the encryption methods are different, their higher ER still depends on the image correlation. However, the ER of our scheme is independent of the image content. As a result, our scheme has a higher ER than others and the average ER reaches 2.53 bpp with sufficient security.

4.2. Reversibility. The reversibility of the reconstructed images can be analysed in two aspects. According to the embedding principle, the main consideration is whether there are data that are discarded during the embedding procedure and cannot be reconstructed directly or indirectly. Also, the peak signal-noise ratio (PSNR) or structural similarity (SSIM) is used to evaluate the distortion degree of the reconstructed image compared with the original image.

The additional data are modulated into load noise before adding to the ciphertext, and the obtained marked ciphertext is equal to a new ciphertext, because the disturbance of the noise to the ciphertext is within the decryption error correction capability. The marked image is directly decrypted, can entirely correct the load noise, and reconstruct the original image, which ensures the reversibility of the proposed scheme.

Table 5 also shows the comparison of reconstructed image quality with other schemes. These results of the PSNR in schemes [22, 25–27, 33, 34] are calculated by comparing the directly decrypted image with the original image, which all have good visual quality. Sometimes, it is necessary to introduce additional operations to recover images lossless, like schemes [16, 23, 32]. Furthermore, we randomly select 100 images from BOSS Base to test the quality of the constructed images. Table 6 gives the results of the PSNR and SSIM in different parameters and EC, where the PSNR reaches infinity when the EC = 300,000 bits, which means there is no difference between reconstructed images and original images. SSIM evaluates the constructed image quality from three metrics of luminance, contrast, and structure. In different parameters and embedding capacities, SSIM reaches the expected value of 1, so the constructed image is lossless. We conclude that our scheme is completely reversible.

Table 6: The PSNR and SSIM of the proposed scheme on 100 randomly selected images from BOSS base.

Metrics	$m = 10,\ t = 53$	$m = 11,\ t = 71$	$m = 12,\ t = 97$
EC (bit)	300,000	300,000	200,000
PSNR	$+\infty$	$+\infty$	$+\infty$
SSIM	1	1	1

4.3. Data Expansion and Complexity. After McEliece encryption, the length of the binary ciphertext sequence is greater than that of the plaintext, which is called data expansion. We define the data expansion rate as follows:

$$\mathrm{EX} = \frac{n}{k} = \frac{2^m}{2^m - m \cdot t}, \tag{17}$$

where the $[m, n, k, t]$ are parameters of McEliece.

In our scheme, the data expansion is caused by encryption and is not related to the embedding operation. However, data expansion is a negative effect in pursuing higher embedding rates; for instance, for fixing $m$and$n$, a larger ER can only be obtained by increasing t, but it will lead to unacceptably data expansion. Therefore, an excellent trade-off obtained between data expansion and embedding rate determines an appropriate parameter t. Figure 7(a) provides a reference basis. Here, it reaches a better trade-off between embedding rate and side information.

To evaluate the time complexity of the scheme, we use the number of the groups performing the embedding operation as a metric and denote the total embedding capacity as TEC, each group embedding capacity as EC. Furthermore, because the embedding production is performed in groups, and groups are independent of each other; thus, as the embedding capacity increases, the increasing of time consumption is linear complexity:

$$\frac{\mathrm{TEC}}{\mathrm{EC}} = \frac{8 \cdot \mathrm{TEC}}{2^m \cdot \mathrm{ER}} \longrightarrow O(N), \tag{18}$$

where the embedding rate (ER) could be regarded as a constant.

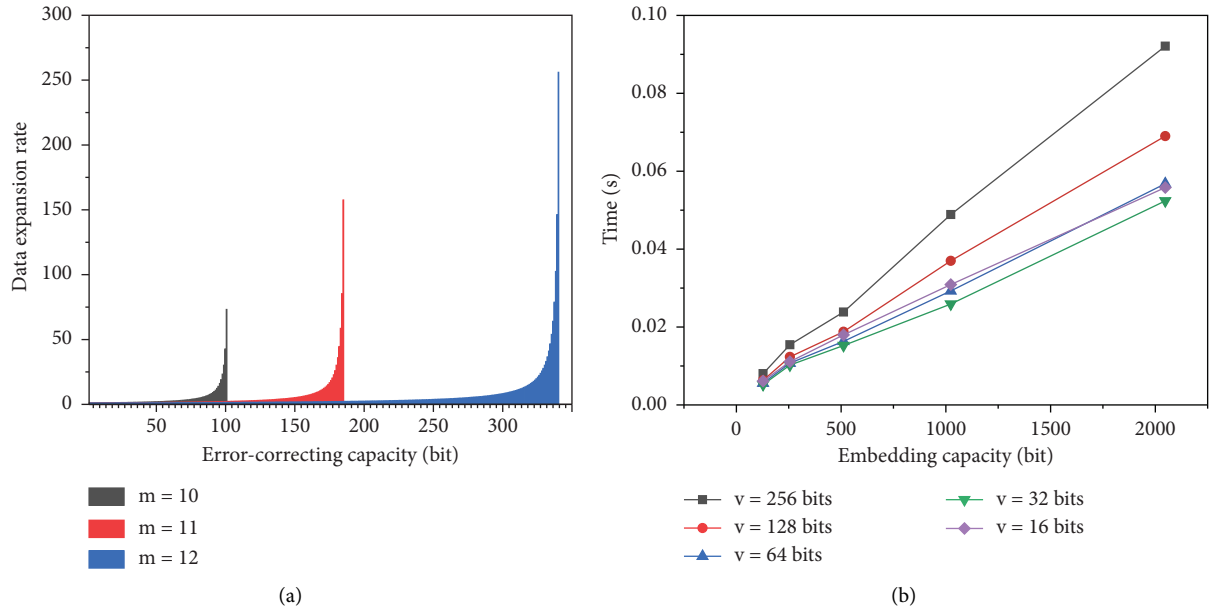


Figure 7: Data expansion and complexity of the proposed scheme: (a) data expansion rate in different encryption parameters and (b) computational cost (in seconds) in different grouping lengths and embedding rates ($m = 11$, $t = 71$).

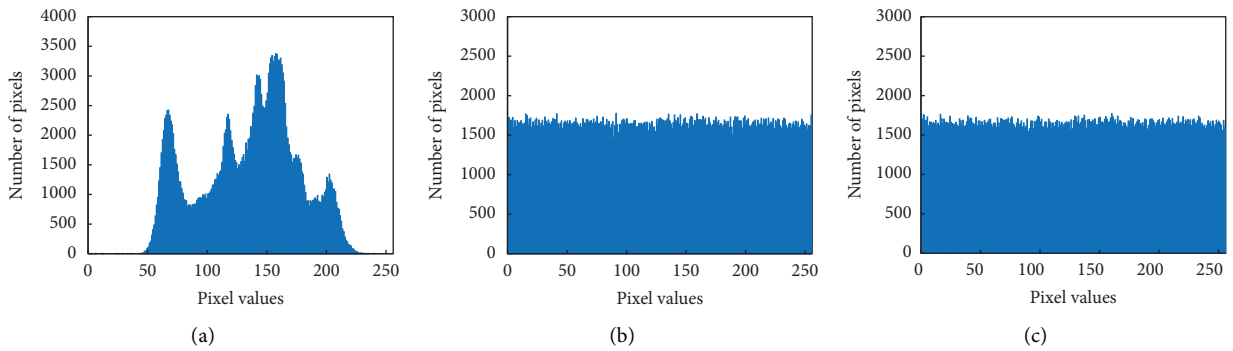


Figure 8: Histogram of images before and after embedding data: (a) histogram of the original image, (b) histogram of the encrypted image, and (c) histogram of the marked image.

Figure 7(b) provides computational cost in different grouping lengths and embedding capacities. When the embedding capacity is fixed, the longer the grouping length v, the more the run-time cost is. However, the computational cost is minimal under the 32 bits grouping for embedding 128 bits, which just costs 0.005233 s.

4.4. Security Analysis. In this part, we evaluate the security of NM-RDHED from the aspects of statistical characteristics of marked images and differential attacks. As a result, the proposed scheme has higher security.

4.4.1. Statistical. As for a secure RDH-ED scheme, the marked images and the encrypted images should have similar statistical properties. To find the difference between an encrypted image and a marked image, Figure 8 gives the histogram of the original, encrypted, and marked images of Lena. It is easy to find that the histograms of the marked image and the encrypted image are similar, and both obey a uniform distribution, unlike the statistical features of the original images. Besides the histogram, correlation is also supposed to be considered. The correlation between neighbouring pixels in nature images is very strong. Figure 9(a) shows a correlation between Lena. We randomly select 3000 pair pixels to test the correlation of the marked image in the horizontal and vertical direction and assess the influence of embedding operation on it. As shown in Figures 9(b) and 9(c), they do not have any correlation. Therefore, the embedding operation does not affect it, and the marked image is secure in statistics.

4.4.2. Differential Attack. Image security encryption theory requires that encrypted images must be extremely sensitive to plaintext and keys; otherwise, they cannot effectively resist differential attacks. Number of pixel change rate

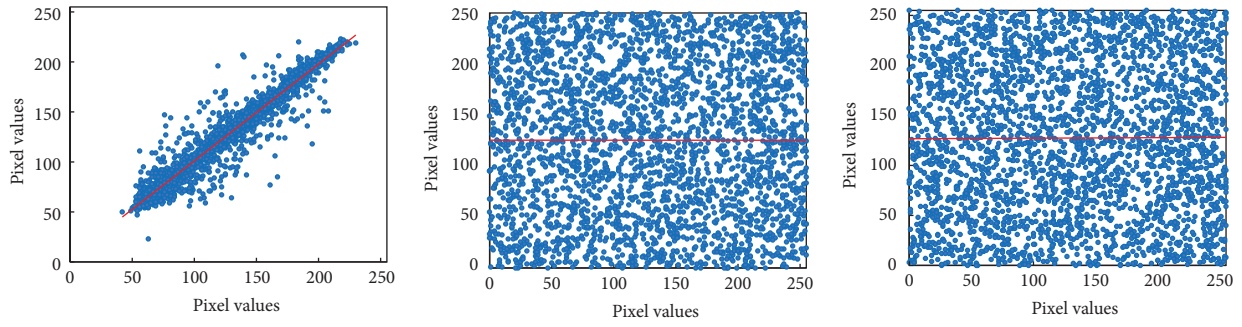


Figure 9: Correlation scatter of images: (a) correlation of original image, (b) horizontal correlation of marked image, and (c) vertical correlation of marked image.

Table 7: Results of Entropy, NPCR, and UACI in the NM-RDHED scheme.

Parameters		Entropy		NPCR (%)	UACI (%)
		Encrypted image	Marked image		
$m = 10$	$t = 53$	7.999664752	7.999677022	99.61791485	33.41627523
	$t = 71$	7.999752149	7.999765369	99.60227857	33.44588781
	$t = 97$	7.999964888	7.999966860	99.60777406	33.45717517
$m = 11$	$t = 53$	7.999490145	7.999527118	99.62612496	33.49962531
	$t = 71$	7.999465042	7.999512357	99.60785974	33.49380102
	$t = 97$	7.999653732	7.999642044	99.61431632	33.41455707
$m = 12$	$t = 53$	7.999303357	7.999342806	99.61652476	33.46878585
	$t = 71$	7.999375217	7.999449063	99.58013714	33.37529066
	$t = 97$	7.999423866	7.999534749	99.60679129	33.37616961

(NPCR) and the normalized average changing intensity (UACI) are used as an important indicator of cryptanalysis. When the image encryption method is secure enough, the sensitivity of the NPCR and UACI to the plaintext is analysed for grey-scale images with 8 bits depth. The expected values of the NPCR and UACI are 99.6094% and 33.4635%, respectively.

Considering an image $\mathbf{I}$, we modify one pixel of it and denote the modified image $\mathbf{I}'$, and encrypt them with the same public key in different settings. Next, during the disguising process, random noise is added to one image, and load noise with additional data is added to the other; two marked images are $\mathbf{I}_m$ and $\mathbf{I}_m'$ obtained. Calculate the NPCR and UACI with them, as listed in Table 7. We can know that the NPCR and UACI are very close to the theoretical values. The embedding scheme does not affect the security of the original encryption algorithm and can effectively resist differential attacks. Meanwhile, the entropy of marked images is close to the limit of entropy 7.99. This is because the load noise is indistinguishable and does not affect the security of the McEliece encryption.

5. Conclusion

This article proves the redundancy room of McEliece encryption that can be used to embed additional data and proposes a new noise modulation-based reversible data hiding in the encrypted domain scheme called NM-RDHED, which is suitable for any signal processing. Any data hider could embed additional data into the encrypted image, but only the receiver with a private key and a data hiding key could extract the embedded data. Compared with other schemes in aspect of the embedding rate, the proposed scheme has a higher ER. Although the side information influences on the ER, an appropriate grouping length makes an excellent trade-off and maintains a higher ER. The reconstructed image with no distortion after direct decryption of a marked image is superior to the state-of-the-art schemes. Our scheme shows better security in both statistical security and resistance to differential attack analysis, and McEliece as a postquantum cryptographic algorithm can resist quantum computing attacks, so the scheme has higher security and meets the demand of RDH-ED for future security development. In the future, we concentrate on reducing the amount of side information and improving the embedding rate.

Data Availability

The BOSSBase database images used in this article are from https://agents.fel.cvut.cz/boss/index.php?mode=view&tmpl=materials, other data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China, under grants nos. 61872384, 62102450, 62102451, and 62202496.

References

[1] 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.

[2] P. Puteaux, S. Y. Ong, K. S. Wong, and W. Puech, "A survey of reversible data hiding in encrypted images: the first 12 years," *Journal of Visual Communication and Image Representation*, vol. 77, Article ID 103085, 2021.

[3] S. Kumar, A. Gupta, and G. S. Walia, "Reversible Data Hiding: A Contemporary Survey of State-Of-The-Art, Opportunities and Challenges," *Applied Intelligence*, vol. 52, pp. 1–34, 2021.

[4] W. Puech, M. Chaumont, and O. Strauss, "A reversible data hiding method for encrypted images," *Proceedings of SPIE, Security, forensics, steganography, and watermarking of multimedia contents X*, vol. 6819, pp. 534–542, 2008.

[5] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, no. 4, pp. 255–258, 2011.

[6] W. Hong, T. S. Chen, and H. Y. Wu, "An improved reversible data hiding in encrypted images using side match," *IEEE Signal Processing Letters*, vol. 19, no. 4, pp. 199–202, 2012.

[7] X. Zhang, "Separable reversible data hiding in encrypted image," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 826–832, 2012.

[8] Z. Yin, B. Luo, and W. Hong, "Separable and error-free reversible data hiding in encrypted image with high payload," *The Scientific World Journal*, vol. 2014, Article ID 604876, 8 pages, 2014.

[9] M. S. Abdul Karim and K. S. Wong, "Data embedding in random domain," *Signal Processing*, vol. 108, pp. 56–68, 2015.

[10] Z. Qian and X. Zhang, "Reversible data hiding in encrypted images with distributed source encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 4, pp. 636–646, 2016.

[11] D. Xiao, Y. Xiang, H. Zheng, and Y. Wang, "Separable reversible data hiding in encrypted image based on pixel value ordering and additive homomorphism," *Journal of Visual Communication and Image Representation*, vol. 45, pp. 1–10, 2017.

[12] H. Ge, Y. Chen, Z. Qian, and J. Wang, "A high capacity multilevel approach for reversible data hiding in encrypted images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 8, pp. 2285–2295, 2019.

[13] C. Yu, X. Zhang, X. Zhang, G. Li, and Z. Tang, "Reversible data hiding with hierarchical embedding for encrypted images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 2, pp. 451–466, 2022.

[14] C. Yu, X. Zhang, G. Li, S. Zhan, and Z. Tang, "Reversible data hiding with adaptive difference recovery for encrypted images," *Information Sciences*, vol. 584, pp. 89–110, 2022.

[15] 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.

[16] P. Puteaux and W. Puech, "A recursive reversible data hiding in encrypted images method with a very high payload," *IEEE Transactions on Multimedia*, vol. 23, pp. 636–650, 2021.

[17] K. Chen and C. C. Chang, "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, 2019.

[18] Y. Wu, Y. Xiang, Y. Guo, J. Tang, and Z. Yin, "An improved reversible data hiding in encrypted images using parametric binary tree labeling," *IEEE Transactions on Multimedia*, vol. 22, no. 8, pp. 1929–1938, 2020.

[19] 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, 2016.

[20] Y. Qiu, Q. Ying, Y. Yang, H. Zeng, S. Li, and Z. Qian, "High-capacity framework for reversible data hiding in encrypted image using pixel prediction and entropy encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 9, pp. 5874–5887, 2022.

[21] Y. Ke, M. Q. Zhang, J. Liu, T. T. Su, and X. Y. Yang, "A multilevel reversible data hiding scheme in encrypted domain based on LWE," *Journal of Visual Communication and Image Representation*, vol. 54, pp. 133–144, 2018.

[22] Y. Ke, M. Q. Zhang, J. Liu, T. T. Su, and X. Y. Yang, "Fully homomorphic encryption encapsulated difference expansion for reversible data hiding in encrypted domain," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2353–2365, 2020.

[23] Y. Kong, M. Zhang, Z. Wang, Y. Ke, and S. Huang, "Reversible data hiding in encrypted domain based on the error-correction redundancy of encryption process," *Security and Communication Networks*, vol. 2022, Article ID 6299469, 17 pages, 2022.

[24] Z. Qian, X. Zhang, Y. Ren, and G. Feng, "Block cipher based separable reversible data hiding in encrypted images," *Multimedia Tools and Applications*, vol. 75, no. 21, pp. 13749–13763, 2016.

[25] Y. C. Chen, C. W. Shiu, and G. Horng, "Encrypted signal-based reversible data hiding with public key cryptosystem," *Journal of Visual Communication and Image Representation*, vol. 25, no. 5, pp. 1164–1170, 2014.

[26] X. Zhang, J. Long, Z. Wang, and H. Cheng, "Lossless and reversible data hiding in encrypted images with public-key cryptography," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1622–1631, 2016.

[27] 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.

[28] C. S. Tsai, Y. S. Zhang, and C. Y. Weng, "Separable reversible data hiding in encrypted images based on paillier cryptosystem," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 18807–18827, 2022.

[29] H. T. Wu, Y. M. Cheung, Z. Zhuang, L. Xu, and J. Hu, "Lossless data hiding in encrypted images compatible with homomorphic processing," *IEEE Transactions on Cybernetics*, pp. 1–14, 2022.

[30] X. Wu, J. Weng, and W. Yan, "Adopting secret sharing for reversible data hiding in encrypted images," *Signal Processing*, vol. 143, pp. 269–281, 2018.

[31] B. Chen, W. Lu, J. Huang, J. Weng, and Y. Zhou, "Secret sharing based reversible data hiding in encrypted images with multiple data-hiders," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 2, pp. 978–991, 2022.

[32] Z. Hua, Y. Wang, S. Yi, Y. Zhou, and X. Jia, "Reversible data hiding in encrypted images using cipher-feedback secret sharing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 8, pp. 4968–4982, 2022.

[33] Y. Ke, M. Zhang, X. Zhang, J. Liu, T. Su, and X. Yang, "A reversible data hiding scheme in encrypted domain for secret image sharing based on Chinese remainder theorem," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 4, pp. 2469–2481, 2022.

[34] M. Yu, H. Yao, and C. Qin, "Reversible data hiding in encrypted images without additional information transmission," *Signal Processing: Image Communication*, vol. 105, Article ID 116696, 2022.

[35] N. J. Alfardan, B. Poettering, and J. Schuldt, "On the security of RC4 in TLS and WPA," *USENIX Security Symposium*, vol. 173, 2013.

[36] L. Chen, S. P. Jordan, Y. K. Liu, D. Moody, and R. Peralta, *Report on post-quantum Cryptography*, NIST, Gaithersburg, MD, USA, 2016.

[37] G. Alagic, J. Alperin-Sheriff, and D. Apon, *Status Report on the Second Round of the NIST post-quantum Cryptography Standardization Process*, US Department of Commerce, NIST, Gaithersburg, MD, USA, 2020.

[38] R. J. McEliece, "A public-key cryptosystem based on algebraic," *Coding Thv*, vol. 4244, pp. 114–116, 1978.

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

[40] Z. Guan, J. Li, L. Huang, X. Xiong, Y. Liu, and S. Cai, "A novel and fast encryption system based on improved Josephus scrambling and chaotic mapping," *Entropy*, vol. 24, no. 3, p. 384, 2022.

[41] M. Li, T. Liang, and Y. J. He, "Arnold Transform Based Image Scrambling Method," in *Proceedings of the 3rd International Conference on Multimedia Technology*, pp. 1309–1316, Guangzhou, China, December 2013.

[42] G. Qi, M. A. Van Wyk, B. J. Van Wyk, and G. Chen, "A new hyperchaotic system and its circuit implementation," *Chaos, Solitons & Fractals*, vol. 40, no. 5, pp. 2544–2549, 2009.

[43] N. Yujun, W. Xingyuan, W. Mingjun, and Z. Huaguang, "A new hyperchaotic system and its circuit implementation," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, no. 11, pp. 3518–3524, 2010.