Reversible Data Hiding Schemes in Encrypted Images Based on the Paillier Cryptosystem

Hefeng Chen[1], Chin-Chen Chang[2], and Kaimeng Chen[1]

(Corresponding author: Chin-Chen Chang)

Computer Engineering College, Jimei University[1]
Xiamen 361021, PR China
Department of Information Engineering and Computer Science, Feng Chia University[2]
Taichung, Taiwan
(Email: alan3c@gmail.com)

(Received Aug. 19, 2018; Revised and Accepted Aug. 10, 2019; First Online Feb. 28, 2020)

Abstract

In this paper, we propose a novel framework for reversible data hiding schemes in encrypted images inspired by the privacy needs of outsourcing data in the cloud service. Our scheme allows the image owner and the data provider to send encrypted images and encrypted data to the data processor separately; then, the data processor can do the embedding without knowing any side information; the receiver would obtain the marked image after decryption and could extract the hidden data and completely recover the original image. By exploiting the Paillier homomorphism and the equivalence of the modular approach, the high capacity of at least 1 bpp and even exceeding 1016 bpp can be achieved at one-time embedding. Then, we extended the first scheme to provide a multi-receiver, reversible data hiding scheme by combining our approach with the(t, w)-threshold secret sharing homomorphism. It is suitable for the application of distributed storage with fault tolerance or the protection of patients' privacy when they are consulting with multiple doctors.

Keywords: Homomorphic Encryption; Reversible Data Hiding; Secret Sharing

1 Introduction

Reversible data hiding (RDH) is a technique that embeds secret data into the cover medium in a reversible manner. In the RDH scheme, the embedded data can be extracted correctly, and, also, the cover medium can be recovered perfectly from the marked data. Prior studies have proposed several approaches for RDH, such as difference expansion [7,11], lossless compression [21], histogram shifting [15], and prediction error expansion [4]. Motivated by the need to preserve privacy in cloud computing and other applications for securely storing or sharing multimedia files with others, the combination of data hiding and encryption has received increasing attention. RDH for encrypted images enables cloud servers to reversibly embed data into images, but no knowledge about image content is available.

The first encrypted image-based RDH scheme was proposed by Puech *et al.* [20], who used the bit substitution method to embed one bit into a block of pixels encrypted by Advanced Encryption Standard. The extraction process is just simple read, and the decryption process is done by analyzing the local standard deviation. In Zhang's scheme [30], the bits of each pixel are encrypted by exclusive-or with pseudo-random bits, and then, the encrypted image is partitioned into blocks. An additional bit is embedded into each block one by one by flipping a portion of the least significant bits (LSBs). The extraction and decryption can be done by examining the fluctuation in natural image blocks. Then, the higher embedding capacity with a lower bit error rate is achieved by defining different evaluation functions based on the spatial correlation of blocks [22], by using a different flipping strategy [12], or by using prediction error [28].

Qin and Zhang [22] proposed the flipped pixels' elaborate selection method to improve the visual effect of the decrypted. Zhou *et al.* [32] proposed a scheme with a high embedding capacity by utilizing a public-key modulation mechanism without sharing the secret data hiding key and a two-class SVM classifier for decoding. In addition, Ma *et al.* [16] reserved room before encryption to obtain large payloads up to 0.5 bit per pixel, and the performance was improved further by considering patch-level sparse representation [6].

However, all of the images are encrypted with symmetric cryptosystem in [1,6,12,16,20,22,28,30,32], making it difficult for them to be processed directly in the encryption domain. This disadvantage can be overcome by introducing the homomorphic encryption. In order to process the encrypted data directly, the special functions called "privacy homomorphism" [23] must be found. In other words, after the ciphertext is processed, an encrypted result is generated that matches the desired plain-text result after decryption. Since the encrypted image can be processed directly, the privacy and confidentiality of the user can be enhanced. Hence, conducting RDH in the homomorphic encryption domain can enrich its availability in cloud computing and other similar scenarios.

Recently, an additive homomorphic Paillier cryptosystem-based RDH scheme [19] also has been investigated [9,14,27,31]. First, Chen *et al.* [9] designed the RDH with the public-key cryptosystem by dividing each pixel value into two portions, *i.e.*, the seven most significant bits (MSBs) and one LSB, and then they performed the encryption using the Paillier cryptosystem. Then, two encrypted LSBs of each encrypted pixel pair are modified to reversibly embed one additional bit following the homomorphism.

Zhang *et al.* [31] used histogram shrink before encryption and used error-correction codes to expand the additional data to achieve reversibility. Wu *et al.* [27] presented two high-capacity RDH schemes, one by doing value expansion on the encrypted pixel values and another by taking advantage of the self-blinding feature of the Paillier encryption. Both embedding capacities are more than 1 bpp. Li *et al.* [14] used histogram shifting in encrypted images to embed bits. Compared with the image RDH algorithms with symmetric cryptography, the proposed algorithms are more suitable for the cloud environment without reducing the security level.

The interpolation-based RDH is also an important work [1,10,13,24]. This paper mainly uses polynomial interpolation technique to realize secret sharing and then solve the RDH problem for multiple receivers.

The (t, w) secret sharing scheme was developed by Shamir [25] based on polynomial interpolation, and it was developed independently by Blakley [5] in 1979 based on geometry. The basic idea is to protect the privacy of information by distribution. In a (t, w) secret sharing scheme, a dealer divides a secret into w shares and the secret is shared among a set of w shareholders, in such a way that any t or greater shareholders can reconstruct the secret, while fewer than t shareholders cannot.

There are other types of secret sharing, *e.g.*, McEliece-Sarwate's scheme [17], which is based on Reed-Solomon codes, and Mignotte's scheme [18] and Asmuth-Bloom's scheme [2], which are based on the Chinese remainder theorem (CRT). In 1987, Benaloh [3] first proposed the concept of secret sharing homomorphism, which allows multiple secrets to be combined by direct computation of shares. This property reduces the need for trust among the agents. Some secret sharing-based, data-hiding algorithms have been presented in literature [8,26]. Recently, Wu *et al.* [29] introduced a model of RDH in encryption domain-based secret sharing. The image content owner encrypts the original image into several shares and sends them to the service provider. The servive provider is responsible for storing and reversibly hiding data into encrypted shares, extracting the hidden data, and sending the encrypted shares to the authenticated receiver who can recover the desired image. This model can be applied to the scenario in which extraction is required for image decryption.

RDH in the encrypted domain is suitable for the scenario in which the image owner and the data hider are not the same person. The image owner would encrypt the medium before transmission, and the data hider can append some additional message into the cipher without knowing the plaintext image. Then, the receiver can recover the original image and extract the embedded data losslessly.

In this paper, we address the issue concerning the separation of the roles of the data provider and the data processor, and both images and data are encrypted before transmission to the data processor. The data processor does not know anything about the image or the hidden data but can integrate them to a new cipher in a way that the receiver can perfectly decrypt the image and extract the data. For example, in electronic-healthcare, medical images and electronic patient records are generated by two different departments, and the information should be encrypted before it is transmitted to the database administrator to protect the patient's privacy.

The database administrator embeds the patient's encrypted record into the corresponding encrypted image to achieve privacy homomorphism. Then, when the doctor receives the marked encrypted medical image, he or she can get the original medical image and data. Our scheme is suitable for the scenario in which image decryption is required for extraction. Also, considering the application scenario after consultation with several doctors, we propose a (t, w) multi-receiver RDH scheme using secret sharing homomorphism to achieve the goal that any t receivers can collaborate with each other by using their shadows to reconstruct the original image and extract the hidden data, which cannot be done unless t or more receivers cooperate.

In short, there are two contributions of our work:

1) Proprose a RDH scheme suitable for data outsourcing, in which the roles of data owner and data hider are separated.

2) Extend the RDH scheme for multi-recever case, combined with the secret sharing technology.

The rest of this paper is organized as follows. Section 2 gives some preliminaries. In Section 3, we review the related works proposed by Chen *et al.* [27] and by Li *et al.* [14]. In Section 4, we propose a high-capacity RDH scheme based on the Paillier cryptosystem. In Section 5, we present another scheme for sharing the marked encrypted image among multiple receivers who have the same decryption key. The performance analysis and the experimental results are shown in Section 6, and our conclusions are made in Section 7.

Table 1: Notations

N	RSA modulus, $N = p \cdot q$, where p and q are two large primes, while $(p-1)/2$ and $(q-1)/2$ are also primes
$\mathbb{Z}_N$	Integers modulo N, $\mathbb{Z}_N = \{0, 1, \ldots, N-1\}$
$\mathbb{Z}_N^*$	Multiplicative group of $\mathbb{Z}_N$, $\mathbb{Z}_N^* = \{r \in \mathbb{Z}_N \vert gcd(r, N) = 1\}$
$\varphi(\cdot)$	Euler's phi function, $\varphi(N) = (p-1)(q-1)$
$\lambda(\cdot)$	Carmichael's function, $\lambda(N) = lcm(\varphi(p), \varphi(q))$
$L(\cdot)$	$L(u) = (u-1)/N, \forall u \in \{u < N^2 \vert u = 1 \, mod \, N\}$
e_{PK}	Encryption algorithm with the receiver's public key PK
d_{SK}	Decryption algorithm with the receiver's private key SK
$\lfloor \cdot \rfloor$	Floor function

2 Preliminaries

Two important techniques were used to design our proposed scheme, *i.e.*, homomorphic cryptosystem and the secret sharing scheme. The former allows direct processing in the encryption domain to reach privacy homomorphism [23], that is, an encrypted result will generate a decryption that matches the desired result without knowing the decryption key. Although Goldwasser-Micali scheme is the classical homomorphic encryption, it only supports additive homomorphism on $\mathbb{Z}_2$ domain, so we finally choose Paillier system. The latter can decompose one secret into shadows that are distributed among shareholders, such that the pooled shadows of specific subsets of users allow the reconstruction of the original secret. To offer sufficient background knowledge of the proposed scheme, these techniques are illustrated as follows. (The notations are listed in Table 1).

2.1 Paillier Homomorphic Cryptosystem

In 1999, Paillier proposed a probabilistic public-key cryptosystem [19] based on the composite residuosity class problem. Paillier's encryption scheme with fast decryption can be described as follows.

2.1.1 Key Generation Phase

Choose an RSA modulus $N = p \cdot q$, where p and q are large primes. Compute Carmichael's function taken on N, *i.e.*, $\lambda = \lambda(N) = lcm(p-1, q-1)$, and choose an element, $g \in \mathbb{Z}_{N^2}^*$, of an order divisible by αN for some α, where $1 \leq \alpha \leq \lambda$.

Now, the public key is $PK = (N, g)$, and the secret key is $SK = \alpha$.

2.1.2 Encryption Phase

The plaintext space is $\mathbb{Z}_N$. Given a plaintext $M < N$, choose $r \in \mathbb{Z}_N^*$ at random, and let the ciphertext be:

$$C = e_{PK}(M) = g^M r^N \, mod \, N^2. \quad (1)$$

2.1.3 Decryption Phase

The plaintext space is $\mathbb{Z}_{N^2}$. Given a ciphertext, $C < N^2$, get the plaintext:

$$M = d_{SK}(C) = \frac{L(C^{\alpha} mod \, N^2)}{L(g^{\alpha} mod \, N^2)} mod \, N, \quad (2)$$

where $L(\mu) = (\mu - 1)/N$.

Based on an appropriate complexity assumption, this system is semantically secure, and it is a trivially additive homomorphism over $\mathbb{Z}_N$, which leads to other identities as we require here:

$$d_{SK}(e_{PK}(M_1) \cdot e_{PK}(M_2) mod \, N^2) = (M_1 + M_2) mod \, N, \quad (3)$$

$$d_{SK}((e_{PK}(M_1))^k mod \, N^2) = (kM_1) mod \, N, \quad (4)$$

$$d_{SK}(e_{PK}(M_1) \cdot g^{M_2} mod \, N^2) = (M_1 + M_2) mod \, N, \quad (5)$$

where $M_1, M_2 \in \mathbb{Z}_N, k \in \mathbb{N}$.

2.2 Shamir's $(t, \; w)$-Threshold Secret Sharing

In 1979, Shamir developed a (t, w)-threshold secret sharing scheme [25] based on polynomial interpolation and the fact that a univariate polynomial $y = f(x)$ of degree $t-1$ is uniquely defined by t points, (x_i, y_i) with distinct x_i, for $i = 1, 2, \cdots, t$. The scheme can decompose one secret into w shadows, with t shadows required to recover the original secret, where $t \leq w$, but no group of $t-1$ shadows can do so. It consists of the following two phases:

2.2.1 Shadow Distribution Phase

The trusted dealer starts with a secret integer, $S \geq 0$, that is to be distributed among w users. Thus, the dealer:

1) Chooses a prime $P > max(w, S)$.

2) Randomly selects $t-1$ independent coefficients $a_1, a_2, \cdots, a_(t-1), 0 \leq a_i \leq P-1$, to constitute a random polynomial with $t-1$ degree over $\mathbb{Z}_P$,

$$f(x) = S + \textstyle\sum_{j=1}^{t-1} a_j x^j \, mod \, P.$$

3) Chooses w distinct non-zero elements of $\mathbb{Z}_P$, denoted as $x_i, 1 \leq i \leq w$.

4) Computes $s_i = f(x_i) mod \, P, 1 \leq i \leq w$, and securely transfers the shadow, s_i, to user U_i, along with the public index x_i.

2.2.2 Secret Reconstruction Phase

Assume that users $U_{i_1}, U_{i_2}, \cdots, U_{i_t}$ pool their shadows to compute the secret S. Their shadows provide t distinct points $(x_{i_j}, s_{i_j})'s$, $1 \leq j \leq t$, which allow the computation of the coefficients of $f(x)$ by Lagrange interpolation. The secret, S, can be expressed as:

$$S = f(0) = \sum_{j=1}^{t} s_{i_j} c_{i_j} mod\, P,$$

where $c_{i_j} = \prod_{1\leq k\leq t, k\neq j} \frac{x_{i_j}}{x_{i_k}-x_{i_j}} mod\, P, 1 \leq j \leq t.$

3 Related Works

By utilizing the redundancy of value representation in the Paillier cryptosystem, Wu *et al.* [27] proposed an encrypted, signal-based RDH for the scenario in which the extraction occurs after decryption. To embed a bit b, a pixel m is mapped to $2m + b$, *i.e.*, $e_{PK}(m)$ is changed to another encrypted value, $e_{PK}(2m + b)$.

There are three parties, *i.e.*, an image owner, a data-hider, and a receiver, corresponding to the three phases. The algorithm runs as follows. In the image encryption phase, for a pixel m, the image owner uses the Paillier cryptosystem to generate the ciphertext, $c = e_{PK}(m)$. In the data embedding phase, to embed a bit b_1, the data-hider sequentially computes:

$$\bar{c} = (c \cdot c) mod\, N^2$$

and

$$c' = \begin{cases} (\bar{c} \cdot e_{PK}(1)) mod\, N^2 & \text{if} \quad b_1 = 1 \\ \bar{c} & \text{if} \quad b_1 = 0 \end{cases},$$

which implies that $c' = e_{PK}(2m + b_1)$. When the Paillier modulus N is chosen to be sufficiently large to ensure that, in data extraction phase, $2m + b_1 < N$, *i.e.* $2m + b_1 mod\, N = 2m + b_1$, the receiver can obtain the correct values of the original pixel, m, and the hidden bit, b_1, by computing:

$$m = \lfloor d_{SK}(c')/2 \rfloor$$

and

$$b_1 = d_{SK}(c') - 2m.$$

The embedding of multiple bits can be accomplished iteratively. For example, if the second bit, b_2, is to be embedded into the encrypted value of the pixel m, based on the encrypted value, c', of the pixel with hidden bit, b_1, the data-hider sequentially computes:

$$\bar{c'} = (c' \cdot c') mod\, N^2$$

and

$$c'' = \begin{cases} (\bar{c'} \cdot e_{PK}(1)) mod\, N^2 & \text{if} \quad b_2 = 1 \\ \bar{c'} & \text{if} \quad b_2 = 0 \end{cases},$$

which implies that $c'' = e_{PK}(2(2m + b_1) + b_2)$.

Therefore, if one wants the embedding rate to reach μ bpp, μ iterations are required, and some room must be vacated for recording associated information, such as the number of iterations.

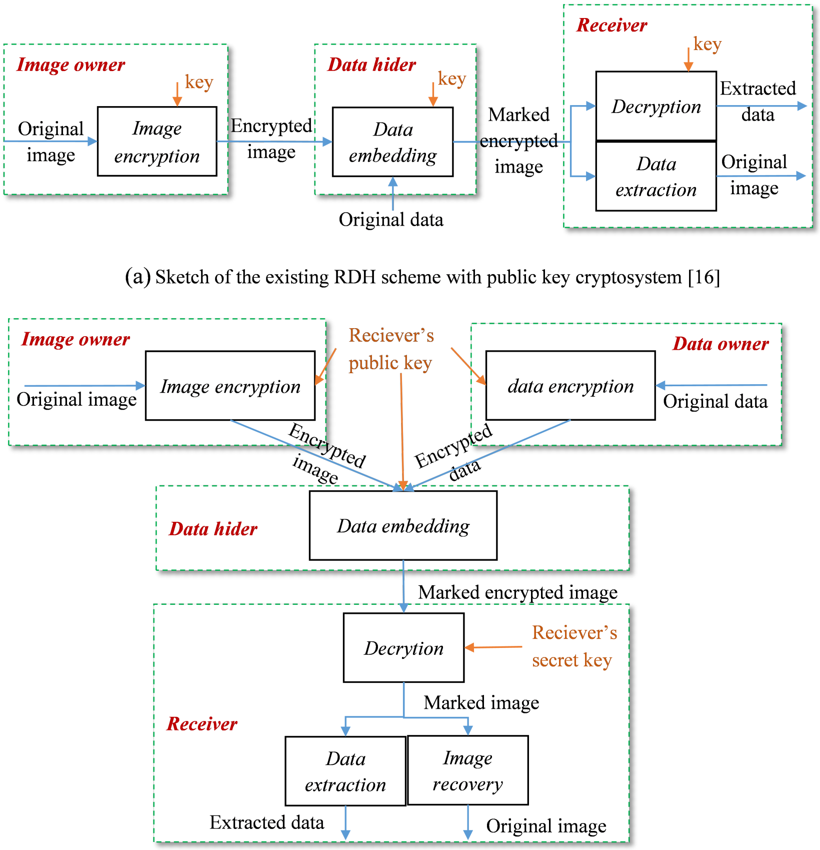
Similarly, Li *et al.* [14] encrypted the image pixel using the Paillier cryptosystem, and then they handled data embedding from the perspective of histogram shifting in the plain domain. First, for embedding one bit per pixel, the histogram of the host image is expanded by a factor of two, *i.e.*, from [0, 255] to [0, 511], so that the zero bins in the expanded histogram with odd numbers are vacated. Second, for embedding a bit, b, into the pixel, m, if the embedded bit b is 1, the corresponding unit of $2m$ in the expanded histogram shifts right by one step. The value $2m$ is processed in the plaintext image, the encryption of which can be obtained by computing $\bar{c} = (e_{PK}(m))^2 mod\, N^2$, and the encrypted value of the pixel m with the hidden bit is obtained by computing $c' = (\bar{c} \cdot g^b) mod\, N^2$. According to the additive homomorphism, c' is a valid encryption of $2m + b$, so reversibility is achieved. When one wants the embedding rate to be 1016 bits, the Paillier modulus N must be at least 1024 bits, and the expansion ratio of the pixel is 2^{1016}.Thus, the calculation in encryption domain is $\bar{c} = (e_{PK}(m))^{2^{1016}} mod\, N^2$.

4 A Reversible Data Hiding Scheme with Single-Receiver

In this section, we present an RDH scheme using the Paillier cryptosystem for a single receiver. Figure 1(b) shows the flowchart of the process. There are five parties in the scheme, *i.e.*, a trusted dealer, an image owner, a data provider, a data processor, and a receiver. The trusted dealer generates the Paillier cryptosystem's public/private key pair, secretly sends a private key to the receiver, and broadcasts the public key to everyone. The image owner and the data provider use the Paillier cryptosystem with the receiver's public key to encrypt each pixel of the host image and the initial data, respectively, and the data provider, who does not know the host image, can modify the pixel values of the ciphertext to embed some additional data into the encrypted image. After receiving the encrypted image with the additional data embedded, the receiver, who has the private key of the cryptosystem, can execute decryption directly to get a marked image and then recover the hidden data and the host image perfectly.

4.1 Initialization Phase

When inputting the payload parameter, K, choose an RSA modulus, N,that is greater than K, such that $K' = \lfloor N/K \rfloor > 255$. Then the image space is $\{0, 1, \cdots, K'-1\}$, and the data space is $\{0, 1, \cdots, K - 1\}$. Choose an element, $g \in \mathbb{Z}^*_{N^2}$, that has an order divisible by αN for some $1 \leq \alpha \leq \lambda(N)$.



(a) Sketch of the existing RDH scheme with public key cryptosystem [16]

(b) Sketch of the proposed RDH scheme with public key cryptosystem

Figure 1: Comparison between the existing scheme and the proposed RDH scheme

The public key $PK = (N, K, g)$ is broadcasted, and the secret key $SK = \alpha$ is sent secretly to the receiver.

4.2 Image Encryption Phase

Given the image $m < K'$, the image owner uses the receiver's public key PK to compute the ciphertext from Equation (1):

$$c_1 = e_{PK}(m) = (g^m r_1^N) mod\, N^2,$$

where $r_1 \in \mathbb{Z}_N^*$ is chosen randomly.

Then, the image owner sends the ciphertext, c_1, with the receiver's public key, PK, to the data processor.

4.3 Data Encryption phase

Given the data $b < K$, the data owner uses the receiver's public key, PK, to compute the ciphertext from Equation (1):

$$c_2 = e_{PK}(b) = (g^b r_2^N) mod\, N^2,$$

where $r_2 \in \mathbb{Z}_N^*$ is chosen randomly.

Then, the image owner sends the ciphertext, c_2, with the receiver's public key, PK, to the data processor.

4.4 Data Hiding Phase

To embed the hidden data, $c_2 < K$, into the ciphertext, c_1, which is encrypted by the reciever's public key, PK, the data processor computes:

$$c = (c_1 \cdot c_2^{k'}) mod\, N^2,$$

and sends the cipher, c, to the receiver.

4.5 Decryption and Extraction Phase

Given the cipher, c, the receiver first decrypts c with the secret key, SK, to get the marked message from Equation (2) as

$$m' = d_{SK}(c) = \frac{L(C^\alpha mod\, N^2)}{L(g^\alpha mod\, N^2)} mod\, N. \quad (6)$$

Then the host image can be obtained by:

$$m = m' mod\, K',$$

and the hidden data can be extracted by:

$$b = \lfloor m'/K' \rfloor.$$

Here we illustrate a simple numerical examples. Suppose the public key $PK = (N, K, g) = (15, 3, 16)$ and the secret key $SK = \alpha = 4$. Given the image pixel

$m = 3$, the image owner can encrypt it with a random number $r_1 = 2$ as: $c_1 = 16^3 \times 2^{15} \bmod 15^2 = 53$. Meanwhile, the data owner can encrypt the data $b = 1$ with a random number $r_2 = 4$ to obtain the encrypted data $c_2 = 16^1 \times 3^{15} \bmod 15^2 = 34$. While the data hider receive the encrypted image and the encrypted data, the hiding is done as $c = 53 \times 34^{\lfloor 15/3 \rfloor} \bmod 15^2 = 122$. When the receiver gets the cipher c, the marked message $m' = \frac{(122^4 \bmod 15^2 - 1)/15}{(16^4 \bmod 15^2 - 1)/15} \bmod 15 = 8$ is obtained first, then the host image pixel can be recovered as $m = 8 \bmod 5 = 3$, the hidden data can be extracted as $b = \lfloor 8/5 \rfloor = 1$.

5 A Multi-Receiver Reversible Data Hiding Scheme

In this section, a reversible data hiding scheme is proposed for sharing data among multiple receivers by combining the homomorphism property of Paillier encryption and polynomial interpolation. The aim of this scheme is to distribute both the image and the hidden data into multiple shadows prior to outsourcing them to the database center. This is necessary because the processing center will embed every data shadow into the responding image shadow to conduct a marked encrypted shadow for w receivers so that more than t receivers who collect their decrypted shadows can recover the host image and the plain data.

The proposed second scheme consists of five phases, *i.e.*, the initialization phase, the image partition and encryption phase, the data partition and encryption phase, the data embedding phase, and the decryption and reconstruction phase.

5.1 Initialization Phase

When the payload parameter, K, is input, the trusted dealer chooses an RSA modulus $N > K$, such that $K' = \lfloor N/K \rfloor > 255$. Then, the dealer selects a prime, $P > max(w, K, K')$ and chooses an element, $g \in \mathbb{Z}^*_{N^2}$, that has an order divisible by αN for some $1 \leq \alpha \leq \lambda(N)$. Choose w non-zero elements, $x_1, x_2, \cdots, x_w \in \mathbb{Z}_P$, randomly, and then x_i is distributed to the receiver R_i as her or his index.

The public key, $PK = (w, N, K, g)$, is broadcasted, while the secret key $SK = \alpha$ is sent secretly to the receivers.

5.2 Image Partition and Encryption Phase

1) Given the image $m < K'$, the image owner randomly selects $t - 1$, independent coefficients, $a_{11}, a_{12}, \cdots, a_{1,t-1} \in \mathbb{Z}_P$, that define the random polynomial over $\mathbb{Z}_P$,

$$f_1(x) = \left(m + \sum_{j=1}^{t-1} a_{1j} x^j\right) mod\, P.$$

2) The image owner computes $s_{1i} = f_1(x_i) mod\, P, 1 \leq i \leq w$, and securely transfers the shadow, s_{1i}, to receiver R_i.

3) The image owner uses the receiver's public key, PK, to compute the cipher shadows by the Paillier cryptosystem,

$$c_{1i} = e_{PK}(s_{1i}),\ i = 1, 2, \cdots, w.$$

Then, the image owner sends the cipher shadow sequence, $(c_{11}, c_{12}, \cdots, c_{1w})$, to the data processor.

5.3 Data Partition and Encryption Phase

1) Given the hidden data $b < K$, the data provider randomly selects $t - 1$ independent coefficients, $a_{21}, a_{22}, \cdots, a_{2,t-1} \in \mathbb{Z}_P$, that define the random polynomial over $\mathbb{Z}_P$,

$$f_2(x) = \left(b + \sum_{j=1}^{t-1} a_{2j} x^j\right) mod\, P. \tag{7}$$

2) The data provider computes $s_{2i} = f_2(x_i) mod\, P, 1 \leq i \leq w$, and securely transfers the shadow s_{2i} to receiver R_i.

3) The receiver uses her or his public key, PK, to compute the cipher shadows by the Paillier cryptosystem,

$$c_{2i} = e_{PK}(s_{2i}), i = 1, 2, \cdots, w. \tag{8}$$

Then, the receiver sends the cipher shadow sequence $(c_{21}, c_{22}, \cdots, c_{2w})$ to the data processor.

5.4 Data Embedding Phase

After obtaining the two cipher shadow sequences, $(c_{11}, c_{12}, \cdots, c_{1w})$ and $(c_{21}, c_{22}, \cdots, c_{2w})$, the data processor computes:

$$c_i = (c_{1i} \cdot (c_{2i})^{K'}) mod\, N^2, i = 1, 2, \cdots, w,$$

and then distributes the marked cipher shadow, c_i, to the receivers R_i, respectively, for $1 \leq i \leq w$.

5.5 Decryption and Reconstruction Phase

Assume that at least t receivers, $R_{i_1}, R_{i_2}, \cdots, R_{i_t}$, pool their shadows and use the receiver's private key, SK, to compute:

$$s'_{i_j} = d_{SK}(c_{i_j}), 1 \leq j \leq t, \tag{9}$$

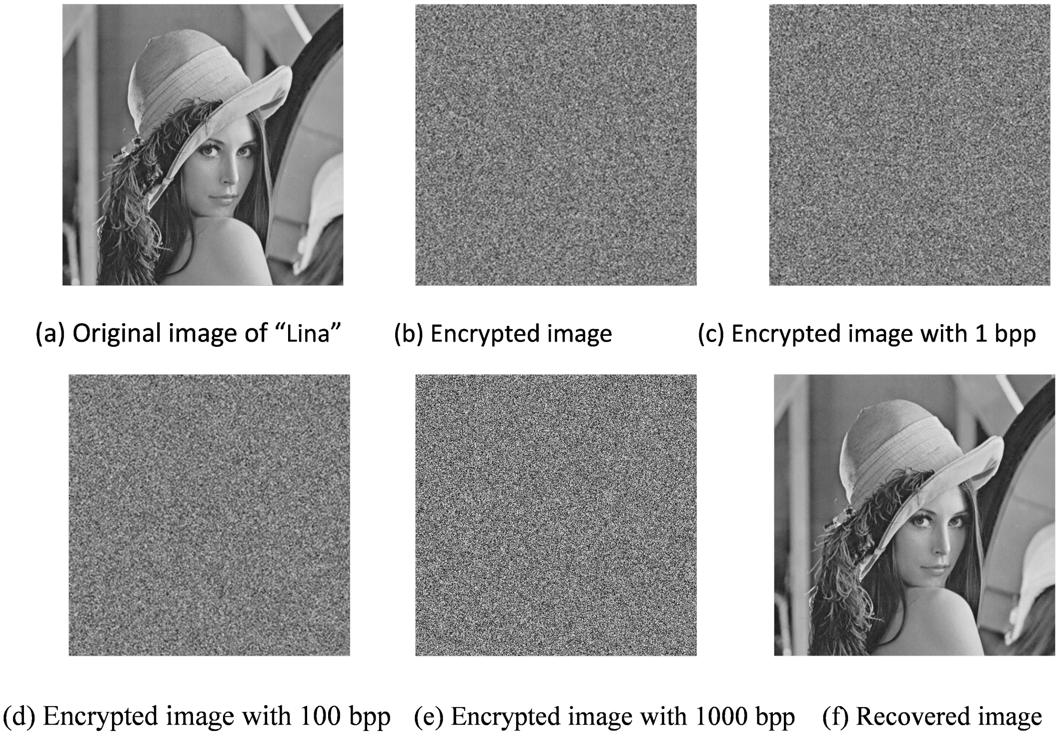
$$m' = \left(\sum_{j=1}^{t} s'_{i_j} \prod_{1 \leq k \leq t, j \neq k} \frac{x_{i_k}}{x_{i_k} - x_{i_j}}\right) mod\, P.$$

Then, the host image can be obtained by:

$$m = m' mod\, K', \tag{10}$$

and the hidden data can be extracted by:

$$b = \lfloor m'/K' \rfloor. \tag{11}$$



(a) Original image of "Lina" (b) Encrypted image (c) Encrypted image with 1 bpp

(d) Encrypted image with 100 bpp (e) Encrypted image with 1000 bpp (f) Recovered image

Figure 2: The original test image Lena, the encrypted image with different hidden data, and the recovered image

6 Performance Analysis

6.1 Verifying Reversibility

The reversibility of the single receiver RDH scheme in Section 4 can be verified easily in a theoretical analysis. Based on the homomorphic properties of the Paillier cryptosystem, which is shown in Equations (4)-(5), we know that Equation (6) is equivalent to $m' = m + K'b \, mod \, N$. Under the condition of $0 \leq m \leq K'-1$ and $0 \leq b \leq K-1$, we have $0 \leq m + K'b \leq K'K - 1 < N$, so that

$$(m + K'b) mod \, N = m + K'b,$$

and then, the original values of m_1 and m_2 can be recovered by Equations (10) and (11), respectively.

In the experimental analysis, we chose the modulus N that was 1024 bits in length, the original "Lena" grayscale, 512×512 image, the encrypted image with no data embedded, the marked encrypted image at different embedding rates (1, 100, and 1000 bpp, respectively), and the perfectly recovered image are shown in Figure 2. The four images in Figures 2(b)-(e) were obtained by performing the arithmetic modulo 256 on the real encrypted images. The same result is also shown in Figure 3 for "Baboon". The test images are came from the USC-SIPI Image Database.

The reversibility of the multi-receiver RDH scheme presented in Section 5 was verified as follows. Assume that t receivers, $R_{i_1}, R_{i_2}, \cdots, R_{i_t}$, honestly pool their shadows. Similar to the analysis above, Equation (9) is equivalent to:

$$s'_{i_j} = (s_{1,i_j} + s_{2,i_j} K') mod \, N = s_{1,i_j} + s_{2,i_j} K', 1 \leq j \leq t,$$

and using Lagrange interpolation, we have:

$$\begin{aligned} m' &= \left(\sum_{j=1}^{t} s'_{i_j} \prod_{1 \leq k \leq t, j \neq k} \frac{x_{i_k}}{x_{i_k} - x_{i_j}} \right) mod \, P \\ &= \left(\sum_{j=1}^{t} (s_{1,i_j} + s_{2,i_j} K') \prod_{1 \leq k \leq t, j \neq k} \frac{x_{i_k}}{x_{i_k} - x_{i_j}} \right) \\ &= \left(\sum_{j=1}^{t} \left(s_{1,i_j} \prod_{1 \leq k \leq t, j \neq k} \frac{x_{i_k}}{x_{i_k} - x_{i_j}} \right) + K' \right. \\ &\quad \left. \cdot \sum_{j=1}^{t} \left(s_{2,i_j} \prod_{1 \leq k \leq t, j \neq k} \frac{x_{i_k}}{x_{i_k} - x_{i_j}} \right) \right) mod \, P \\ &= (f_1(0) + K' \cdot f_2(0)) mod \, P = m + K'b. \end{aligned}$$

Therefore, Equations (7) and (8) hold, *i.e.*, the host image can be recovered exactly and the hidden data can be extracted correctly.

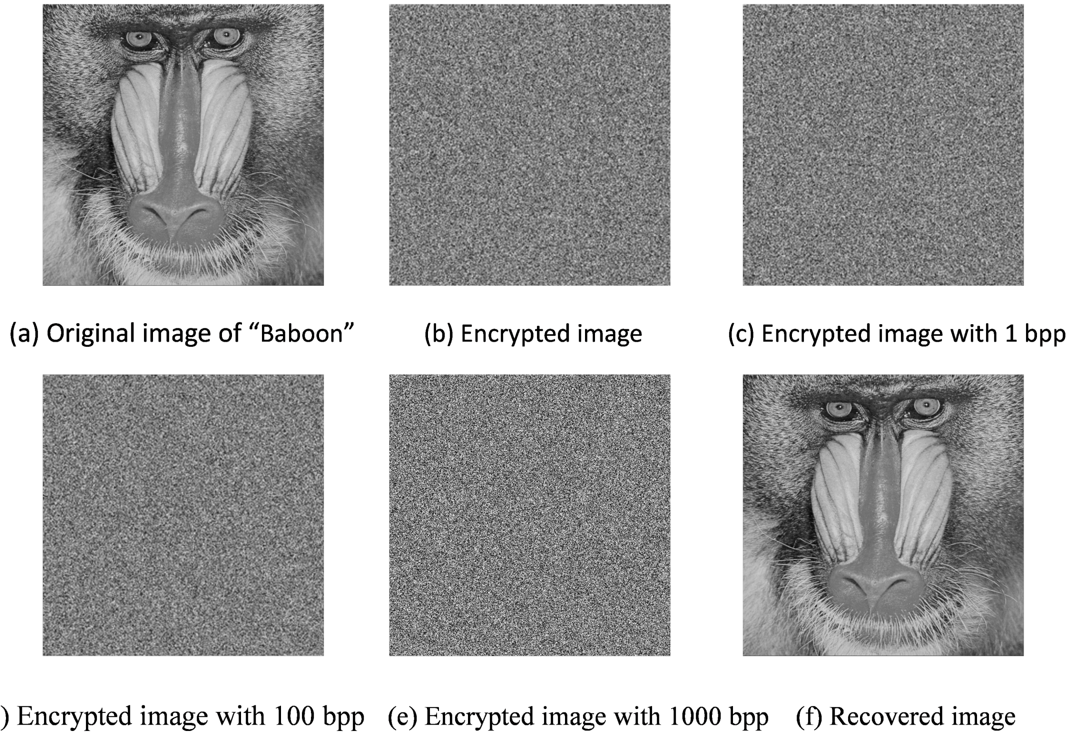
Figure 4 shows the illustration of the multi-receiver RDH scheme with (2,3)-secret sharing. Here we illustrate a small example for (3,5)-secret sharing reconstruction. Suppose that $P = 17$, $t = 3$, $w = 5$; and the i -th receiver's public index is $x_i = i$, for $1 \leq i \leq 5$. Suppose that three shares $(1, 8)$, $(3, 10)$ and $(5, 11)$ are pooled. Writing the polynomial $f(x)$ as $f(x) = a_0 + a_1 x + a_2 x^2$, then we have three linear equations in $\mathbb{Z}_{17}$:

$$\begin{cases} a_0 + a_1 + a_2 = 8 \\ a_0 + 3a_1 + 9a_2 = 10 \\ a_0 + 5a_1 + 8a_2 = 11 \end{cases}$$

This system has a unique solution in $\mathbb{Z}_{17}$: $a_0 = 13$, $a_1 = 10$ and $a_2 = 2$. Therefore the secret key is $f(0) = a_0 = 13$.

6.2 Embedding Capacity

The embedding capacity depends on the payload parameter, K, and up to $\lfloor log_2 K \rfloor$ bits can be hidden per pixel



(a) Original image of "Baboon"

(b) Encrypted image

(c) Encrypted image with 1 bpp

(d) Encrypted image with 100 bpp

(e) Encrypted image with 1000 bpp

(f) Recovered image

Figure 3: The original test image Baboon, the encrypted image with different hidden data, and the recovered image

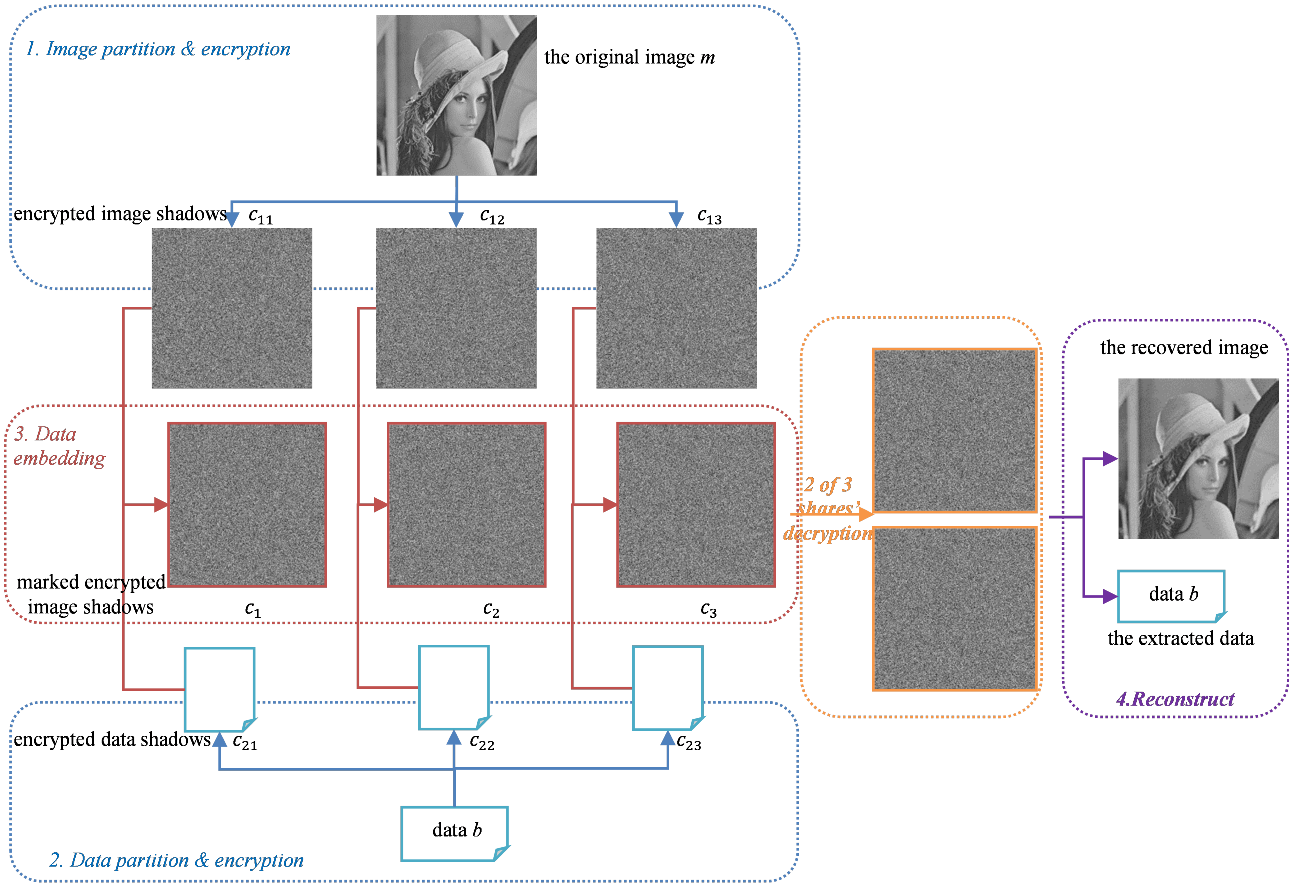


Figure 4: The illustration of the multi-receiver RDH scheme with (2, 3) - secret sharing

Table 2: Performance comparison

Method	Cryptosystem	Embedding capacity(bpp)	preprocess before encryption	Separability of data encryption & embedding
Ma *et al.* [16]	Stream cipher	0.5	yes	no
Zhang *et al.* [31]	Paillier encryption	< 1	no	no
Wu *et al.* [27] ($\|N\| = 1024$)	Paillier encryption	1016	no	no
Li *et al.* [14] ($\|N\| = 1024$)	Paillier encryption	1016	yes	no
Singh *et al.* [26]	secret sharing	≤ 2	no	no
Proposed Scheme ($\|N\| = 1024$)	Paillier encryption	$\|K\| \in [1, 1016]$	no	yes

and perfectly extracted with the appropriate modulus. As a grey-level pixel value from 0 to 255 can be represented with 8 bits, when a big modulus, N, with the bit length of 1024 was used, one pixel can embed up to 1016 bits, even when the length of N is only 9 bits, and 1 bit per pixel can be embedded and correctly extracted. The size of the modulus is related to the scale of the value expansion and the security of the scheme, so parameters can be chosen adaptively by the trade off between efficiency and security. The performance of the proposed algorithm was compared with those of several other algorithms [14, 16, 26, 27, 31], as shown in Table 2. When an 8-bit pixel value was encrypted into a 2048-bit big integer for N with 1024 bits in the Paillier cryptosystem, the embedding capacities of the proposed algorithms is much higher than those in [16,26,31]; although [27] and [14] attained the same capacity, the former must conduct data hiding 1016 iteratively, and the latter must perform extra processing of the images before encryption.

7 Conclusions

In this paper, we proposed two RDH schemes with large embedding capacity, *i.e.*,

1) One that is suitable for a single receiver;

2) One for multiple receivers.

These two schemes have the following common features:

1) Compared to the traditional scheme, we do the role separation between the data provider and data-hider for more application scenarios, and the hidden data also are transmitted in encrypted form;

2) The high embedding rate can be achieved adaptively according to requirements, ranging from 1 bpp to even more than 1016 bpp, which is irrelevant to the pixel distribution of the test image;

3) The schemes do not require an extra processing step before encryption;

4) The embedding rate is independent of the pixel distribution of different natural images;

5) Both the encryption key and the data hiding key are the receiver's public key, and the extraction of the data is done after decrypting the marked encrypted image with the corresponding private key.

In addition, the multi-receiver RDH scheme distributes trust among several receivers, the marked, encrypted image is shared among w receivers, and the host image and hidden data cannot be extracted unless t or more receivers cooperate. It was assumed that all of the receivers have the same private key, and this inflexibility may be improved in the future work by considering multi-secret sharing or proxy encryption.

Acknowledgments

This work was partly supported by the Natural Science Foundation of Fujian Province, China (Grant No. 2018J01537), the Education and Scientific Research Project for Young Middle-aged Teachers of Fujian Province, China (Grant No. JAT190314), and the Science and Technology project of Xiamen Municipal (Grant No. 3502Z20173028). The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Abd-Eldayem and M. Mohamed, "A proposed security technique based on watermarking and encryption for digital imaging and communications in medicine," *Egyptian Informatics Journal*, vol. 14, pp. 1–13, Mar. 2013.

[2] A. Asmuth and J. Bloom, "A modular approach to key safeguarding," *IEEE Transactions on Information Theory*, vol. 30, pp. 208–210, Mar. 1983.

[3] J. C. Benaloh, "Secret sharing homomorphisms: Keeping shares of a secret," in *Advances in Cryptology (CRYPTO'86)*, pp. 251–260, Aug. 1986.

[4] K. Bharanitharan and C. C. Chang, H. R. Yang, and Z. H. Wang, "Efficient pixel prediction algorithm for reversible data hiding," *International Journal of Network Security*, vol. 18, pp. 750–757, July 2016.

[5] G. R. Blakley, "Safeguarding cryptographic keys," in *Proceedings of American Federation of Information Processing Societies National Computer Conference (AFIPS'79)*, pp. 313–317, June 1979.

[6] X. Cao, X. Wei L. Du, D. Meng, and X. Guo, "High capacity reversible data hiding in encrypted images by patch-level sparse representation," *IEEE Transactions on Cybernetics*, vol. 46, pp. 1132–1143, Mar. 2016.

[7] C. C. Chang, Y. H. Huang, and T. C. Lu, "A difference expansion based reversible information hiding scheme with high stego image visual quality," *Multimedia Tools and Applications*, vol. 76, pp. 12659–12681, May 2017.

[8] C. C. Chang, C. C. Lin, C. H. Lin, and Y. H. Chen, "A novel secret image sharing scheme in color images using small shadow images," *Information Sciences*, vol. 178, pp. 2433–2447, June 2008.

[9] Y. Chen, C. 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.

[10] M. Chen X. Zeng G. Biao, Z. Chen and Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Transactions on Information Forensics and Security*, vol. 5, pp. 187–193, Mar. 2010.

[11] S. F. Chiou, Y. C. Lu, I-En Liao, and M. S. Hwang, "An efficient reversible data hiding scheme based on SMVQ," *Imaging Science Journal*, vol. 61, no. 6, pp. 467–474, 2013.

[12] W. Hong, T. Chen, J. Chen, Y. Kao, H. Wu, and M. Wu, "Reversible data embedment for encrypted cartoon images using unbalanced bit flipping," in *Proceedings of the 4th International Conference in Swarm Intelligence*, pp. 208–214, June 2013.

[13] M. Khosravi and M. Yazdi, "A lossless data hiding scheme for medical images using a hybrid solution based on ibrw error histogram computation and quartered interpolation with greedy weights," *Neural Computing and Applications*, vol. 30, pp. 2017–2028, Oct. 2018.

[14] M. Li and Y. Li, "Histogram shifting in encrypted images with public key cryptosystem for reversible data hiding," *Signal Processing*, vol. 130, pp. 190–196, Nov. 2017.

[15] L. Liu, C. C. Chang, and A. Wang, "Reversible data hiding scheme based on histogram shifting of n-bit planes," *Multimedia Tools and Applications*, vol. 75, pp. 11311–11326, Sep. 2016.

[16] 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 Security*, vol. 8, pp. 553–562, Mar. 2013.

[17] R. J. McEliece and D.V. Sarwate, "On sharing secrets and reed-solomon codes," *Communications of the ACM*, vol. 24, pp. 583–584, Sep. 1981.

[18] M. Mignotte, "How to share a secret," in *Proceedings of the Workshop on Cryptography*, pp. 371–375, Mar. 1982.

[19] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology*, pp. 223–238, May 1999.

[20] W. Puech, M. Chaumont, and O. Strauss, "A reversible data hiding method for encrypted images," in *Proceedings of SPIE*, vol. 6819, pp. 1–9, Mar. 2008.

[21] C. Qin and Y. C. Hu, "Reversible data hiding in VQ index table with lossless coding and adaptive switching mechanism," *Signal Processing*, vol. 129, pp. 48–55, Dec. 2016.

[22] C. Qin and X. Zhang, "Effective reversible data hiding in encrypted image with privacy protection for image content," *Journal of Visual Communication and Image Representation*, vol. 31, pp. 154–164, Aug. 2015.

[23] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," *Foundations of Secure Computation*, vol. 4, no. 11, pp. 169–179, 1978.

[24] H. Rostami, M. Khosravi and S. Samadi, *Enhancing the Binary Watermark-Based Data Hiding Scheme Using an Interpolation-Based Approach for Optical Remote Sensing Images*, 2019. DOI: 10.4018/978-1-5225-7033-2.ch014.

[25] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, pp. 612–613, Nov. 1979.

[26] P. Singh and B. Raman, "Reversible data hiding based on shamir's secret sharing for color images over cloud," *Information Sciences*, vol. 422, pp. 77–97, Jan. 2018.

[27] H. T. Wu, Y. M. Cheung, and J. W. Huang, "Reversible data hiding in paillier cryptosystem," *Journal of Visual Communication and Image Representation*, vol. 40, pp. 765–771, Oct. 2016.

[28] X. Wu and W. Sun, "High-capacity reversible data hiding in encrypted images by prediction error," *Signal Processing*, vol. 104, pp. 387–400, Nov. 2014.

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

[30] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, pp. 255–258, Apr. 2011.

[31] X. Zhang, J. Wang, Z. Wang, and H. Cheng, "Lossless and reversible data hiding in encrypted images with public key cryptography," *IEEE Transactions on Circuits Systems Video Technology*, vol. 26, pp. 1622–1631, Sep. 2016.

[32] 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 Systems Video Technology*, vol. 26, pp. 441–452, Mar. 2016.

Biography

Hefeng Chen received the B.S. and M.S. degrees in mathematics from Xiamen University, China, in 2005 and 2008, respectively, and the Ph.D. degree in cryptography from Xidian University, Xi'an, China, in 2016. She was a visiting scholar with Feng Chia University, Taiwan, in 2018. She is currently an assistant professor with the Computer Engineering College, Jimei University, Xiamen, China. Her current research interests include cryptography and information hiding.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. He served in National Chung Cheng University from 1989 to 2005. Since February 2005, he has been a Chair Professor with Feng Chia University. He is a Fellow of IEEE and a Fellow of IEE, UK. His current research interests include computer cryptography, image compression, and data structures.

Kaimeng Chen received the B.Sc. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2010 and 2016, respectively. He was a visiting scholar with Feng Chia University, Taiwan, in 2018. He is currently an assistant professor with the Computer Engineering College, Jimei University, Xiamen, China. He is a principle investigator of a project of the National Science Foundation of Fujian Province, China. His research interests include data hiding, image processing, databases on new hardware, hybrid storage, and non-volatile memory technology.