mathematics

MDPI

Article

A Novel CDMA-Based Secret Message Multiplexing Scheme by Image Data Hiding †

Chih-Hung Lee [1], Kun-Sheng Sun [1], Ji-Hwei Horng [1,*] and Chin-Chen Chang [2,*]

[1] Department of Electronic Engineering, National Quemoy University, Kinmen 89250, Taiwan
[2] Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan
* Correspondence: horng@email.nqu.edu.tw (J.-H.H.); ccc@o365.fcu.edu.tw (C.-C.C.)
† This paper is an extended version of our paper published in 2022 3rd Asia Service Sciences and Software Engineering Conference (ASSE' 22), Macau, 24–26 February 2022; pp. 52–59.

Abstract: In this paper, based on the concept of Code Division Multiple Access (CDMA), we propose an innovative secret message multiplexing scheme by digital image data hiding. Multiple secret messages distributed for several participants are encoded into a CDMA codeword through CDMA multiplexing and then embedded into a grayscale image. At the receiving side, each intended recipient will be assigned a distinct spreading code as the secret key for data extraction. In this way, even the authorized user can only extract his or her own secret message using an exclusive secret key and has no knowledge about other user's secret. This characteristic helps to ensure data security. Experimental results demonstrate that the scheme is also resistant to Gaussian noise attack, salt-and-pepper noise attack, and cropping attacks. Those merits sustain the feasibility of the proposed scheme.

Keywords: Code Division Multiple Access (CDMA); CDMA multiplexing; secret multiplexing; secret sharing; data hiding

MSC: 68U10

check for updates

Citation: Lee, C.-H.; Sun, K.-S.; Horng, J.-H.; Chang, C.-C. A Novel CDMA-Based Secret Message Multiplexing Scheme by Image Data Hiding. *Mathematics* **2023**, *11*, 704. https://doi.org/10.3390/math11030704

Academic Editor: Jakub Nalepa

Received: 29 December 2022
Revised: 27 January 2023
Accepted: 28 January 2023
Published: 30 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

With the rapid development of information and communication technologies (ICT), smartphones have been an indispensable part of our daily life. Large amounts of data are transmitted on the Internet every day. Information security is becoming more and more important. In recent years, a lot of image-steganographic methods, also known as data hiding techniques, have been proposed to hide the secret data in the cover images. These techniques can be categorized into three major classes, which are the original domain hiding [1–10], the compressed space domain hiding [11–14], and the transform domain hiding [15–18]. By embedding the secret data into a digital image with slight modification and transmitting the image on the Internet, the security of the secret data is improved. Nevertheless, among these techniques, there exists a common weakness. The secret data are stored in a single image. Being the only information carrier, once the image is lost or damaged, the secret data cannot be recovered correctly. To solve this problem, the secret sharing method is a feasible approach.

The concept of secret sharing was proposed separately by Shamir [19] and Blakley [20] in 1979. It was called the (k, n) threshold scheme. A secret is split into n pieces ("shadows" or "shares") and those pieces can be distributed to n participants at different locations. The secret can only be reconstructed from k or more pieces ($k \leq n$). In 1995, Naor and Shamir [21] extended the secret sharing technique to the image field and proposed the visual cryptography technique, which was also called visual secret sharing (VSS). A secret image is divided into n shadows or shares to be held by n participants. The secret image

can be revealed by stacking k or more shares. VSS schemes utilize the human visual system in the reconstruction of the secret image but suffer from pixel expansion problem. Thereafter, lots of novel secret image sharing (SIS) techniques were proposed to improve the VSS performance, such as the EMD-based scheme [22], Sudoku matrix scheme [23], turtle shell matrix scheme [24], maze matrix scheme [25], fractal matrix scheme [26], and so forth [27–31]. Those schemes utilize mathematical computation and well-designed reference matrices to construct meaningful shares and recover the secret image.

In contrast to the secret sharing, we previously proposed an innovative secret message multiplexing scheme applied to a QR code [32], in which the secret extraction can be achieved by each participant independently without cooperating with others. This scheme is inspired by the concept of Code Division Multiple Access (CDMA) and enables a secret distributor delivering multiple secret messages to several participants simultaneously via a single QR Code. All the secret messages for different users are encoded into a CDMA codeword and embedded in the padding region of the cover QR code. By this way, everyone can read the public message of a marked QR code by any standard QR code reader, but only the authorized participant can extract the dedicated secret message using his or her specific secret key and has no knowledge about others' secret. Since the embedding capacity of the QR code is limited, this scheme is only feasible for small amount of secret message delivering.

In this paper, we will implement the concept of secret multiplexing in the digital image and propose a novel CDMA-based secret message multiplexing scheme by image data hiding. In this scheme, we still consider a scenario where there is a secret distributor and several participants. The secret distributor is going to deliver multiple secret messages to all participants simultaneously via a single digital image. The secret messages for all users are encoded into a compound CDMA codeword and embedded in the grayscale image by modulo operation. At the receiving side, each participant will be assigned an exclusive spreading code. This spreading code serves as the secret key to extract secret data. The cross-correlation between any two different secret keys is equal to zero. Due to this orthogonal property, only the authorized recipient can decode the respective hidden secret message using a specific secret key. Therefore, data security is well ensured. At the same time, with the help of spreading codes, a secret data bit is spread to several bits. These characteristics make the proposed scheme robust against some kinds of noise attacks such as Gaussian noise attack and salt-and-pepper noise attack.

The novelty and contributions of this work mainly include the following aspects. (1) We introduce a novel CDMA-based secret message multiplexing scheme applicable to digital images, in which multiple secret messages for different participants are embedded into a single image by CDMA multiplexing and modulo operation. (2) The theoretical PSNR of the proposed scheme is analysed and compared with experimental results. It is verified that the PSNR performance is independent of the characteristics of the cover image. (3) We apply the Gaussian noise attack, salt-and-pepper noise attack, and cropping attacks to the marked images. The bit-error rate (BER) and logotype secret images are utilized to evaluate the robustness of the proposed scheme. We also provide theoretical BER for the Gaussian noise attack.

The rest of this paper is organized as follows. The technology of CDMA multiplexing is introduced in Section 2. The proposed CDMA-based secret multiplexing scheme is presented in Section 3. Section 4 demonstrates the experimental results. Finally, conclusions are made in Section 5.

2. CDMA Multiplexing

Code Division Multiple Access (CDMA) [33–38] is a channel access method employing spread spectrum technology. Being a multiplexing scheme in essence, CDMA is also known as Code Division Multiplexing (CDM). Multiplexing is the sharing of a communication channel or bandwidth. It is the process in which multiple signals coming from multiple sources are combined and transmitted over a shared communication link. The receiver

recovers the separate signals by a process called demultiplexing. Compared to CDM, the term CDMA is mentioned more often to emphasize the asynchronous and random nature of multiuser connections.

The spread spectrum technique has been developed for a long time. It was formerly used in military communications to achieve anti-jamming. In spread spectrum communication, the bandwidth occupied by a single transmitted signal is much higher than in systems using conventional modulation methods. This bandwidth spread is accomplished by means of a spreading code, which is independent of the data. A very popular approach is to multiply the data signal with a fast-spreading signal, which is also named the spreading code. This technique is known as Direct-Sequence (DS) Spread Spectrum. Figure 1a displays the waveforms of binary data signal, spreading signal, and the transmitted signals in a Direct-Sequence Spread Spectrum (DSSS) system. As seen in Figure 1a, the binary antipodal spreading signal comprises eight chips with values $1, 1, 1, 1, -1, 1, -1, -1$ in sequence. The chip rate is 8 times that of the bit rate, which means the bandwidth of the transmitted signals will be 8 times larger than that of the data signal.

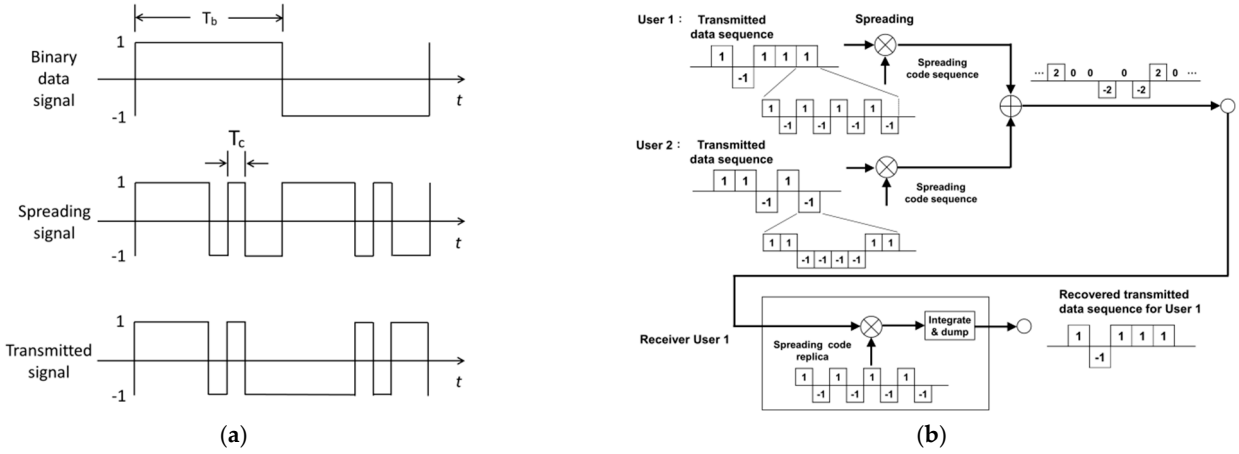


Figure 1. Diagrammatic illustration of the Spread Spectrum and CDMA techniques. (**a**) Signal waveforms in a Direct-Sequence Spread Spectrum (DSSS) system. (**b**) Block diagram of a DS-CDMA communication system.

In the case that multiple users share the same radio spectrum but use different codes to discriminate respective transmitted signals, we called it Code Division Multiple Access (CDMA). At present, Direct-Sequence CDMA (DS-CDMA) has been widely applied to mobile communications. In the following, we will focus on DS-CDMA technology. The spreading codes for CDMA could be classified into two categories, orthogonal codes and pseudorandom (PN) codes. For PN codes, such as Maximum Length sequences and Gold codes, there is a low correlation between the codes, while orthogonal codes, such as Hadamard codes, have a cross-correlation equal to zero.

Here, an illustrative example of DS-CDMA communication system is shown in Figure 1b where the 8-bit Hadamard codes are used as spreading codes and two users are displayed. In the transmitter, the binary data signal of each user is multiplied by a specific spreading code, and they are superimposed together to form the DS-CDMA signal. In the receiver, the received signal is again multiplied by a spreading code and integrated over one data bit interval. Note that this operation is equivalent to taking the inner product. A CDMA receiver can retrieve the original data signal only by multiplying the received signal with the same code as the one used during transmission.

A more detailed example is demonstrated. Suppose the number of concurrent users in the system is four, and the four users' data are $A = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$, $B = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$, $C = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$, $D = \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}$, each with three bits. Every user will be assigned a four-bit Hadamard code as the spreading code.

In the transmitter side, the steps for CDMA multiplexing are listed below.

Step 1. Process the bit value conversion with $1 \rightarrow 1$ and $0 \rightarrow -1$ on users' data. The resultant antipodal bits are $A' = \begin{bmatrix} 1 & -1 & 1 \end{bmatrix}$, $B' = \begin{bmatrix} 1 & 1 & -1 \end{bmatrix}$, $C' = \begin{bmatrix} -1 & -1 & 1 \end{bmatrix}$, and $D' = \begin{bmatrix} -1 & 1 & 1 \end{bmatrix}$.

Step 2. Generate Hadamard codes using Hadamard matrix.

$$H_1 = [1], H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, H_{2^{k+1}} = \begin{bmatrix} H_{2^k} & H_{2^k} \\ H_{2^k} & -H_{2^k} \end{bmatrix}$$

The 4×4 Hadamard matrix H_4 is as follows.

$$H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

Each row represents a distinct spreading code. Those four mutually orthogonal spreading codes are assigned to four users in order.

$$C_A = [1\,1\,1\,1],\ C_B = [1\ -1\,1\ -1],\ C_C = [1\,1\ -1\ -1],\ C_D = [1\ -1\ -1\,1]$$

Step 3. For each user, perform the multiplication between the user data and the spreading code bit by bit.

$$\mathrm{A_C} = [1\,1\,1\,1,\ -1\ -1\ -1\ -1,\ 1\,1\,1\,1],\ \mathrm{B_C} = [1\ -1\,1\ -1,\ 1\ -1\,1\ -1,\ -1\,1\ -1\,1],$$
$$\mathrm{C_C} = [-1\ -1\,1\,1,\ -1\ -1\,1\,1,\ 1\,1\ -1\ -1],\ \mathrm{D_C} = [-1\,1\,1\ -1,\ 1\ -1\ -1\,1,\ 1\ -1\ -1\,1]$$

Step 4. Add the spread data together to form the CDMA signal.

$$M = A_c + B_c + C_c + D_c = [0\,0\,4\,0,\ 0\ -4\,0\,0,\ 2\,2\ -2\,2]$$

At the receiving side, the steps for CDMA demultiplexing are listed below.

Step 1. Divide the CDMA signal into three groups. Each group is composed of four digits. Perform inner product operation of CDMA codeword groups and spreading codes C_A, C_B, C_C, and C_D individually. The results are $[4,\ -4,\ 4]$, $[4,\ 4,\ -4]$, $[-4,\ -4,\ 4]$, and $[-4,\ 4,\ 4]$.

Step 2. Make decisions with the threshold set to zero. Accordingly, $[4,\ -4,\ 4]$, $[4,\ 4,\ -4]$, $[-4,\ -4,\ 4]$, and $[-4,\ 4,\ 4]$ are decoded as $\begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$, $\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$, $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$, and $\begin{bmatrix} 0 & 1 & 1 \end{bmatrix}$, respectively. It is evident that all users' data are retrieved successfully.

3. The Proposed Secret Message Multiplexing Scheme

In this section, we will introduce the proposed CDMA-based secret multiplexing scheme. The scheme is split into the secret embedding procedure and the extraction procedure. Besides, the analysis of embedding capacity and theoretical PSNR are also presented in this section.

3.1. Secret Embedding Procedure

In this subsection, we design a secret multiplexing scheme where there is a secret distributor to deliver individual secret messages to N participants via a digital image. Each participant can only extract his or her own secret message using a correct secret key. For illustrative purposes, we used the Hadamard matrix of order N to generate secret keys. Hadamard codes have excellent cross-correlation properties but the structure is regular and well-known. To reduce the risk of eavesdropping in realistic applications, besides Hadamard codes, other types of PN codes, such as Maximum Length sequences and Gold codes, could also be used as the secret keys [34].

Maximum Length sequences, also known as m-sequences, can be generated using linear generator polynomials of degree m as given by

$$g(x) = g_m x^m + g_{m-1} x^{m-1} + g_{m-2} x^{m-2} + \cdots + g_1 x + g_0, \tag{1}$$

where $g(x)$ is a primitive polynomial and the m-sequence generated has a period of $n = 2^m - 1$. For instance, $x^5 + x^3 + 1$ is one of six usable primitive polynomials to generate m-sequences of length $n = 2^5 - 1 = 31$.

The periodic autocorrelation function of m-sequences is

$$R_{kk}(j) = \begin{cases} n, j = 0, \\ -1, 1 \leq j \leq n-1, \end{cases} \tag{2}$$

where j denotes the number of cyclic shifts of the m-sequence.

As m increases, the number of m-sequences of length $n = 2^m - 1$ is enlarged. The lowest achievable cross-correlation between two different m-sequences is -1. Moreover, due to the excellent autocorrelation properties, different offsets of the same m-sequence can also be used by different participants. The combinations of different primitive polynomials and cyclic shifts could provide a large number of secret keys.

A set of Gold codes is constructed from a preferred pair of m-sequences, $\mathbf{a} = [a_1 a_2 \ldots a_n]$ and $\mathbf{b} = [b_1 b_2 \ldots b_n]$, by taking the modulo$-2$ sum of $\mathbf{a}$ with the n cyclically shifted versions of $\mathbf{b}$ and vice versa. Thus, including the original sequences $\mathbf{a}$ and $\mathbf{b}$, a total of $n + 2$ Gold codes with period $n = 2^m - 1$ and a cross-correlation as low as -1 is generated.

Regardless of m-sequences, Gold codes, or Hadamard codes, when the number of codes with a specified length is large enough, a subset of available secret keys could be chosen to accommodate all participants. In this case, data security is further improved.

To fit with the characteristics of the Hadamard matrix, N is assumed to be power-of-2. The flowchart of secret embedding procedure is shown in Figure 2. Suppose the N individual secret messages $S_1, S_2, \ldots S_N$ are all L-bit long. After converting to an antipodal signal and multiplied by the secret key, every secret message will be spread to $L \times N$-bit long. Then, all the N sets of spread bits are added together to form the CDMA codeword with $L \times N$ digits. Note that because of signal superimposition, the amplitude of a CDMA codeword is enlarged. Obviously, the value of a codeword digit lies on the range $[-N, +N]$, which can be represented as a number in the $(2N + 1)$-ary notational system. Let the original image I be a grayscale image with size $H \times W$. The image is rearranged in the raster scan order into a one-dimensional sequence $\{x_n\}$ of length $H \times W$. Here, we assume that $H \times W$ is divisible by N and $L \times N$ equals $H \times W$. In this case, the embedding capacity will reach the maximum value. Note that when N is not power-of-2 or $H \times W$ cannot be divided by N, the scheme is still applicable only with some capacity reduction.

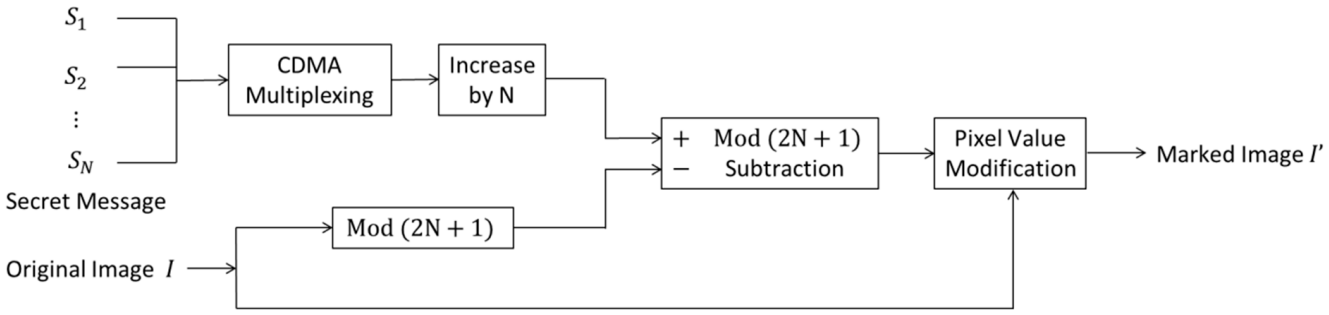


Figure 2. A flowchart of the secret embedding procedure.

To embed a $(2N+1)$-ary number into a grayscale pixel, we define a CDMA codeword mapping function f for the pixel value with modulo operation.

$$f(x_i) = x_i \bmod (2N+1) \tag{3}$$

where x_i denotes the i-th term of the pixel sequence and the converted result $f_i \equiv f(x_i)$ is a $(2N+1)$-ary number.

To facilitate embedding, N is added to each CDMA codeword digit such that the value range is shifted to $[0,\ 2N]$. Next, we compare the i-th digit of the modified CDMA codeword, sc_i $(1 \le i \le L \times N)$, with f_i. No modification is needed if sc_i is equal to f_i. When $sc_i \neq f_i$, we calculate the difference value

$$d_i = sc_i - f_i \bmod (2N+1) \tag{4}$$

If d_i is not larger than N, we increase the value of x_i by d_i ; otherwise, decrease the value of x_i by $(2N+1-d_i)$.

After all pixels are modified, the one-dimensional pixel sequence $\{x'_n\}$ is restored to the marked image I' with size $H \times W$ and the embedding process is complete.

Note that after data embedding, it is possible that the grayscale values of some pixels in the marked image exceed the upper bound (255 for an 8-bit grayscale image) or the lower bound (0 for an 8-bit grayscale image). This is caused by the addition or subtraction operations performing on pixel values close to 255 or 0. To avoid overflow or underflow from occurring, a straightforward procedure is adopted. When the modified pixel value x'_i is less than zero, $2N+1$ is added to x'_i; when x'_i is larger than 255, $2N+1$ is subtracted from x'_i.

The secret embedding procedure is summarized in the following steps.

Step 1. Perform CDMA multiplexing on the N individual L-bit secret messages S_1, S_2, ... S_N to form a CDMA codeword with $L \times N$ digits. For each digit in the codeword, increase the value by N.

Step 2. Rearrange the original image I into a one-dimensional sequence $\{x_n\}$ of length $H \times W$. Map each pixel value x_i to a $(2N+1)$-ary number f_i accoridng to Equation (3).

Step 3. Calculate the difference value between the CDMA codeword digit sc_i and the modular pixel value f_i according to Equation (4).

Step 4. Compute the marked pixel value x'_i by

$$x'_i = \begin{cases} x_i + d_i, & d_i \le N, \\ x_i - (2N+1-d_i), & \text{otherwise.} \end{cases} \tag{5}$$

when overflow/underflow occurs, do

$$x'_i = \begin{cases} x'_i + (2N+1), & x'_i < 0, \\ x'_i - (2N+1), & x'_i > 255. \end{cases} \tag{6}$$

Step 5. Convert the one-dimensional marked pixel sequence $\{x'_n\}$ to the marked image I' with size $H \times W$.

Here, we take the four-user example described in the previous section to further explain our proposed secret multiplexing scheme. The whole CDMA codeword contains 12 digits. For simplicity, just take the first four digits [0 0 4 0], for data embedding. The corresponding original secret messages for four users are 1, 1, 0, and 0, respectively. First, $N = 4$ is added to [0 0 4 0]. The resulting CDMA codeword vector is denoted as $S_c = [4\ 4\ 8\ 4]$. Then, we select a group of four consecutive pixels from the 512×512 grayscale image Lena to form the vector $X = [162\ 157\ 163\ 161]$. The mapping function f converts vector X to a 9-ary vector $X_f = [0\ 4\ 1\ 8]$. Next, calculate the difference between S_c and X_f. The result is $D = [4\ 0\ 7\ 5]$. It means that among the four digits of X, only the second digit need

not be modified, while the first digit should be increased by 4 and the third and fourth digits would be decreased by 2 and 4, respectively. Therefore, the modified pixel vector is $X' = [166\ 157\ 161\ 157]$. At this instant, the four secret message bits $(1,\ 1,\ 0,\ 0)$, each belonging to one individual user, have been embedded into the pixel vector through the proposed CDMA multiplexing scheme.

3.2. Secret Extraction Procedure

At the receiving side, the secret message can be easily extracted in a manner similar to the secret embedding procedure. The flowchart of secret extraction procedure is shown in Figure 3. By raster scanning the marked image I', we get the one-dimensional pixel sequence $\{x'_n\}$. For each pixel, perform the modulo operation according to Equation (3). The results correspond to the $(2N+1)$-ary CDMA codeword. After being decreased by N, the codeword is divided into L groups with each group composed of N digits. Thereafter, the CDMA demultiplexing process is performed on every group. The process mainly involves the inner product operation of a codeword group and a secret key. Based on a threshold value of zero, the value of the inner product is converted to a binary form of secret message. It is noteworthy that this CDMA demultiplexing is performed by each participant separately. Since the cross-correlation between any two secret keys equals zero, only the authorized user can extract the dedicated secret message with a correct secret key. Thus, unauthorized data access is prevented, and data confidentiality is ensured.

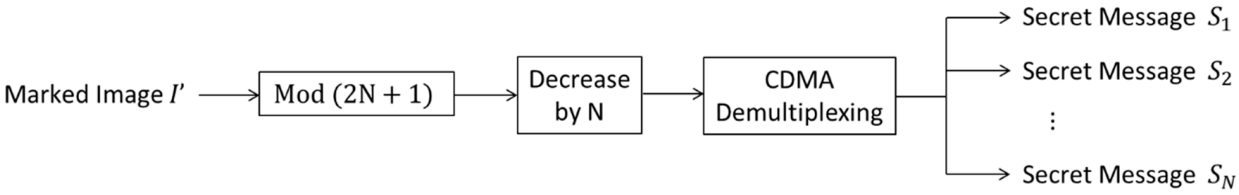


Figure 3. The flowchart of secret extraction procedure.

The secret extraction procedure is summarized in the following steps.

Step 1. Rearrange the marked image I' into a one-dimensional sequence $\{x'_n\}$ of length $H \times W$.

Step 2. Map each pixel value x'_i to a $(2N+1)$-ary number f'_i accoridng to Equation (3) and decrease f'_i by N. The results represent the original CDMA codeword.

Step 3. Divide the CDMA codeword into L groups with each group composed of N digits. Perform CDMA demultiplexing group-wise with a specific secret key to retrieve the individual L-bit secret message S_i $(1 \le i \le N)$.

Take the same example described in the secret embedding procedure subsection to illustrate the process of secret message extraction.

For the pixel vector $X' = [166\ 157\ 161\ 157]$ of the marked image, the modulo result $[4\ 4\ 8\ 4]$ implies that $[0\ 0\ 4\ 0]$ is the original CDMA codeword vector. Recall that the four secret keys are $[1\ 1\ 1\ 1]$, $[1\ -1\ 1\ -1]$, $[1\ 1\ -1\ -1]$, and $[1\ -1\ -1\ 1]$. They are used to calculate the inner product with $[0\ 0\ 4\ 0]$ individually. The results are 4, 4, -4, and -4. We observe that the converted binary forms $(1,\ 1,\ 0,\ 0)$ coincide with the original four secret message bits. In other words, the extraction result is correct.

3.3. Embedding Capacity

In the proposed secret multiplexing scheme, according to the total number of users N, the value of modification for each pixel will lie within the range of $[-N, +N]$. Therefore, a $(2N+1)$-ary CDMA codeword digit can be hidden in a single pixel. This strategy is similar to the well-known exploiting modification direction (EMD) method [39]. In the EMD method, to carry a $(2N+1)$-ary secret digit, a group of N pixels must be used and among those N pixels, at most only one pixel value is increased or decreased by 1. Owing

to the larger pixel value modification, our strategy will lead to greater distortion in the marked images. Nevertheless, the benefit is the increase of the embedding capacity.

On the basis of assumptions that N is power-of-2 and the image size $H \times W$ is divisible by N, the embedding capacity can reach the maximum value by setting the length of secret message bits for all users an identical value L equal to $H \times W/N$. In that case, by summing up all users' capacities, the total embedding capacity is $H \times W$ and the total embedding rate (ER) measured in bits per pixel (bpp) is 1 bpp. Accordingly, for any individual participant, the embedding rate is $1/N$ bpp. That is to say, the total embedding capacity is fixed and shared by all the participants. In general, when N is power-of-2 but the image size $H \times W$ is not divisible by N, there will exist some pixels not used for data embedding. Hence, the total embedding rate will be slightly less than 1 bpp.

3.4. PSNR Analysis

Generally, the pixel values of original images can be regarded as randomly distributed; it is reasonable to suppose that after converted to a digit in the $(2N+1)$-ary notational system by modulo operation, the occurrence of any number within the range of $[0,\ 2N]$ is equally likely. Likewise, to embed a $(2N+1)$-ary CDMA codeword digit into a pixel, the value of modification is uniformly distributed between -N and N. Theoretically, the mean-square-error (MSE) is given by

$$\mathrm{MSE} = \frac{1}{2N+1}\sum_{k=-N}^{k=N} k^2 = \frac{N(N+1)}{3}. \tag{7}$$

The peak-signal-to-noise-ratio (PSNR) for an 8-bit grayscale image is defined by

$$\mathrm{PSNR} = 10\log_{10}\frac{255^2}{\mathrm{MSE}}\ (\mathrm{dB}). \tag{8}$$

By applying the result in Equation (7) to Equation (8), the expected PSNR value of a marked image is given by

$$\mathrm{PSNR} = 10\log_{10}\frac{3\times 255^2}{N(N+1)}\ (\mathrm{dB}). \tag{9}$$

Apparently, the number of users N dominates the mean-square-error and the PSNR value. A larger N means that more users are involved in the the process of CDMA multiplexing and longer spreading codes (or secret keys) are adopted to provide better data confidentiality for participants' secret. But inevitably, it also lowers the PSNR value and reduces the image quality. The expected PSNR values at different values of N are listed in Table 1.

Table 1. The expected PSNR value (dB) versus number of users N.

	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
PSNR (dB)	45.12	39.89	34.33	28.56	22.67

4. Experimental Results and Discussion

In this section, several experiments are carried out to demonstrate the performance of the proposed algorithm. Performance evaluation based on the visual quality of marked images and robustness against noise and cropping attacks are given. Four commonly used grayscale images sized 512×512 are chosen as the test images: Lena, Baboon, Airplane, and Peppers, respectively. These images are shown in Figure 4. All experiments are simulated with MATLAB software on PC Windows operating system.



(**a**) (**b**) (**c**) (**d**)

Figure 4. The four grayscale test images. (**a**) Lena; (**b**) Baboon; (**c**) Airplane; (**d**) Peppers.

4.1. Image Quality

The peak-signal-to-noise-ratio (PSNR) is used to measure the visual quality of marked images. The PSNR definition for an 8-bit grayscale image is given in Equation (8) previously. In ours experiments, the mean-square-error (MSE) of marked image I' with respect to the cover image I is given as

$$\text{MSE} = \frac{1}{H \times W} \sum_{i,j} (I'_{ij} - I_{ij})^2. \tag{10}$$

The PSNR values for the four test images are listed in Table 2. We can see that the PSNR decreases with the increase in number of users. The major reason is that hiding a large number of participants' respective secret messages by CDMA multiplexing leads to greater distortion in the marked images. That is, a higher multiplexing level yields a lower PSNR value. This phenomenon indicates that there is a tradeoff between imperceptibility and multiplexity.

Table 2. PSNR values (dB) versus number of users N for four test images.

Images	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
Lena	45.13	39.91	34.38	28.65	22.66
Airplane	45.14	39.95	34.26	28.86	22.16
Baboon	45.11	39.90	34.32	28.66	22.37
Peppers	45.13	39.91	34.30	28.30	22.25

We also observe that the results closely coincide with the expected PSNR values in Table 1, which means that the PSNR performance is independent of the characteristics of the original cover image. Take Lena image for example. Figure 5 shows the original Lena image and the marked images with different values of N. As shown in the figures, with the number of users N less than 16, the PSNR is kept above 34 dB and the visual quality of the marked image is quite good. Even when N reaches 16, the visual distortion is minor and the PSNR is still higher than 28 dB.

4.2. Robustness against Noise Attacks

Noise can be treated as a random intensity variation present in an image during the process of image acquisition and transmission. It can significantly deteriorate the image quality. In this subsection, to test the robustness of the proposed scheme, two kinds of noise attacks, Gaussian noise attack and the salt-and-pepper noise attack, are considered. The marked Lena images with different values of N are taken as the test images.

For the Gaussian noise attack, based on the 8-bit grayscale image values, three different values of σ (noise standard deviation) are chosen, which are 2.55, 8.06, and 18.03, respectively. The corresponding noise variance values with pixel values normalized to the range [0, 1] are 0.0001, 0.001, and 0.005, respectively. Figure 6 displays the tainted Lena images with number of users N equal to 16. The degradation in image quality is unobvious

compared to the noise-free marked image. The PSNR values for the marked Lena image with different values of N corrupted by Gaussian noise are listed in Table 3 and plotted in Figure 7. For each σ value, with the increase in number of users, the PSNR decline with respect to the noise-free marked image is mitigated. In the case that σ is 2.55 and N equals 16 or 32, the PSNR decrease is even negligible.



Figure 5. Original and marked Lena images with different values of N. (**a**) Original Lena image; (**b**) $N = 2$; (**c**) $N = 4$; (**d**) $N = 8$; (**e**) $N = 16$; (**f**) $N = 32$.



Figure 6. Marked Lena images with N = 16 corrupted by Gaussian noise. (**a**) σ = 0 (noise-free); (**b**) σ = 2.55; (**c**) σ = 8.06; (**d**) σ = 18.03.

Table 3. PSNR values (dB) versus number of users N for the Lena image under Gaussian noise.

Noise σ	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
0	45.13	39.91	34.38	28.65	22.66
2.55	38.80	36.92	33.31	28.32	22.57
8.06	29.86	29.56	28.65	26.26	21.92
18.03	23.00	22.93	22.73	21.98	19.88

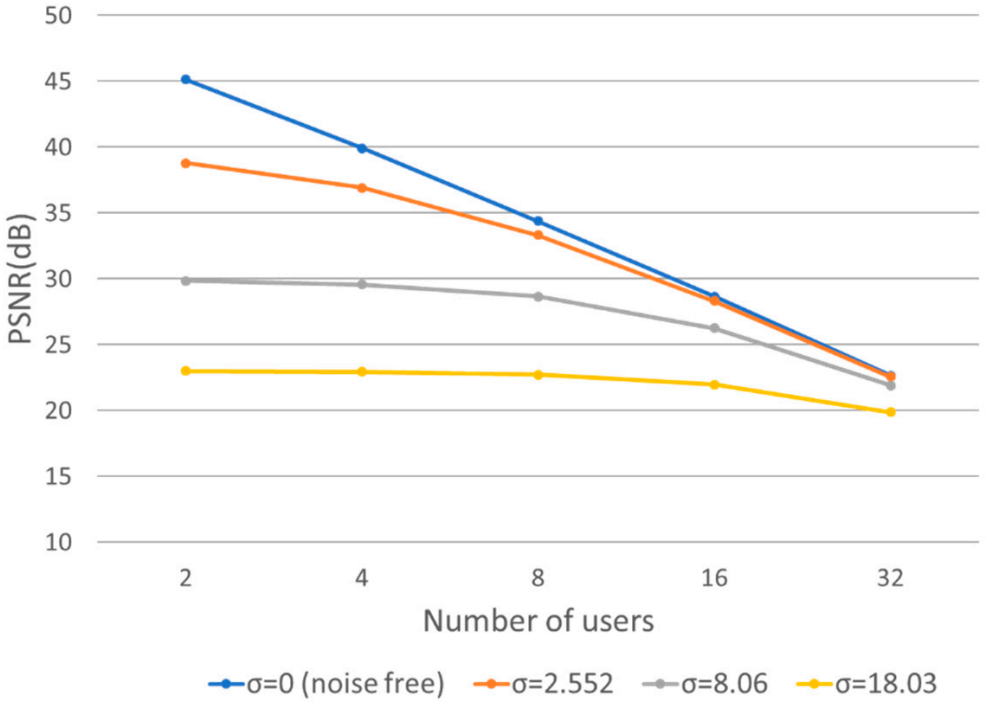


Figure 7. PSNR values versus number of users N for the Lena image under Gaussian noise.

Besides PSNR, to further evaluate the robustness of the proposed scheme, the bit-error rate (BER) between the extracted secret message and the embedded secret message is calculated as as:

$$\text{BER} = \frac{\text{Error bits}}{\text{Total number of embedded bits}}. \tag{11}$$

At the secret message extraction stage, the inner product operation of an N-digit CDMA codeword group and a secret key is performed. For a noise-free marked image, with a correct secret key, the result value will be N or $-N$. When the data is corrupted by Gaussian noise of variance σ^2, the inner product operation will result in a Gaussian random variable with mean N or $-N$ and variance $N\sigma^2$ if the effect of modulo operation is ignored. The error occurs when the noise component is less than $-N$ corresponding to mean N or the noise component is greater than N corresponding to mean $-N$. By symmetry of the Gaussian distribution, the probabilities are the same in either case. Hence, the error probability can be calculated as:

$$P_b = \int_N^\infty \frac{1}{\sqrt{2\pi N\sigma^2}} \exp\left(\frac{-x^2}{2N\sigma^2}\right) dx. \tag{12}$$

with variable substitution, Equation (12) can be rewritten as:

$$P_b = \int_{\frac{\sqrt{N}}{\sigma}}^\infty \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-u^2}{2}\right) du, \tag{13}$$

where the integrand is the standard Gaussian distribution function with mean 0 and variance 1.

When the number of users N grows, the longer secret key will lead to a larger inner product value. From Equations (12) and (13), we can observe that a larger N helps to reduce the error probability. This is the benefit of CDMA multiplexing. The BER values of the extracted secret from the Lena image with different values of N under Gaussian noise are listed in Table 4 and the corresponding expected BER values calculated according to Equation (12) are listed in Table 5. For comparison, those values are also plotted in Figure 8. It is observed that when the number of users increases, BER will decrease and

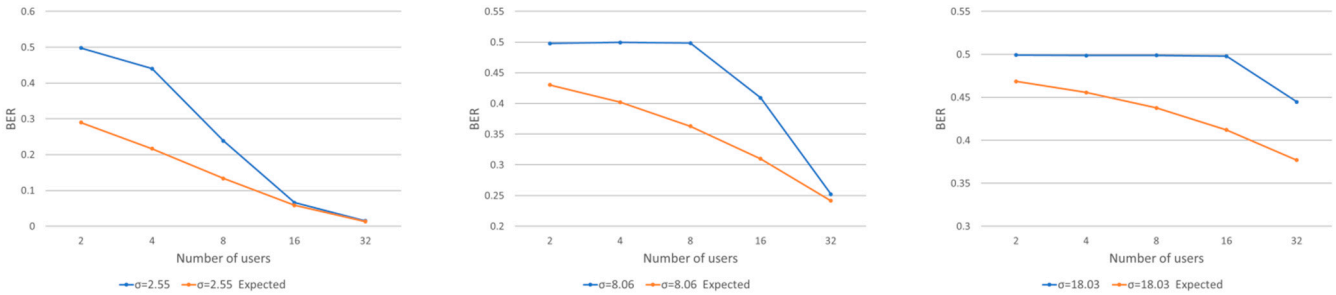
get closer to its expected value, especially for moderate values of σ such as 2.55 and 8.06. The reason is that for small value of N or larger value of σ, modulo wrap-around of the tainted pixel value occurs more frequently, causing the random variable to deviate from the Gaussian distribution.

Table 4. BER of the extracted secret from the Lena image versus number of users N under Gaussian noise.

Noise σ	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
2.55	0.4982	0.4404	0.2386	0.0660	0.0150
8.06	0.4980	0.4994	0.4985	0.4093	0.2521
18.03	0.4993	0.4987	0.4988	0.4979	0.4448

Table 5. Expected BER of the extracted secret versus number of users N under Gaussian noise.

Noise σ	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
2.55	0.2896	0.2164	0.1337	0.0584	0.0133
8.06	0.4304	0.4021	0.3629	0.3099	0.2415
18.03	0.4687	0.4558	0.4377	0.4122	0.3769



(**a**) (**b**) (**c**)

Figure 8. BER of the extracted secret from the Lena image versus number of users N under Gaussian noise. (**a**) σ = 2.55; (**b**) σ = 8.06; (**c**) σ = 18.03.

For the salt-and-pepper noise attack, three different density values are chosen, which are 0.1, 0.15, and 0.2, respectively. Figure 9 displays the tainted Lena images with number of users N equal to 16. For each corrupted image, evidently, the image quality is worse than the noise-free marked image. The PSNR values for the marked Lena image contaminated with the salt-and-pepper noise are listed in Table 6 and plotted in Figure 10a. Compared with the noise-free marked image, the PSNR drop is significant. However, for each density value, as N increases from 2 to 32, little has changed in the PSNR. The BER values of the extracted secret from the Lena image with different values of N under the salt-and-pepper noise are listed in Table 7 and plotted in Figure 10b. In contrast to Gaussian noise, for each density value, BER increases gradually with the increase in number of users. Similarly, for each N value, BER increases with the increase in noise density. The maximum BER is about 0.36 with noise density equal to 0.2 and N equal to 32. Owing to the impulse nature of the salt-and-pepper noise, when the noise occurs, it is very likely that all the N secret bits embedded in the N-digit CDMA codeword group will be destroyed. Thus, when the number of users N increases, BER is expected to increase accordingly.

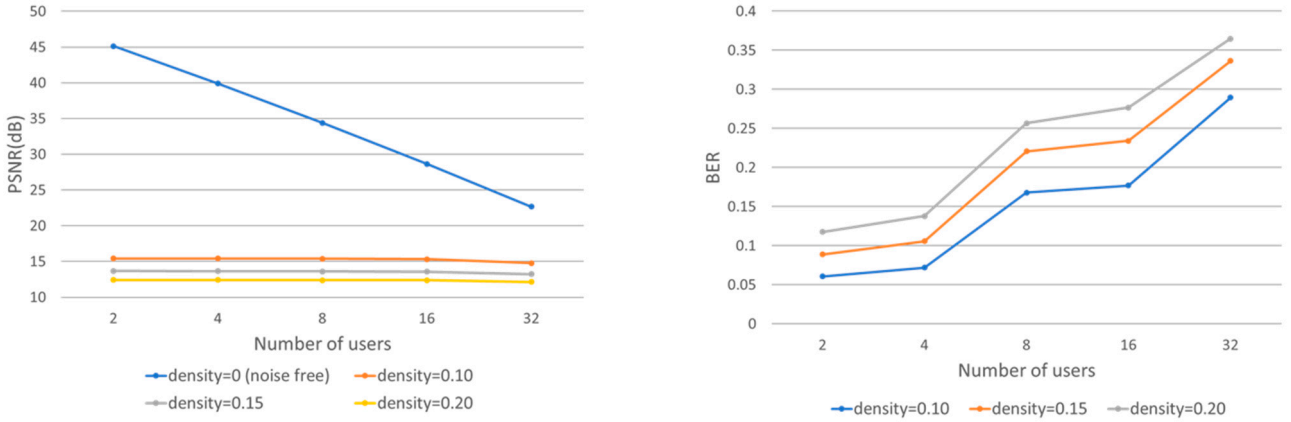


(**a**) (**b**) (**c**) (**d**)

Figure 9. Marked Lena images with N=16 corrupted by salt-and-pepper noise. (**a**) density = 0 (noise-free); (**b**) density = 0.10; (**c**) density = 0.15; (**d**) density = 0.20.

Table 6. PSNR values (dB) versus number of users N for the Lena image under salt-and-pepper noise.

Noise Density	N = 2	N = 4	N = 8	N = 16	N = 32
0	45.13	39.91	34.38	28.65	22.66
0.10	15.42	15.43	15.37	15.30	14.76
0.15	13.67	13.64	13.62	13.56	13.24
0.20	12.42	12.42	12.38	12.38	12.13



(**a**) (**b**)

Figure 10. (**a**) PSNR values versus number of users N for the Lena image under salt-and-pepper noise. (**b**) BER of the extracted secret from the Lena image versus N under salt-and-pepper noise.

Table 7. BER of the extracted secret from the Lena image versus number of users N under salt-and-pepper noise.

Noise Density	N = 2	N = 4	N = 8	N = 16	N = 32
0.10	0.0605	0.0718	0.1680	0.1770	0.2894
0.15	0.0887	0.1057	0.2206	0.2342	0.3365
0.20	0.1174	0.1378	0.2567	0.2767	0.3648

Based on the previous analysis, we know that data loss may occur under the two kinds of noise attacks. Next, we will take logotype binary images as the secret data, which can tolerate some amounts of bit errors. Several logotype binary images, each for a distinct participant, are embedded into the Lena image. The secret extraction results of two logotype secret images sized 90 × 90 corrupted by Gaussian noise and the salt-and-pepper noise are listed in Tables 8 and 9, respectively. For Gaussian noise, the visual quality of both images improves with the increase in N. When N reaches 16, even the four Chinese characters in the second image become recognizable. As for the salt-and-pepper noise, the visual quality

of both images decreases with the increase in N, the Chinese characters in the second image are visible with smaller values of N. The results coincide with the above BER analysis well. Since the characteristics of Gaussian noise and the salt-and-pepper noise are quite different, the preferable N values for noise reduction are also different. In resistance against noise attacks, the N value, which represents the multiplexity level, plays a major role.

Table 8. The secret extraction results of two logotype secret images at different values of N under Gaussian noise ($\sigma = 2.55$).

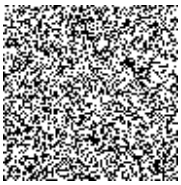
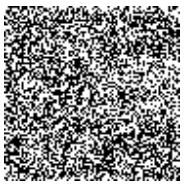
Secret Images	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
					
					

Table 9. The secret extraction results of two logotype secret images at different values of N under salt-and-pepper noise (density = 0.1).

Secret Images	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$
					
					

4.3. Robustness against Cropping Attacks

In the process of transmission and storage, digital images may suffer some portions of data loss. To test the robustness against cropping attacks, four different cropping sizes, 50×50, 100×100, 170×170, and 256×256, are applied to the 512×512 marked images. The cropped portions of sizes 50×50 and 170×170 are located at the upper left corner while the cropped portions of sizes 100×100 and 256×256 are located at the middle of the image. We take several logotype binary images as the secret images and consider two kinds of N values, which are 4 and 16. First, the original 90×90 logotype secret images are rescaled to fit the size of applied images. For $N = 4$, the rescaled secret images are 256×256 in size. For $N = 16$, the rescaled secret images are 128×128 in size. Then, all the secret images are scrambled by Arnold transform with five iterations and embedded into the Lena image through CDMA multiplexing. Next, the cropping attacks are performed. Figures 11 and 12 display the cropped Lena images with number of users N equal to 4 and

16 separately. The corresponding secret extraction results of two logotype secret images are listed in Tables 10 and 11, respectively.



(**a**) (**b**) (**c**) (**d**)

Figure 11. Marked Lena images with $N = 4$ corrupted by cropping attacks. (**a**) 50 × 50 cropped; (**b**) 100 × 100 cropped; (**c**) 170 × 170 cropped; (**d**) 256 × 256 cropped.



(**a**) (**b**) (**c**) (**d**)

Figure 12. Marked Lena images with $N = 16$ corrupted by cropping attacks. (**a**) 50 × 50 cropped; (**b**) 100 × 100 cropped; (**c**) 170 × 170 cropped; (**d**) 256 × 256 cropped.

Table 10. The extracted secret images under cropping attacks for $N = 4$.

Secret Images	50 × 50	100 × 100	170 × 170	256 × 256
				
	BER = 0.0029	BER = 0.0120	BER = 0.0342	BER = 0.0785
				
	BER = 0.0018	BER = 0.0077	BER = 0.0212	BER = 0.0503

Notice that the secret images are scrambled in advance of CDMA multiplexing and data embedding. With a fixed N, it is observed that BER for each secret image is roughly in proportion to the cropping size and independent of cropping positions. Next, BER performances at two different N values are compared. For the second secret image, the BER performance is insensitive to the N value. However, the situation in the first secret image is very different, BER at $N = 4$ is much lower than that at $N = 16$. The reason comes from the difference in secret keys. The first secret key is composed of N 1's while the second secret key is composed of $N/2$ 1's and $N/2$ -1's. In the experiment, the cropped region is

replaced by a white block. After modulo operation and decreased by N, 255 will be mapped to different values, -1 for $N = 4$ and 8 for $N = 16$. For the first secret image, the pixels in the cropped region will be decoded as bit 0 for $N = 4$ and bit 1 for $N = 16$, respectively. Since there are lots of black pixels in the first secret image, this will lead to fewer error bits for $N = 4$ and more error bits for $N = 16$. As for the second secret image, the pixels in the cropped region will be decoded as bit 1 regardless of the N value since the decision threshold is set to zero. Due to the large amount of white pixels in the second secret image, this will lead to fewer error bits and lower BER. Besides, we can see that the extracted secret image is still recognizable even if cropping attacks cause some portions of data loss. This shows that the proposed scheme is robust to cropping attacks.

Table 11. The extracted secret images under cropping attacks for $N = 16$.

Secret Images	50 × 50	100 × 100	170 × 170	256 × 256
				
	BER = 0.0071	BER = 0.0292	BER = 0.0793	BER = 0.1733
				
	BER = 0.0017	BER = 0.0081	BER = 0.0215	BER = 0.0499

5. Conclusions

Based on the technology of Code Division Multiple Access (CDMA), an innovative secret message multiplexing scheme by digital image data hiding techniques is proposed in this paper. Multiple secret messages distributed for several participants are encoded into a CDMA codeword through CDMA multiplexing and then embedded into a grayscale image. At the receiving side, each authorized recipient will be assigned a distinct spreading code as the secret key for data extraction. The cross-correlation between any two different secret keys is equal to zero. Due to this orthogonal property, only the authorized user can decode the respective hidden secret message using a specific secret key. These characteristics help the proposed scheme to possess strong data security. Experimental results also demonstrate that the proposed scheme is robust to cropping attacks and noise attacks, such as the Gaussian noise attack and the salt-and-pepper noise attack. In the case that several different secret messages must be embedded and transmitted simultaneously, the proposed scheme provides an efficient and feasible solution.

Author Contributions: Conceptualization, J.-H.H. and C.-H.L.; methodology, J.-H.H. and C.-H.L.; software, K.-S.S.; validation, J.-H.H. and C.-H.L.; formal analysis, C.-H.L.; resources, J.-H.H.; data curation, K.-S.S.; writing—original draft preparation, C.-H.L.; writing—review and editing, C.-H.L.; visualization, C.-H.L.; supervision, C.-C.C.; project administration, C.-C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wu, H.; Huang, F. Reversible data hiding based on partitioning the prediction values. In *Digital Forensics and Watermarking*; Springer: Cham, Switzerland, 2020; Volume 12022, pp. 195–203. [CrossRef]
2. Gao, X.; Pan, Z.; Gao, E.; Fan, G. Reversible data hiding for high dynamic range images using two-dimensional prediction-error histogram of the second time prediction. *Signal Process.* **2020**, *173*, 107579. [CrossRef]
3. Chang, C.-C.; Li, C.-T.; Chen, K. Privacy-Preserving Reversible Information Hiding Based on Arithmetic of Quadratic Residues. *IEEE Access* **2019**, *7*, 54117–54132. [CrossRef]
4. Chang, C.-C.; Liu, Y. Fast turtle shell-based data embedding mechanisms with good visual quality. *J. Real-Time Image Process.* **2019**, *16*, 589–599. [CrossRef]
5. Lin, C.-C.; Chang, C.-C.; Wang, Z.-M. Reversible Data Hiding Scheme Using Adaptive Block Truncation Coding Based on an Edge-Based Quantization Approach. *Symmetry* **2019**, *11*, 765. [CrossRef]
6. Chang, C.-C.; Lu, T.-C.; Horng, G.; Huang, Y.-H.; Hsu, Y.-M. A high payload data embedding scheme using dual stego-images with reversibility. In Proceedings of the 2013 9th International Conference on Information, Communications & Signal Processing, Tainan, Taiwan, 10–13 December 2013; pp. 1–5. [CrossRef]
7. Malik, A.; Wang, H.; Khan, A.N.; Chen, Y.; Chen, Y. A novel lossless data hiding scheme in homomorphically encrypted images. In *Digital Forensics and Watermarking (Lecture Notes in Computer Science)*; Springer: Cham, Switzerland, 2019; Volume 12022.
8. Puteaux, P.; Trinel, D.; Puech, W. High-capacity data hiding in encrypted images using MSB prediction. In Proceedings of the 2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA), Oulu, Finland, 12–15 December 2016; pp. 1–6. [CrossRef]
9. Horng, J.-H.; Lin, J.; Liu, Y.; Chang, C.-C. 3D Multilayered Turtle Shell Models for Image Steganography. *Comput. Model. Eng. Sci.* **2020**, *125*, 879–906. [CrossRef]
10. Lin, J.; Horng, J.H.; Liu, Y.; Chang, C.C. An Anisotropic Reference Matrix for Image Steganography. *J. Vis. Commun. Image Represent.* **2021**, *74*, 102969. [CrossRef]
11. Lin, C.-C.; Liu, X.-L.; Tai, W.-L.; Yuan, S.-M. A novel reversible data hiding scheme based on AMBTC compression technique. *Multimedia Tools Appl.* **2015**, *74*, 3823–3842. [CrossRef]
12. Hong, W.; Ma, Y.-B.; Wu, H.-C.; Chen, T.-S. An efficient reversible data hiding method for AMBTC compressed images. *Multimedia Tools Appl.* **2017**, *76*, 5441–5460. [CrossRef]
13. Horng, J.-H.; Chang, C.-C.; Li, G.-L. Steganography Using Quotient Value Differencing and LSB Substitution for AMBTC Compressed Images. *IEEE Access* **2020**, *8*, 129347–129358. [CrossRef]
14. Li, Y.; Yao, S.; Yang, K.; Tan, Y.-A.; Zhang, Q. A High-Imperceptibility and Histogram-Shifting Data Hiding Scheme for JPEG Images. *IEEE Access* **2019**, *7*, 73573–73582. [CrossRef]
15. Xuan, G.; Chen, J.; Zhu, J.; Shi, Y.Q.; Ni, Z.; Su, W. Lossless data hiding based on integer wavelet transform. In Proceedings of the 2002 IEEE Workshop on Multimedia Signal Processing, St. Thomas, VI, USA, 9–11 December 2002; pp. 312–315.
16. Li, F.; Mao, Q.; Chang, C.C. A reversible data hiding method based on IWT and the Sudoku method. *Int. J. Netw. Secur.* **2014**, *18*, 410–419.
17. Yang, C.Y.; Hu, W.C. Reversible data hiding in the spatial and frequency domains. *Int. J. Image Process.* **2010**, *3*, 373–384.
18. Zhang, H.; Hu, L. A data hiding scheme based on multidirectional line encoding and integer wavelet transform. *Signal Process. Image Commun.* **2019**, *78*, 331–344. [CrossRef]
19. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
20. Blakley, G.R. Safeguarding cryptographic keys. In Proceedings of the American Federation of Information Processing Societies National Computer Conference, New York, NY, USA, 4–7 June 1979; Volume 48, pp. 313–317.
21. Naor, M.; Shamir, A. Visual cryptography. In *Advances in Cryptology—EUROCRYPT (Lecture in Computer Science)*; Springer: Berlin, Germany, 1994; Volume 950.
22. Chang, C.C.; Kieu, T.D.; Chou, Y.C. Reversible data hiding method using two steganographic images. In Proceedings of the TENCON-IEEE Region 10th Conference, Taipei, Taiwan, 30 October–2 November 2007; pp. 1–4.
23. Chang, C.C.; Lin, P.Y.; Wang, Z.H.; Li, M.C. A Sudoku-based secret image sharing method with reversibility. *J. Commun.* **2010**, *5*, 5–12. [CrossRef]
24. Liu, Y.; Chang, C.-C. A turtle shell-based visual secret sharing scheme with reversibility and authentication. *Multimed. Tools Appl.* **2018**, *77*, 25295–25310. [CrossRef]
25. Chang, C.C.; Horng, J.H.; Shih, C.S.; Chang, C.C. A maze matrix-based secret image sharing scheme with cheater detection. *Sensors* **2020**, *20*, 3802. [CrossRef]
26. Gao, K.; Horng, J.-H.; Chang, C.-C. A Novel (2, 3) Reversible Secret Image Sharing Based on Fractal Matrix. *IEEE Access* **2020**, *8*, 174325–174341. [CrossRef]
27. Liu, Z.; Zhu, G.; Wang, Y.G.; Yang, J.; Kwong, S. A Novel (t, s, k, n)-Threshold Visual Secret Sharing Scheme Based on Access Structure Partition. *ACM Trans. Multimed. Comput. Commun. Appl.* **2021**, *16*, 1–21. [CrossRef]
28. Charoghchi, S.; Mashhadi, S. Three (t,n)-secret image sharing schemes based on homogeneous linear recursion. *Inf. Sci.* **2021**, *552*, 220–243. [CrossRef]
29. Ding, W.; Liu, K.; Yan, X.; Liu, L. Polynomial-Based Secret Image Sharing Scheme with Fully Lossless Recovery. *Int. J. Digit. Crime Forensics* **2018**, *10*, 120–136. [CrossRef]

30. Liu, L.; Lu, Y.; Yan, X.; Ding, W.; Xuan, Q. A Lossless Polynomial-Based Secret Image Sharing Scheme Utilizing the Filtering Operation. *Adv. Intell. Syst. Comput.* **2020**, *895*, 129–139.
31. Horng, J.-H.; Chen, S.-S.; Chang, C.-C. A k,n-Threshold Secret Image Sharing Scheme Based on a Non-Full Rank Linear Model. *Mathematics* **2022**, *10*, 524. [CrossRef]
32. Lee, C.-H.; Sun, K.-S.; Wang, Y.-Q.; Horng, J.-H.; Chang, C.-C. Secret Message CDMA Multiplexing via QR Codes. In Proceedings of the 2022 3rd Asia Service Sciences and Software Engineering Conference (ASSE' 22), Macao, 24–26 February 2022; Association for Computing Machinery: New York, NY, USA, 2022; pp. 52–59. [CrossRef]
33. Viterbi, A.J. *CDMA: Principles of Spread Spectrum Communications*; Addison-Wesley: Boston, MA, USA, 1995.
34. Proakis, J.G. *Digital Communications*, 4th ed.; McGraw Hill: New York, NY, USA, 2001.
35. Rappaport, T.S. *Wireless Communications: Principles and Practice*, 2nd ed.; Prentice Hall: Hoboken, NJ, USA, 2001.
36. Pickholtz, R.; Schilling, D.; Milstein, L. Theory of Spread-Spectrum Communications—A Tutorial. *IEEE Trans. Commun.* **1982**, *30*, 855–884. [CrossRef]
37. Pickholtz, R.; Milstein, L.; Schilling, D. Spread spectrum for mobile communications. *IEEE Trans. Veh. Technol.* **1991**, *40*, 313–322. [CrossRef]
38. Moshavi, S. Multi-user detection for DS-CDMA communications. *IEEE Commun. Mag.* **1996**, *34*, 124–136. [CrossRef]
39. Zhang, X.P.; Wang, S.Z. Efficient Steganographic Embedding by Exploiting Modification Direction. *IEEE Commun. Lett.* **2006**, *10*, 781–783. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.