mathematics

MDPI

Article

A (k, n)-Threshold Secret Image Sharing Scheme Based on a Non-Full Rank Linear Model

Ji-Hwei Horng [1], Si-Sheng Chen [2,3,*] and Chin-Chen Chang [3,*]

[1] Department of Electronic Engineering, National Quemoy University, Kinmen 89250, Taiwan; horng@email.nqu.edu.tw
[2] School of Big Data and Artificial Intelligence of Fujian Polytechnic Normal University, Fuzhou 350030, China
[3] Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan
* Correspondence: sschen358@mail.fcu.edu.tw (S.-S.C.); ccc@o365.fcu.edu.tw (C.-C.C.)

Abstract: Secret image sharing is a hot issue in the research field of data hiding schemes for digital images. This paper proposes a general (k, n) threshold secret image sharing scheme, which distributes secret data into n meaningful image shadows based on a non-full rank linear model. The image shadows are indistinguishable from their corresponding distinct cover images. Any k combination of the n shares can perfectly restore the secret data. In the proposed scheme, the integer parameters (k, n), with $k \leq n$, can be set arbitrarily to meet the application requirement. The experimental results demonstrate the applicability of the proposed general scheme. The embedding capacity, the visual quality of image shadows, and the security level are satisfactory.

Keywords: secret image sharing; data hiding; meaningful shadow images; non-full rank linear model

Citation: 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. https://doi.org/10.3390/math10030524

Academic Editor: Radi Romansky

Received: 15 January 2022
Accepted: 5 February 2022
Published: 7 February 2022

1. Introduction

With the rapid development of portable devices and wireless communication, a huge amount of information is transmitted via the Internet in daily life. To protect the security of private information, encryption techniques are applied [1,2], which transform the private data into a meaningless ciphertext with a secret key. However, the meaningless ciphertext may arouse the eavesdropper's suspicion. Researchers turn to develop some techniques which hide private data into meaningful media such as digital images or videos. The technique of embedding secret data in a digital cover image to produce a stego image is called image steganography. Conventional methods for image steganography include the least significant bits (LSB) substitution [3], the exploiting modifying direction (EMD) method [4], the difference expansion (DE) [5], the histogram shifting (HS) [6], and some other approaches [7,8]. The secret data are transmitted under the cover of a meaningful digital image, without attracting the eavesdropper's suspicion.

The current image steganography techniques suffer from a problem that the stego images are vulnerable to tampering attack. Slight changes to a stego image may completely disrupt the decryption result. To improve the flexibility of decryption, in 1995, Naro and Shamir [9] proposed the visual secret sharing (VSS), also called visual cryptography, in which they encrypted a secret image into n shares and recovered the secret image through stacking not less than t shares. Later, many improved versions of visual secret sharing were proposed [10–13]. The main weakness of such schemes is the low quality of the restored secret image.

As a result, a series of secret image sharing schemes was proposed [14–19] that used mathematical computation to recover the secret image instead of stacking. A common approach is to apply the polynomial-based secret sharing scheme. In a (k, n) polynomial-based secret image sharing scheme, the dealer divides a pixel or a pixel block of the secret image into n shadows via a polynomial function. In such schemes, we can consider that the

secret image is encrypted with polynomial-based secret sharing into n noise-like images. Due to the continuous feature of a polynomial, the reconstructed secret image is distorted by numerical errors.

Another series of secret image sharing schemes focusing on sharing secret into multiple image shadows were proposed [20–22]. The image shadows are indistinguishable from the cover images and the secret can be perfectly recovered. In 2007, Chang et al. [20] introduced the first version of such schemes. The secret message is transmitted through two meaningful image shadows, and it can be recovered only when both image shadows are obtained. Although these schemes distribute the secret data into multiple image shadows, all shadows are required in the secret extraction phase.

Some works on (k, n)-threshold secret image sharing have been proposed [23–26], in which the secret data can be completely recovered by no less than k shares. However, shadow images are vulnerable to the steganalysis because they are generated based on the same cover image. Recently, some secret image sharing schemes using different meaningful cover images have been proposed [27,28]. In these methods, (k, n) is a fixed pair of predefined constants.

The motivation of this research is to find a general secret image sharing scheme, in which parameters (k, n) with $k < n$ can be arbitrarily chosen according to different application requirements, and the n distinct cover images can be arbitrarily assigned too. We first present a (k, n)-threshold secret sharing scheme based on a non-full rank linear model. A secret segment of k entries is used to generate a solution vector of n entries. By using any combination of k entries from the solution vector, the complete solution vector can be recovered and the secret segment can be restored.

Then, this scheme is applied to a digital image transmission system. In the first step, the secret data is divided into secret vectors too. Then, the solution vectors are generated in the same way as the proposed secret sharing scheme. To implement it on the image transmission system, each solution vector is further applied to modulate a vector constituted by pixel values collected from different cover images. In the final step, the resulting modulated vector is recorded to the image shadows. In the recovery phase, only a predefined number of shadows are required to restore the complete secret.

The remainder of this paper is organized as follows. In Section 2, we introduce the (k, n) secret sharing scheme based on a non-full rank linear model. The proposed (k, n)-threshold secret image sharing scheme is described in Section 3. The experimental results and theoretic analysis are given in Section 4. Finally, we conclude this paper in Section 5.

2. (k, n)-Threshold Secret Sharing Scheme Based on Non-Full Rank Linear Model

In this section, we first discuss the solution of a non-full rank linear model over a finite integer field. Then, an application of recovering a solution vector with missing entries is introduced. Based on this mathematical model, we propose a generalized (k, n)-threshold secret sharing scheme based on a non-full rank linear model.

2.1. Non-Full Rank Linear Model over a Finite Integer Field

Assume p is a prime number, z_p is a field of integers modulo p; n and k are two positive integers with $n \geq k$. The equations $\pi_1, \pi_2, \cdots, \pi_{n-k}$ are $(n-k)$ linearly independent equations of n variables over the field z_p, which are formulated as

$$\pi_i : \ a_{i,1}x_1 + a_{i,2}x_2 + \cdots + \cdots a_{i,n}x_n = b_i \bmod p, \ i = 1, 2, \cdots, n-k. \tag{1}$$

Let

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{i,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n-k,1} & a_{n-k,2} & \cdots & a_{n-k,n} \end{pmatrix}, b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-k} \end{pmatrix}. \tag{2}$$

Then, the solution space of the $(n-k)$ linear equations Equation (1) is given by

$$H = \left\{ x \middle| x = (x_1, x_2, \cdots, x_n)^T, Ax = b \bmod p \right\}. \tag{3}$$

This system of linear equations can also be written in matrix form as

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{i,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n-k,1} & a_{n-k,2} & \cdots & a_{n-k,n} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-k} \end{pmatrix} \bmod p. \tag{4}$$

According to linear algebra, a system of n linearly independent equations with n variables $x = (x_1, x_2, \cdots, x_n)^T$, a unique solution vector x can be determined. The current non-full rank $(n-k<n)$ system of linearly independent equations leads to a solution space of rank $n-(n-k)=k$.

We can manipulate the augmented matrix $(A|b)$ by using elementary row operations over field z_p to obtain the simplest row form as

$$\begin{pmatrix} 1 & 0 & \cdots & 0 & c_{1,n-k+1} & \cdots & c_{1,n+1} \\ 0 & 1 & \cdots & 0 & c_{2,n-k+1} & \cdots & c_{2,n+1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & c_{n-k,n-k+1} & \cdots & c_{n-k,n+1} \end{pmatrix}. \tag{5}$$

Then, we obtain an equivalent system of linearly independent equations

$$\begin{cases} x_1 = -c_{1,n-k+1}x_{n-k+1} - c_{1,n-k+2}x_{n-k+2} - \cdots - c_{1,n}x_n + c_{1,n+1} \bmod p \\ x_2 = -c_{2,n-k+1}x_{n-k+1} - c_{2,n-k+2}x_{n-k+2} - \cdots - c_{2,n}x_n + c_{2,n+1} \bmod p \\ \vdots \\ x_{n-k} = -c_{n-k,n-k+1}x_{n-k+1} - c_{n-k,n-k+2}x_{n-k+2} - \cdots - c_{n-k,n}x_n + c_{n-k,n+1} \bmod p \end{cases}. \tag{6}$$

Let $\beta = (\beta_1, \beta_2, \ldots, \beta_{n-k}, 0, 0, \cdots, 0)^T$ be a particular solution for Equation (6). To solve the unknown entries, we substitute β into Equation (6) and obtain

$$\begin{cases} \beta_1 = c_{1,n+1} \\ \beta_2 = c_{2,n+1} \\ \vdots \\ \beta_{n-k} = c_{n-k,n+1} \end{cases}. \tag{7}$$

To solve a set of linearly independent homogeneous solutions of Equation (6), let the k homogeneous solutions be

$$(\alpha_1, \alpha_2, \cdots, \alpha_k) = \begin{pmatrix} \alpha_{1,1} & \alpha_{2,1} & \cdots & \alpha_{k,1} \\ \alpha_{1,2} & \alpha_{2,1} & \cdots & \alpha_{k,2} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{1,n-k} & \alpha_{2,n-k} & \cdots & \alpha_{k,n-k} \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & 0 \\ 0 & 0 & \cdots & 1 \end{pmatrix}. \tag{8}$$

To solve the unknown entries of each solution α_i, we can substitute them into the homogeneous equation that corresponds to Equation (6) and obtain

$$\begin{cases} \alpha_{i,1} = -c_{1,n-k+j} \bmod p \\ \alpha_{i,2} = -c_{2,n-k+j} \bmod p \\ \quad\quad\vdots \\ \alpha_{i,n-k} = -c_{n-k,n-k+j} \bmod p \end{cases}, i = 1, 2, \ldots, k. \tag{9}$$

The solution space H can be defined by using the homogeneous and particular solutions as

$$H = \left\{ x \middle| x = \sum_{i=1}^{k} m_i \alpha_i + \beta \bmod p \right\}, \text{ where } m_i \in z_p. \tag{10}$$

Suppose a particular solution vector with missing entries $x' = \left(x'_1, x'_2, \cdots, x'_k\right)^T$ is given, where $(n-k)$ entries of a solution $x = (x_1, x_2, \cdots, x_n)^T \in H$ are missing. We can recover the missing entries by the following procedures.

Without the loss of generality, let us assume the available entries are the first k entries. We first manipulate the original augmented matrix $(A|b)$ with elementary row operations into the following form

$$\begin{pmatrix} h_{1,1} & \cdots & h_{1,k} & 1 & 0 & \cdots & 0 & h_{1,n+1} \\ h_{2,1} & \cdots & h_{2,k} & 0 & 1 & \cdots & 0 & h_{2,n+1} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ h_{n-k,1} & \cdots & h_{n-k,k} & 0 & 0 & \cdots & 1 & h_{n-k,n+1} \end{pmatrix}. \tag{11}$$

Then, the missing entries can be obtained by

$$\begin{cases} x_{k+1} = -h_{1,1}x_1 - h_{1,2}x_2 - \cdots - h_{1,k}x_k + h_{1,n+1} \bmod p \\ x_{k+2} = -h_{2,1}x_1 - h_{2,2}x_2 - \cdots - h_{2,k}x_k + h_{2,n+1} \bmod p \\ \quad\quad\vdots \\ x_n = -h_{n,1}x_1 - h_{n,2}x_2 - \cdots - h_{n,k}x_k + h_{n,n+1} \bmod p \end{cases}. \tag{12}$$

Each solution vector can be represented as

$$x = m_1\alpha_1 + m_2\alpha_2 + \cdots + m_k\alpha_k + \beta \bmod p. \tag{13}$$

By substituting the obtained particular and homogeneous solutions into Equation (13), we can get

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-k} \\ \vdots \\ x_n \end{pmatrix} = m_1 \begin{pmatrix} \alpha_{1,1} \\ \alpha_{1,2} \\ \vdots \\ \alpha_{1,n-k} \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} + m_2 \begin{pmatrix} \alpha_{2,1} \\ \alpha_{2,2} \\ \vdots \\ \alpha_{2,n-k} \\ 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} + \cdots + m_k \begin{pmatrix} \alpha_{k,1} \\ \alpha_{k,2} \\ \vdots \\ \alpha_{k,n-k} \\ 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} + \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{n-k} \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \bmod p. \tag{14}$$

By expanding the last k rows, we can also get the coefficients of the given vector x'.

$$\begin{cases} m_1 = x_{n-k+1} \\ m_2 = x_{n-k+2} \\ \quad\vdots \\ m_k = x_n \end{cases}. \tag{15}$$

From the above analysis, Equations (5)–(7) give the solution steps to obtain a particular solution, and Equations (8) and (9) give the solution steps to obtain a set of linearly independent homogeneous solutions. Thus, any solution vector of the given $(n-k)$ linearly independent equations of n variables can be represented with a linear combination of homogeneous and particular solutions as Equation (10). In addition, Equations (11) and (12) give the solution steps to recover the missing $(n-k)$ entries of a solution vector from k available entries; Equations (13)–(15) give a one-to-one mapping between a solution vector and the coefficients of the linear combination. Based on this model, suppose coefficients $(m_1, m_2, \cdots, m_k)$ are the secret, we can produce a vector $(x_1, x_2, \cdots, x_n)^T$, which includes n secret shares, using Equation (13). When k shares among n are available, we can restore the remaining shares and calculate the secret from the restored complete set of shares.

2.2. (k, n). -Threshold Secret Sharing Scheme Based on the Non-Full Rank Linear Model

A general scheme for (k, n) threshold secret sharing based on a non-full rank linear model is proposed in this section. It is divided into three phases: the setup, the share generation, and the secret extraction with authentication.

2.2.1. Setup

The scenario is that a dealer embeds secret data S into n shares $X_1, X_2, \cdots, X_n$ and distributes them to n participants. To recover secret data, at least k participants should cooperate by providing their shares. The dealer first selects $(n-k)$ linearly independent equations with n variables over the field z_p as

$$\pi_i: \ a_{i,1}x_1 + a_{i,2}x_2 + \cdots + \cdots a_{i,n}x_n = b_i \bmod p, \ i = 1, 2, \cdots, n-k. \tag{16}$$

Then, a set of linearly independent homogeneous solutions $\{\alpha_1, \alpha_2, \cdots, \alpha_k\}$ and a particular solution β can be obtained by applying Equations (5)–(9).

2.2.2. Share Generation

The secret data S to be shared is divided into l secret segments of k entries denoted by $m_j = \left(m_{j,1}, m_{j,2}, \cdots, m_{j,k}\right)$, where $m_{j,r}(1 \le r \le k) \in z_p$, $1 \le j \le l$. For each secret message segment m_j, a solution vector x_j can be generated by

$$x_j = F(m_j) = (m_{j,1} + \gamma_{j,1})\alpha_1 + (m_{j,2} + \gamma_{j,2})\alpha_2 + \cdots + \left(m_{j,k} + \gamma_{j,k}\right)\alpha_k + \beta \bmod p, \tag{17}$$

where $\gamma_{j,r}(1 \le r \le k)$ is a series of integers over z_p generated by a pre-shared data hiding key. This integer series encrypt each secret entry by adding a random offset. By rearranging the collection of synthesized vectors x_j $(1 \le j \le l)$, the dealer generates n secret shares as

$$X_i = (x_1(i), \ x_2(i), \cdots, x_l(i))^T, 1 \le i \le n, \tag{18}$$

where $x_j(i)$ denotes the i-th entry of the j-th synthesized vector x_j. Finally, these shares are distributed to the n participants.

2.2.3. Secret Extraction with Authentication

Suppose k faithful participants provide their shares as $X_1, X_2, \cdots, X_k$. The combiner first rearranges the k shares into l vectors as

$$x'_j = (x'_j(1), \ x'_j(2), \cdots, x'_j(k))^T, 1 \le j \le l, \tag{19}$$

where $x'_j(i)$ denotes the j-th entry of the i-th share. Then, the vector x'_j with missing entries can be restored into the complete solution x_j by using Equations (11) and (12).

To extract secret data, the combiner first reproduces the random sequence $\gamma_{j,r}(1 \le r \le k)$ by the shared data hiding key. Then, the secret segments $m_j(1 \le j \le l)$ can be extracted by substituting the complete solution vectors $x_j(1 \le j \le l)$ back into Equation (17), as

$$\begin{pmatrix} x_j(1) \\ x_j(2) \\ \vdots \\ x_j(n-k) \\ \vdots \\ \vdots \\ \vdots \\ x_j(n) \end{pmatrix} + (m_{j,1}+\gamma_{j,1}) \begin{pmatrix} \alpha_{1,1} \\ \alpha_{1,2} \\ \vdots \\ \alpha_{1,n-k} \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \cdots + (m_{j,k}+\gamma_{j,k}) \begin{pmatrix} \alpha_{k,1} \\ \alpha_{k,2} \\ \vdots \\ \alpha_{k,n-k} \\ 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} + \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{n-k} \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \bmod p. \tag{20}$$

By expanding the last k rows, we can get the secret segments

$$\begin{cases} m_{j,1} = x_j(n-k+1) - \gamma_{j,1} \\ m_{j,2} = x_j(n-k+2) - \gamma_{j,2} \\ \vdots \\ m_{j,k} = x_j(n) - \gamma_{j,k} \end{cases}. \tag{21}$$

When additional doubtful shares are available, they can be authenticated by using the restored missing entries based on the trustworthy shares.

2.3. Demonstration

Let us assume $k = 2$, $n = 4$, and $p = 5$. That is, we are going to demonstrate a (2,4)-threshold secret sharing scheme. The dealer selects two linearly independent equations with four variables, as

$$\begin{aligned} \pi_1 &: 1x_1 + 3x_2 + 1x_3 + 2x_4 = 1 \bmod 5, \\ \pi_2 &: 2x_1 + 4x_2 + 0x_3 + 1x_4 = 3 \bmod 5. \end{aligned} \tag{22}$$

That is, $A = \begin{pmatrix} 1 & 3 & 1 & 2 \\ 2 & 4 & 0 & 1 \end{pmatrix}$ and $b = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$.

2.3.1. Particular and Homogeneous Solutions

Manipulate the augmented matrix with elementary row operations (see Appendix A) over the field Z_p to obtain a row simplest augmented matrix as

$$\begin{aligned} (A|b) = \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 2 & 4 & 0 & 1 & 3 \end{array}\right) &\overset{R_2=R_2+3R_1}{\rightarrow} \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 3 & 3 & 2 & 1 \end{array}\right) \\ \overset{R_2=2R_2}{\rightarrow} \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right) &\overset{R_1=R_1+2R_2}{\rightarrow} \left(\begin{array}{cccc|c} 1 & 0 & 3 & 0 & 0 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right). \end{aligned} \tag{23}$$

Then, by applying Equation (7), we can get a particular solution $\beta = (0, 2, 0, 0)^T$. By applying Equations (8) and (9), we can get a set of linearly independent homogeneous solutions

$$\alpha_1 = (-3 \bmod 5, -1 \bmod 5, 1, 0) = (2, 4, 1, 0), \tag{24}$$

$$\alpha_2 = (-0 \bmod 5, -4 \bmod 5, 0, 1) = (0, 1, 0, 1). \tag{25}$$

2.3.2. Share Generation

For each secret segment $m_j = (m_{j,1}, m_{j,2})$, the solution vector x_j can be generated by

$$x_j = (m_{j,1} + \gamma_{j,1})\alpha_1 + (m_{j,2} + \gamma_{j,2})\alpha_2 + \beta \bmod 5. \tag{26}$$

Let the secret data be $m_1 = (4,0)$, $m_2 = (1,1), m_3 = (4,2)$, and the random integer series generated by the data hiding key be $(\gamma_{1,1}, \gamma_{1,2}) = (3,\ 2)$, $(\gamma_{2,1}, \gamma_{2,2}) = (4,\ 2)$, $(\gamma_{3,1}, \gamma_{3,2}) = (2,3)$, then the generated vectors are

$$x_1 = \begin{pmatrix} x_1(1) \\ x_1(2) \\ x_1(3) \\ x_1(4) \end{pmatrix} = (4+3)\begin{pmatrix} 2 \\ 4 \\ 1 \\ 0 \end{pmatrix} + (0+2)\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 2 \\ 2 \end{pmatrix}, \quad (27)$$

$$x_2 = \begin{pmatrix} x_2(1) \\ x_2(2) \\ x_2(3) \\ x_2(4) \end{pmatrix} = (1+4)\begin{pmatrix} 2 \\ 4 \\ 1 \\ 0 \end{pmatrix} + (1+2)\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 3 \end{pmatrix}, \quad (28)$$

$$x_3 = \begin{pmatrix} x_3(1) \\ x_3(2) \\ x_3(3) \\ x_3(4) \end{pmatrix} = (4+2)\begin{pmatrix} 2 \\ 4 \\ 1 \\ 0 \end{pmatrix} + (2+3)\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \\ 1 \\ 0 \end{pmatrix}. \quad (29)$$

The dealer rearranges the vector entries into secret shares and distributes them to the participants.

$$X_1 = (4,\ 0,\ 2)^T, X_2 = (2,\ 0,\ 1)^T, X_3 = (2,\ 0,\ 1)^T, X_4 = (2,\ 3,\ 0)^T. \quad (30)$$

2.3.3. Authentication

At the combiner side, suppose that only shares $X_1 = (4,\ 0,\ 2)^T$ and $X_2 = (2,\ 0,\ 1)^T$ are trustworthy, while X_3 and X_4 are doubtful. The combiner can reproduce the remaining shares and restore the secret data by using the faithful ones. He/she first manipulates $(A|b)$ by using elementary row operations, so that the submatrix constituted by column 3 and 4 becomes an identity matrix as

$$(A|b) = \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 2 & 4 & 0 & 1 & 3 \end{array}\right) \overset{R_1 = R_1 + 3R_2}{\rightarrow} \left(\begin{array}{cccc|c} 2 & 0 & 1 & 0 & 0 \\ 2 & 4 & 0 & 1 & 3 \end{array}\right). \quad (31)$$

Then, the combiner rearranges the available entries into $x_1' = (4,2)^T$, $x_2' = (0,0)^T$, and $x_3' = (2,1)^T$. Finally, these vectors with missing entries are substituted into Equation (31) as

$$\begin{cases} x_1(3) = -2 \times 4 - 0 \times 2 + 0 \bmod 5 = 2 \\ x_1(4) = -2 \times 4 - 4 \times 2 + 3 \bmod 5 = 2 \end{cases}. \quad (32)$$

$$\begin{cases} x_2(3) = -2 \times 0 - 0 \times 0 + 0 \bmod 5 = 0 \\ x_2(4) = -2 \times 0 - 4 \times 0 + 3 \bmod 5 = 3 \end{cases}. \quad (33)$$

$$\begin{cases} x_3(3) = -2 \times 2 - 0 \times 1 + 0 \bmod 5 = 1 \\ x_3(4) = -2 \times 2 - 4 \times 1 + 3 \bmod 5 = 0 \end{cases}. \quad (34)$$

Therefore, the complete solution vectors are $x_1 = (4,2,2,2)^T$, $x_2 = (0,0,0,3)^T$, $x_3 = (2,1,1,0)^T$, and the remaining shares should be $X_3 = (2,\ 0,\ 1)^T$, $X_4 = (2,\ 3,\ 0)^T$. If a share provided by a suspicious participant does not match the result, then he/she is a cheater.

2.3.4. Secret Recovery

To restore secret data, the combiner first manipulates the given linear equations to obtain the particular and homogeneous solutions as the share generation phase. Thus, the solution vectors should be

$$x_j = (m_{j,1} + \gamma_{j,1}) \begin{pmatrix} 2 \\ 4 \\ 1 \\ 0 \end{pmatrix} + (m_{j,2} + \gamma_{j,2}) \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} \bmod 5. \tag{35}$$

Therefore, the secret segments can be obtained by substituting the restored shares X_3, X_4 and the integer series $\gamma_{j,r}, 1 \le j \le 2,\ 1 \le r \le 2$, generated by the shared data hiding key into the last two rows of Equation (35), as

$$\begin{cases} m_{1,1} + \gamma_{1,1} = x_1(3) \\ m_{1,2} + \gamma_{1,2} = x_1(4) \end{cases} \rightarrow \begin{cases} m_{1,1} = x_1(3) - \gamma_{1,1} \bmod 5 = 2 - 3 \bmod 5 = 4 \\ m_{1,2} = x_1(4) - \gamma_{1,2} \bmod 5 = 2 - 2 \bmod 5 = 0 \end{cases}. \tag{36}$$

$$\begin{cases} m_{2,1} = x_2(3) - \gamma_{2,1} \bmod 5 = 0 - 4 \bmod 5 = 1 \\ m_{2,2} = x_2(4) - \gamma_{2,2} \bmod 5 = 3 - 2 \bmod 5 = 1 \end{cases}. \tag{37}$$

$$\begin{cases} m_{3,1} = x_3(3) - \gamma_{3,1} \bmod 5 = 1 - 2 \bmod 5 = 4 \\ m_{3,2} = x_3(4) - \gamma_{3,2} \bmod 5 = 0 - 3 \bmod 5 = 2 \end{cases} \tag{38}$$

The recovered secret segments are $m_1 = (4, 0),\ m_2 = (1, 1), m_3 = (4, 2)$, which are the same as embedded.

3. (k, n)-Threshold Secret Image Sharing Scheme Based on Linear Model

In this section, the (k, n)-threshold secret sharing scheme discussed above is applied to the platform of digital image transmission. By using the proposed secret sharing scheme, we can hide secret data in three distinct meaningful images and produce three shadows with imperceptible changes. While the shadow images are transmitted to n different participants, the secret data can be recovered only if more than k participants share their shadows. We present our scheme with the following subsections: system overview, setup, shadow image generation, secret data extraction with authentication, and a demonstration.

3.1. System Overview

The flowchart of the proposed scheme is shown in Figure 1. In the proposed scheme, secret data S are converted into a sequence of p-ary digits first. Then, this sequence is divided into segments denoted by $S = \left\{ m_j \middle| m_j = \left(m_{j,1},\ m_{j,2}, \cdots, m_{j,k} \right)^T \in z_p^k, j = 1, 2, \cdots, l \right\}$, where l is the total number of secret segments. We randomly choose n distinct meaningful cover images $I_1, I_2, \ldots, I_n$ with size $w \times h > l$. A vector sequence is produced from the cover images and denoted by $I_{vec} = \left\{ y_j \middle| y_j = \left(y_j(1), y_j(2), \ldots, y_j(n) \right)^T, j = 1, 2, \cdots, w \times h \right\}$, where each vector y_j is a collection of n pixel values retrieved from corresponding positions of the n images. To embed secret data, we apply the proposed (k, n)-threshold secret sharing scheme to generate a vector $x_j = \left(x_j(1), x_j(2), \ldots x_j(n) \right)^T$ of n digits using a secret segment m_j of k digits. After all secret segments are converted, the resulting vector sequence $X = \{ x_j | j = 1, 2, \cdots, l \}$, is applied to modulate $I_{vec} = \{ y_j | j = 1, 2, \cdots, w \times h \}$ and obtain the sequence $\hat{I}_{vec} = \{ \hat{y}_j | j = 1, 2, \cdots, w \times h \}$, correspondingly. Finally, the secret image shadows are generated by distributing the modulated sequence back into n meaningful images $\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_n$ sized $w \times h$. In the secret data recovery phase, at least k trustworthy shadows should be available. These k shadows are rearranged into a vector sequence with missing entries $\hat{I}'_{vec} = \left\{ \hat{y}'_j \middle| \hat{y}'_j = \left(\hat{y}'_j(1), \hat{y}'_j(2), \ldots, \hat{y}'_j(k) \right)^T, j = 1, 2, \cdots, w \times h \right\}$. Then, a sequence of vectors with missing entries $\hat{X}' = \left\{ \hat{x}'_j \middle| \hat{x}'_j = \left(\hat{x}'_j(1), \hat{x}'_j(2), \ldots, \hat{x}'_j(k) \right)^T, j = 1, 2, \cdots, l \right\}$ are

demodulated. The missing entries can be recovered by using Equations (11) and (12). If there are suspicious participants, the one who provides a share that mismatches the reproduced shadow is a cheater. Finally, the secret data can be recovered using Equations (20) and (21).

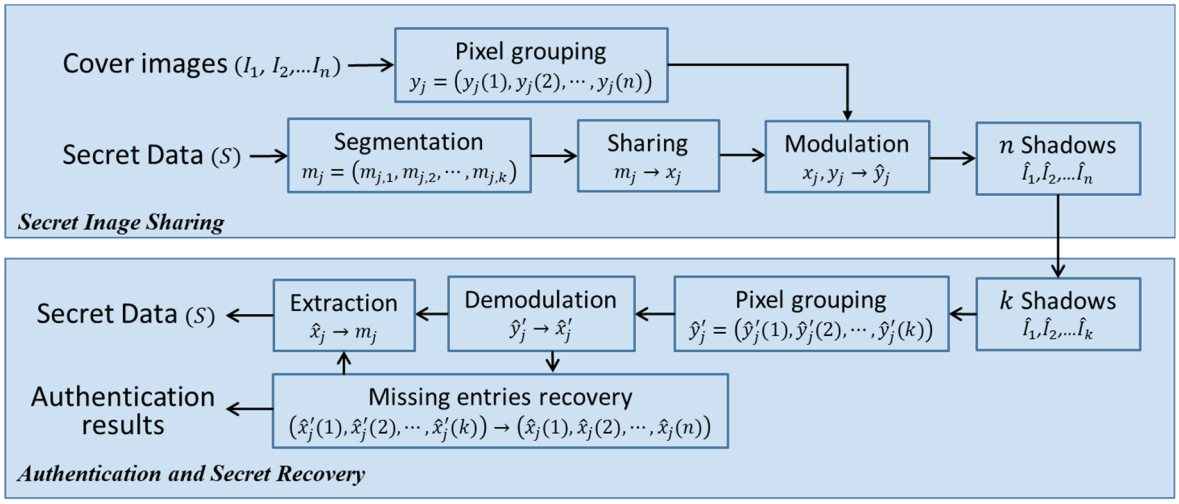


Figure 1. Flowchart of the proposed scheme.

3.2. Setup

The dealer decides the number of the participants n, the threshold k, and the prime number p. He/she selects $(n-k)$ linearly independent equations of n variables over the field z_p

$$\pi_i : a_{i,1}x_1 + a_{i,2}x_2 + \cdots + a_{i,n}x_n = 0 \bmod p, \text{ where } i = 1, 2, \cdots, (n-k). \tag{39}$$

We can rewrite them in matrix form as

$$Ax = b \bmod p. \tag{40}$$

Find a set of linearly independent homogeneous solutions $\{\alpha_1, \alpha_2, \cdots, \alpha_k\}$ and a particular solution β by applying Equations (5)–(9).

3.3. Shadow Image Generation

Let us assume that the secret data to be embedded are denoted as S and the n distinct meaningful cover images with size $w \times h$ are denoted as $I_1, I_2, \cdots, I_n$. The shadow images can be generated by the following steps.

Step 1. Rearrange the cover images $I_1, I_2, \cdots, I_n$ into vector sequence as

$$I_{vec} = \left\{ y_j \middle| y_j = \left(y_j(1), y_j(2), \ldots y_j(n)\right)^T, j = 1, 2, \cdots, w \times h \right\}. \tag{41}$$

Step 2. Convert the secret data stream S into a sequence of p-ary digits and divide them into segments denoted by $S = \left\{ m_j \middle| m_j = \left(m_{j,1}, m_{j,2}, \cdots, m_{j,k}\right)^T \in z_p^k, j = 1, 2, \cdots, l \right\}$, where $l \leq w \times h$ is the total number of secret segments.

Step 3. Apply $m_j = (m_{j,1}, m_{j,2}, \cdots, m_{j,k})^T, j = 1, 2, \cdots, l$ to generate vectors $x_j = \left(x_j(1), x_j(2), \ldots x_j(n)\right)^T, j = 1, 2, \cdots, l$ using Equation (17). Compute the residual vectors of I_{vec} by

$$\overline{x}_j = \left(\overline{x}_j(1), \overline{x}_j(2), \ldots \overline{x}_j(n)\right)^T \left(y_j(1), y_j(2), \ldots y_j(n)\right)^T \bmod p, \text{where } j = 1, 2, \cdots, l. \tag{42}$$

Compute the stego vector sequence $\hat{I}_{vec} = \left\{\hat{y}_j \middle| \hat{y}_j = \left(\hat{y}_j(1), \hat{y}_j(2), \ldots, \hat{y}_j(n)\right)^T, j = 1, 2, \cdots, l\right\}$ by

$$\hat{y}_j(i) = \begin{cases} y_j(i) + \left[\left(x_j(i) - \overline{x}_j(i)\right) \bmod p\right], \text{ if } \left(x_j(i) - \overline{x}_j(i)\right) \bmod p < p/2\,, \\ y_j(i) - \left[\left(\overline{x}_j(i) - x_j(i)\right) \bmod p\right], \text{ otherwise.} \end{cases} \tag{43}$$

When overflow/underflow is encountered, do

$$\hat{y}_j(i) = \begin{cases} \hat{y}_j(i) + p, \; if \; \hat{y}_j(i) < 0, \\ \hat{y}_j(i) - p, \; if \; \hat{y}_j(i) > 255. \end{cases} \tag{44}$$

Append the rest of the unmodulated vectors $y_j, j = l+1, l+2, \cdots, w \times h$ to constitute $\hat{I}_{vec} = \left\{\hat{y}_j \middle| j = 1, 2, \cdots, w \times h\right\}$.

Step 4. Distribute the modulated sequence $\hat{I}_{vec} = \left\{\hat{y}_j \middle| j = 1, 2, \cdots, w \times h\right\}$ back into n meaningful shadow images $\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_n$ sized $w \times h$.

3.4. Secret Data Extraction with Authentication

Suppose the n meaningful shadow images $\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_n$ sized $w \times h$ are transmitted to n participants and now k faithful participants provide their shadow images to a combiner. The combiner can recover secret data and authenticate the rest shadows by the following steps. Without loss of generality, let us suppose that the k trustworthy shadows are $\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_k$.

Step 1. Convert k shadow images $\hat{I}_1, \hat{I}_2, \ldots, \hat{I}_k$ into vectors of k-tuples as

$$\hat{I}'_{vec} = \left\{\hat{y}'_j \middle| \hat{y}'_j = \left(\hat{y}'_j(1), \hat{y}'_j(2), \ldots, \hat{y}'_j(k)\right)^T, j = 1, 2, \cdots, w \times h\right\}. \tag{45}$$

Step 2. Compute the residual vectors

$$\hat{x}'_j = \left(\hat{y}'_j(1), \hat{y}'_j(2), \ldots, \hat{y}'_j(k)\right)^T \bmod p, \text{where } j = 1, 2, \cdots, l. \tag{46}$$

Step 3. Substitute each vector $\hat{x}'_j = \left(\hat{x}'_j(1), \hat{x}'_j(2), \ldots, \hat{x}'_j(k)\right)^T$ into Equations (11) and (12), the missing entries can be recovered to obtain its corresponding complete solution vector $\hat{x}_j = \left(\hat{x}_j(1), \hat{x}_j(2), \ldots, \hat{x}_j(n)\right)^T$.

Step 4. Convert a doubtful shadow $\hat{I}_r$ into sequence $\hat{y}^*_j(r), j = 1, 2, \ldots, w \times h$. Then, compute its residual sequence $\hat{x}^*_j(r), j = 1, 2, \ldots, l$. If $\hat{x}^*_j(r) = \hat{x}_j(r), \; j = 1, 2, \ldots, l$, the authentication is passed; else, it is a tampered shadow.

Step 5. Extract the secret data by applying $\hat{x}_j = \left(\hat{x}_j(n-k+1), \ldots, \hat{x}_j(n)\right)^T$, $j = 1, 2, \ldots, l$ to Equation (21).

3.5. Demonstration

We use the example model given in Section 2.3 to demonstrate how the proposed secret sharing scheme can be applied to image transmission applications. That is, $k = 2$, $n = 4$, $p = 5$, and the two linearly independent equations with four variables are

$$\pi_1 : x_1 + 3x_2 + x_3 + 2x_4 = 1 \bmod 5 \text{ and } \pi_2 : 2x_1 + 4x_2 + 0x_3 + x_4 = 3 \bmod 5. \tag{47}$$

As analyzed in Section 2.3.1, $\alpha_1 = (2,\ 4, 1, 0)^T$ and $\alpha_2 = (0,\ 1, 0, 1)^T$ are two linearly independent homogeneous solutions, and $\beta = (0, 2, 0, 0)^T$ is a particular solution.

Assume that $m_1 = (2, 1)$ and $m_2 = (3, 0)$ are two 5-ary secret segments, and the four cover images are $I_1 = (135, 136)$, $I_2 = (161, 162)$, $I_3 = (201, 200)$, and $I_4 = (55, 58)$. Two random segments $\gamma_1 = (4, 0)$ and $\gamma_2 = (1, 2)$ are generated by the data hiding key.

For the secret segments m_1 and m_2, we generate their corresponding solution vectors by using Equation (17) and obtain

$$x_1 = F(m_1) = (2+4)\begin{pmatrix} 2 \\ 4 \\ 1 \\ 0 \end{pmatrix} + (1+0)\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 1 \\ 1 \end{pmatrix}, \quad (48)$$

$$x_2 = F(m_2) = (3+1)\begin{pmatrix} 2 \\ 4 \\ 1 \\ 0 \end{pmatrix} + (0+2)\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 3 \\ 0 \\ 4 \\ 2 \end{pmatrix}. \quad (49)$$

The residual vectors are

$$\overline{x}_1 = (135, 161, 201, 55)^T \bmod 5 = (0, 1, 1, 0)^T, \quad (50)$$

$$\overline{x}_2 = (136, 162, 200, 58)^T \bmod 5 = (1, 2, 0, 3)^T. \quad (51)$$

Then, the stego vectors are

$$\hat{y}_1 = \begin{pmatrix} 135 + (2-0)\text{mod}5 \\ 161 + (2-1)\text{mod}5 \\ 201 + (1-1)\text{mod}5 \\ 55 + (1-0)\text{mod}5 \end{pmatrix} = \begin{pmatrix} 137 \\ 162 \\ 201 \\ 56 \end{pmatrix}, \hat{y}_2 = \begin{pmatrix} 136 + (3-1)\text{mod}5 \\ 162 - (2-0)\text{mod}5 \\ 200 - (0-4)\text{mod}5 \\ 58 - (3-2)\text{mod}5 \end{pmatrix} = \begin{pmatrix} 138 \\ 160 \\ 199 \\ 57 \end{pmatrix}. \quad (52)$$

Finally, the four shadow images are

$$\hat{I}_1 = (137, 138),\ \hat{I}_2 = (162, 160), \hat{I}_3 = (201, 199), \hat{I}_4 = (56, 57). \quad (53)$$

The shares $\hat{I}_1,\ \hat{I}_2,\ \hat{I}_3,\ \hat{I}_4$ are distributed to the four participants, respectively.

If a combiner obtains trustworthy shadow images $\hat{I}_1 = (137, 138)$ and $\hat{I}_2 = (162, 160)$, he/she first calculates the residual vectors as

$$\hat{x}_1' = (137, 162)^T \text{mod } 5 = (2, 2)^T, \hat{x}_2' = (138, 160)^T \text{mod } 5 = (3, 0)^T. \quad (54)$$

Then, the missing entries can be recovered using Equations (11,12). For the current case, manipulations are demonstrated in Equation (31) as

$$\begin{cases} x_1(3) = -2 \times 2 - 0 \times 2 + 0 \bmod 5 = 1 \\ x_1(4) = -2 \times 2 - 4 \times 2 + 3 \bmod 5 = 1 \end{cases}. \quad (55)$$

$$\begin{cases} x_2(3) = -2 \times 3 - 0 \times 0 + 0 \bmod 5 = 4 \\ x_2(4) = -2 \times 3 - 4 \times 0 + 3 \bmod 5 = 2 \end{cases}. \quad (56)$$

To extract secret data, the random segments $\gamma_1 = (4, 0)$ and $\gamma_2 = (1, 2)$ are generated by the shared data hiding key first. The secret data can be extracted using Equation (21) as

$$m_1 = (1-4, 1-0) = (2, 1), m_2 = (4-1, 2-2) = (3, 0). \quad (57)$$

4. Experimental Results and Discussions

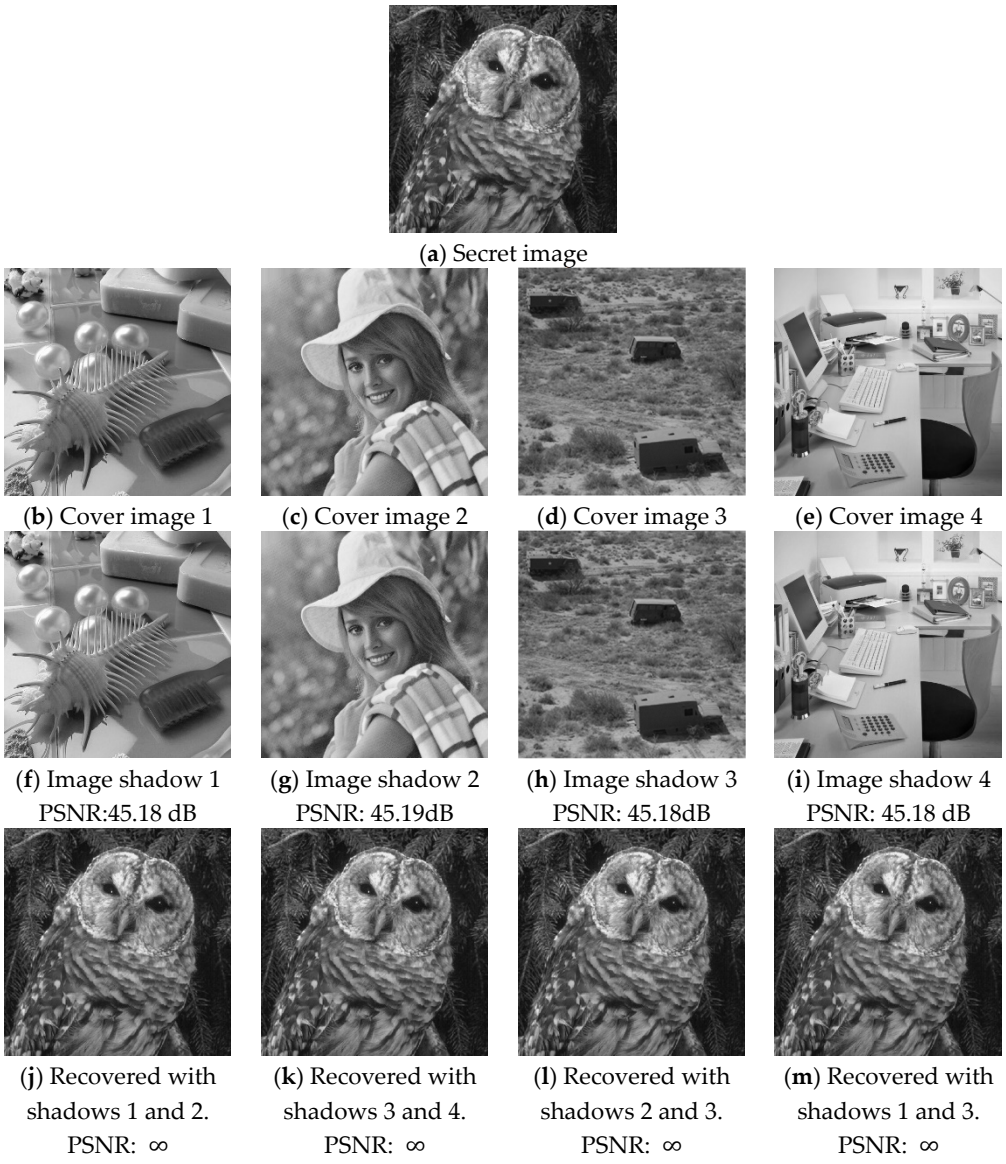
In this section, we use two secret image sharing schemes with different combinations of (k, n) values to demonstrate the applicability of the proposed approach. Performance evaluation based on the embedding capacity and the visual quality of image shadows are given. The security level of the proposed scheme is also analyzed.

4.1. Demonstration of Applicability

In this subsection, we implement a (2,4)-threshold and a (3,5)-threshold secret image sharing schemes to verify the applicability and the generalizability of the proposed approach.

4.1.1. (2,4)-Threshold Secret Image Sharing Scheme

In our first implementation, the parameter settings are $k = 2$, $n = 4$, $p = 5$, and the two linearly independent equations given in Equation (47) are applied. The secret data are assumed to be a secret image, as shown in Figure 2a, which is a grayscale image sized 384×384. The four distinct cover images are shown in Figure 2b–e, which are all grayscale images sized 512×512. All pixel values in our experiments are recorded in eight bits.



(**a**) Secret image

(**b**) Cover image 1 (**c**) Cover image 2 (**d**) Cover image 3 (**e**) Cover image 4

(**f**) Image shadow 1 PSNR:45.18 dB (**g**) Image shadow 2 PSNR: 45.19dB (**h**) Image shadow 3 PSNR: 45.18dB (**i**) Image shadow 4 PSNR: 45.18 dB

(**j**) Recovered with shadows 1 and 2. PSNR: ∞ (**k**) Recovered with shadows 3 and 4. PSNR: ∞ (**l**) Recovered with shadows 2 and 3. PSNR: ∞ (**m**) Recovered with shadows 1 and 3. PSNR: ∞

Figure 2. The experimental results of the (2,4)-threshold secret image sharing.

To fit the parameter settings, the pixels of the secret image are divided into groups of four pixels each. Thirty-two bits of each pixel group are converted into fourteen 5-ary digits and then grouped into seven secret segments of two digits. Each secret segment is applied to generate a solution vector of four entries and used to modulate four cover pixels. Therefore, four pixel-values of the secret image are distributed in fourteen pixels for each cover image. The generated image shadows are shown in Figure 2f–i. The changes with respect to their corresponding cover images are imperceptible. The peak-signal-to-noise-ratio (PSNR) given below the figures is defined by

$$\text{PSNR} = 10 \ \log_{10}\left(\frac{255^2}{e_{mse}}\right) \ (\text{dB}), \tag{58}$$

where e_{mse} denotes the mean-square-error between the cover image I_j and the shadow image $\hat{I}_j$ defined by

$$e_{mse} = \frac{1}{w \times h} \sum_{j=1}^{w \times h} \left(\hat{I}_j - I_j\right)^2. \tag{59}$$

The recovered secret images by using different combinations of image shadow pairs are given in Figure 2j–m. In all cases, the secret image can be perfectly recovered.

4.1.2. (3,5)-Threshold Secret Image Sharing Scheme

In our second implementation, the parameter settings are $k = 3$, $n = 5$, and $p = 5$, and the two linearly independent equations are given by

$$\pi_1 : 2x_1 + 2x_2 + x_3 + 3x_4 + x_5 = 1 \bmod 5. \pi_2 : 3x_1 + 4x_2 + 3x_3 + 4x_4 + x_5 = 3 \bmod 5. \tag{60}$$

The solution space H of π_1 and π_2 over z_5 is formulated as

$$H = c_1 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 4 \\ 1 \end{pmatrix} + c_2 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 3 \\ 4 \end{pmatrix} + c_3 \begin{pmatrix} 0 \\ 0 \\ 1 \\ 3 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \\ 2 \\ 0 \end{pmatrix} \bmod 5, \tag{61}$$

where $c_1, c_2, c_3 \in z_5$. Then, solution vector generation function is given by

$$x_j = F(m_j) = (m_{j,1} + \gamma_{j,1}) \begin{pmatrix} 1 \\ 0 \\ 0 \\ 4 \\ 1 \end{pmatrix} + (m_{j,2} + \gamma_{j,2}) \begin{pmatrix} 0 \\ 1 \\ 0 \\ 3 \\ 4 \end{pmatrix} + (m_{j,3} + \gamma_{j,3}) \begin{pmatrix} 0 \\ 0 \\ 1 \\ 3 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \\ 2 \\ 0 \end{pmatrix} \bmod p. \tag{62}$$

The secret image sized 512×438 is given in Figure 3a; the five distinct cover images sized 512×512 are given in Figure 3b–f. To fit the parameter settings, pixels of the secret image are divided into groups of six pixels each. Forty-eight bits of each pixel group are converted into twenty-one 5-ary digits, and then grouped into seven secret segments of three digits. Each secret segment is applied to generate a solution vector of five entries and is used to modulate five cover pixels. Therefore, six pixel-values of the secret image are distributed in thirty-five pixels for each cover image. The generated image shadows are shown in Figure 3g–k. The changes with respect to their corresponding cover images are again imperceptible. The recovered secret images using different combinations of image shadows are given in Figure 3l–p. In all cases, the secret image can be perfectly recovered.

4.2. Performance Evaluation

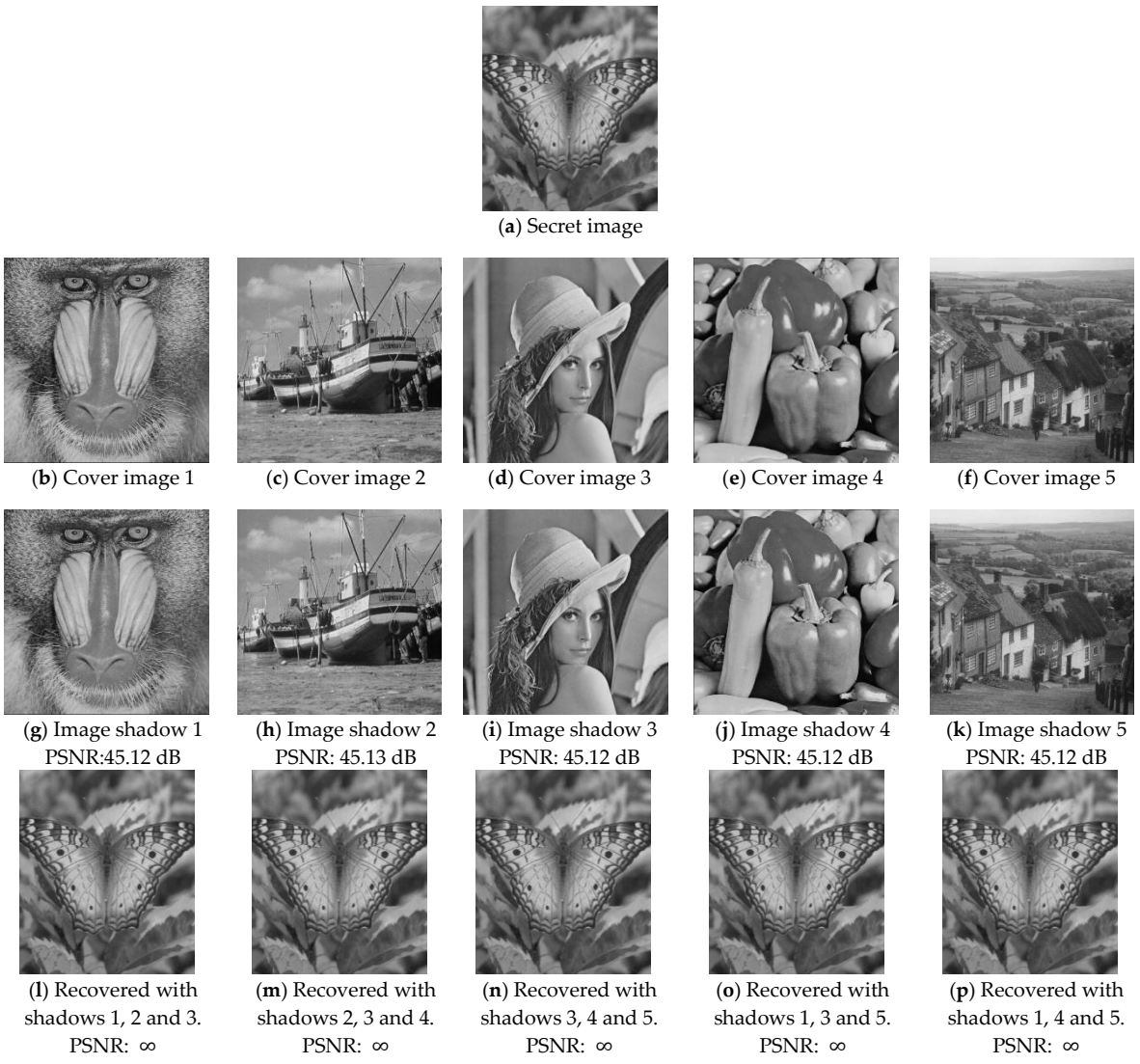
In our (k, n) threshold secret image sharing scheme, k digits of p-ary numbers are embedded in n pixels of each image shadow. The embedding capacity (EC) can therefore be measured in bits per pixel (bpp) of an image shadow by

$$\text{EC} = \log_2 p^k \text{ (bpp)}. \tag{63}$$

When the total number of shares n increases, the total amount of image shadows to be transmitted also increases. However, an increase in the number of shares can improve the flexibility of secret recovery. Recall that any k shares among the n shadows can perfectly recover the secret.

For each cover pixel, Equation (43) always modulates the pixel value with an integer deviation within the range of $[-(p-1)/2 : (p-1)/2]$. Suppose the deviation value is randomly distributed, the mean-square-error e_{mse} of modulation distortion can be estimated by

$$e_{mse} = \frac{1}{p}\sum_{d=-(p-1)/2}^{(p-1)/2} d^2 = \frac{(p-1)(p+1)}{12}. \tag{64}$$



(**a**) Secret image

(**b**) Cover image 1 (**c**) Cover image 2 (**d**) Cover image 3 (**e**) Cover image 4 (**f**) Cover image 5

(**g**) Image shadow 1 PSNR:45.12 dB (**h**) Image shadow 2 PSNR: 45.13 dB (**i**) Image shadow 3 PSNR: 45.12 dB (**j**) Image shadow 4 PSNR: 45.12 dB (**k**) Image shadow 5 PSNR: 45.12 dB

(**l**) Recovered with shadows 1, 2 and 3. PSNR: ∞ (**m**) Recovered with shadows 2, 3 and 4. PSNR: ∞ (**n**) Recovered with shadows 3, 4 and 5. PSNR: ∞ (**o**) Recovered with shadows 1, 3 and 5. PSNR: ∞ (**p**) Recovered with shadows 1, 4 and 5. PSNR: ∞

Figure 3. The experimental results of the (3,5)-threshold secret image sharing.

When the pixels of a cover image are not fully exploited to embed secret data, we define the payload ratio as

$$r = \frac{N_e}{N_t}, \tag{65}$$

where N_e and N_t denote the number of pixels exploited and the number of total pixels, respectively. For the two schemes implemented in Section 4.1, the payload ratios can be calculated by

$$r_{(2,4)} = [(384 \times 384)/4(\text{groups}) \times (7 \times 4)(\text{pixels/group})]/(512 \times 512 \times 4) = 0.9844, \tag{66}$$

$$r_{(3,5)} = [(512 \times 438)/6(\text{groups}) \times (7 \times 5)(\text{pixels/group})]/(512 \times 512 \times 5) = 0.9980. \tag{67}$$

Under the payload ratio r, the mean-square-error e_{mse} is proportionally corrected to $r \times e_{mse}$. Thus, the expected PNSR value of an image shadow with a payload ratio r is given by

$$\mathrm{PSNR}(r) = 10\log_{10}\left(\frac{255^2}{r \times e_{mse}}\right) = 10\log_{10}\left(\frac{12 \times 255^2}{r(p-1)(p+1)}\right)(\mathrm{dB}). \tag{68}$$

The expected PSNR values with respect to different payload ratios and prime numbers are listed in Table 1. The corresponding evolution curves are plotted in Figure 4, where the curves for four different prime numbers are given. According to the theoretic analysis, the expected PSNR values of the two implemented cases are 45.1889 dB and 45.1290 dB, respectively, which sharply coincide with our experimental values.

Table 1. The expected PSNR value (dB) with specific ratio, r

	r									
p	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
3	59.89	56.88	55.12	53.87	52.90	52.11	51.44	50.86	50.35	49.89
5	55.12	52.11	50.35	49.10	48.13	47.348	46.67	46.09	45.58	45.12
7	52.11	49.10	47.34	46.09	45.12	44.33	43.66	43.08	42.57	42.11
11	48.13	45.12	43.36	42.11	41.14	40.35	39.68	39.10	38.59	38.13

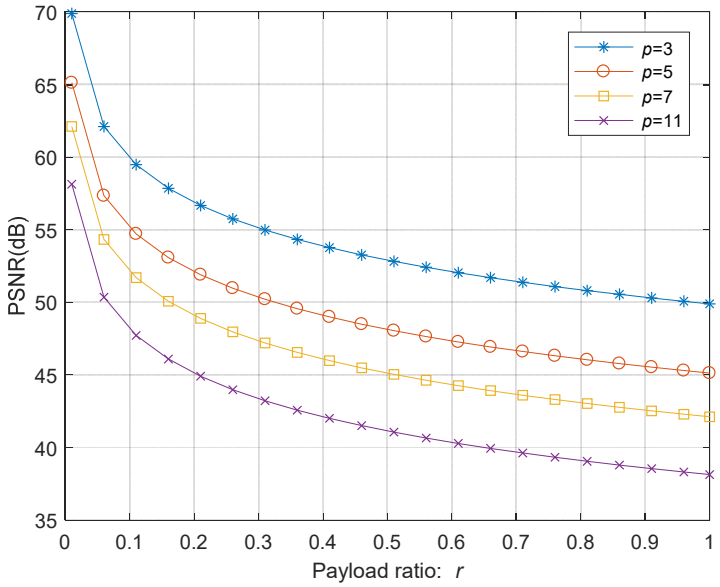


Figure 4. The expected PSNR value of an image shadow with respect to the payload ratio.

4.3. Security Analysis

Various steganalysis techniques [29–32] have been proposed to detect the existence of secret information in a digital image. The general idea of steganalysis is to detect the abnormal statistical feature of a doubtful image. However, there are various types of data hiding methods for images. To successfully detect the embedding of secret, special steganalysis techniques are devised to deal with each target steganography. For example, the RS steganalysis [29] is applied to detect the LSB substitution. For data hiding in encrypted images, people usually use pixel value entropy or gray level histogram to analyze the existence of a secret message.

The data hiding method of the proposed secret sharing scheme is essentially based on the modulus function. Two steganalysis tools, relative entropy [30] and pixel-value differencing [31,32], are suitable for testing the security level of our scheme. The first tool is the relative entropy proposed by Cachin [30]. To measure the difference between cover

image I_c and image shadow I_s, we accumulate their probability distributions $P_c(x)$ and $P_s(x)$, respectively. Then, their relative entropy $D(I_c, I_s)$ is calculated by

$$D(I_c, I_s) = \sum_{x=0}^{255} P_c(x) \log_2 \left[\frac{P_c(x)}{P_s(x)} \right]. \tag{69}$$

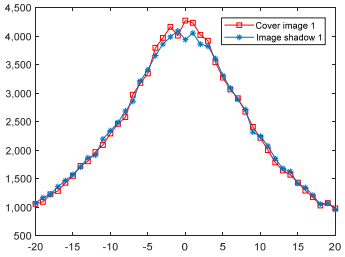
If $D(I_c, I_s) \leq \varepsilon$, by definition, the data hiding scheme is ε-secure against passive attacking. A smaller value of ε means a better security level. We apply the relative entropy measure to our (3,5)-threshold secret image sharing scheme with $p = 5$. The experimental results are listed in Table 2, where the relative entropy values for half embedded and fully embedded cases are given. All values listed in the table are very close to zero, which implies the image shadows of the proposed scheme are robust under a passive attack.

Table 2. Relative entropy values under half embedded and fully embedded cases.

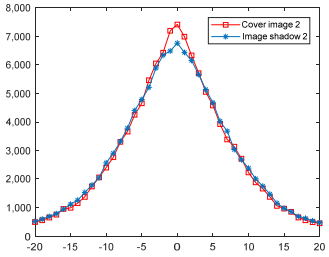
Cover Images	Half Embedded			Fully Embedded		
	$E(I_c)$	$E(I_s)$	$D(I_c,I_s)$	$E(I_c)$	$E(I_s)$	$D(I_c,I_s)$
Baboon	7.3579	7.3543	0.0049	7.3579	7.3415	0.0185
Boat	7.1914	7.2050	0.0052	7.1914	7.2129	0.0184
Lena	7.4455	7.4482	0.0007	7.4455	7.4510	0.0014
Peppers	7.5944	7.5978	0.0006	7.5944	7.5998	0.0021
Goldhill	7.4778	7.4839	0.0036	7.4778	7.4829	0.0058

The second steganalysis applied is the pixel-value differencing analysis [31,32]. The neighboring pixel-values of a natural image are highly continuous. The continuity is disrupted by most data hiding schemes. If the pixel-value differencing histogram (PDH) of an image is unusually flat, it is highly doubtful. To investigate the security level of the proposed scheme, we apply the PDH analysis to a set of five image shadows produced by our (3,5)-threshold secret image sharing scheme with $p = 5$. The secret image given in Figure 5a is embedded into five distinct cover images. The PDHs of the five pairs of cover images and their corresponding image shadows are plotted in Figure 5b–f. The PDH curves of image shadows are very close to their corresponding cover images, which indicates the proposed scheme is secure under PDH steganalysis.

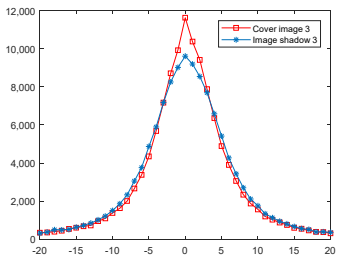
(**a**) Secret image



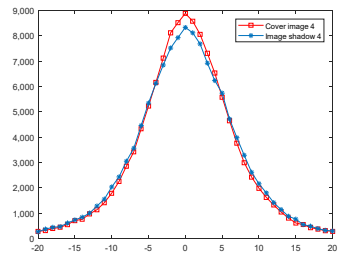
(**b**) Baboon



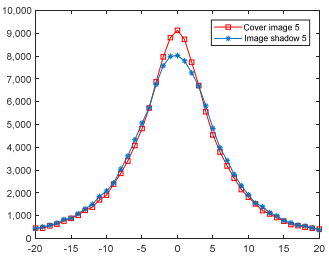
(**c**) Boat



(**d**) Lena



(**e**) Peppers



(**f**) Goldhill

Figure 5. The PVD histogram analysis between the cover images and image shadows.

5. Conclusions

In this paper, we propose a general (k, n)-threshold secret image sharing scheme with meaningful image shadows. We first present a (k, n)-theshold secret sharing scheme based on a non-full rank linear model. It is then applied to the platform of image transmission system. Using n secret shares to modulate n distinct cover images, we can produce n image shadows. A combiner can completely recover the secret message by collecting at least k image shadows.

To demonstrate the applicability to any combination of (k, n) parameter values, we use two example models (2,4) and (3,5) to demonstrate our scheme. Experimental results confirm the applicability of the proposed scheme. Moreover, the embedding capacity and visual quality of image shadows are satisfactory. Embedding capacity and theoretic image quality under different parameter settings are also analyzed. Finally, we use two steganalysis tools to show the security level of our scheme.

The proposed scheme is based on the pixel-value modification in the spatial domain, which is suitable for digital images of a bitmap format. Our future work will focus on the implementation of the general secret sharing scheme to different cover media, such as JPEG images, encrypted images, and QR code images, which are more commonly applied in Internet applications.

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

Funding: This research was funded by the Ministry of Science and Technology of Taiwan, grant number MOST 110-2221-E-507-003.

Data Availability Statement: Not applicable.

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

Appendix A

The elementary row operations over the field z_p include row switching, row multiplication, and row addition as in conventional linear algebra, except that each resulting integer number should be modulated by p as

$$\forall a, b \in z_p,\ a + ba + b \bmod p \forall a, b \in z_p,\ a \times ba \times b \bmod p \tag{A1}$$

For the addition operation, if $a, b \in z_p$ and $a + b = 0 \bmod p$, then $-a = b \bmod p$; for the multiplication operation, if $a, b \in z_p$ and $a \times b = 1 \bmod p$, then $a^{-1} = b \bmod p$. Next, we use the derivation of Equation (23) in Section 2.3.1 to demonstrate the details of the elementary row operations. The two linearly independent equations are

$$\pi_1 : 1x_1 + 3x_2 + 1x_3 + 2x_4 = 1 \bmod 5, \pi_2 : 2x_1 + 4x_2 + 0x_3 + 1x_4 = 3 \bmod 5. \tag{A2}$$

That is, $A = \begin{pmatrix} 1 & 3 & 1 & 2 \\ 2 & 4 & 0 & 1 \end{pmatrix}$ and $b = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$. The corresponding augmented matrix is

$$(A|b) = \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 2 & 4 & 0 & 1 & 3 \end{array}\right) \tag{A3}$$

To transfer the first two columns into an identity matrix, first multiply row 1 by 3 and add to row 2 as

$$\left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 2 & 4 & 0 & 1 & 3 \end{array}\right) \overset{R_2=R_2+3R_1}{\rightarrow} \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 5 & 13 & 3 & 7 & 6 \end{array}\right) \bmod 5 = \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 3 & 3 & 2 & 1 \end{array}\right), \tag{A4}$$

where $R_2 = R_2 + 3R_1$ above the operation arrow denotes the operation detail. Then, scale row 2 by 2 as

$$\left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 3 & 3 & 2 & 1 \end{array}\right) \overset{R_2=2R_2}{\rightarrow} \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 6 & 6 & 4 & 2 \end{array}\right) \bmod 5 = \left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right). \tag{A5}$$

Finally, multiply row 2 by 2 and add to row 1 as

$$\left(\begin{array}{cccc|c} 1 & 3 & 1 & 2 & 1 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right) \overset{R_1=R_1+2R_2}{\rightarrow} \left(\begin{array}{cccc|c} 1 & 5 & 3 & 10 & 5 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right) \bmod 5 = \left(\begin{array}{cccc|c} 1 & 0 & 3 & 0 & 0 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right). \tag{A6}$$

Thus, the final row simplest augmented matrix over the field z_5 is

$$(A|b) \rightarrow \left(\begin{array}{cccc|c} 1 & 0 & 3 & 0 & 0 \\ 0 & 1 & 1 & 4 & 2 \end{array}\right). \tag{A7}$$

References

1. Pareek, N.K.; Patidar, V.; Sud, K.K. Image encryption using chaotic logistic map. *Image Vis. Comput.* **2006**, *24*, 926–934. [CrossRef]
2. Zhu, Z.L.; Zhang, W.; Wong, K.W.; Yu, H. A chaos-based symmetric image encryption scheme using a bit-level permutation. *Inf. Sci.* **2011**, *181*, 1171–1186. [CrossRef]
3. Chan, K.C.; Cheng, L.M. Hiding data in images by simple LSB substitution. *Pattern Recognit.* **2004**, *37*, 469–474. [CrossRef]
4. Zhang, X.; Wang, S. Efficient steganographic embedding by exploiting modification direction. *IEEE Commun. Lett.* **2006**, *10*, 781–783. [CrossRef]
5. Tian, J. Reversible data embedding using a difference expansion. *IEEE Trans. Circuits Syst. Video Technol.* **2003**, *13*, 890–896. [CrossRef]
6. Ni, Z.; Shi, Y.Q.; Ansari, N.; Su, W. Reversible data hiding. *IEEE Trans. Circuits Syst. Video Technol.* **2006**, *16*, 354–361. [CrossRef]
7. Chang, C.C. Neural Reversible steganography with long short-term memory. *Secur. Commun. Netw.* **2021**, *2021*, 5580272. [CrossRef]
8. Chang, C.C.; Li, C.T.; Shi, Y.Q. Privacy-Aware reversible watermarking in cloud computing environments. *IEEE Access* **2018**, *6*, 70720–70733. [CrossRef]
9. Naor, M.; Shamir, A. Visual cryptography. *Lect. Notes Comput. Sci.* **1995**, *950*, 1–12. [CrossRef]
10. Nakajima, M.; Yamaguchi, Y. Extended visual cryptography for natural images. *WSCG* **2002**, *10*, 303–310. [CrossRef]
11. Patil, S.; Rao, J. Extended visual cryptography for color shares using random number generators. *Int. J. Adv. Res. Comput. Commun. Eng.* **2012**, *1*, 399–410.
12. Blundo, C.; De Santis, A.; Naor, M. Visual cryptography for grey level images. *Inf. Process. Lett.* **2000**, *75*, 255–259. [CrossRef]
13. 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]
14. Ulutas, M.; Ulutas, G.; Nabiyev, V.V. Medical image security and EPR hiding using Shamir's secret sharing scheme. *J. Syst. Softw.* **2011**, *84*, 341–353. [CrossRef]
15. Charoghchi, S.; Mashhadi, S. Three (t,n)-secret image sharing schemes based on homogeneous linear recursion. *Inf. Sci.* **2021**, *552*, 220–243. [CrossRef]
16. Liu, Y.; Yang, C. Scalable secret image sharing scheme with essential shadows. *Signal Process. Image Commun.* **2017**, *58*, 49–55. [CrossRef]
17. Yan, X.; Li, J.; Pan, Z.; Zhong, X.; Yang, G. Multiparty verification in image secret sharing. *Inf. Sci.* **2021**, *562*, 475–490. [CrossRef]
18. 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]
19. 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. [CrossRef]
20. Chang, C.-C.; Kieu, T.; Chou, Y.-C. Reversible data hiding scheme using two steganographic images. In Proceedings of the TENCON 2007–2007 IEEE Region 10 Conference, Taipei, Taiwan, 30 October–2 November 2007; pp. 1–4. [CrossRef]
21. Chen, S.; Chang, C.C. Reversible data hiding based on three shadow images using rhombus magic matrix. *J. Vis. Commun. Image Represent.* **2021**, *76*, 103064. [CrossRef]
22. Qin, C.; Chang, C.C.; Hsu, T.J. Reversible data hiding scheme based on exploiting modification direction with two steganographic images. *Multimed. Tools Appl.* **2015**, *74*, 5861–5872. [CrossRef]
23. Lin, P.Y.; Lee, J.S.; Chang, C.C. Distortion-free secret image sharing mechanism using modulus operator. *Pattern Recognit.* **2009**, *42*, 886–895. [CrossRef]
24. Lin, P.Y.; Chan, C.S. Invertible secret image sharing with steganography. *Pattern Recognit. Lett.* **2010**, *31*, 1887–1893. [CrossRef]
25. Yadav, M.; Singh, R. Essential secret image sharing approach with same size of meaningful shares. *Multimed. Tools Appl.* **2021**. [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. Chang, C.C.; Chen, Y.H.; Wang, H.C. Meaningful secret sharing technique with authentication and remedy abilities. *Inf. Sci.* **2011**, *181*, 3073–3084. [CrossRef]
28. Gao, K.; Horng, J.H.; Chang, C.C. An authenticatable (2, 3) secret sharing scheme using meaningful share images based on hybrid fractal matrix. *IEEE Access* **2021**, *9*, 50112–50125. [CrossRef]
29. Fridrich, J.; Goljan, M.; Du, R. Reliable detection of LSB steganography in color and grayscale images. In Proceedings of the 2001 Workshop on Multimedia and Security: New Challenges, Ottawa, ON, Canada, 5 October 2001; pp. 27–30. [CrossRef]
30. Cachin, C. An information-theoretic model for steganography. *Lect. Notes Comput. Sci.* **1998**, *1525*, 306–318. [CrossRef]
31. Arabia, S. Pixel-Value Differencing Steganography: Attacks and Improvements. In Proceedings of the ICCIT 2012, Chittagong, Bangladesh, 22–24 December 2012; pp. 757–762.
32. Joo, J.C.; Lee, H.Y.; Bui, C.N.; Yoo, W.Y.; Lee, H.K. Steganalytic measures for the steganography using pixel-value differencing and modulus function. *Lect. Notes Comput. Sci.* **2008**, *5353 LNCS*, 476–485. [CrossRef]