mathematics

MDPI

Article

An Implementation of Image Secret Sharing Scheme Based on Matrix Operations

Zihan Ren, Peng Li * and Xin Wang

Department of Mathematics and Physics, North China Electric Power University, Baoding 071003, China; renzihan@ncepu.edu.cn (Z.R.); wstar@ncepu.edu.cn (X.W.)
* Correspondence: peng.li@ncepu.edu.cn

Abstract: The image secret sharing scheme shares a secret image as multiple shadows. The secret image can be recovered from shadow images that meet a threshold number. However, traditional image secret sharing schemes generally reuse the Lagrange's interpolation in the recovery stage to obtain the polynomial in the sharing stage. Since the coefficients of the polynomial are the pixel values of the secret image, it is able to recover the secret image. This paper presents an implementation of the image secret sharing scheme based on matrix operations. Different from the traditional image secret sharing scheme, this paper does not use the method of Lagrange's interpolation in the recovery stage, but first identifies the participants as elements to generate a matrix and calculates its inverse matrix. By repeating the matrix multiplication, the polynomial coefficients of the sharing stage are quickly derived, and then the secret image is recovered. By theoretical analysis and the experimental results, the implementation of secret image sharing based on matrix operation is higher than Lagrange's interpolation in terms of efficiency.

Keywords: image secret sharing; linear algebra; matrix operations; Lagrange's polynomials

MSC: 94A62

Citation: Ren, Z.; Li, P.; Wang, X. An Implementation of Image Secret Sharing Scheme Based on Matrix Operations. *Mathematics* **2022**, *10*, 864. https://doi.org/10.3390/math10060864

Academic Editor: Angel Martín-del-Rey

Received: 24 January 2022
Accepted: 6 March 2022
Published: 9 March 2022

1. Introduction

Secret sharing is an important research direction in the field of modern cryptography. The use of a secret sharing scheme to transmit or save the secret information can prevent excessive concentration of power on the one hand, and ensure the integrity, reliability and security of secret information on the other. Only when the attacker has a certain number of shared keys is it possible to recover the original secret information correctly, so the recovery of the original secret information is very difficult. Moreover, when some shared keys are lost or damaged due to subjective or objective factors, the remaining participants can still jointly recover the secret information through the shared keys they hold. Thus, the secret sharing technology has a wide range of applications in data security, key management, bank network management and other aspects.

A secret sharing scheme with a threshold of (k, n) means sharing a secret message into n shares, where any k or more shares can recover the secret. Shamir [1] first introduced this concept in 1979, in which a secret message is embedded as a constant term of a polynomial of order $k-1$, and random numbers are used for the other coefficients of the polynomial. Thien and Lin [2] proposed an image secret sharing scheme, (k, n)-SIS, based on Shamir's scheme in 2002. K secret pixels are used at a time embedded into the coefficients of a $(k-1)$ order polynomial in their scheme. In 2004, Lin and Tsai [3] used a steganography approach to embed the shadow image in [2] in a steganographic image, which improves the security of the scheme and has verifiable functionality. In 2007, Yang et al. [4], based on [3], used a hash function instead of parity to rearrange the positions of shared bits and authentication bits to improve the authentication capability and quality of steganographic images. Polynomial based image secret sharing schemes have attracted a

lot of attention, and new secret sharing algorithms based on polynomial sharing schemes have been developed.

The concept of an image secret sharing scheme with essential participants, like (t, s, k, n)-ESIS, was first proposed in 2013 by Li et al. [5]. It was designed to address the inconsistent power of participants in the practical application of image secret sharing schemes. Essential participants have higher levels of power than non-essential participants when performing image restoration. This concept has since been implemented by many researchers in different ways [6–10], such as polynomial differentiation in (k, n)-SIS to obtain new polynomials [8]. Considering that a dynamic threshold k has a greater role in some cases, Yuan et al. first proposed a polynomial-based image secret sharing scheme with variable thresholds in 2016 [11]. Liu has improved on this by allowing the threshold to be varied over a range of values [12].

In recent years, image secret sharing schemes with new functions have been researched. In 2016, Maroti Deshmukh et al. proposed a (n, n)-Multi Secret Image Sharing Scheme [13], which uses Boolean XOR and Modular Arithmetic to share n secret images as n shares. This idea has since been continuously improved. [14–19] In response to the verifiability of participant identity and shadow, researchers also proposed a verifiable ISS on the polynomial-based ISS [20–23]. In 2017, Adnan Gutub et al. proposed a counting-based secret sharing technique for multimedia applications [24]. By performing parallel counting on binary secret information, the sharing and recovery of secret information can be achieved quickly. This method provides a new idea for fast sharing of binary images.

For polynomial-based image secret sharing, it often uses Lagrange's interpolation when performing secret image recovery to reconstruct one polynomial and thus obtain the grayscale values of the original secret image. A simple way to implement Lagrange's interpolation and construct the polynomial is by solving a linear system of equations. Many researchers just simply adopted this method in the revealing process of the SIS scheme [1–3,5,8,10–12,16,17,19,22,23]. In practice, many secret image pixels and a large amount of data require solving many linear systems until all pixels are recovered. This requires a large amount of computation, which shows the disadvantage of low efficiency when encrypting larger images. Actually, these linear systems are only different in their constant terms. They have the same coefficient matrix and the coefficient matrix is invertible. This coefficient matrix is only determined by IDs of the participants involved in the revealing process.

This paper proposes a more efficient implementation method of SIS based on matrix operation: using matrix multiplication instead of Lagrange's interpolation based on the properties of polynomial coefficients. This not only greatly reduces the computational complexity but also improves the efficiency of secret image recovery.

The structure of this article is as follows. The classical secret image sharing scheme proposed by Thien and Lin is reviewed in Section 2. In Section 3, the implementation of the SIS scheme based on a matrix operation is proposed, and Section 4 shows the advantages of our method in efficiency compared with the SIS scheme with Lagrange's interpolation by experiments. Finally, concise conclusions are formulated in Section 5.

2. Thien and Lin's Secret Image Sharing Scheme

2.1. Secret Sharing Stage

In Thien and Lin's image secret sharing scheme, the threshold is k, the number of participants is n, and the participant identifiers are $id_1, id_2, \ldots, id_n$. The secret image to be shared is S and its number of pixels is w. The allocator first takes n distinct non-zero elements ($id_1, id_2, \ldots, id_n$) in the finite field as identifiers for the n participants. $id_1, id_2, \ldots, id_i$ and its correspondence with each participant is public. The allocator takes the grayscale value of the first to kth pixels in S as the value of $a_0, a_1, \ldots, a_{k-1}$, forming the polynomial:

$$f(x) = (a_0 + a_1 x + a_2 x^2 + \cdots + a_{(k-1)} x^{(k-1)}) mod p \quad (1)$$

The allocator generates n secrets to share with n participants based on each participant's identifier as the first pixel ${S^i}_1$ of each shadow image ${S^i}_1$

$$ {S^i}_1 = f(id_i), i = 1, \cdots, n \tag{2} $$

The allocator then repeats the above operation for the kth + 1st to 2kth pixel in S as the value of $a_0, a_1, \ldots, a_{k-1}$ to obtain ${S^i}_2$, $i = 1, \ldots,$ n. And so on until all pixels in S have been secretly shared to obtain ${S^i}_{w/k}$, $i = 1, \ldots,$ n.

The specific algorithm of Thien and Lin's image secret sharing scheme is described in Algorithm 1.

Algorithm 1 Thien and Lin's secret image sharing scheme

Input: $S, id_1, id_2, \ldots, id_n$
Output: $S^1, S^2, \ldots, S^n$
Step1: Let $j = 1$
Step2: The k pixels from $k(j-1) + 1$ to $k(j-1) + k$ are selected in S and assigned to $a_0, a_1, \ldots, a_{k-1}$
Step3: Construct the polynomial (1) where $p > 0$
Step4: Calculate ${S^i}_j = f(id_i)$, $i = 1, \ldots, n$.
Step5: If $j < w/k$, let $j = j + 1$ and skip to Step2; if $j < w/k$, skip to Step6.
Step6: Using ${S^i}_j$, $j = 1, \ldots, w/k$ as the pixel gray value of image S^i and obtain the shadow image S^i, $i = 1, \ldots, n$.

2.2. Secret Recovery Stage

In Thien and Lin's image secret recovery scheme [2], secret recovery is performed using k shadow images $S^1, S^2, \ldots, S^n$, $id_1, id_2, \ldots, id_n$ are known, taking the first pixel of each shadow image $S^1, S^2, \ldots, S^n$.

From the Lagrange's interpolation formula,

$$ f(x) = \sum_{i=1}^{k} f(x_i) \prod_{h \neq i}^{1 \le h \le k} \frac{(x - x_h)}{(x_i - x_h)} \tag{3} $$

we have,

$$ f(x) = \sum_{i=1}^{k} {S^i}_1 \prod_{h \neq i}^{1 \le h \le k} \frac{(x - id_h)}{(id_i - id_h)} \tag{4} $$

The grayscale value of the first to kth pixel of the secret message as a secret image (a_0, $a_1, \ldots, a_{k-1}$) can be obtained according to the coefficient of $f(x)$. Take the second pixel of each shadow image (${S^1}_2, {S^2}_2, \ldots, {S^k}_2$) for Lagrange's interpolation.

$$ f(x) = \sum_{i=1}^{k} {S^i}_2 \prod_{h \neq i}^{1 \le h \le k} \frac{(x - id_h)}{(id_i - id_h)} \tag{5} $$

From the coefficients of $f(x)$, obtain $a_0, a_1, \ldots, a_{k-1}$ as the kth + 1st to 2kth pixels in the secret image. Hence, the above process is repeated until all pixels in S have been recovered to obtain the secret image S.

The specific algorithm of Thien and Lin's image secret recovery scheme is described in Algorithm 2.

Algorithm 2 Thien and Lin's secret image recovery scheme

Input: $id_1, id_2, \ldots, id_k$; $S^1, S^2, \ldots, S^k$.
Output: S
Step1: Denote the jth pixel gray value of S^i as ${S^i}_j$ and let $j = 1$.
Step2: Substitute $id_1, id_2, \ldots, id_k$ and ${S^1}_j, {S^2}_j, \ldots, {S^k}_j$ into the Lagrange's interpolation polynomial (4).
Step3: Take the coefficient $a_0, a_1, \ldots, a_{k-1}$ of $f(x)$, let $S_{k(j-1)+1} = a_0$, $S_{k(j-1)+2} = a_1, \ldots, S_{k(j-1)+k} = a_{k-1}$.
Step4: If $j < w/k$, let $j = j + 1$ and skip to Step2; if $j = w/k$, then skip to Step5.
Step5: Get the secret image S

3. Implementation of SIS with Matrix Operation

3.1. Motivation

An image secret sharing scheme is a technique to share a secret image among a group of participants. In a polynomial-based SIS scheme, the shadows are generated by polynomials, and the secret pixels are derived from the coefficients of a Lagrange interpolating polynomial. Actually, to determine the coefficients of a Lagrange interpolating polynomial, we need to solve linear equations. In a real life application, sometimes the secret image has a large size with a high resolution, such as the CT and MR medical image, and a confidential image in military or in commerce. Therefore, in the revealing process, we need to repeatedly solve linear equations many times. This operation is inefficient, since solving linear equations has computation complexity $O(n^3)$. Therefore, improving the efficiency of the secret sharing scheme is crucial for the practical application of image secret sharing.

In the case of Thien and Lin's experiments, due to the large number of pixels in the secret image, the sharing and recovery of the image secret was done in many iterations. The *id* value in each operation is the same, in particular during the secret image recovery stage, each time the same *id* value is substituted into the Lagrange's interpolation polynomial calculation. Optimization and improvement are attempted for this property.

First, in the image secret sharing process as shown in Algorithm 1, Step2, Step3 constructs $f(x) = (a_0 + a_1x + a_{2\times}{}^2 + \ldots + a_{(k-1)}x^{(k-1)}) mod p$ and computes ${S^i}_j = f(id_i)$ separately at $i = 1, \ldots, n$. This process is equivalent to calculating:

$$\begin{cases} a_0 + a_1 id_1 + a_2 id_1^2 + \cdots + a_{k-1} id_1^{k-1} = {S^1}_j \\ a_0 + a_1 id_2 + a_2 id_2^2 + \cdots + a_{k-1} id_2^{k-1} = {S^2}_j \\ \vdots \\ a_0 + a_1 id_n + a_2 id_n^2 + \cdots + a_{k-1} id_n^{k-1} = {S^n}_j \end{cases} \tag{6}$$

Equation (6) can be rewritten in the form of the product of matrices in linear algebraic terms:

$$\begin{pmatrix} 1 & id_1 & id_1^2 & \cdots & id_1^{k-1} \\ 1 & id_2 & id_2^2 & \cdots & id_2^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & id_n & id_n^2 & \cdots & id_n^{k-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{k-1} \end{pmatrix} = \begin{pmatrix} {S^1}_j \\ {S^2}_j \\ \vdots \\ {S^n}_j \end{pmatrix} \tag{7}$$

Denote that,

$$D = \begin{pmatrix} 1 & id_1 & id_1^2 & \cdots & id_1^{k-1} \\ 1 & id_2 & id_2^2 & \cdots & id_2^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & id_n & id_n^2 & \cdots & id_n^{k-1} \end{pmatrix} \tag{8}$$

$$A = \begin{pmatrix} a_0 & a_1 & \cdots & a_{k-1} \end{pmatrix}^T \tag{9}$$

$$R = \begin{pmatrix} {S^1}_j & {S^2}_j & \cdots & {S^n}_j \end{pmatrix}^T \tag{10}$$

The formula (7) can be written as: $DA = R$.

In the image secret recovery process shown in Algorithm 2, Step2 and Step3 substitute $id_1, id_2, \ldots, id_k$ and ${S^1}_j, {S^2}_j, \ldots, {S^k}_j$ into a Lagrange's interpolation polynomial, and the obtained coefficients $a_0, a_1, \ldots, a_{k-1}$ of function $f(x)$ are used as the pixel values of the secret image S. The purpose of this process is to solve for the value of $a_0, a_1, \ldots, a_{k-1}$.

Solving for the value of $a_0, a_1, \ldots, a_{k-1}$ in terms of linear algebra means solving for the matrix A. If any k rows are taken out of the matrix D to form a new matrix F, the pixel values involved in the recovery form a new vector G. Then we have $FA = G$. Calculate the inverse matrix F^{-1} of F, both sides of the equation $FA = G$ are simultaneously multiplied left by F^{-1} to obtain $F^{-1}FA = F^{-1}G$. Thus, we get $A = F^{-1}G$ and $a_0, a_1, \ldots, a_{k-1}$.

This paper provides a procedure for solving F^{-1} in [25] is as follows:

Step1: Construct e_1, e_2, e_3 as shown (11) below.

Step2: Solve the system of linear equations separately to obtain $x_1, x_2, \ldots, x_k$.

Step3: Get $F^{-1} = (x_1, x_2, \ldots, x_k)$.

$$e_1 = \begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}^T, e_2 = \begin{pmatrix} 0 & 1 & \cdots & 0 \end{pmatrix}^T, \cdots, e_k = \begin{pmatrix} 0 & 0 & \cdots & 1 \end{pmatrix}^T \quad (11)$$

$$Fx_1 = e_1, Fx_2 = e_2, \cdots, Fx_k = e_k \quad (12)$$

The following scheme is proposed after the theoretical derivation of the feasibility of matrix operations (see Section 3.2).

3.2. Agrithm of Secret Image Sharing Scheme with Matrix Operatio n

The implementation method of matrix operation-based image secret sharing scheme proposed in this paper is as follows: the allocator substitutes $id_1, id_2, \ldots, id_n$ into the (8) construction matrix D, and the grayscale values of the first to kth pixels in S are substituted as the values of $a_0, a_1, \ldots, a_{k-1}$ to form the vector A as shown in (9). The allocator calculates $R = DA$ to obtain the vector R as shown. The allocator can generate n secret messages to share with n participants based on each participant's identifier as the first pixel of each shadow image S^i: ${S^i}_1$.

The allocator then repeats the above operation by using the grayscale values of the kth + 1st to 2kth pixels in S as the value of $a_0, a_1, \ldots, a_{k-1}$ to obtain ${S^i}_2$, $i = 1, \ldots, n$. Until all pixels in S have been shared to obtain ${S^i}_{w/k}$, $i = 1, \ldots, n$. Thus, get the shadow image $S_1, S_2, \ldots, S_n$.

The proposed algorithm of secret image sharing based on matrix operation (ISSMO) is shown in Algorithm 3. When $n = 5$, $k = 3$, the schematic diagram of the sharing process is shown in Figure 1.

Algorithm 3 Image secret sharing algorithm based on matrix operations

Input: S; $id_1, id_2, \ldots, id_n$
Output: $S^1, S^2, \ldots, S^k$
Step1: Construct matrix D from (8).
Step2: Let $j = 1$
Step3: Select the k pixels from $k(j-1) + 1$ to $k(j-1) + k$ in S and assign $a_0, a_1, \ldots, a_{k-1}$ to each pixel
Step4: Construct the column vector A and calculate $R = DA$.
Step5: Get ${S^1}_j, {S^2}_j, \ldots, {S^n}_j$
Step6: If $j < w/k$, let $j = j + 1$ and jump to Step3; if $j = w/k$, skip to Step7.
Step7: The shadow image $S_1, S_2, \ldots, S_n$ is obtained by using ${S^i}_1, {S^i}_2, \ldots, {S^i}_{w/k}$ as the pixel grayscale value of image S^i

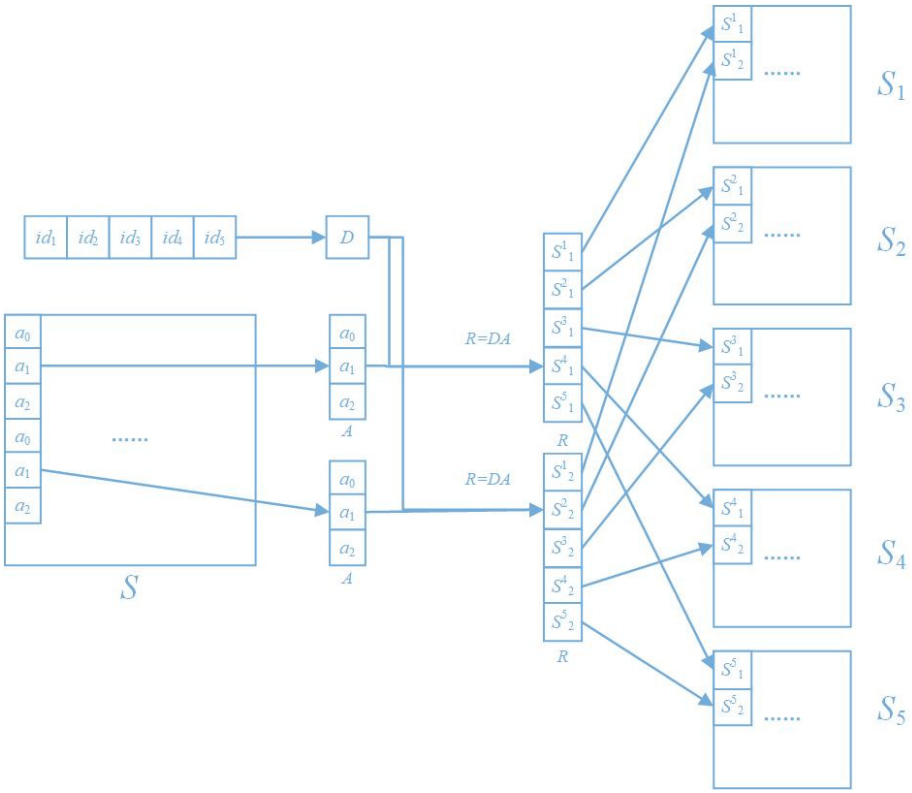

Figure 1. The schematic diagram of the sharing process.

When using k shadow images $S^1, S^2, \ldots, S^k$ for secret recovery, its corresponding participant identifiers $id_1, id_2, \ldots, id_n$ are known. First, the matrix F is obtained by taking the k rows corresponding to $id_1, id_2, \ldots, id_k$ in the matrix D.

$$F = \begin{pmatrix} 1 & id_1 & id_1^2 & \cdots & id_1^{k-1} \\ 1 & id_2 & id_2^2 & \cdots & id_2^{k-1} \\ 1 & id_3 & id_3^1 & \cdots & id_3^{k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & id_k & id_k^2 & \cdots & id_k^{k-1} \end{pmatrix} \tag{13}$$

Next, follow the method mentioned in Section 3.1 to derive F^{-1}. Take the first pixel ${S^1}_1, {S^2}_1, \ldots, {S^k}_1$ of each shadow image to form the vector

$$G = \begin{pmatrix} {S^1}_1 & {S^2}_1 & \cdots & {S^k}_1 \end{pmatrix}^T \tag{14}$$

Then calculate $A = G^{-1}R$ to get the column vector A. From the coefficients of A, we can get the secret information $a_0, a_1, \ldots, a_{k-1}$, as the gray value of k pixels from 1 to k of the secret image.

Next, the second pixel ${S^1}_2, {S^2}_2, \ldots, {S^k}_2$ of each shadow image is taken to form the vector $G = ({S^1}_2, {S^2}_2, \ldots, {S^k}_2)^T$, and $A = F^{-1}G$ is calculated to obtain the column vector A. From the coefficients of A, the secret information $a_0, a_1, \ldots, a_{k-1}$, as the grayscale values of kth + 1st to 2kth pixels of the secret image, is obtained. The above process is repeated until all pixels of S are secretly recovered and obtain the secret image S.

In this paper, the proposed image secret sharing algorithm based on matrix operations is shown in Algorithm 4. A schematic diagram of the recovery process is shown in Figure 2. We assume that the identities of the participants involved in recovery are id_1, id_2, and id_3.

Algorithm 4 Image secret recovery algorithm based on matrix operations

Input: $id_1, id_2, \ldots, id_k, S^1, S^2, \ldots, S^k$
Output: S

Step1: The matrix F is obtained by taking out the k rows corresponding to $id_1, id_2, \ldots, id_k$ in the matrix D.
Step2: Calculate the inverse matrix F^{-1} of F according to the method in Section 3.1. *Motivation*
Step3: Denote the jth pixel gray value of S^i as $S^i{}_j$ and let $j = 1$.
Step4: Construct $G = (S^1{}_j, S^2{}_j, \ldots, S^k{}_j)^T$ and compute $A = F^{-1}\mathrm{G}$ to obtain the column vector A.
Step5: Get $a_0, a_1, \ldots, a_{k-1}$ from (9), let $S_{k(j-1)+1} = a_0, S_{k(j-1)+2} = a_0, \ldots, S_{k(j-1)+k} = a_{k-1}$
Step6: If $j < w/k$, then let $j = j + 1$ and skip to Step4; if $j = w/k$, skip to Step7.
Step7: Obtain S by using $S_1, S_2, \ldots, S_w$. as the pixel gray value of the image S.

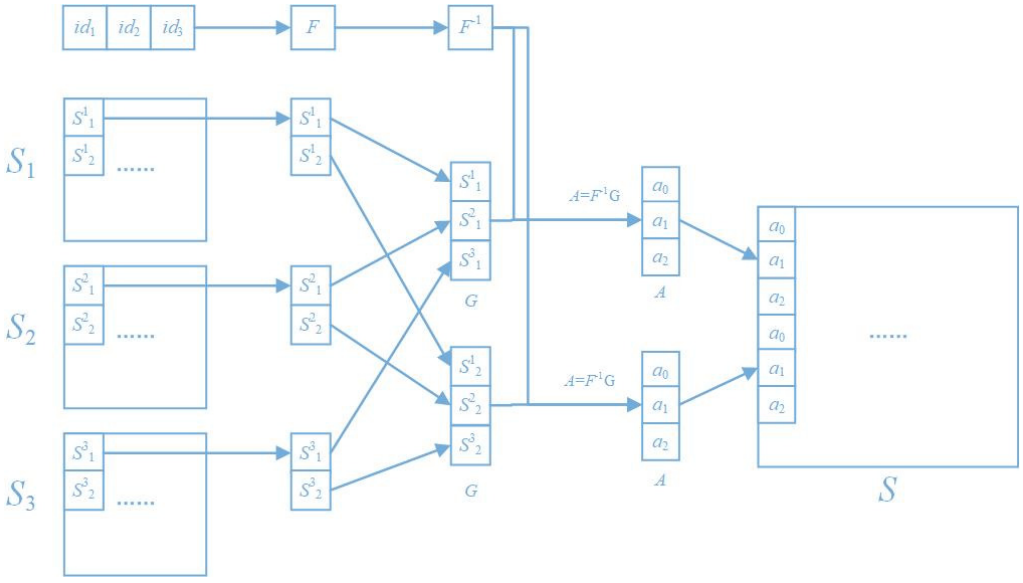

Figure 2. The schematic diagram of the recovery process.

3.3. Example of (3, 3)-SIS

This section will assign $id_1 = 1$, $id_2 = 2$, $id_3 = 3$ to each participant's identity without loss of generality under the finite fields $GF(2^8)$ and $GF(251)$, respectively. Share a set of secret pixels through an image secret sharing scheme with threshold conditions: $k = 3$, $n = 3$. The shared secret pixel grayscale values are: 25, 156, 7, 210, 54, 134.The secret image recovery algorithm based on matrix operations is shown more specifically by the following example.

3.3.1. Example of (3, 3)-SIS Scheme in $GF(2^8)$

In the SISMO sharing process, first construct matrix D according to the value of each participant's identity.

$$D = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \tag{15}$$

Construct the column vector $A_1 = (25\ 156\ 7)^T$ using the first three numbers in the secret message. Through $R_1 = DA_1$ to get the column vector R_1.

$$R_1 = DA_1 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} 25 \\ 156 \\ 7 \end{pmatrix} = \begin{pmatrix} 130 \\ 32 \\ 187 \end{pmatrix} \tag{16}$$

The first element of the three shadow images $S^1{}_1 = 130$, $S^2{}_1 = 32$, $S^3{}_1 = 187$ can be obtained according to the elements of the vector R. Then use the last three numbers in the

secret message to construct the column vector $A_2 = (210\ 54\ 134)^T$, and R_2 is obtained by calculate $R_2 = DA_2$.

In the recovery process, first construct the matrix F according to the formula.

$$F = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \tag{17}$$

Then construct e_1, e_2, e_3.

$$e_1 = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T, e_2 = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T, e_3 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T \tag{18}$$

Solve the linear equations $Fx_1 = e_1, Fx_2 = e_2, \ldots, Fx_k = e_k$ separately as follows.

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} x_{1,1} \\ x_{1,2} \\ x_{1,3} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} x_{2,1} \\ x_{2,2} \\ x_{2,3} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{19}$$
$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} x_{3,1} \\ x_{3,2} \\ x_{3,3} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Get x_1, x_2, x_3 and calculate F^{-1} as follows.

$$x_1 = \begin{pmatrix} 1 \\ 122 \\ 122 \end{pmatrix}, x_2 = \begin{pmatrix} 1 \\ 245 \\ 244 \end{pmatrix}, x_3 = \begin{pmatrix} 1 \\ 143 \\ 142 \end{pmatrix} \tag{20}$$

$$F^{-1} = (x_1\ \ x_2\ \ x_3) = \begin{pmatrix} 1 & 1 & 1 \\ 122 & 245 & 143 \\ 122 & 244 & 142 \end{pmatrix} \tag{21}$$

The next step is to construct $G_1 = (130\ 32\ 187)^T$ and calculate $A_1 = F^{-1}G_1$.

$$A_1 = F^{-1}G_1 = \begin{pmatrix} 3 & 248 & 1 \\ 123 & 4 & 124 \\ 126 & 250 & 126 \end{pmatrix} \begin{pmatrix} 130 \\ 32 \\ 187 \end{pmatrix} = \begin{pmatrix} 25 \\ 156 \\ 7 \end{pmatrix} \tag{22}$$

The first three numbers of the secret information are known. Next is to build $G_2 = (98\ 156\ 44)^T$ and calculate $A_2 = F^{-1}G_2$.

$$A_2 = F^{-1}G_2 = \begin{pmatrix} 3 & 248 & 1 \\ 123 & 4 & 124 \\ 126 & 250 & 126 \end{pmatrix} \begin{pmatrix} 98 \\ 156 \\ 44 \end{pmatrix} = \begin{pmatrix} 210 \\ 54 \\ 134 \end{pmatrix} \tag{23}$$

So far, the complete secret information is obtained: 25, 156, 7, 210, 54, 134.

3.3.2. Example of (3, 3)-SIS Scheme in GF (251)

In the secret sharing process, first construct the matrix D as shown,

$$D = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \tag{24}$$

Then construct the column vector $A_3 = (25\ 156\ 7)^T$, calculate R_3 as follows and get the first pixel of the three shadow images: ${S^1}_1 = 188$, ${S^2}_1 = 114$, ${S^3}_1 = 54$.

$$R_3 = DA_3 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} 25 \\ 156 \\ 7 \end{pmatrix} = \begin{pmatrix} 188 \\ 114 \\ 54 \end{pmatrix} \tag{25}$$

Next, construct the column vector $A_4 = (210\ 54\ 134)^T$ and calculate R_4 according to the following formula, ${S^1}_2 = 147$, ${S^2}_2 = 101$, ${S^3}_2 = 72$, we can get $S_1 = (188\ 147)^T$, $S_2 = (144\ 101)^T$, $S_3 = (54\ 72)^T$,

$$R_4 = DA_4 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} 210 \\ 54 \\ 134 \end{pmatrix} = \begin{pmatrix} 147 \\ 101 \\ 72 \end{pmatrix} \tag{26}$$

In the recovery process, first construct the matrix F and e_1, e_2, e_3 as shown in the formulas:

$$F = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \tag{27}$$

$$e_1 = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^T, e_2 = \begin{pmatrix} 0 & 1 & 0 \end{pmatrix}^T, e_3 = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}^T \tag{28}$$

Solve the linear equations $Fx_1 = e_1$, $Fx_2 = e_2$, ... , $Fx_k = e_k$ separately as shown:

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} x_{1,1} \\ x_{1,2} \\ x_{1,3} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} x_{2,1} \\ x_{2,2} \\ x_{2,3} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{29}$$
$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \end{pmatrix} \begin{pmatrix} x_{3,1} \\ x_{3,2} \\ x_{3,3} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Get x_1, x_2, x_3 and F^{-1} as shown:

$$x_1 = \begin{pmatrix} 3 \\ 123 \\ 126 \end{pmatrix}, x_2 = \begin{pmatrix} 248 \\ 4 \\ 250 \end{pmatrix}, x_3 = \begin{pmatrix} 1 \\ 124 \\ 126 \end{pmatrix} \tag{30}$$

$$F^{-1} = (x_1\ \ x_2\ \ x_3) = \begin{pmatrix} 3 & 248 & 1 \\ 123 & 4 & 124 \\ 126 & 250 & 126 \end{pmatrix} \tag{31}$$

Then construct $G_3 = (118\ 114\ 54)^T$ and calculate A_3, construct $G_4 = (147\ 101\ 72)^T$ and calculate A_4 as follows:

$$A_3 = F^{-1}G_3 = \begin{pmatrix} 3 & 248 & 1 \\ 123 & 4 & 124 \\ 126 & 250 & 126 \end{pmatrix} \begin{pmatrix} 118 \\ 114 \\ 54 \end{pmatrix} = \begin{pmatrix} 25 \\ 156 \\ 7 \end{pmatrix} \tag{32}$$

Finally, the secret information: 25, 156, 7, 210, 54, 134 is obtained. Thus it can be seen that the restoration process is more effective compared with the original secret information.

3.4. Computational Efficiency Analysis

The complicated step of the calculation in this scheme is to construct the matrix D and calculate D^{-1}. Since id_1, id_2, ... , id_n and its correspondence with P_1, P_2, ... , P_n are public

in the secret sharing process, the process of constructing the matrix D and calculating D^{-1} only needs to be done before recovering the first set of pixels. In this way, the process of recovering all secret pixels afterwards only needs to be performed to calculate $A = D^{-1}S$, which greatly reduced the computation complexity of the secret recovery. In this section, the efficiency of the two algorithms is analyzed in terms of the number of operations (+, −, ×, ÷) required by the two schemes.

3.4.1. Efficiency Analysis of Thien and Lin's (k, n)-SIS Scheme

In Thien and Lin's image secret sharing process, as described in Algorithm 1, no operation occurred in Step1 to Step3 and it takes place in Step4, the calculation of a polynomial of degree $k-1$ is performed. The number of operations that have occurred is:

$$O_1S_1 = \frac{1}{2}nk^2 + \frac{1}{2}nk - n \tag{33}$$

Step5 makes a judgment and loops Step1 to Stpe4 for w/k times; no operation occurs in Step6. After collating the above steps, the number of operations occurred during the secret sharing of images of Thien and Lin is obtained as:

$$O_1S = \frac{w}{k}O_1S_1 = \frac{1}{2}wnk + \frac{1}{2}wn - wnk^{-1} \tag{34}$$

In Thien and Lin's image secret recovery process Algorithm 2, no operation occurs in Step1. Step2 involves the calculation of Lagrange's interpolation polynomials, which is equivalent to constructing a linear equation system first, and then solving it. The number of operations that occurred in this step is:

$$O_1R_1 = \left(\frac{1}{2}\mathrm{k}^3 + \frac{1}{2}k^2\right) + \left(\frac{2}{3}k^3 + \frac{1}{2}k^2 - \frac{7}{6}k\right) = \frac{7}{6}k^3 + k^2 - \frac{7}{6}k \tag{35}$$

No operation occurred in Step3, Step1~Step3 are looped for w/k times after the judgment of Step4. Collating all the steps, the number of operations occurred during the Thien and Lin's image secret recovery process is:

$$O_1R = \frac{w}{k}O_1R_1 = \frac{7}{6}wk^2 + wk - \frac{7}{6}w \tag{36}$$

Integrating the number of operations of the sharing process with the recovery process, the total number of operations occurring in Thien and Lin's image secret sharing scheme is:

$$O_1 = O_1S + O_1R = \frac{7}{6}wk^2 + \left(\frac{1}{2}n + 1\right)wk + \frac{1}{2}wn - \frac{7}{6}w - wnk^{-1} \tag{37}$$

3.4.2. Efficiency Analysis of (k, n)-SIS Scheme Based on Matrix Operations

In Algorithm 3, a secret sharing process for images based on matrix operations, the number of operations that occur in Step1 is:

$$O_2S_1 = \frac{1}{2}nk^2 - \frac{3}{2}nk + n \tag{38}$$

No operation occurs in Step2 and Step3 and one matrix multiplication operation occurs in Step4. Since it is a matrix of n rows and k columns multiplied by a column vector of k rows, the number of addition operations that occur is:

$$O_2S_2 = 2nk - 2n \tag{39}$$

No operation occurred in step5 and Step7, and Step6 made a judgment and cycled Step3 to Step5 w/k times. In summary, the total number of operations that occurred in Algorithm 3, a secret sharing process for images based on matrix operations, is:

$$O_2S = O_2S_1 + \frac{w}{k}O_2S_2 = \frac{1}{2}nk^2 - \frac{3}{2}nk + 2wn + n - 2wnk^{-1} \tag{40}$$

And in the matrix operation-based image secret recovery process Algorithm 4, the number of operations occurring in Step1 is:

$$O_2R_1 = \frac{1}{2}k^3 - \frac{3}{2}k^2 + k \tag{41}$$

In Step2, the process of solving the system of linear equations is performed k times and the total number of operations performed is:

$$O_2R_2 = \frac{2}{3}k^4 + \frac{1}{2}k^3 - \frac{7}{6}k^2 \tag{42}$$

No operation occurs in Step3. The number of operations that occur in Step4 is:

$$O_2R_3 = 2k^2 - k \tag{43}$$

In Step5, no operation occurred. Step6 made a judgment and looped Step4 and Step5 w/k times. No operation occurred in Step7. To sum up, the total number of operations occurred in Algorithm 4 of the image secret recovery process based on matrix operations is:

$$O_2R = O_2R_1 + O_2R_2 + \frac{w}{k}O_2R_3 = \frac{2}{3}k^4 + k^3 - \frac{8}{3}k^2 + (1+2w)k - w \tag{44}$$

By integrating the number of operations of the sharing process with the recovery process, the total number of operations occurring in the implementation of image secret sharing scheme based on matrix operations is:

$$\begin{aligned} O_2 &= O_2S + O_2R \\ &= \tfrac{2}{3}k^4 + k^3 + \left(\tfrac{n}{2} - \tfrac{8}{3}\right)k^2 + \left(2w - \tfrac{3}{2}n + 1\right)k + 2wn - w + n - wnk^{-1} \end{aligned} \tag{45}$$

3.4.3. Operation Time Comparison

In this paper, the number of operations of the classical secret sharing scheme compared with the secret sharing scheme proposed in this paper is plotted using MATLAB 2016a. Figure 3 shows the image of the number of operations performed by the two different schemes varying with the value of k, where w = 256 × 256, n = 10, and k is taken from 2 to 10.

And Figure 4 shows the images of the number of operations performed by the two schemes with different w when w is taken from 128 × 128 to 512 × 512 and k = 5, n = 10.

According to the comparison of operation times as shown in Figures 3 and 4, the number of operations of the implementation method proposed in this paper is smaller than that of Thien and Lin's image secret sharing scheme in the range of threshold values from 2 to 10 and secret image sizes from 128 × 128 to 512 × 512. Moreover, the difference between the number of operations of both methods is proportional to the threshold and secret image size.

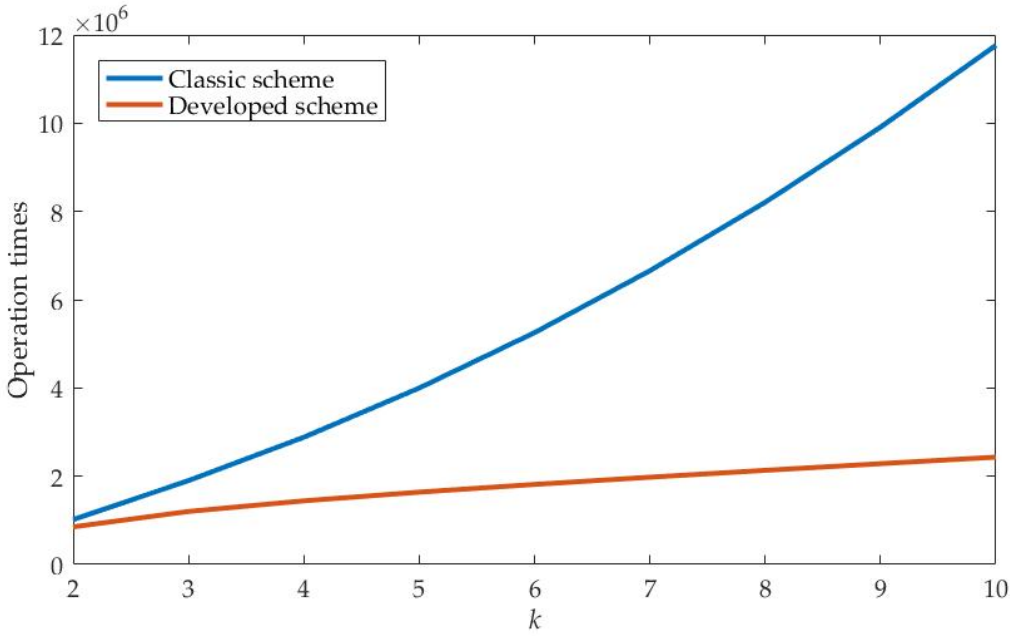

Figure 3. Comparison of the number of operations performed by the two schemes on the change of k.

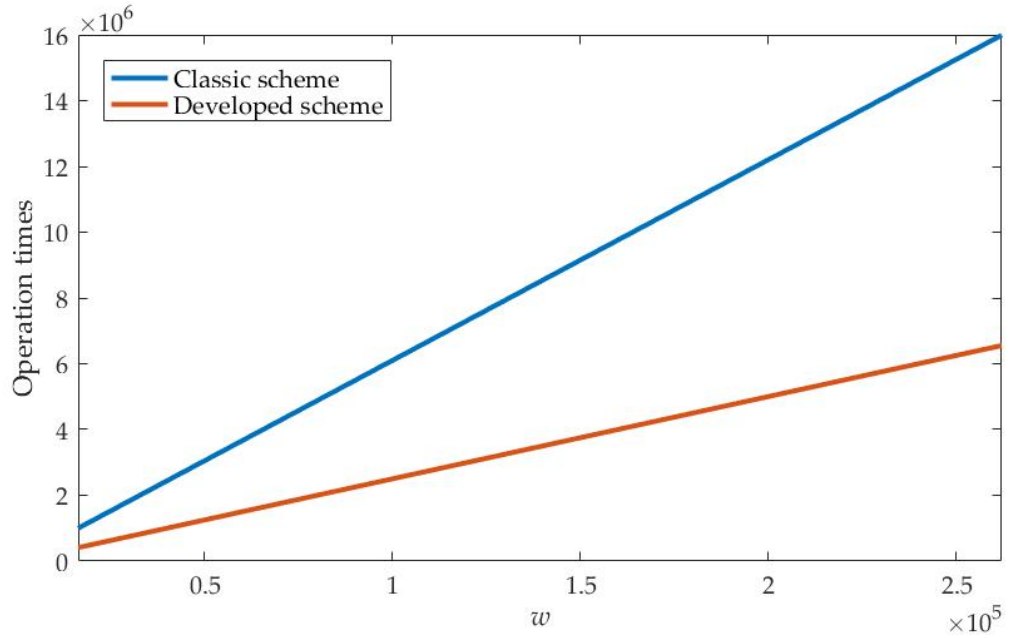

Figure 4. Comparison of the number of operations performed by the two schemes with respect to changes in w.

4. Experiments and Comparisons

4.1. Feasibility Experiment of the Implementation Proposed in This Paper

The implementation of image secret sharing scheme based on matrix operations proposed in this paper was programmed using MATLAB 2016a. The (5, 7)-secret sharing is performed on the original image Figure 5a with image size 256 × 256 pixels, and the values of $id_1, id_2, \ldots, id_7$ are: $id_i = i, i = 1, \ldots, 7$.

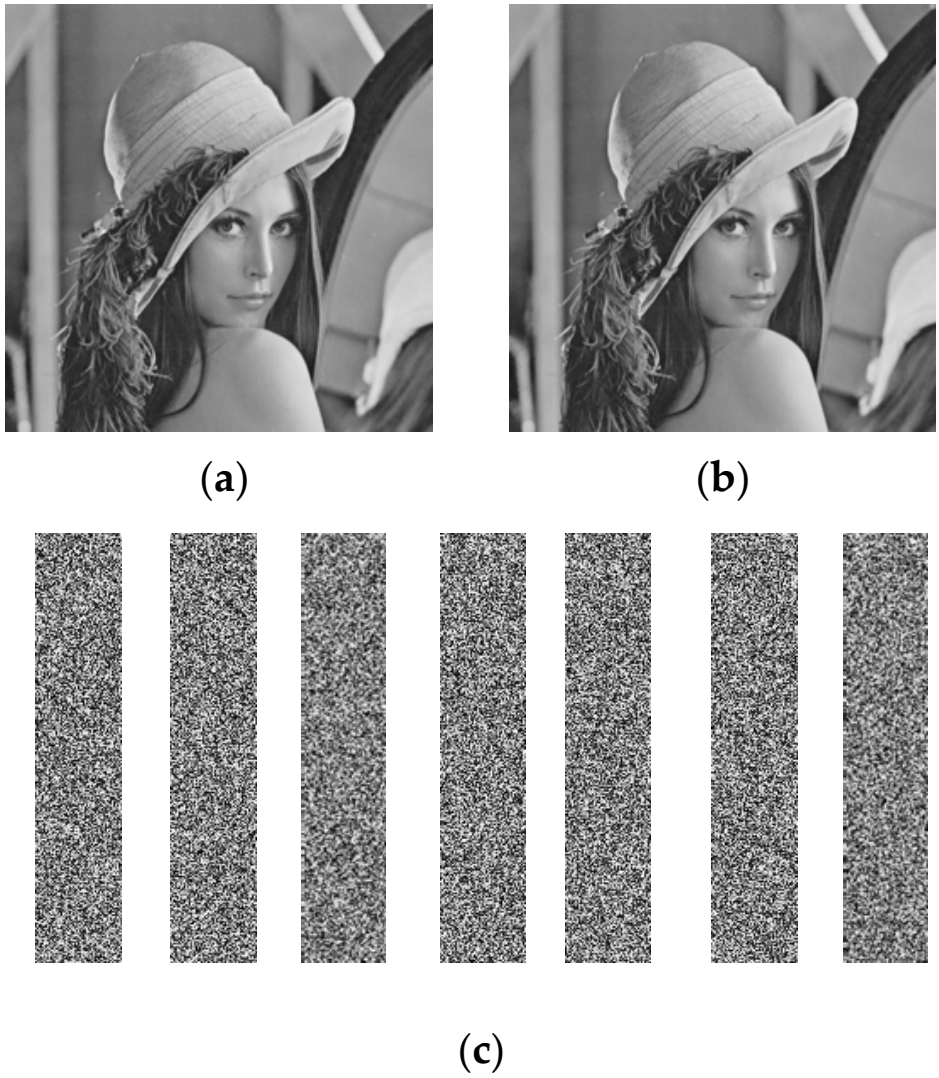

Figure 5. The proposed implementation of (5,7)-ISS scheme based on matrix operations: (**a**) the secret image; (**b**) the recovered image; (**c**) 7 shadows.

The shadow images obtained from the experiment are shown in Figure 5b. The secret image is recovered according to the matrix operation-based image secret recovery algorithm, and the recovered image obtained is shown in Figure 5c. Comparing the recovered Figure 5a with Figure 5c shows that the image secret sharing scheme based on matrix operation proposed in this paper is effective.

4.2. Comparison Experiment of Computing Efficiency under Different Thresholds

In this experiment, a 256 × 256 size image is used for secret sharing, and the same shares are selected for secret sharing using two different implementation methods: Thien and Lin's image secret sharing scheme based on Lagrange's polynomials, and the implementation of image secret sharing scheme based on matrix operations for secret sharing and recovery. The running times of the two schemes are calculated separately.

The experiment used MATLAB2016 for programming, and selected $GF(2^8)$ as the finite field. The irreducible polynomial used in the operation is $x^8 + x^4 + x^3 + x^2 + x$+1.The running time is recorded when the threshold is k = 2, 3, 4, 5 respectively, as shown in the following Table 1, and the comparison of the running time is shown in Figure 6.

Table 1. Running time with different threshold in $GF(2^8)$.

	Threshold k = 2	Threshold k = 3	Threshold k = 4	Threshold k = 5
Developed scheme Running time (second)	21.3185	16.4230	13.5538	12.4131
Classic scheme Running time (second)	50.0028	57.6745	66.0707	65.6301

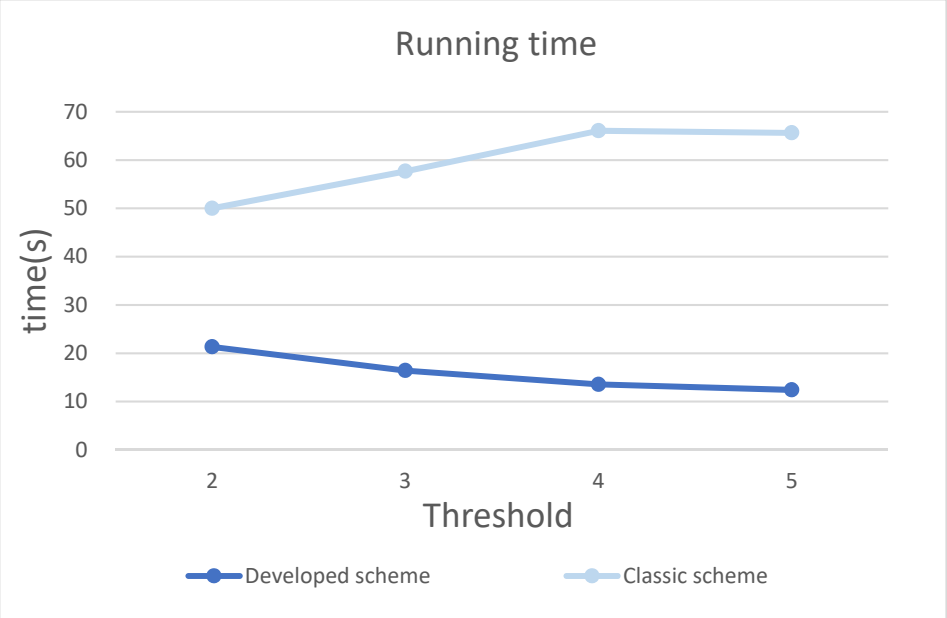

Figure 6. Running time comparison chart with different threshold in $GF(2^8)$.

Under the finite field $GF(251)$, the running times for threshold values of $k = 2, 3, 4, 5$ are recorded respectively as shown in the following Table 2, and the running time comparison graphs are shown in Figure 7.

Table 2. Running time in different threshold in $GF(251)$.

	Threshold k = 2	Threshold k = 3	Threshold k = 4	Threshold k = 5
Developed scheme Running time (second)	0.3453	0.2960	0.2448	0.2524
Classic scheme Running time (second)	0.4062	0.3633	0.3287	0.3622

4.3. Comparison Experiments of Computing Efficiency under Secret Image Recovery of Different Sizes

This experiment used images of different sizes for secret sharing. The same shares are selected to perform (5, 5) secret sharing and recovery using two different secret sharing schemes: the classical Thien and Lin's image secret sharing scheme (Classic scheme), and the matrix operation-based secret sharing scheme (Developed scheme) proposed in this paper, and the running times of the two are calculated separately.

The experiments were programmed using MATLAB2016 with the finite field selection $GF(2^8)$. The irreducible polynomial used in the operation is $x^8 + x^4 + x^3 + x^2 + x+1$.The running times when the secret image size is 128 × 128, 192 × 192, 256 × 256, 384 × 384, 512 × 512 are recorded respectively as shown in the following Table 3, and the running time comparison graphs are shown in Figure 8.

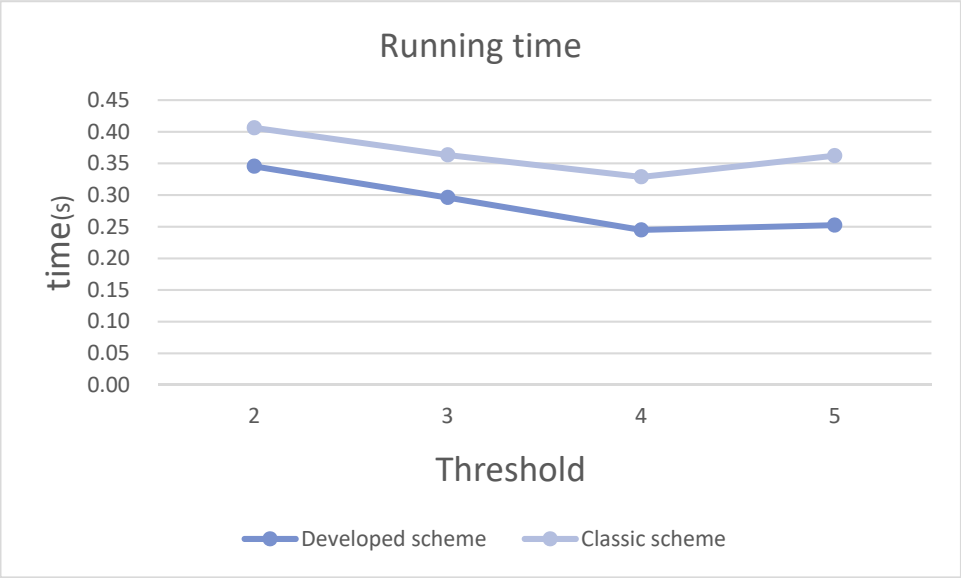

Figure 7. Running time comparison chart with different threshold in $GF(251)$.

Table 3. Running times with different sizes in $GF(2^8)$.

	Secret Image Size 128 × 128	**Secret Image Size 192 × 192**	**Secret Image Size 256 × 256**	**Secret Image Size 384 × 384**	**Secret Image Size 512 × 512**
Developed scheme Running time (second)	3.1263	5.7193	9.3425	11.3432	38.2335
Classic scheme Running time (second)	16.9145	36.3079	64.6712	151.6507	269.1568

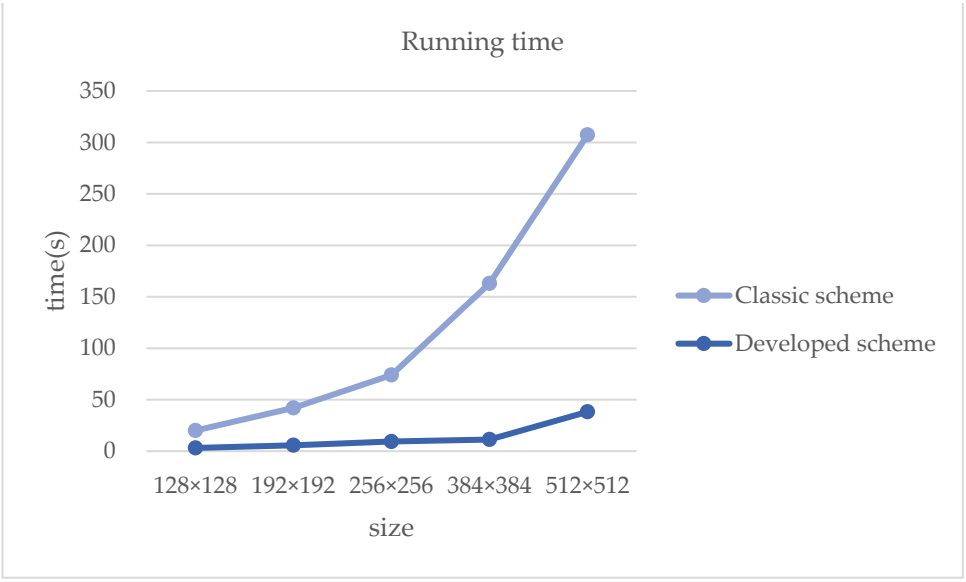

Figure 8. Running time comparison chart with different size in $GF(2^8)$.

Under the finite field $GF(251)$, the running time when the recording size is 128 × 128, 192 × 192, 256 × 256, 384 × 384, 512 × 512 is shown in the following Table 4, and the running time comparison chart is shown in Figure 9.

Table 4. Running times with different sizes in $GF(251)$.

	Secret Image Size 128 × 128	**Secret Image Size 192 × 192**	**Secret Image Size 256 × 256**	**Secret Image Size 384 × 384**	**Secret Image Size 512 × 512**
Developed scheme Running time (second)	0.1517	0.1778	0.2208	0.3558	0.5600
Classic scheme Running time (second)	0.2008	0.2203	0.4060	0.5803	0.9693

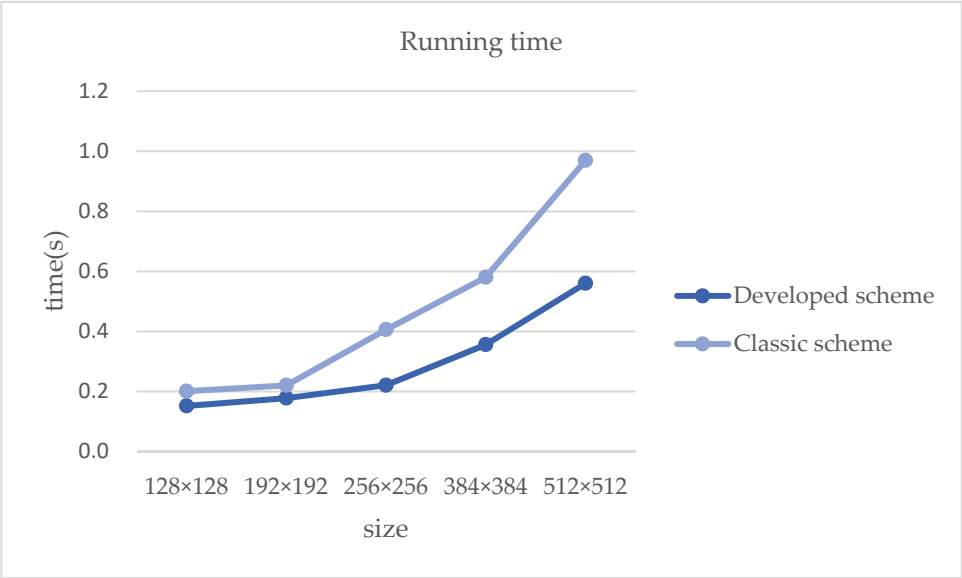

Figure 9. Running time comparison chart with different sizes in $GF(251)$.

4.4. Analysis of Experimental Results

A comparison of the operation times of the two schemes under $GF(251)$ and $GF(2^8)$ finite fields with different thresholds k is shown respectively in Section 4.2. Section 4.3 shows the comparison of the computation time of the two schemes under two finite fields ($GF(251)$ and $GF(2^8)$) with different secret image sizes. Experiments show that the operation time of the matrix operation-based secret image sharing scheme proposed in this paper is smaller than that of the polynomial interpolation-based secret image sharing scheme. And the time difference between the two schemes is proportional to the threshold value k and the size of the secret image. The experimental results are consistent with the theoretical results in Section 3.4.

5. Conclusions

Thien and Lin's ISS scheme uses Lagrange's interpolation polynomials in the secret reconstruction process, which requires a lot of calculations in practice, which affects the efficiency of the schemes. This paper proposes an implementation method of a secret sharing scheme based on matrix operations instead of the traditional polynomial calculation method to reduce the number of operations in the revealing process. Compared with Lagrange's interpolation, the proposed implementation method has better performance through theoretical analysis, which is also confirmed by experimental results. The implementation in this paper is still valid for general polynomial based secret sharing schemes, and may have better results for image secret sharing schemes with different functions. For example, for an image secret sharing scheme with essential participants, the implementation of this paper may provide new ideas for building essential participants. This direction still deserves further research.

Author Contributions: Conceptualization, Z.R. and P.L.; Formal analysis, Z.R. and X.W.; Methodology, Z.R.; Writing–original draft, Z.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Natural Science Foundation of Hebei Province (Grant number: F2019502173), National Natural Science Foundation of China (Grant number: 61602173) and the Fundamental Research Funds for Central Universities (Grant number: 2019MS116).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the editor and the anonymous reviewers for their valuable comments.

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
2. Lin, T. Secret image sharing. *Comput. Graph.* **2002**, *26*, 765–770.
3. Lin, C.C.; Tsai, W.H. Secret image sharing with steganography and authentication. *J. Syst. Softw.* **2004**, *73*, 405–414. [CrossRef]
4. Yang, C.N.; Chen, T.S.; Yu, K.H.; Wang, C.C. Improvements of image sharing with steganography and authentication. *J. Syst. Softw.* **2007**, *80*, 1070–1076. [CrossRef]
5. Ma, K.Y. Essential secret image sharing scheme with different importance of shadows. *J. Vis. Commun. Image Represent.* **2013**, *24*, 1106–1114.
6. Yan, X.; Wang, S.; Niu, X.; Yang, C.N. *Essential Visual Cryptographic Scheme with Different Importance of Shares*; Springer International Publishing: Berlin/Heidelberg, Germany, 2014.
7. Chen, C.C. Essential Secret Image Sharing Scheme. In Proceedings of the CSCIST 2010, CrossStrait Conference on Information Science and Technology, Qinhuangdao, China, 9–13 July 2010; Department of Information Management, Hsuan Chuang University: Hsinchu, Taiwan, 2010.
8. Li, P.; Liu, Z. An Improved Essential Secret Image Sharing Scheme with Smaller Shadow Size. *Int. J. Digit. Crime Forensics* **2018**, *10*, 78–94. [CrossRef]
9. Li, P.; Yin, L.; Ma, J. Visual Cryptography Scheme with Essential Participants. *Mathematics* **2020**, *8*, 838. [CrossRef]
10. Yadav, M.; Singh, R. Essential secret image sharing approach with same size of meaningful shares. *Multimed. Tools Appl.* **2021**, *2021*, 1–18. [CrossRef]
11. Yuan, L.; Li, M.; Guo, C.; Hu, W.; Luo, X. Secret Image Sharing Scheme with Threshold Changeable Capability. *Math. Probl. Eng.* **2016**, *2016*, 9576074. [CrossRef]
12. Liu, Y.X.; Yang, C.N.; Wu, C.M.; Sun, Q.D.; Bi, W. Threshold changeable secret image sharing scheme based on interpolation polynomial. *Multimed. Tools Appl.* **2019**, *78*, 18653–18667. [CrossRef]
13. Deshmukh, M.; Nain, N.; Ahmed, M. An (n, n)-Multi Secret Image Sharing Scheme Using Boolean XOR and Modular Arithmetic. In Proceedings of the IEEE International Conference on Advanced Information Networking & Applications, Crans-Montana, Switzerland, 23–25 March 2016.
14. Singh, N.; Tentu, A.N.; Basit, A.; Venkaiah, V.C. Sequential secret sharing scheme based on Chinese remainder theorem. In Proceedings of the 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), Chennai, India, 15–17 December 2016.
15. Rajput, M.; Deshmukh, M. Secure (n, n + 1)-Multi Secret Image Sharing Scheme Using Additive Modulo. *Procedia Comput. Sci.* **2016**, *89*, 677–683. [CrossRef]
16. Deshmukh, M.; Nain, N.; Ahmed, M. A Novel Approach for Sharing Multiple Color Images by Employing Chinese Remainder Theorem. *J. Vis. Commun. Image Represent.* **2017**, *49*, 291–302. [CrossRef]
17. Basit, A.; Kumar, N.C.; Venkaiah, V.C.; Moiz, S.A.; Tentu, A.N.; Naik, W. Multi-stage Multi-secret Sharing Scheme for Hierarchical Access Structure. In Proceedings of the International Conference on Computing, Communication and Automation (ICCCA2017), Greater Noida, India, 5–6 May 2017.
18. Meng, K.; Miao, F.; Huang, W.; Xiong, Y. Tightly coupled multi-group threshold secret sharing based on Chinese Remainder Theorem. *Discret. Appl. Math.* **2019**, *268*, 152–163. [CrossRef]
19. Bisht, K.; Deshmukh, M. A novel approach for multilevel multi-secret image sharing scheme. *J. Supercomput.* **2021**, *77*, 12157–12191. [CrossRef]
20. Liu, Y.; Yang, C.N. Scalable secret image sharing scheme with essential shadows. *Signal Process. Image Commun.* **2017**, *58*, 49–55. [CrossRef]
21. Liu, Y.; Yang, C.; Wang, Y.; Zhu, L.; Ji, W. Cheating Identifiable Secret Sharing Scheme Using Symmetric Bivariate Polynomial. *Inf. Sci.* **2018**, *453*, 21–29. [CrossRef]

22. Liu, Y.; Sun, Q.; Yang, C.N. (k, n) secret image sharing scheme capable of cheating detection. *Eurasip J. Wirel. Commun. Netw.* **2018**, *2018*, 72. [CrossRef]
23. Yan, X.; Gong, Q.; Li, L.; Yang, G.; Lu, Y.; Liu, J. Secret image sharing with separate shadow authentication ability. *Signal Processing Image Commun.* **2019**, *82*, 115721. [CrossRef]
24. Gutub, A.; Al-Juaid, N.; Khan, E. Counting-based secret sharing technique for multimedia applications. *Multimed. Tools Appl.* **2019**, *78*, 5591–5619. [CrossRef]
25. Strang, G. Matrices and Gaussian Elimination. In *Linear Algebra and Its Applications*, 4th ed.; Thomson Learning: Stamford, CT, USA, 2005; pp. 52–53.