cryptography

MDPI

Article

A (k, n)-Threshold Progressive Visual Secret Sharing without Expansion

Ying-Yu Chen, Bo-Yuan Huang and Justie Su-Tzu Juan *

Department of Computer Science and Information Engineering, National Chi Nan University, Nantou 54561, Taiwan; s99321514@mail1.ncnu.edu.tw (Y.-Y.C.); s103321007@mail1.ncnu.edu.tw (B.-Y.H.)

* Correspondence: jsjuan@ncnu.edu.tw

Received: 31 August 2018; Accepted: 25 September 2018; Published: 27 September 2018

Abstract: Visual cryptography (VC) encrypts a secret image into n shares (transparency). As such, we cannot see any information from any one share, and the original image is decrypted by stacking all of the shares. The general (k, n)-threshold secret sharing scheme (SSS) can similarly encrypt and decrypt the original image by stacking at least k ($\leq n$) shares. If one stack is fewer than k shares, the secret image is unrecognizable. Another subject is progressive visual secret sharing, which means that when more shares are progressively stacked, the combined share becomes clearer. In this study, we constructed an advanced scheme for (k, n)-threshold SSS that can be encrypted in VC for any positive integers $n \geq k \geq 2$ through the method of combination, and the size of each share is the same as that of the original image. That is, no pixel expansion is required. Our scheme is novel, and the results from the theoretical analysis and simulation reveal that our scheme exhibits favorable contrast to that of other related schemes.

Keywords: visual cryptography; (k, n)-threshold; secret sharing; progressive; information security

1. Introduction

Visual Cryptography (VC), a type of the visual secret sharing (VSS) scheme, involves the encoding of secret information into a message and can be decoded visually without computer assistance. In 1979, Blakley [1] and Shamir [2] introduced the (k, n)-threshold secret sharing scheme (SSS), in which a secret image was encoded into n (≥ 2) shares, and only when k ($\geq n$) shares were superimposed upon one another would the secret message be revealed. Furthermore, when fewer than k shares were collected, the secret information could not be decoded.

In 1995, Naor and Shamir proposed four types of (k, n)-threshold SSSs [3], but their schemes exhibited the expansion problem with each share. Some studies [3–9] related to VC have exhibited the common problem of pixel expansion. Pixel expansion occurs when each share is larger than the input secret image. Therefore, greater n values lead to greater expansion values. To remedy the restrictions of pixel expansion, a new scheme was proposed by Fang et al. in 2008 [10]. In their scheme, they adopted a Hibert curve [11] and two queues to enable all generated shares to remain the same size as the secret image. However, the size of the secret images was restricted to $2^r \times 2^r$ pixels, where r is a positive integer. Subsequently, studies have been conducted on the nonexpansion VSS scheme problem. In 2009, Shyu proposed a random grid (RG)-based (n, n)-threshold VSS scheme for binary, gray-level, and color images [12] in accordance with an idea proposed by [13]. Chen and Tsao developed an RG-based (2, n)-threshold VSS scheme [14] in the same year. Later, some researchers considered the use of XOR operations in their (k, n)-threshold VSS schemes [15,16]. In 2015, Ref. [17] designed a (2, 2)-threshold VSS scheme that encrypted a secret image four-pixel blocks at a time. The main idea was inspired by [3], but no pixel expansion occurred. However, they neither analyzed the security of their scheme nor calculated the contrast of the recovered image.

At the same time, some schemes related to the progressive VSS (PVSS) scheme have been proposed in recent years [18–29]. The term "progressive" indicates that the greater the number of shares stacked together, the clearer the restored image is. In 2011, Hou and Quan proposed a new (2, n)-threshold PVSS scheme [22] that seemed to be inspired by the (2, n)-threshold scheme of [3]. In the same year, Chen and Juan established a general (4, n)-threshold PVSS scheme [23]. Chen and Tsao [24] produced more general results for the (k, n)-threshold PVSS scheme with unexpanded shares in 2011. They designed their scheme using a random grid [13]. Several studies [25–27] have subsequently continued and extended the work on the scheme [24]. Moreover, Wan et al. proposed a VSS scheme based on a QR code (VSSQR) with a (k, n)-threshold in 2018 [28]. In the VSSQR scheme, the output shares were all valid QR codes that could be scanned and decoded through the utilization of a QR code reader. The secret image can be revealed through stacking and XOR decryptions. Because the (k, n)-threshold PVSS scheme they applied in the VSSQR scheme was [24], its performance was similar. Yan et al. designed their (k, n)-threshold PVSS scheme with unexpanded shares [29] based on [22]. However, the performance of these schemes can still be improved.

Some studies on the secret-image sharing problem have not been based on VC. Most of these schemes can recover high-quality images but require complicated cryptographic computations. Liu et al. used three Boolean operations, bit-level XOR, COV(1, 7, 3) from (7, 4), a Hamming code, and COV(2, 8, 4) from an (8, 4) shortened Hamming code, to propose three (k, n)-threshold progressive secret-image sharing schemes [30]. Their scheme operations are much more than typically efficient. However, the secret image cannot be recovered unless human vision is aided by additional computation.

All of the foregoing studies have confronted either the problem of pixel expansion or the topic of performance. As a result, this study proposed a novel (k, n)-threshold PVSS scheme with unexpanded shares and higher contrast. The remainder of the study is organized as follows. In Section 2, we present relevant research, and the details of the main proposed scheme are provided in Section 3. The experimental results are presented in Section 4. Analyses and comparisons are provided in the last section.

2. Related Works

Two properties are essential to a VSS scheme: visually recognizable and security [19]. For measuring the visual quality, which determines how well human eyes can recognize the recovered image, the recovered secret image S' corresponding to the original secret image S is evaluated through contrast. Contrast is commonly defined in the following two respects [8,10,18,19]

Definition 1. *Light transmittance of the image S is denoted as* $T(S)$, *and* $T(S) = w/t = 1 - (b/t)$, *where b is the number of the black pixel, and w is the number of the white pixel, and* $t = w + b$.

Definition 2. *The contrast of the recovered secret image S′ corresponding to the original secret image S is defined as*

$$\alpha_1 = \frac{T(S'[S_0]) - T(S'[S_1])}{t} \text{ and } \alpha_2 = \frac{T(S'[S_0]) - T(S'[S_1])}{1 + T(S'[S_1])}$$

where $T(S'[S_0])$ *(resp.,* $T(S'[S_1])$*) denotes the average light transmittance of the area in S′ which correspond to the white (resp., black) area of the original secret image S; and t is referred to the pixel expansion.*

Regardless of whether we use the definition of α_1 or α_2, larger values produce higher visual quality. In the following three definitions, we use α to represent α_1 or α_2.

Definition 3 (Visually recognizable). *In a (k, n)-threshold VSS scheme, the recovered secret image S' could be recognized as the corresponding original secret image S if $\alpha > 0$ when staking more than or equal to k shares.*

Definition 4 (Security). *In a (k, n)-threshold VSS scheme, the scheme is secure if $\alpha = 0$ when staking less than k shares, which means no information of S could be recognized through S'.*

For "progressive," we provide the following definition.

Definition 5 (Progressive). *In a (k, n)-threshold PVSS scheme, when the number of shares collected is greater than or equal to k, the more the share is staked, the larger the contrast α will be.*

In 1995, Naor and Shamir proposed a general (k, n)-threshold VSS scheme [8]. In their scheme, they used the stacking rules presented in Table 1, where 0 (or 1) represents white (or black) and the secret image can be restored by stacking the shares together. The size of the shares generated in [8] was $tm \times n$ pixels, whereas the size of the shares in our scheme was $m \times n$ pixels, where m (n) is the width (height) of the secret image, and t is a positive integer.

Table 1. Stacking rules of two random pixels.

Secret Image	Share 1	Share 2	Staked Share
0	0	0	0
	1	1	1
1	0	1	1
	1	0	1

Some types (including (2, 2), (2, n), (k, k), and (k, n)) of general (k, n)-threshold VSS schemes have been introduced as follows.

The first type is a (2, 2)-threshold VSS scheme. A secret image is encoded in the two shares, and only when the two shares are stacked together can the secret image be seen. Based on Table 1, they construct the two matrices C_0 and C_1,

$$P_0 = \{\text{all the matrices obtained by permuting the column of } C_0 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}\}$$

$$P_1 = \{\text{all the matrices obtained by permuting the column of } C_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\}$$

where C_0 represents the matrix for the white pixel, and C_1 represents the matrix for the black pixel. Entries in each matrix C_0 or C_1 are classified into two values, 0 and 1, where 1 is the black pixel and 0 is the white pixel. The first row in each matrix refers to the pixel of share 1, and the pixel of share 2 is indicated in the second row. In this type, t is equal to 2. As a result, the size of all shares is $2m \times n$ pixels, where m is the width and n is the height of the secret image.

The second type is a (2, n)-threshold VSS scheme. A dealer provides a share of the secret to n users, but only when specific conditions are fulfilled are users able to restore the confidential information in the secret image from their shares. If the users collect at least two shares and stack them together, then they can obtain the secret. The two matrices C_0 and C_1 were designed as follows. Matrices C_0 and C_1 are $n \times n$ matrices.

$$P_0 = \{\text{all the matrices obtained by permuting the column of } C_0 = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix}\}$$

$$P_1 = \{\text{all the matrices obtained by permuting the column of } C_1 = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}\}$$

In total, n shares are generated, and each row in either C_0 or C_1 is randomly selected by each share. We can see that each row contains one black pixel and $n-1$ white pixels. When any k (≥ 2) shares are stacked together, C_0 contains one black and $n-1$ white pixels, and C_1 contains k black and $n-k$ white pixels. The difference between C_0 and C_1 reveals the secret image. The size of the shares is $nm \times n$ pixels.

The (k, k)-threshold VSS scheme is the third type; in this scheme, a secret is divided into k shares, and the secret information is not revealed when fewer than k shares are stacked. Furthermore, two $k \times 2^{k-1}$ matrices C_0 and C_1 are constructed. The sum of the 1's in each column in C_0 is even, and the sum of the 1's in each column in C_1 is odd. With similar characteristics to those described in the (2, n)-threshold, the contrast in this case is equal to $1/2^{k-1}$, and the size of the shares is $2^{k-1}m \times n$ pixels.

Let C be a (k, k)-threshold VSS scheme and H be a collection of k-wise independent hash functions [31–33]. Noar and Shamir constructed the k-out-of-n VSS scheme C' from C and H. In this (k, n)-threshold VSS scheme, the size of the shares is $(n^k \times 2^{k-1}m) \times n$ pixels, and the contrast reaches $2e^{-k}/\sqrt{2\pi k}$. As shown above, all schemes suffered from pixel expansion, regardless of the type.

For managing share expansion, Fang et al. proposed a scheme [10] that could generate nonexpanded shares and in which the combination of shares was the same size as the secret image. They used C_0 and C_1, constructed with a VSS scheme, and the two queues were created using a Hilbert curve [11]. A Hilbert curve is a specific path that passes through $2^k \times 2^k$ pixels for any positive integer k, as illustrated in Figure 1.

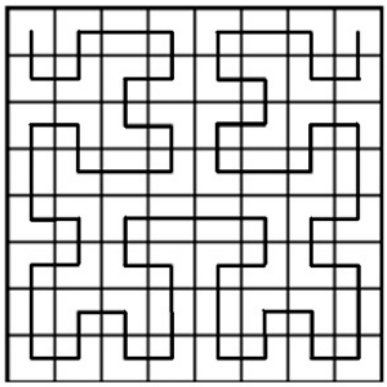


Figure 1. Hilbert-curve for $r = 3$.

When employing a Hilbert curve for a square secret image, all pixels are scanned sequentially. In [10], these researchers prepared two queues, one for storing white pixels and one for storing black pixels. Accordingly, these researchers proposed two algorithms, one that was one dimension and another that was two dimensions, in which the one-dimensional algorithm concerns the line and

the two-dimensional algorithm concerns the image. However, problems occur when an image is not $2^k \times 2^k$, and some of the remaining pixels will be difficult to process.

In 2011, Chen and Tsao [24] proposed a (k, n)-threshold PVSS scheme with unexpanded shares based on a random grid. Wu and Sun [25], Guo et al. [26], and Yan et al. [27] have all continued and extended work on the scheme [24]. Of these studies, in 2018 Yan et al. reported the most favorable results. Their primary algorithm was as follows (Algorithm 1 (YLY scheme)), where the symbol $\oplus$ denotes the Boolean XOR operation.

Algorithm 1. (YLY scheme [27])

Input: A binary secret image S with size $w \times h$ pixels and the threshold parameters n and k
Output: n shares $SC_1, SC_2, \ldots, SC_n$, each with size $w \times h$ pixels
Step 1: For each position $(i, j) \in \{(i, j) \mid 1 \le i \le w, 1 \le j \le h\}$, repeat Steps 2–6.
Step 2: Select $b_1, b_2, \ldots, b_k \in \{0, 1\}$ randomly.
Step 3: If $S(i, j) = b_1 \oplus b_2 \oplus \ldots \oplus b_k$, go to Step 5; else go to Step 4
Step 4: Randomly select $p \in \{1, 2, \ldots, k\}$ let $b_p = 1 - b_p$ (that is, $0 \to 1$ or $1 \to 0$).
Step 5: Set $b_{k+1} = b_1, b_{k+2} = b_2, \ldots, b_{2k} = b_k, b_{2k+1} = b_1, \ldots$ if $(n \bmod k) = 0$, $b_n = b_k$ else $b_n = b_{(n \bmod k)}$.
Step 6: Randomly rearrangement $b_1, b_2, \ldots, b_n$ to $SC_1(i, j), SC_2(i, j), \ldots, SC_n(i, j)$.
Step 7: Output the n shares $SC_1, SC_2, \ldots, SC_n$.

In 2011, Chen et al. proposed another $(4, n)$-threshold VSS scheme [23] using a combinatorial idea. For the two matrices C_0 and C_1 constructed in [23], fewer than four shares stacked together produce a transmittance of C_0 equal to that of C_1; therefore, no secret is revealed. When superimposing more than three shares, the transmittance of C_0 is greater than that of C_1. As a result, the secret from the stacked shares is revealed. Their scheme exhibited favorable performance in comparison with the schemes proposed in [3,10,24]. We used a similar concept and devised an advanced (k, n)-threshold PVSS scheme in this study.

3. Our Scheme

Before detailing our proposed (k, n)-threshold progressive SSS, some definitions and lemmas must be provided. For convenience, let

$$\begin{pmatrix} n \\ j \end{pmatrix} = \begin{cases} \frac{n!}{(n-j)! \times j!}, & 0 \le j \le n \\ 0, & \text{otherwise} \end{cases}$$

Definition 6. *The Hamming weight of a column vector (respectively, a row vector) is the sum of each entry in this column (respectively, row).*

Definition 7. *An $n \times m$ 0–1 matrix $M(n, j)$ is totally symmetric where $m = \begin{pmatrix} n \\ j \end{pmatrix}$ if each column has the same Hamming weight j and has different column vector among each column.*

Definition 8. *An $n \times (m_A + m_B)$ combined matrix $[A \mid\mid B]$ is obtained by concatenating matrix A and B, where A is an $n \times m_A$ matrix, and B is an $n \times m_B$ matrix. A matrix $[a \times A \mid\mid b \times B]$ is generated by concatenating A for a times and B for b times.*

Lemma 1. *Given an $n \times m$ totally symmetric matrix $A = M(n, j)$, $f_i(A)$ is defined as the Hamming weight of the row vector that results from applying the OR operation for any i rows in A, where $i = 1, 2, \ldots, n$. As a result, $f_i(A) = f_i(M(n, j)) = \begin{pmatrix} n \\ j \end{pmatrix} - \begin{pmatrix} n-i \\ j \end{pmatrix}$.*

Proof of Lemma 1. The number of columns in $M(n, j) = \begin{pmatrix} n \\ j \end{pmatrix}$. When we selected any i rows in matrix A, the resultant entry from applying OR operation for these row vectors remains zero if one column vector has all zeros in these i rows. For any one column vector with all zeros in these i rows, the j ones must be present in the other $n - i$ rows. As a result, the number of those column vectors is $\begin{pmatrix} n-i \\ j \end{pmatrix}$. □

Lemma 2. *$f_i(A \,||\, B) = f_i(A) + f_i(B)$ for any two totally symmetric matrices A and B with sizes $n \times m_1$ and $n \times m_2$, respectively.*

Proof of Lemma 2. A matrix $[A \,||\, B]$ is obtained by concatenating matrices A and B, where $A = M(n, j_A)$ and $B = M(n, j_B)$. Then, $f_i(A \,||\, B) = f_i([M(n, j_A) \,||\, M(n, j_B)])$. Because using a Hamming weight for the row vector resulting from applying the OR operation for some i rows of $[A \,||\, B]$ is equal to that $[A \,||\, B]$ be divided into two parts A, B and doing the same thing to these two parts, then add those two results together. For this reason, $f_i([M(n, j_A) \,||\, M(n, j_B)]) = f_i([M(n, j_A)]) + f_i([M(n, j_B)])$. That is, $f_i(A \,||\, B) = f_i(A) + f_i(B)$. □

Let $[A_1 \,||\, A_2 \,||\, ... \,||\, A_k] = [[\ldots\ [A_1 \,||\, A_2] \,||\, A_3] \,||\, \ldots \,||\, A_k]$ for any totally symmetric matrix $A_1, A_2, \ldots, A_k$, where the size of A_i is $n \times m_i$ for $i = 1, 2, \ldots, k$. We have the following corollary.

Corollary 3. *For all k totally symmetric matrices $A_1, A_2, \ldots, A_k$, where $A_p = M(n, j_p)$, for some $0 \leq j_p \leq n$ for any $1 \leq p \leq k$, let $[A_1 \,||\, A_2 \,||\, ... \,||\, A_k] = B$. Then, $f_i(B) = f_i(A_1) + f_i(A_2) + \ldots + f_i(A_k)$ for any $1 \leq i \leq n$.*

Definition 9. *For any $1 \leq p \leq k$, given totally symmetric matrices $A_p = M(n, j_p)$, where $0 \leq j_p \leq n$, let $[A_1 \,||\, A_2 \,||\, \ldots \,||\, A_k] = B$, where B is an $n \times m$ matrix. Define $T(B, i) = 1 - (f_i(B)/m)$, where $1 \leq i \leq n$.*

When conducting a (k, n)-threshold SSS, two matrices, C_0 and C_1, should first be constructed according to the white and black pixels, respectively, in the secret image. In our scheme, the following two conditions must be met, in accordance with Definitions 2–4, because no pixel expansion to occur.

$$T(C_0,t) = T(C_1,t), \text{ for } 1 \leq t \leq k-1 \tag{1}$$

$$T(C_0,t) > T(C_1,t), \text{ for } t \geq k \tag{2}$$

In the first condition, when adopting the OR operation for any t rows of C_0, the Hamming weight is equal to that of C_1 for $1 \leq t \leq k - 1$. This indicates that when stacking fewer than k shares together, the light transmittance between C_0 and C_1 is the same. As a result, the secret image is not revealed under this condition. Regarding the second condition, C_0 has a higher light transmittance than C_1 when t is greater than or equal to k, enabling us to decode the confidential information in the secret image. According to these two rules, we provide the following main algorithm (Algorithm 2), referred to as the CHJ scheme. The case in [23] is a special case for our scheme, where $k = 4$. Proof of the scheme's accuracy is presented in Section 3.1.

Algorithm 2. (CHJ scheme)

Input: A binary secret image S with size $w \times h$ pixels and the threshold parameters n and k
Output: n shares $R_1, R_2, \ldots, R_n$, each with size $w \times h$ pixels

if (k mod 2 == 1)

$$C_0 = [M(n,2) \,||\, \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} \times M(n,0) \,||\, \sum_{t=1}^{(k-3)/2} (\begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \times M(n,\ n-2t+1))]$$

$$C_1 = [(n-k+1) \times M(n,1) \,||\, \begin{pmatrix} n-3 \\ k-3 \end{pmatrix} \times M(n,n) \,||\, \sum_{t=1}^{(k-3)/2} (\begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \times M(n,\ n-2t))]$$

$$m = \begin{pmatrix} n \\ 2 \end{pmatrix} + \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} + \sum_{t=1}^{(k-3)/2} (\begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \begin{pmatrix} n \\ n-2t+1 \end{pmatrix})]$$

else

$$C_0 = [M(n,2) \,||\, \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} \times M(n,0) \,||\, \begin{pmatrix} n-3 \\ k-3 \end{pmatrix} \times M(n,n) \,||\, \sum_{t=1}^{k/2-2} (\begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \times M(n,\ n-2t))]$$

$$C_1 = [(n-k+1) \times M(n,1) \,||\, \sum_{t=1}^{k/2-2} (\begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \times M(n,\ n-2t+1))]$$

$$m = \begin{pmatrix} n \\ 2 \end{pmatrix} + \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} + \begin{pmatrix} n-3 \\ k-3 \end{pmatrix} + \sum_{t=1}^{k/2-2} (\begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \begin{pmatrix} n \\ n-2t \end{pmatrix})]$$

```
for (int j = 1; j ≤ h; j++)
    for (int i = 1; i ≤ w; i++)
        x = random(1, m)
        for (int t = 1; t ≤ n; t++)
            if (S(i, j) == 0)
                R_t(i, j) = C_0(t, x)
            Else
                R_t(i, j) = C_1(t, x)
```

3.1. Proof of the CHJ Scheme

In this subsection, we aim to determine the accuracy of the CHJ scheme. That is, we want to demonstrate that the CHJ scheme is a (k, n)-threshold PVSS scheme.

Theorem 4. *In the CHJ scheme, stacking at least k shares reveals the secret; while stacking fewer than k shares does not reveal the secret. The CHJ scheme is therefore a (k, n)-threshold VSS scheme.*

We used light transmittance to prove that the secret cannot be discerned if fewer than k shares are stacked, but the image can be viewed if at least k shares are stacked together. The proof is divided into two cases according to whether k is odd or even. In other words, we must prove that our scheme satisfies (1) and (2). We know that $T(A, i) = 1 - (f_i(A)/m)$ when $A = C_0$ or C_1; therefore, two formulas must be discussed in each case: the denominator m and the numerator $f_i(A)$. Because the entire proof of Theorem 4 is mathematical and encompasses more contexts, we only present an outline for proving the CHJ scheme in this study. First, regardless of how many shares are stacked, we prove that the denominator m of the light transmittance for the white and black pixels of the stacked image is always the same. That is,

$$\begin{pmatrix} n \\ 2 \end{pmatrix} + \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} + \sum_{t=1}^{(k-3)/2} \begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \begin{pmatrix} n \\ n-2t+1 \end{pmatrix} = n(n\ \ k+1) + \begin{pmatrix} n-3 \\ k-3 \end{pmatrix} + \sum_{t=1}^{(k-3)/2} \begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \begin{pmatrix} n \\ n-2t \end{pmatrix} \text{ when } k \text{ is odd}$$

and

$$\begin{pmatrix} n \\ 2 \end{pmatrix} + \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} + \begin{pmatrix} n-3 \\ k-3 \end{pmatrix} + \sum_{t=1}^{k/2-2} \begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \begin{pmatrix} n \\ n-2t \end{pmatrix} = n(n-k+1) + \sum_{t=1}^{k/2-2} \begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \begin{pmatrix} n \\ n-2t+1 \end{pmatrix} \text{ when } k \text{ is even.}$$

Second, we prove that the numerators of the light transmittance $f_i(A)$ for the white and black pixels of the stacked image are different when we stack at least k shares ($i \geq k$). However, the light transmittance $f_i(A)$ for the white and black pixels of the stacked image is the same when stacking fewer than k shares ($i < k$). After providing such proof, we certified the CHJ scheme is a (k, n)-threshold VSS sharing scheme. Notably, the principle of mathematical induction and the Pascal theorem, $\begin{pmatrix} n \\ r \end{pmatrix} = \begin{pmatrix} n-1 \\ r \end{pmatrix} + \begin{pmatrix} n-1 \\ r-1 \end{pmatrix}$ for any positive integer $1 \leq r \leq n$, are used in the proof.

Theorem 5. *In the CHJ scheme, when we stack at least k shares, progressively stacking more shares results in greater clarity in the recovered image. That is, the CHJ scheme is demonstrably a PVSS scheme.*

Proof of Theorem 5. According to Definition 5, we were required to demonstrate that α is larger when stacking more shares (more than k). Here, we only use the α_1 in Definition 2 to represent α, and the proof for α_2 is similar to this proof. We therefore prove that the difference between the light transmittance of the white and black pixels is greater when we stack more shares. That is, $[T(C_0, i+1) - T(C_1, i+1)]$ is greater than $[T(C_0, i) - T(C_1, i)]$ for $i \geq k$. Because $[T(C_0, i+1) - T(C_1, i+1)] - [T(C_0, i) - T(C_1, i)] = [f_{i+1}(C_1)/m - f_{i+1}(C_0)/m] - [f_i(C_1)/m - f_i(C_0)/m]$, we can prove $[f_{i+1}(C_1) - f_{i+1}(C_0)] - [f_i(C_1) - f_i(C_0)] > 0$. We divide the proof into two cases for when k is odd and even. The proof for the second case, when k is even, is similar to that of the first case; we therefore prove only the first case, for when k is odd. □

We know, by definition, that $f_i(M(n, j)) = \begin{pmatrix} n \\ j \end{pmatrix} - \begin{pmatrix} n-i \\ j \end{pmatrix}$. Recall that when k is odd,

$$C_0 = [M(n,\ 2)\ ||\ \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} \times\ M(n,\ 0)\ ||\ \sum_{t=1}^{\frac{k-3}{2}} \left(\begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \times M(n,\ n-2t+1) \right)],$$

$$C_1 = [(n-k+1) \times\ M(n,\ 1)||\begin{pmatrix} n-3 \\ k-3 \end{pmatrix} \times\ M(n,\ n)\ ||\ \sum_{t=1}^{\frac{k-3}{2}} \left(\begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \times M(n,\ n-2t) \right)].$$

We first compute the difference between $f_x(C_1)$ and $f_x(C_0)$. By Corollary 3,

$$\begin{aligned} f_x(C_1) - f_x(C_0) &= \left\{ (n-k+1) \left[\begin{pmatrix} n \\ 1 \end{pmatrix} - \begin{pmatrix} n-x \\ 1 \end{pmatrix} \right] + \begin{pmatrix} n-3 \\ k-3 \end{pmatrix} \left[\begin{pmatrix} n \\ n \end{pmatrix} - \begin{pmatrix} n-x \\ n \end{pmatrix} \right] \right. \\ &\quad \left. + \sum_{t=1}^{\frac{k-3}{2}} \left[\begin{pmatrix} n-2t-3 \\ k-2t-3 \end{pmatrix} \left(\begin{pmatrix} n \\ n-2t \end{pmatrix} - \begin{pmatrix} n-x \\ n-2t \end{pmatrix} \right) \right] \right\} \\ &\quad - \left\{ \left[\begin{pmatrix} n \\ 2 \end{pmatrix} - \begin{pmatrix} n-x \\ 2 \end{pmatrix} \right] + \begin{pmatrix} n-k+2 \\ 2 \end{pmatrix} \left[\begin{pmatrix} n \\ 0 \end{pmatrix} - \begin{pmatrix} n-x \\ 0 \end{pmatrix} \right] \right. \\ &\quad \left. + \sum_{t=1}^{\frac{k-3}{2}} \left[\begin{pmatrix} n-2t-2 \\ k-2t-2 \end{pmatrix} \left(\begin{pmatrix} n \\ n-2t+1 \end{pmatrix} - \begin{pmatrix} n-x \\ n-2t+1 \end{pmatrix} \right) \right] \right\} \end{aligned}$$

$$
\begin{aligned}
&f_x(C_1) - f_x(C_0) \\
&= \left\{ (n-k+1)\binom{n}{1} + \binom{n-3}{k-3}\binom{n}{n} + \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-3}{k-2t-3}\binom{n}{n-2t} \right] - \binom{n}{2} \right. \\
&\left. - \binom{n-k+2}{2}\binom{n}{0} - \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n}{n-2t+1} \right] \right\} \\
&- \left\{ (n-k+1)\binom{n-x}{1} + \binom{n-3}{k-3}\binom{n-x}{n} \right. \\
&+ \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-3}{k-2t-3}\binom{n-x}{n-2t} \right] - \binom{n-x}{2} - \binom{n-k+2}{2}\binom{n-x}{0} \\
&\left. - \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n-x}{n-2t+1} \right] \right\} \\
&= \{\text{the denominator of } T(C_1, x) - \text{the denominator of } T(C_0, x)\} \\
&-(n-k+1)\binom{n-x}{1} - 0 - \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-3}{k-2t-3}\binom{n-x}{n-2t} \right] + \binom{n-x}{2} \\
&+ \binom{n-k+2}{2} + \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n-x}{n-2t+1} \right] \\
&= \binom{n-x}{2} + \binom{n-k+2}{2} - (n-k+1)\binom{n-x}{1} \\
&+ \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n-x}{n-2t+1} - \binom{n-2t-3}{k-2t-3}\binom{n-x}{n-2t} \right]
\end{aligned}
$$

Replace x by $i + 1$ and i, we have

$$
\begin{aligned}
&[f_{i+1}(C_1) - f_{i+1}(C_0)] \quad -[f_i(C_1) - f_i(C_0)] \\
&= \left\{ \binom{n-i-1}{2} + \binom{n-k+2}{2} - (n-k+1)\binom{n-i-1}{1} \right. \\
&\left. + \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n-i-1}{n-2t+1} - \binom{n-2t-3}{k-2t-3}\binom{n-i-1}{n-2t} \right] \right\} \\
&- \left\{ \binom{n-i}{2} + \binom{n-k+2}{2} - (n-k+1)\binom{n-i}{1} \right. \\
&\left. + \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n-i}{n-2t+1} - \binom{n-2t-3}{k-2t-3}\binom{n-i}{n-2t} \right] \right\}
\end{aligned}
$$

We can simplify the equation to $(i-k+2) - \sum_{t=1}^{\frac{k-3}{2}} \left[\binom{n-2t-2}{k-2t-2}\binom{n-i-1}{n-2t} - \binom{n-2t-3}{k-2t-3}\binom{n-i-1}{n-2t-1} \right]$. Because $i \geq k$ (≥ 3), so $n-i-1 \leq n-k-1 < n-k+2 \leq n-2t-1 < n-2t$, for any $t = 1, 2, \ldots, (k-3)/2$. And $\binom{n-i-1}{n-2t} = \binom{n-i-1}{n-2t-1} = 0$. Therefore, this equation becomes $i - k + 2 > 0$, and $[T(C_0, i+1) - T(C_1, i+1)] > [T(C_0, i) - T(C_1, i)]$ can be concluded for when k is odd.

4. Experimental Results

This study proposes an advanced PVSS scheme, the CHJ scheme, that can encode a secret image into n (≥ 2) shares without pixel expansion, and in which the hidden secret can be restored only when k ($\leq n$) shares are stacked together. To demonstrate the feasibility of the proposed CHJ scheme, we conducted five experiments in the following order: (4, 5)-, (4, 6)-, (4, 7)-, (5, 6)-, and (5, 7)-threshold PVSS schemes.

4.1. Simulation 1: (4, 5)-Threshold PVSS Scheme

In the first experiment, the parameter k was set at 4, and n was set at 5; C_0 and C_1 could be generated by the CHJ scheme described in Section 3 as follows.

$$C_0 = [M(n,\ 2)\ ||\ (\mathrm{n}\ -\ 3)\ \times\ \mathrm{M}(n,\ n)\ ||\ (\frac{\mathrm{n}^2 - 5\mathrm{n} + 6}{2}) \times\ \mathrm{M}(n,\ 0)] = [M(5,\ 2)\ ||\ 2\ \times\ \mathrm{M}(5,\ 5)\ ||\ 3\ \times\ \mathrm{M}(5,\ 0)]$$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$C_1 = [(n - 3) \times M(n, 1)\ ||\ M(n, n - 1)] = [2 \times M(5, 1)\ ||\ M(5, 4)]$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

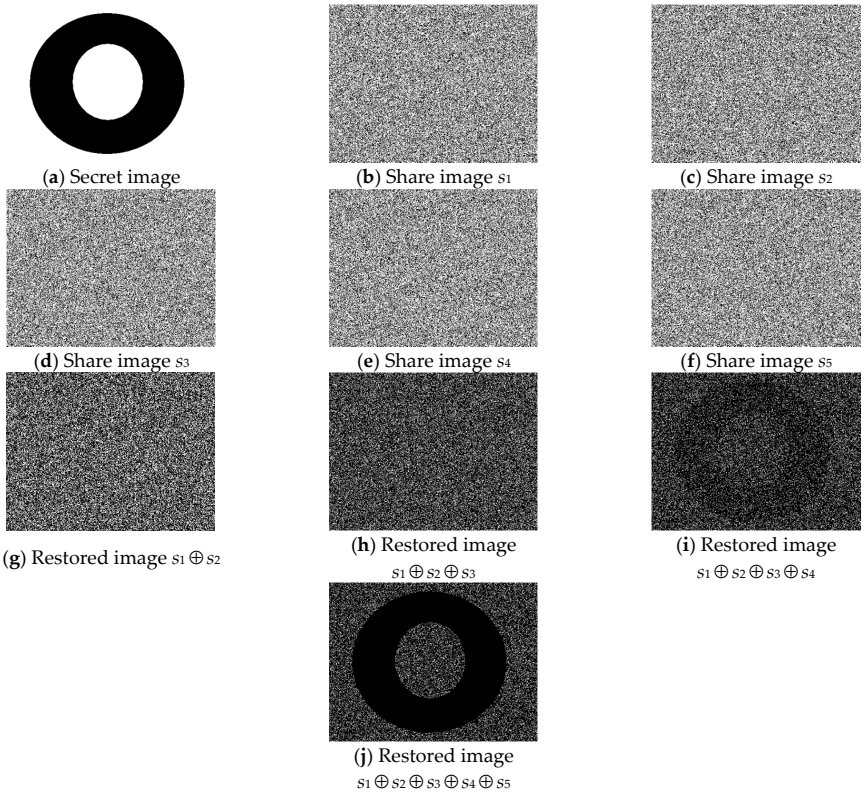
Figure 2a is the input secret image with size 449 × 341 pixels, and the five shares (G_1, G_2, G_3, G_4, and G_5) generated by the CHJ scheme are displayed in Figure 2b–f. The result of stacking the two shares is presented in Figure 2g. The result of stacking any other two shares was nearly the same; therefore, we only present one combined image of stacking G_1 and G_2. Any two shares stacked together revealed no information regarding the secret image. The simulation of superimposing three shares is presented in Figure 2h, and we only display the combination of G_1, G_2, and G_3. Furthermore, the confidential message could not be seen in such a case. Figure 2i presents the result of stacking G_1, G_2, G_3, and G_4. From watching this simulation, we were able to distinguish the secret image. By superimposing all of the shares, we achieved greater clarity when viewing the secret image, as displayed in Figure 2j. A greater number of stacked shares (must greater than or equal to the threshold value k) produced a clearer restored image.

4.2. Simulation 2: (4, 6)-Threshold PVSS Scheme

In the second experiment, we encrypted a secret image with the size of 449 × 341 pixels into six shares (G_1, G_2, G_3, G_4, G_5, and G_6), as displayed in Figure 3b–g, respectively. Regarding the threshold value k = 4, we expected that the secret image could be seen when stacking more than three shares. The result of stacking G_1 and G_2 is depicted in Figure 3h, and Figure 3i is the result of stacking G_1, G_2, and G_3 together. The result of stacking G_1, G_2, G_3, and G_4 is displayed in Figure 3j; stacking of G_1, G_2, G_3, G_4, and G_5 is displayed in Figure 3k; and stacking of G_1, G_2, G_3, G_4, G_5, and G_6 is presented in Figure 3l. The secret image can be seen in Figure 3j–l with varying clarity.

4.3. Simulation 3: (4, 7)-Threshold PVSS Scheme

Regarding the (4, 7)-threshold PVSS scheme, the image with the size 449 × 341 pixels presented in Figure 4a was encoded into seven individual shares, referred to as G_1, G_2, G_3, G_4, G_5, G_6, and G_7. Figure 4b–h present the cipher shares generated with the CHJ scheme. The result of stacking G_1 and G_2 is displayed in Figure 4i, and Figure 4j is the result of stacking G_1, G_2, and G_3 together. Figure 4k is the result of stacking G_1, G_2, G_3, and G_4, and the result of stacking G_1, G_2, G_3, G_4, and G_5 together is presented in Figure 4l. Figures 4m and 4n are the results of stacking G_1, G_2, G_3, G_4, G_5, and G_6 and stacking all of the shares together, respectively. Only Figure 4k–n reveals information regarding the secret image.



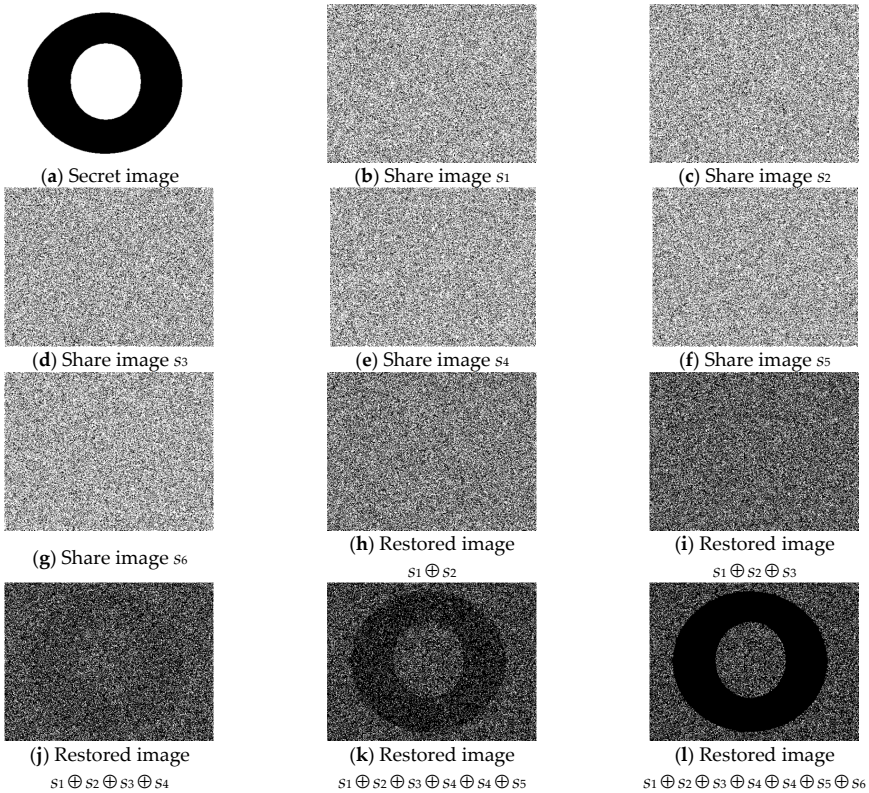
(**a**) Secret image (**b**) Share image s_1 (**c**) Share image s_2

(**d**) Share image s_3 (**e**) Share image s_4 (**f**) Share image s_5

(**g**) Restored image $s_1 \oplus s_2$ (**h**) Restored image $s_1 \oplus s_2 \oplus s_3$ (**i**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4$

(**j**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5$

Figure 2. The experimental results of the proposed (4, 5)-threshold PVSS scheme.



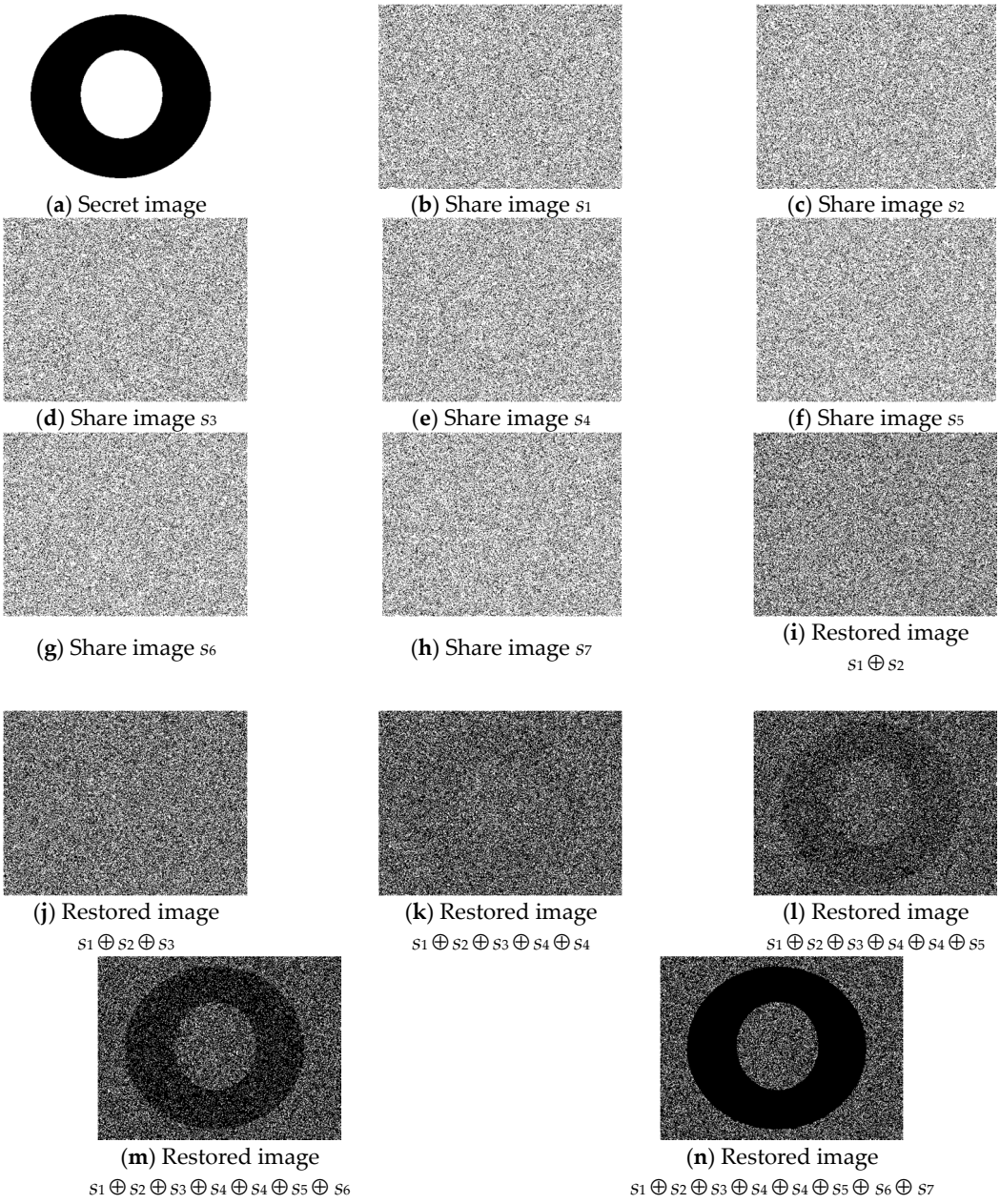
(**a**) Secret image (**b**) Share image s_1 (**c**) Share image s_2

(**d**) Share image s_3 (**e**) Share image s_4 (**f**) Share image s_5

(**g**) Share image s_6 (**h**) Restored image $s_1 \oplus s_2$ (**i**) Restored image $s_1 \oplus s_2 \oplus s_3$

(**j**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4$ (**k**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5$ (**l**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5 \oplus s_6$

Figure 3. The experimental results of the proposed (4, 6)-threshold VSS scheme.



(**a**) Secret image (**b**) Share image s_1 (**c**) Share image s_2

(**d**) Share image s_3 (**e**) Share image s_4 (**f**) Share image s_5

(**g**) Share image s_6 (**h**) Share image s_7 (**i**) Restored image $s_1 \oplus s_2$

(**j**) Restored image $s_1 \oplus s_2 \oplus s_3$ (**k**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_4$ (**l**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_4 \oplus s_5$

(**m**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_4 \oplus s_5 \oplus s_6$ (**n**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_4 \oplus s_5 \oplus s_6 \oplus s_7$

Figure 4. The experimental results of the proposed (4, 7)-threshold VSS scheme.

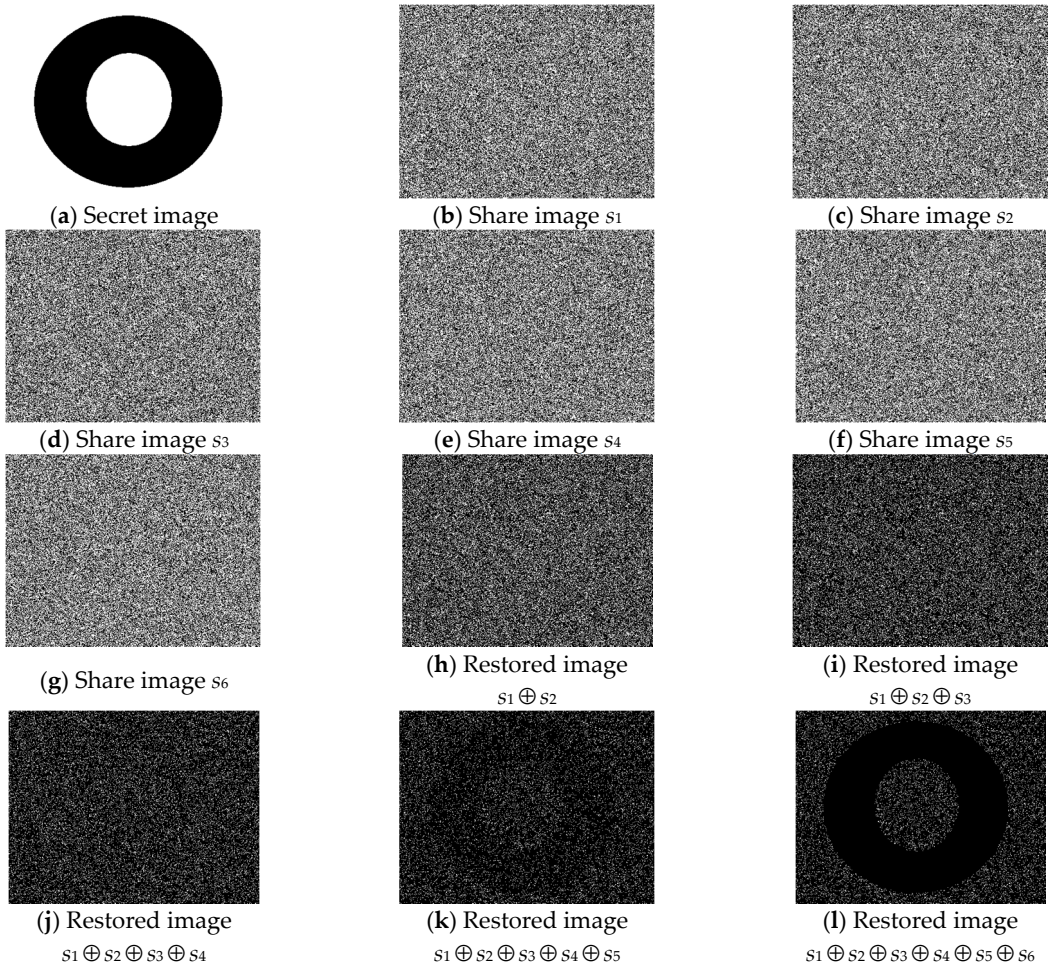
4.4. Simulation 4: (5, 6)-Threshold PVSS Scheme

In the fourth experiment, we encrypted the image with the size of 449 × 341 pixels displayed in Figure 5a; Figure 5b–g are the cipher shares (G_1, G_2, G_3, G_4, G_5, and G_6, respectively). When stacking G_1 and G_2, we achieved the result depicted in Figure 5h. Staking the three shares G_1, G_2, and G_3 produced the result displayed in Figure 5i. The result of stacking G_1, G_2, G_3, and G_4 is shown in Figure 5j. The result of stacking five shares together, G_1, G_2, G_3, G_4, and G_5, is displayed in Figure 5k. Figure 5l reveals the result of stacking all of the shares together. For the (5, 6)-threshold PVSS scheme, the secret image could only be seen when more than four shares were stacked.

4.5. Simulation 5: (5, 7)-Threshold PVSS Scheme

In the last experiment, which involved the (5, 7)-threshold PVSS scheme, an image with a size of 449 × 341 pixels, as presented in Figure 6a, was encrypted into seven shares (G_1, G_2, G_3, G_4, G_5, G_6,

and G_7) as displayed in Figure 6b–h, respectively. The results of stacking fewer than five shares are indicated in Figure 6i–k. Figures 6l and 6m present the results of stacking G_1, G_2, G_3, G_4, and G_5 and G_1, G_2, G_3, G_4, G_5, and G_6, respectively. When stacking all of the shares, the result presented in Figure 6n. Similarly, when more than four shares were stacked together, we could see the secret image.



(**a**) Secret image (**b**) Share image s_1 (**c**) Share image s_2
(**d**) Share image s_3 (**e**) Share image s_4 (**f**) Share image s_5
(**g**) Share image s_6 (**h**) Restored image $s_1 \oplus s_2$ (**i**) Restored image $s_1 \oplus s_2 \oplus s_3$
(**j**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4$ (**k**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5$ (**l**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5 \oplus s_6$

Figure 5. The experimental results of the proposed (5, 6)-threshold VSS scheme.

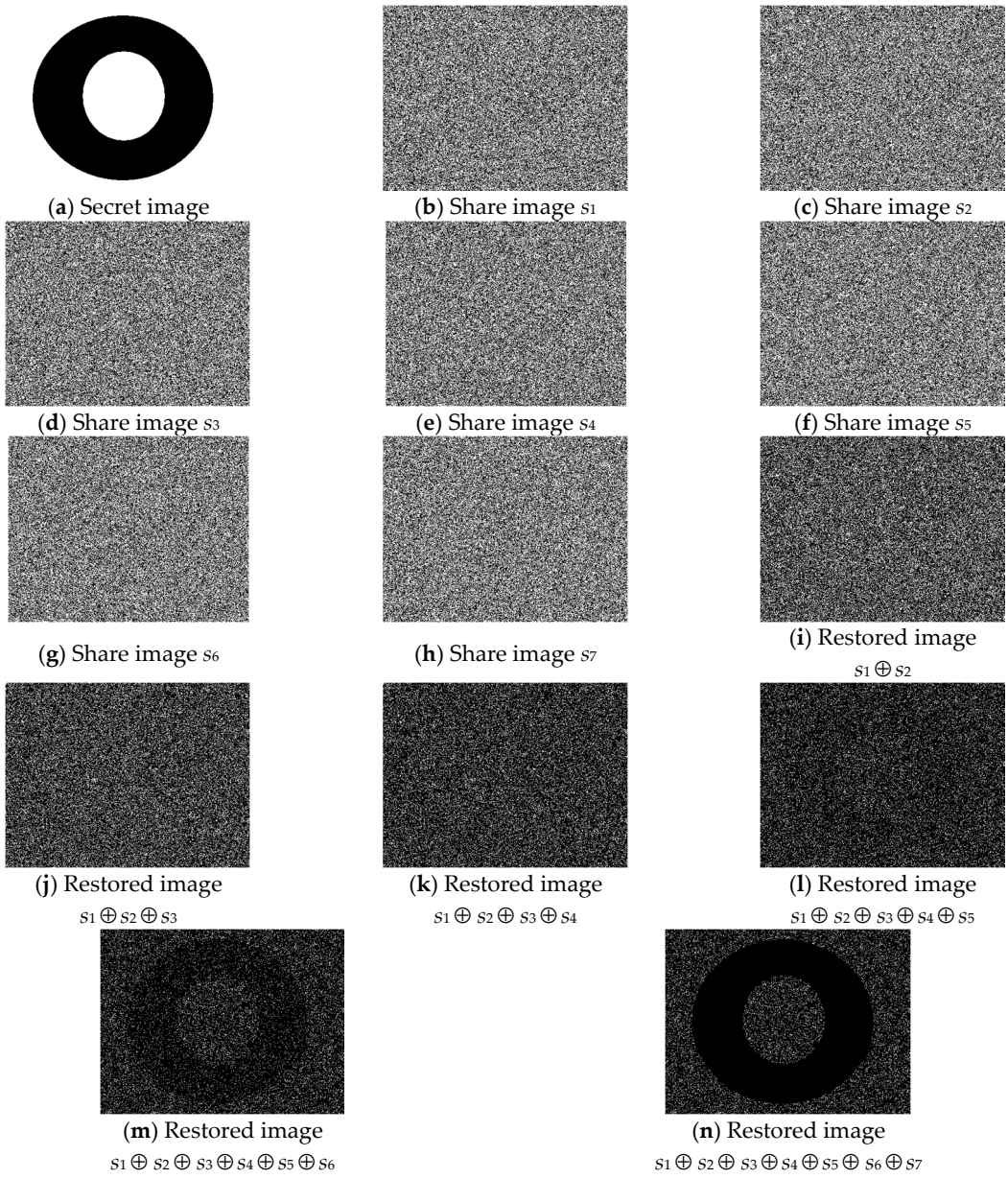
5. Analysis and Comparison

In this paper, we provide an advanced (k, n)-threshold PVSS scheme for any integer $2 \leq k \leq n$. We constructed the CHJ scheme through a technique of combination. The visual and security analyses of our scheme are detailed in Sections 5.1 and 5.2, respectively.

5.1. Visual Analysis

In Section 2, we define two contrasts α_1 and α_2 using the definition of light transmittance when stacking k shares. Because the CHJ scheme has no pixel expansion and in accordance with the design of the algorithm, we use $T(C_0, k)$ and $T(C_1, k)$ to rewrite the definition of contrast.

Property 6. *The contrast* $\alpha_1 = T(C_0, k) - T(C_1, k)$, *and* $\alpha_2 = (T(C_0, k) - T(C_1, k))/(1 + T(C_1, k))$, *when staking k shares, where* C_0 *and* C_1 *are the designed matrices for white and black pixels, respectively, in the proposed CHJ scheme.*



(**a**) Secret image (**b**) Share image s_1 (**c**) Share image s_2

(**d**) Share image s_3 (**e**) Share image s_4 (**f**) Share image s_5

(**g**) Share image s_6 (**h**) Share image s_7 (**i**) Restored image $s_1 \oplus s_2$

(**j**) Restored image $s_1 \oplus s_2 \oplus s_3$ (**k**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4$ (**l**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5$

(**m**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5 \oplus s_6$ (**n**) Restored image $s_1 \oplus s_2 \oplus s_3 \oplus s_4 \oplus s_5 \oplus s_6 \oplus s_7$

Figure 6. The experimental results of the proposed (5, 7)-threshold VSS scheme.

Because [3,10] use the definition of α_1 for contrast, we offer a comparison of α_1 in their scheme and our scheme according to the k and n values in Table 2. Notably, the contrast of the shares generated by [10] was the same as that in the basic scheme that they applied. Because they applied the scheme developed in [3], their scheme exhibits the same contrast performance as [3]. For cases in which $k = 2$ or 3, the contrast of our scheme was the same as that in [3]; therefore, we only discuss the condition of $n \geq k \geq 4$. The value of α_1 in the (4, 5)-threshold SSS of [3] approached $1/4261$, whereas the contrast α_1 in our scheme was $3/15 - 2/15 = 1/15$. For any (k, n) such that $n \geq k \geq 4$, our proposed scheme exhibited superior contrast performance in comparison with (Table 2) [3]. For the scheme in [10], the contrast was dependent on which basic scheme was applied. For example, if [10] is based on [3], then the contrast is the value presented in the second column in Table 2. Furthermore, the contrast is the same as in our scheme if [10] is based on our proposed CHJ scheme.

Table 2. The comparison of contrast α_1 between Ref. [3] and our scheme.

Contrast α_1	Ref. [3]	Our Proposed CHJ Scheme
(2, 2)-threshold SSS	1/2	1/2
(3, 3)-threshold SSS	1/4	1/4
(4, 5)-threshold SSS	≅1/4261	1/15
(4, 6)-threshold SSS	≅1/4261	1/24
(5, 6)-threshold SSS	≅1/12820	1/30
(6, 8)-threshold SSS	≅1/152200	1/128
(7, 8)-threshold SSS	≅1/887707	1/175

For compare to our scheme, those in [24–27,29] have all used the second definition of α_2 for contrast; thus, we conducted another comparison, as presented in Table 3. In Table 3, those values for [29] were calculated by averaging the two values used in their paper. Because no results were reported for the (4, 6)- or (5, 6)-threshold in [29], we wrote "N/A" in the relevant place in Table 3. In [24], the researchers provided a formula for estimating the contrast when stacking t shares together: $(2 \times \binom{t}{k}) / ((2^t + 1) \times \binom{n}{k} - \binom{t}{k})$, where $k \leq t \leq n$. Therefore, we obtained those values for [24] in Table 3 by directly calculating the formula for different k, t, and n values. In [27], the researchers calculated the value of α_2 from their experimental result for all schemes ([24–27]). Therefore, we obtained those values for [25–27], as presented in Table 3. Unfortunately, the value for a large n (≥ 6) was not provided, but we reached the following conclusions based on the algorithms. (1) When $n = k = t$, the contrast α_2 is equal to $1/2^{k-1}$ for all of these schemes. (2) When $N = \lfloor n/k \rfloor = 1$, the schemes in [24,26] are the same. (3) When $q = n \bmod k = 1$, the values of α_2 of the schemes in [25,27] are the same. (4) The contrast of the scheme in [27] is always greater than or equal to that of the others. The experimental results in [27] support the results of our theoretical analysis. We then compared the contrast α_2 of the scheme in [27] with that of our scheme. Although in [27] the exact value α_2 for general (k, n, t) was difficult to obtain (The Principle of Inclusion and Exclusion was required), obtaining the value for $k = t$ was relatively simple. The value was $2p/(2^t + 1 - p)$, where $p = (N + 1)^q N^{k-q} / \binom{n}{k}$. Therefore, some information in Table 3 could be provided.

From the data listed in Table 3, we determined that when $n > 4$, the contrast of the scheme in [24] is small, indicating inadequate performance. In [25–27,29], when $k < 4$, the contrast of the scheme in [29] is similar to that of our scheme. However, when n and k are both ≥ 4 and the value of k is close to that of n, the contrast decreases, as indicated in the bold rows of Table 3. Because our scheme demonstrated higher contrast, it demonstrates superior performance for larger k values. Therefore, our scheme offers superior contrast of stacked images under the general (k, n)-threshold for larger k and n values.

Table 3. The comparison of contrast when stacking t shares together.

Contrast α_2	Ref. [29]	Ref. [24]	Ref. [25]	Ref. [26]	Ref. [27]	Proposed CHJ Scheme
(3, 3)-threshold, $t = 3$	0.24957	0.25	0.249555	0.250342	0.24965	0.25
(3, 4)-threshold, $t = 3$	0.10286	0.0571	0.111718	0.05811	0.11335	0.0909
(3, 4)-threshold, $t = 4$	0.33267	0.125	0.250875	0.126323	0.25314	0.3333
(4, 4)-threshold, $t = 4$	0.12515	0.125	0.124552	0.124741	0.12451	0.125
(3, 5)-threshold, $t = 3$	0.05569	0.0224	0.062646	0.022415	0.0854	0.0454
(3, 5)-threshold, $t = 4$	0.17037	0.0481	0.136758	0.048008	0.18894	0.1578
(3, 5)-threshold, $t = 5$	**0.37517**	0.0625	0.250588	0.062215	0.24851	**0.375**
(4, 5)-threshold, $t = 4$	**0.04495**	**0.0238**	**0.048006**	**0.023466**	**0.04691**	**0.0666**
(4, 5)-threshold, $t = 5$	**0.166885**	**0.0625**	**0.126651**	**0.0616**	**0.12536**	**0.2**
(4,6)-threshold, $t = 4$	**N/A**	**0.0078**	**N/A**	**0.0078**	**0.031873**	**0.0333**
(4, 6)-threshold, $t = 5$	**N/A**	**0.0204**	**N/A**	**0.0204**	**N/A**	**0.11111**
(4, 6)-threshold, $t = 6$	**N/A**	**0.03125**	**N/A**	**0.03125**	**N/A**	**0.25**
(5, 6)-threshold, $t = 5$	**N/A**	**0.01010**	**0.020408**	**0.01010**	**0.020408**	**0.03125**
(5, 6)-threshold, $t = 6$	**N/A**	**0.03125**	**N/A**	**0.03125**	**N/A**	**0.1**

The Hibert curve [11] was not used in our scheme; therefore, unlike in [10], the size of the secret image has no restrictions. The method used in [3] has the problem of pixel expansion; because the shares generated by our scheme are all the same size as the input image, no expansion occurs. The scheme in [23] can only satisfy the conditions of the (4, n)-threshold; however, we can meet the general conditions of the (k, n)-threshold. Notably, [23] is a special case of our proposed scheme. For greater k and n values, the contrast is relatively small for the schemes in [24–27,29]; therefore, our proposed scheme is preferable. Moreover, the output shares of the VSSQR scheme [28] proposed in 2018 all possessed valid QR codes. In comparing their scheme with others, for equality, we only discuss the scheme which the secret image can be revealed visually through stacking. In this kind of view, the performance of their scheme is similar to that of [24]. All comparisons between ours and related schemes are presented in Table 4.

Table 4. The comparison of our scheme with some related schemes.

Reference	[3]	[10]	[23]	[24]	[25–27]	[28]	[29]	Proposed CHJ Scheme
Free size	Yes	No	Yes	Yes	Yes	No	Yes	Yes
Without pixel expansion	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
The value of α	Small	Small/Large	Large	Small	Large when $k < 4$	Small	Large when $k < 4$	Large
For every (k, n)	Yes	Yes	(4, n)	Yes	Yes	Yes	Yes	Yes
Shares are QR code	No	No	No	No	No	Yes	No	No

5.2. Security Analysis

A correlation test was performed to evaluate the security of a secret-image sharing scheme. This test has been widely used in secret-image sharing, such as in [15,34,35]. Through this test, we were able to determine whether high correlations existed among pixels in each share of our proposed scheme. Four types of correlations were observed, those between two horizontally adjacent pixels, two vertically adjacent pixels, two diagonally adjacent pixels, and two vice-diagonally adjacent pixels in the input secret images, with shares computed as usual. The correlation coefficient of two adjacent pixels $R(x, y)$ can be obtained by using the following formula, where x and y are the values of two adjacent pixels in one image, and $w \times h$ is the number of pairs of adjacent pixels:

$$R(x,y) = \frac{S_{xy}}{S_x \times S_y}, S_{xy} = \frac{1}{w \times h} \sum_{i=1}^{w \times h} (x_i - E(x)) \times (y_i - E(y))$$

$$S_x = \sqrt{\frac{1}{w \times h} \sum_{i=1}^{w \times h} (x_i - E(x))^2}, E(x) = \frac{1}{w \times h} \sum_{i=1}^{w \times h} x_i$$

$$S_y = \sqrt{\frac{1}{w \times h} \sum_{i=1}^{w \times h} (y_i - E(y))^2}, E(y) = \frac{1}{w \times h} \sum_{i=1}^{w \times h} y_i$$

The correlation test involved 448 × 341 (or 449 × 340, or 448 × 340) pairs of two neighboring pixels (horizontally adjacent pixels, vertically adjacent pixels, diagonally adjacent pixels, and vice-diagonally adjacent pixels) for the first experiment, described in Section 4. Four types of correlation coefficients of one secret image and five shares are presented in Table 5.

As indicated in Table 5, the correlation coefficients of the five shares were nearly 0 (irrespective of whether the value was negative or positive), and the restored image exhibited similar results when fewer than k (4, in this example) shares were stacked, indicating a weak correlation among all of the pixels in each share and unauthorized restored shares. Therefore, the shares can be viewed as random shares. Correlation test results from Experiments 2–5 are presented in Table 6. Here we only list the correlation coefficients of share one, because the results for the other shares are all similar to share one. Note that, the secret images were all identical to that in Experiment 1 (already shown in

Table 5). As indicated by the results, no share revealed sufficient information regarding the input image. From the results of these correlation tests, we conclude that all of the shares generated by our scheme were merely noise-like, demonstrating that the secret message could not be discerned in any share.

Table 5. The correlation coefficients of two adjacent pixels in Simulation 1.

Image	Horizontal	Vertical	Diagonal	Vice-Diagonal
Secret image	0.980633	0.987411	0.978314	0.977105
Share image s_1	0.004277	0.003554	0.001016	−0.005866
Share image s_2	0.001302	0.002463	−0.000905	0.003459
Share image s_3	−0.000577	0.005377	−0.002523	−0.001825
Share image s_4	−0.000266	0.003001	−0.000825	−0.000814
Share image s_5	−0.002421	−0.000262	0.000475	−0.002781
Restored image $s_1 \oplus s_2$	0.003155	−0.003003	0.000593	−0.000530
Restored image $s_1 \oplus s_2 \oplus s_3$	0.001182	−0.001205	0.003247	−0.001327

As a result, the secret image is revealed when at least k shares are stacked in our proposed scheme and increases in clarity in accordance with the number of shares stacked together. That is, our scheme is a (k, n)-threshold PVSS scheme. The data in Tables 2–6 indicate the following four advantages of our proposed CHJ scheme: no pixel expansion, greater contrast α, and the abilities to process an image of any size and for any values of k or n. In practical use, it is a more flexible method for encoding a secret image.

Table 6. The correlation coefficients of neighboring pixels for the first share in Simulation 1~ 5.

Image	Horizontal	Vertical	Diagonal	Vice-Diagonal
(4, 5) Share image s_1	0.004277	0.003554	0.001016	−0.005866
(4, 6) Share image s_1	−0.002590	0.002540	0.001554	−0.001326
(4, 7) Share image s_1	−0.001515	0.001431	0.002974	−0.001740
(5, 6) Share image s_1	−0.003995	−0.000352	0.000663	0.000839
(5, 7) Share image s_1	0.003781	0.002251	−0.002108	0.003725

Author Contributions: All authors discussed the main idea and scientific contribution. Y.-Y.C. provided the presented idea, the main algorithm and performed the experiments. B.-Y.H. contributed in analyzed some of the results, and drafted the manuscript. J.S.-T.J. analyzed the results, modified the manuscript, and supervised the whole project.

Funding: This research was funded by the Ministry of Science and Technology of the Republic of China, grant number: MOST 106-2221-E-260-009.

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

References

1. Blakley, G.R. Safeguarding cryptographic keys. In Proceedings of the AFIPS, New York, NY, USA, 4–7 June 1979; pp. 313–317.
2. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
3. Naor, M.; Shamir, A. *Visual cryptography. Eurocrypt94, Lecture Notes in Computer Science*; Springer: Berlin, Germany, 1995; Volume 950, pp. 1–12.
4. Blundo, C.; Santis, A.D.; Stinson, D.R. On the contrast in visual cryptography schemes. *J. Cryptol.* **1999**, *12*, 261–289. [CrossRef]
5. Eisen, P.A.; Stinson, D. Threshold visual cryptography schemes with specified whiteness levels of reconstructed pixels. *Des. Codes Cryptogr.* **2002**, *25*, 15–61. [CrossRef]
6. Hofmeister, T.; Krause, M.; Simon, H.U. Contrast-optimal k out of n secret sharing schemes in visual cryptography. *Theory Comput. Sci.* **2000**, *240*, 471–485. [CrossRef]

7. Lin, C.C.; Tsai, W.H. Secret multimedia information sharing with data hiding capability by simple logic operations. *Pattern Recognit. Image Anal.* **2004**, *14*, 594–600.
8. Linial, N.; Nisan, N. Aprroximate inclusion-exlusion. *Combinatorica* **1990**, *10*, 349–365. [CrossRef]
9. Yang, C.N. New visual secret sharing schemes using probabilistic method. *Pattern Recognit. Lett.* **2004**, *25*, 481–495. [CrossRef]
10. Fang, W.-P.; Lin, S.-J.; Li, J.-C. Visual cryptography (VC) with nonexpanded shadow images: A hilbert-curve approach. In Proceedings of the IEEE International Conference on Intelligence and Security Informatics, Taipei, Taiwan, 17–20 June 2008; pp. 271–272.
11. Hilbert, D. Üeber die stetige abbildung einer linie aufein flächenstück. *Math. Ann.* **1891**, *38*, 459–460. [CrossRef]
12. Shyu, S.-J. Image encryption by multiple random grids. *Pattern Recognit.* **2009**, *42*, 1582–1596. [CrossRef]
13. Kafri, O.; Keren, E. Encryption of pictures and shapes by random grids. *Opt. Lett.* **1987**, *12*, 377–379. [CrossRef] [PubMed]
14. Chen, T.H.; Tsao, K.H. Visual secret sharing by random grids revisited. *Pattern Recognit.* **2009**, *42*, 2203–2217. [CrossRef]
15. Wu, X.; Sun, W. Random grid-based visual secret sharing with abilities of OR and XOR decryptions. *J. Vis. Commun. Image Represent.* **2013**, *24*, 48–62. [CrossRef]
16. Nag, A.; Biswas, S.; Sarkar, D.; Sarka, P.P. Secret image sharing scheme based on a boolean operation. *Cybern. Inf. Technol.* **2014**, *14*, 98–113. [CrossRef]
17. Al-Tamimi, A.G.T.; Gaafar, A. A New Simple Non-Expansion Algorithm for (2, 2)-Visual Secret Sharing Scheme. *Int. J. Comput. Appl.* **2015**, *113*, 3.
18. Chen, S.K.; Lin, J.C. Fault-tolerant and progressive transmission of images. *Pattern Recognit.* **2005**, *38*, 2466–2471. [CrossRef]
19. Fang, W.P. Multi-layer progressive secret image sharing. In Proceedings of the 7th WSEAS, Athens, Greece, 24–26 August 2007; pp. 112–116.
20. Fang, W.P.; Lin, J.C. Progressive viewing and sharing of sensitive images. *Pattern Recognit. Image Anal.* **2006**, *16*, 638–642. [CrossRef]
21. Chen, Z.-H.; Lee, Y.-S. Yet another friendly progressive visual secret sharing scheme. In Proceedings of the 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Kyoto, Japan, 12–14 September 2009; pp. 353–356.
22. Hou, Y.-C.; Quan, Z.-Y. Progressive visual cryptography with unexpanded shares. *IEEE Trans. Circuits Syst. Video Technol.* **2011**, *21*, 1760–1764. [CrossRef]
23. Chen, Y.Y.; Juan, J.S.T. A 4 out of n Secret Sharing Scheme in Visual Cryptography without Expansion. In Proceedings of the International Conference on Foundations of Computer Science (FCS), Stanford, CA, USA, 18–21 July 2011.
24. Chen, T.H.; Tsao, K.H. Threshold visual secret sharing by random grids. *J. Syst. Softw.* **2011**, *84*, 1197–1208. [CrossRef]
25. Wu, X.; Sun, W. Improving the visual quality of random grid based visual secret sharing. *Signal Process.* **2013**, *93*, 977–995. [CrossRef]
26. Guo, T.; Liu, F.; Wu, C. Threshold visual secret sharing by random grids with improved contrast. *J. Syst. Softw.* **2013**, *86*, 2094–2109. [CrossRef]
27. Yan, X.; Liu, X.; Yang, C.N. An enhanced threshold visual secret sharing based on random grids. *J. Real-Time Image Process.* **2018**, *14*, 61–73. [CrossRef]
28. Wan, S.; Lu, Y.; Yan, X.; Wang, Y.; Chang, C. Visual secret sharing scheme for (k, n) threshold based on QR code with multiple decryptions. *J. Real-Time Image Process.* **2018**, *14*, 25–40. [CrossRef]
29. Yan, X.; Wang, S.; Niu, X. Threshold progressive visual cryptography construction with unexpanded shares. *Multimed. Tools Appl.* **2016**, *75*, 8657–8674. [CrossRef]
30. Liu, Y.X.; Yang, C.N.; Wu, S.Y.; Chou, Y.S. Progressive (k, n) secret image sharing schemes based on Boolean operations and covering codes. *Signal Process. Image Commun.* **2018**, *66*, 77–86. [CrossRef]
31. Alon, N.; Spencer, J. *The Probabilistic Method*; Wiley: Hoboken, NJ, USA, 1992.
32. Carter, J.L.; Wegman, M.N. Universal classes of hash functions. *J. Comput. Syst. Sci.* **1979**, *18*, 143–154. [CrossRef]

33. Wegman, M.N.; Carter, J.L. New hash functions and their use in authentication and set equality. *J. Comput. Syst. Sci.* **1981**, *22*, 265–279. [CrossRef]
34. Chang, C.; Lin, C.; Lin, C.; Chen, Y. A novel secret image sharing scheme in color images using small shadow images. *Inf. Sci.* **2008**, *178*, 2433–2447. [CrossRef]
35. Shankar, K.; Eswaran, P. A new *k* out of *n* secret image sharing scheme in visual cryptography. In Proceedings of the 10th International Conference on IEEE Intelligent Systems and Control (ISCO2016), Coimbatore, India, 7–8 January 2016; pp. 1–6.