mathematics

MDPI

Article

Visual Cryptography Scheme with Essential Participants

Peng Li *, Liping Yin and Jianfeng Ma

Department of Mathematics and Physics, North China Electric Power University, Baoding 071003, China; yinlipingytlx@163.com (L.Y.); jianfma@163.com (J.M.)

* Correspondence: peng.li@ncepu.edu.cn

Received: 23 April 2020; Accepted: 18 May 2020; Published: 22 May 2020

Abstract: Visual cryptography scheme (VCS) shares a binary secret image into multiple shadows printed on transparencies. Stacking shadows can visually decode the secret image without computational resources. Specifically, a (k, n) threshold VCS ((k, n)-VCS) shares a secret image into n shadows, stacking any k shadows can reveal the secret image by human visual system, while any less than k shadows cannot decode any information regarding the secret image. In practice, some participants (essentials) play more important roles than others (non-essentials). In this paper, we propose a (t, s, k, n) VCS with essential participants (so called (t, s, k, n)-EVCS). The secret image is shared into n shadows with s essentials and n-s non-essentials. Any k shadows, including at least t essentials, can reveal the secret image. The proposed scheme is constructed from a monotonic (K, N)-VCS. The condition and optimal choice of (K, N)-VCS to construct (t, s, k, n)-EVCS are given by solving integer programming model. The experimental results are conducted to verify the feasibility of our scheme.

Keywords: visual secret sharing; secret image sharing; visual cryptography; integer programming; essential shadows

1. Introduction

Visual cryptography scheme (VCS) is a technique for sharing a secret image among the participants. The revealing process of the secret image can be implemented by stacking operation without computation. The first VCS was proposed by Naor and Shamir [1] in 1994. A (k, n) threshold VCS ((k, n)-VCS) shares a binary secret image into n shadows printed on transparencies, which are assigned to n participants, respectively. Stacking any k shadows can reveal the secret image by human visual system without computation. The advantage of VCS is its easy revealing. Stacking shadows without computational resources can reveal the secret image. However, the disadvantages are the large shadow size expansion and the degraded visual quality of the revealed image. Many researchers were dedicated on improving performance of VCS [2–4], and proposed VCS with different properties, like VCS for color images [5–7], VCS for multiple secret images [8,9], VCS with meaningful shadows [10,11], and random grid-based VCS (RGVCS) [12–15], et.al.

A (k, n)-VCS is called the monotonic VCS if it can reveal the secret image by stacking more than k shadows. Otherwise, it is called the non-monotonic VCS. Jin et al. proposed progressive VCS [16]. Stacking more shadows can decode secret image with better visual quality. Most of existing VCSs do not distinguish the roles of each shadow. However, in practice, some shadows are more important than others according to the status of the participants. Arumugam et al. [17] proposed (k, n)-VCS with one essential participant and n-1 non-essential participants. In the revealing process, any k participants, including the essential one, can reveal the secret image. Without the essential one, the secret image cannot be revealed, even with all other non-essentials. Guo et al. [18] extended the scheme

of Arumugam et al. [17] and proposed a (t, k, n)-VCS with t essential participants, namely (t, k, n)-EVCS that is constructed from a known (k-t, n-t)-VCS and a known optimal (t, t)-VCS. A qualified set of shadows should contain k shadows, including t essential ones.

Another category of secret image sharing scheme is based on polynomial. Thien and Lin [19] proposed a (k, n) threshold secret image sharing (SIS) scheme. The secret pixels are embedded into the coefficients of a (k−1)-degree polynomial to generate shadow pixels. With any k shadows, the secret image can be decoded by Lagrange interpolation. When compared with VCS, polynomial based SIS can reveal the original grayscale secret image by computation. Many researchers proposed many polynomial based SIS [20–23]. Li et al. [24] first presented the concept of secret image sharing scheme with essential participants (ESIS), and proposed (t, s, k, n)-ESIS. For a (t, s, k, n)-ESIS, the secret image is shared into n shadows with s essentials and n-s non-essentials. A qualified set of shadows should contain k shadows, including at least t essentials. Many researchers proposed different (t, s, k, n)-ESIS schemes to achieve smaller shadow size and equal shadow size [25–28]. Liu et al. [29] combined the scalable secret image sharing scheme with ESIS, so that k or more shadows, including at least t essential shadows, can gradually restore the secret image, while restoring the whole secret image requires the participation of all s essential shadows.

In this paper, we propose general (t, s, k, n) visual cryptography scheme with essential participants (EVCS). For a (t, s, k, n)-EVCS, the secret image is shared into n shadows with s essentials and n-s non-essentials. Stacking any k shadows, including at least t essentials, can reveal the secret image. The proposed (t, s, k, n)-EVCS is constructed from a monotonic (K, N)-VCS based on integer programming. When compared with ESIS, the revealing process of the proposed EVCS does not need complicated mathematical operation. EVCS has potential application when some participants are accorded special privileges due to their status or importance, e.g., heads of government, managers of company, high-level corporate officers, major employers, etc. For example, in a nuclear-powered submarine under the ocean, the missile launch code is shared by (2, 2, 4, 6)-EVCS into two essential shadows for the commander and the executive commander and four non-essentials for four other decision members. The missile launch code can be decoded if and only if at least four participants, including the commander and the executive commander, have the agreement on the launch of the missile, and stacking their shadows. EVCS can also be applied in key exchange or key distribution when exchanging message in a public secure network [30,31].

The layout of this paper is as follows. In next section, we present some preliminaries of VCS. In Section 3, we propose our (t, s, k, n)-EVCS based on integer programming. Section 4 provides experimental results and comparisons and Section 5 concludes the paper.

2. Related Works

In this section, we briefly introduce relevant concepts of (k, n)-VCS and Yan et al.'s random grid based VCS (RGVCS) [32].

2.1. Access Structure of (k, n)-VCS

Let $P = \{1, 2, \cdots, n\}$ be the set of all participants and 2^P is the power set of P. Let qualified sets Γ_{Qual} be the collection of the set of participants that can recover the secret, forbidden sets Γ_{Forb} be the collection of the set of participants that cannot recover the secret. $(\Gamma_{Qual}, \Gamma_{Forb})$ constitutes an access structure, where $\Gamma_{Qual} \subseteq 2^P$, $\Gamma_{Forb} \subseteq 2^P$, and $\Gamma_{Qual} \cap \Gamma_{Forb} = \varnothing$.

Definition 1 ([33]). *A (k, n)-VCS with access structure $(\Gamma_{Qual}, \Gamma_{Forb})$ is monotonic if the following conditions are satisfied.*

(1) *Γ_{Qual} is monotonic increasing, i.e. if a subset of Q can reveal the secret, then the participants in Q can reveal the secret as well.*

(2) *Γ_{Forb} is monotonic decreasing, i.e. if $F \in \Gamma_{Forb}$ cannot reveal the secret, then any subset of F cannot reveal the secret as well.*

(3) $\Gamma_{Qual} \cup \Gamma_{Forb} = 2^p$

For a (k, n)-VCS, a qualified set should contain at least k participants. Stacking any k shadows can reveal the secret image. If stacking more than k shadows can still reveal the secret, the (k, n)-VCS is also called monotonic (k, n)-VCS.

Usually, a (k, n)-VCS is constructed by a pair of matrices, called basis matrices M_0 and M_1. Let $S \subseteq P$, $M|S$ is a submatrix generated by restricting matrix M on the rows of S. Let $OR(M)$ denote the vector generated by performing OR operation on the rows of matrix M. Let $w(a)$ denote the Hamming weight of vector a. Formally, we have the definition of monotonic (k, n)-VCS, as follows.

Definition 2. *Two binary matrices M_0 and M_1 with the size $n \times m$ can be used as basis matrices of a monotonic (k, n)-VCS if and only if the following conditions satisfied.*

(Contrast condition). For any $S \subseteq P$ and $|S| \geq k$, we have $w(OR(M_0|S)) < w(OR(M_1|S))$.

(Security condition). For any $S \subseteq P$ and $1 \leq |S| < k$, we have $w(OR(M_0|S)) = w(OR(M_1|S))$.

For a (k, n)-VCS with basis matrices M_0 and M_1, if the secret pixel is white (resp. black), permute the columns of M_0 (resp. M_1), and then assign its n rows to n shadows, respectively. Since each shadow receives m pixels for sharing each secret pixel, the shadows size is m times of the secret image. m is also called size expansion. The visual quality of the revealed image is usually degraded, and it is evaluated by the contrast defined, as follows.

$$\alpha = \left(w(OR(M_1\big|S)) - w(OR(M_0\big|S))\right)/m$$

where $S \subseteq P$ and $|S| \geq k$. By contrast condition of the definition of VCS, we know that α is larger than 0 and no more than 1. When the contrast is 1, the revealed image has the perfect visual quality. The larger value of the contrast, the better visual quality of the revealed image.

Example 1. *The example of (3, 4)-VCS.*

Here we show a (3, 4)-VCS while using the following basis matrices presented in [33].

$$M_0 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} \text{ and } M_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

The size expansion m is 6, which means the generated shadows have the size six times of the secret image, as we can see from the basis matrices. The contrast value α when stacking three shadows is 1/6. When stacking four shadows, the contrast value α is increased to 1/3. Therefore, the (3, 4)-VCS with above basis matrices is monotonic. Figure 1 shows the experimental results of (3, 4)-VCS. Stacking any three or four shadows can reveal the secret image.

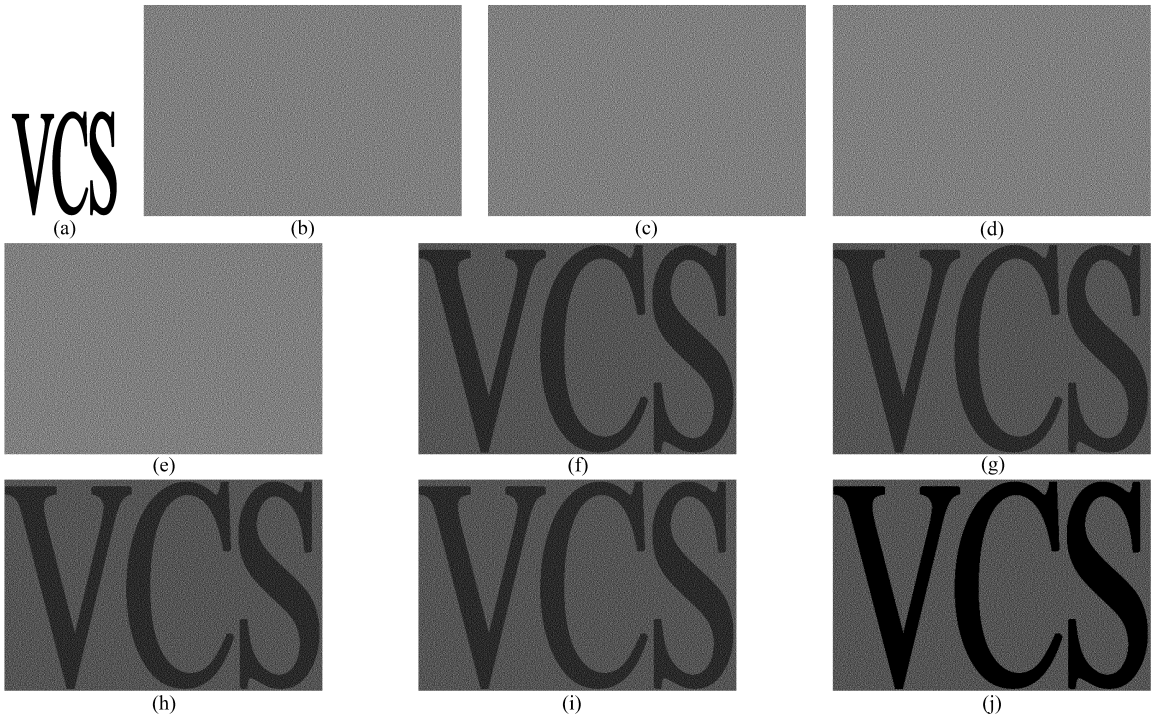

Figure 1. The experimental results of (3, 4)-VCS. (**a**) the secret image; (**b**–**e**) four generated shadows; (**f**) revealed image by shadow 1, 2 and 3; (**g**) revealed image by shadow 1, 2, and 4; (**h**) revealed image by shadow 1, 3, and 4; (**i**) revealed image by shadow 2, 3, and 4; and, (**j**) revealed image by shadow 1, 2, 3, and 4.

2.2. Yan et al.'s RGVCS

Kafri and Keren first presented RG-based VCS [34]. Each shadow is noise-like and it has the same size as the secret image. The revealing operation is also stacking shadows. First, we briefly introduce the generation of (2, 2)-RGVCS, as described below.

Step 1: Randomly generate the first shadow SC_1 with the same size as secret image S.

Step 2: Calculate the corresponding $SC_2(i, j)$ according to $S(i, j)$(the value of pixels in the secret image), as described in Equation (1).

$$SC_2(i,j) = \begin{cases} SC_1(i,j) & if\ S(i,j) = 0 \\ \overline{SC_1(i,j)} & if\ S(i,j) = 1 \end{cases} \tag{1}$$

Step 3: Repeat Step2 until all pixels in S are processed.

Finally, the revealed image obtained by stacking shadows ($S' = SC_1 \otimes SC_2$ as in Equation (2), where $\otimes$ denotes the Boolean OR operation) SC_1 and SC_2 can be directly recognized by the human visual system.

$$S'(i,j) = SC_1(i,j) \otimes SC_2(i,j) = \begin{cases} SC_1(i,j) \otimes SC_1(i,j) & if\ S(i,j) = 0 \\ SC_1(i,j) \otimes \overline{SC_1(i,j)} = 1 & if\ S(i,j) = 1 \end{cases} \tag{2}$$

Many (k, n)-RGVCS schemes have been proposed based on (2, 2)-RGVCS. Their similarity is to repeat the above process for the first k bits, but the difference is the disposal of the last n-k bits. Yan et al. [32] proposed a novel (k, n)-RGVCS, which makes full use of n-k random bits to improve the visual quality of the recovered image. Their (k, n)-RGVCS is also a progressive VCS. Better visual quality of the revealed secret image will be gained by stacking more shadows. The algorithm of Yan et al.'s (k, n)-RGVCS is given, as follows (Algorithm 1).

Algorithm 1. Yan et al.'s RGVCS.

Input: secret image S, the threshold parameters (k, n)
Output: n shadows $SC_1, SC_2, \ldots, SC_n$

A1-1: For each pixel $S(i, j)$ in the secret image S, repeat Steps 2–4.
A1-2: Apply the above conventional (2, 2)-RGVCS to encrypt the pixel $S(i, j)$, then b_1 and b'_2 are obtained. b'_2 is encrypted in the same way. Repeat the above operation until $b_1, b_2, \ldots, b'_k (= b_k)$ are obtained.
A1-3: For $b_l (k + 1 \le l \le n)$, if $l \bmod k = x, (0 \le x \le k - 1)$, then $b_l = b_k$.
A1-4: Redistribute $b_1, b_2, \ldots, b_n$ to $SC_1(i, j), SC_2(i, j), \ldots, SC_n(i, j)$ randomly.
A1-5: Output n shadows $SC_1, SC_2, \ldots, SC_n$.

Example 2. *The experiment of Yan et al.'s (3, 6)-RGVCS*

An experiment of (3, 6) threshold of scheme with secret image "VCS" is conducted in order to demonstrate the Yan et al.'s algorithm. Figure 2a shows the secret image. Figure 2b–g show six shadows. Figure 2h shows the revealed image by 2 shadows. Figure 2i–l show the revealed image by stacking 3, 4, 5, and 6 shadows, respectively. Apparently, better visual quality of the recovered secret will be gained by stacking more shadow images. The results show that Yan et al.'s scheme satisfies monotonicity, as described in Definition 1.

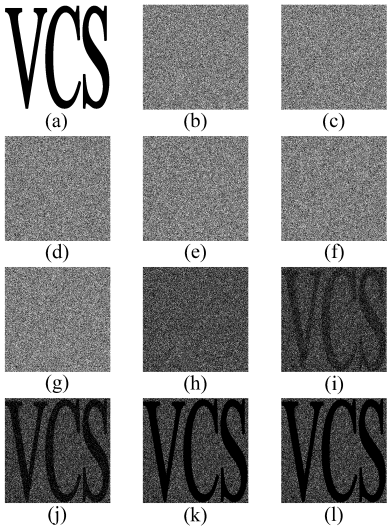

Figure 2. An experiment of Yan et al.'s (3, 6)-VCS. (**a**) secret image; (**b**–**g**) six shadows; (**h**) revealed image by two shadows; (**i**) revealed image by three shadows; (**j**) revealed image by four shadows; (**k**) revealed image by five shadows; and, (**l**) revealed image by six shadows.

3. The Proposed (*t*, *s*, *k*, *n*)-EVCS Based on Integer Programming

3.1. The Definition of (t, s, k, n)-EVCS

In traditional (k, n)-VCS, a qualified subset of participants should have any k or more participants. The roles of each participant are the same. However, there are many examples in practical situations where some participants are given privileges because of their status or importance, such as heads of government, company managers, etc. Therefore, it is reasonable for us to consider giving special powers to some participants in VCS. The proposed (t, s, k, n)-EVCS shares the secret image into n shadows with s essentials and n-s non-essentials. Stacking any k shadows, including at least t essential one, can reveal the secret image. A qualified subset of participants should have at least k shadows, including t essentials. Let $EP = \{1, 2, \ldots, s\}$ and $NEP = \{s + 1, s + 2, \ldots, n\}$ denote the set of essential

participants and non-essential participants, respectively. Subsequently, we can derive all qualified sets Γ_{Qual} of (t, s, k, n)-EVCS, as follows.

$$\Gamma_{Qual} = \{Q | Q \subseteq P, |Q| \geq k \text{ and } |Q \backslash NEP| \geq t\} \tag{3}$$

If a subset of participant does not belong to the qualified sets Γ_{Qual}, it belongs to forbidden sets. Hence, we have forbidden sets Γ_{Forb} of (t, s, k, n)-EVCS.

$$\Gamma_{Forb} = \left\{ S \middle| S \subseteq P \text{ and } S \notin \Gamma_{Qual} \right\} \tag{4}$$

Subsequently, (t, s, k, n)-EVCS can be defined if and only if the access structure satisfies Equations (3) and (4).

We only consider non-trivial EVCS, which cannot be reduced to a threshold VCS. For the relationships among t, s, k, and n of (t, s, k, n)-EVCS, we have the following facts.

(1) t, s, k and n are all integers no less than 1, and $t \leq s \leq n$, $t \leq k \leq n$.
(2) $k > t$. Otherwise, (t, s, k, n)-EVCS is reduced to (t, s)-VCS.
(3) $k < n$. Otherwise, (t, s, k, n)-EVCS is reduced to (n, n)-VCS.
(4) If $s = n$, (t, s, k, n)-EVCS is reduced to (t, n)-VCS. Hence, $s < n$. If $s = n - 1$, then there is only one non-essential participant. A qualified set of participants contains k members, including at least t essentials and k - t non-essentials. We have $k - t \leq 1$. Since $k > t$, then $k = t + 1$. If $s = t$, then $k = s + 1 = n$, and (t, s, k, n)-EVCS is reduced to (n, n)-VCS. Otherwise, $s \geq t + 1$, which means that there are more than t essentials. Since there is only one non-essential participant, any $t + 1$ participants must contain at least t essentials and they can reveal the secret image. Afterwards, (t, s, k, n)-EVCS is reduced to $(t + 1, n)$-VCS. Overall, we have $s \leq n$-2.
(5) $k - t < n - s$. The number of non-essentials is $n - s$, and the largest number of non-essentials in a qualified set is $k - t$. Obviously, $k - t \leq n - s$. If $k - t = n - s$, then any k participants will contain at least $k - (n - s) = k - (k - t) = t$ essentials. Subsequently, (t, s, k, n)-EVCS is reduced to (k, n)-VCS. Therefore, we have $k - t < n - s$.

Finally, we have the relationships among t, s, k and n of (t, s, k, n)-EVCS are shown, as follows:

$$\begin{cases} t \leq s \leq n - 2 \\ t < k < n \\ k - t < n - s \\ t, s, k \text{ and } n \text{ are integers no less than } 1 \end{cases} \tag{5}$$

3.2. Constructing (t, s, k, n)-EVCS Based on Integer Programming

The idea for constructing (t, s, k, n)-EVCS is that we generate the shadows by a monotonic (K, N)-VCS. The secret image is first shared into N shadows by (K, N)-VCS. Subsequently, each essential (non-essential) shadow of EVCS is obtained by the superposition of ω_1 (ω_2) shadows of VCS. Obviously, we have

$$N = s\omega_1 + (n - s)\omega_2 \tag{6}$$

Since essential shadow is more important than the non-essential shadow of EVCS, ω_1 must be larger than ω_2.

Figure 3 shows the diagram of generating shadows of EVCS by the shadows of a monotonic VCS.

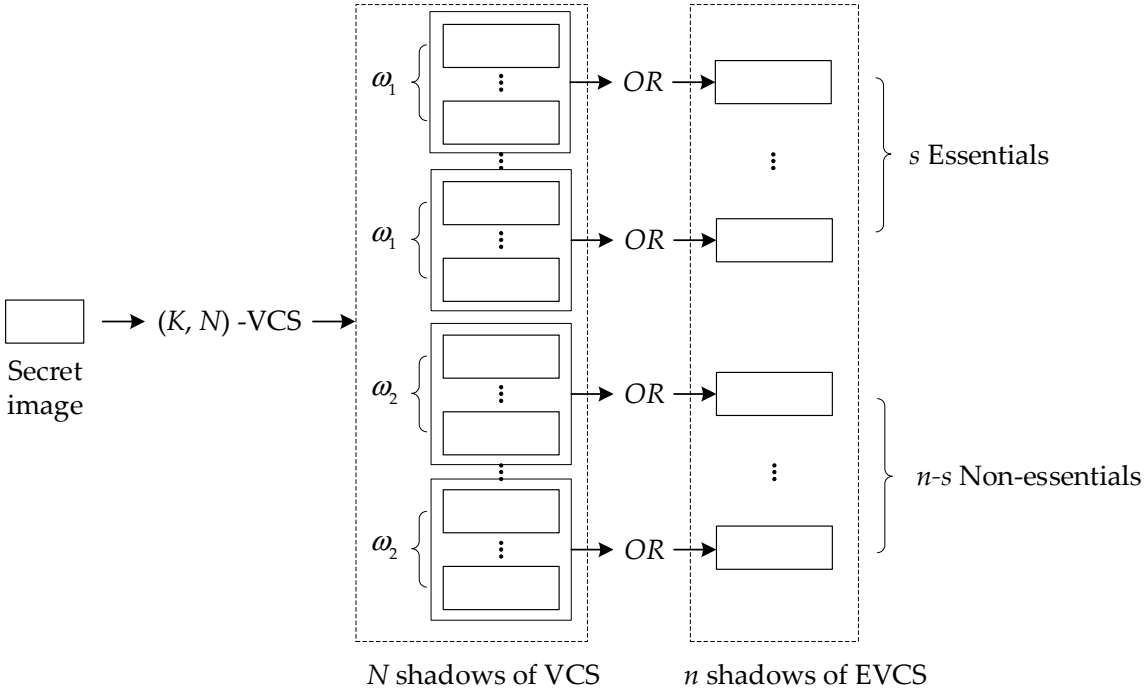

Figure 3. The diagrammatical representation of the proposed (*t*, s, *k*, *n*)-EVCS.

For (*K*, *N*)-VCS, a qualified subset of shadows should have at least *K* shadows. Each essential (non-essential) shadow of EVCS represents the stacking result of ω_1 (ω_2) shadows of (*K*, *N*)-VCS. A qualified subset of shadows of EVCS should contribute no less than *K* shadows of (*K*, *N*)-VCS to satisfy the contrast condition of (*K*, *N*)-VCS. Any forbidden subset of shadows of EVCS should contribute less than *K* shadows of (*K*, *N*)-VCS to satisfy the security condition of (*K*, *N*)-VCS. Therefore, our task is determining the proper values of ω_1, ω_2, *K*, and *N* to be used for constructing (*t*, *s*, *k*, *n*)-EVCS. In this paper, we get the values of ω_1, ω_2, *K*, and *N* by solving an integer programming model.

For (*t*, *s*, *k*, *n*)-EVCS, we first need to build the relationship among the values of ω_1, ω_2, *K*, and *N*. These parameters should satisfy the following conditions.

(1) For all parameters ω_1, ω_2, *K*, and *N* to make sense, we need to restrict that $\omega_1 \geq 1$, $\omega_2 \geq 1$, $K \geq 1$, and $N \geq K$.

(2) Essential shadows are more important than non-essential shadows. In another word, an essential shadow can contribute more shadows of VCS than a non-essential shadow. Hence, ω_1 must be larger than ω_2. That is

$$\omega_1 - \omega_2 \geq 1 \tag{7}$$

(3) By Equation (3), we have that a qualified set should contain any *k* shadows, including at least *t* essentials. In another word, *k* shadows of EVCS, including *t* essential ones, can contribute at least *K* shadows of VCS. Subsequently, we have

$$t\omega_1 + (k-t)\omega_2 \geq K \tag{8}$$

Obviously, Equation (8) guarantees that any *k* shadows of EVCS, including more than *t* essential ones, can also contribute at least *K* shadows of VCS.

(4) By Equations (3) and (4), the secret image cannot be recovered with less than *t* essential shadows. In another word, the threshold value *K* of (*K*, *N*)-VCS is still not satisfied, even if $t-1$ essential shadows and all *n-s* non-essential shadows are gathered. Subsequently, we have the following inequality.

$$(t-1)\omega_1 + (n-s)\omega_2 \leq K-1 \tag{9}$$

(5) By Equations (3) and (4), the secret image cannot be recovered with less than k shadows.

If $s \geq k$, then any $k-1$ essential shadows cannot contribute enough shadows of VCS. In order to satisfy the security condition of VCS, we have

$$(k-1)\omega_1 \leq K-1 \tag{10}$$

If $s < k$, then any s essential shadows and k-$s-1$ non-essential shadows cannot contribute enough shadows of VCS. To satisfy the security condition of VCS, we have

$$s\omega_1 + (k-1-s)\omega_2 \leq K-1 \tag{11}$$

For (K, N)-VCS, the larger values of K and N may reduce the visual quality of the revealed image, and complicate the sharing process. Therefore, we want to obtain as small values of K and N as possible. In general, the objective function is:

$$\min K+N \tag{12}$$

We generate the following integer programming models (IPM) by combining the constraint conditions and objective function.

(IPM I): When $s < k$, the corresponding integer programming model is:

$$\begin{cases} \min K + s\omega_1 + (n-s)\omega_2 \\ t\omega_1 + (k-t)\omega_2 - K \geq 0 \\ (t-1)\omega_1 + (n-s)\omega_2 - K \leq -1 \\ s\omega_1 + (k-1-s)\omega_2 - K \leq -1 \\ \omega_1 - \omega_2 \geq 1 \\ \omega_1 \geq 1 \\ \omega_2 \geq 1 \\ K \geq 1 \end{cases} \tag{13}$$

(IPM II): When $s \geq k$, the corresponding integer programming model is:

$$\begin{cases} \min K + s\omega_1 + (n-s)\omega_2 \\ t\omega_1 + (k-t)\omega_2 - K \geq 0 \\ (t-1)\omega_1 + (n-s)\omega_2 - K \leq -1 \\ (k-1)\omega_1 - K \leq -1 \\ \omega_1 - \omega_2 \geq 1 \\ \omega_1 \geq 1 \\ \omega_2 \geq 1 \\ K \geq 1 \end{cases} \tag{14}$$

3.3. Determine the Parameters by Solving IPMs

We need to solve IPM I or IPM II to determine the values of ω_1, ω_2, K and N. Before we solve IPM, we divide the relationship among t, s and k into six cases: (Case 1) $t = s$, $s < k$; (Case 2) $t \neq s$, $s < k$; (Case 3) $s - t = 1$, $s = k$; (Case 4) $s - t \neq 1$, $s = k$; (Case 5) $k - t = 1$, $s > k$; (Case 6) $k - t = 1$, $s > k$. Figure 4 shows the diagram of the division for different cases. For Case 1 and Case 2, we need to solve IPM I. For the other cases, we need to solve IPM II.

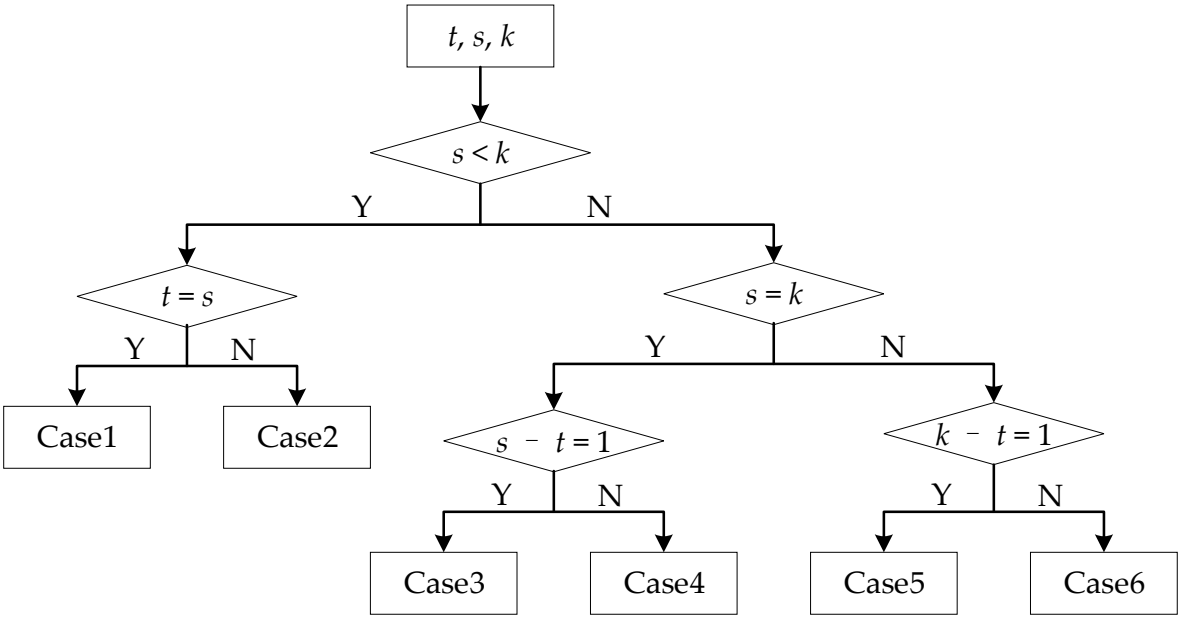

Figure 4. The diagram of the division for different cases.

Now we solve IPM according to the six cases, respectively.

(Case 1) t = s, s < k.

For this case, we need to solve IPM I. First, we convert IPM I into standard form by generalizing the sighs of ω_1, ω_2 and K to x_1, x_2, and x_3. Subsequently, we have new IPM as follows.

$$\begin{cases} \max -tx_1 + (t-n)x_2 - x_3 \\ tx_1 + (k-t)x_2 - x_3 - x_4 = 0 \\ (1-t)x_1 + (t-n)x_2 + x_3 - x_5 = 1 \\ -tx_1 + (t+1-k)x_2 + x_3 - x_6 = 1 \\ x_1 - x_2 - x_7 = 1 \\ x_1 - x_8 = 1 \\ x_2 - x_9 = 1 \\ x_3 - x_{10} = 1 \\ x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10} \geq 0 \end{cases} \tag{15}$$

where x_4, x_5, x_6, x_7, x_8, x_9, and x_{10} are non-negative residual variables (slack variables). We use the dual simplex method to solve above IPM. Equation (15) is converted to the following form to obtain the initial feasible basis of the dual problem.

$$\begin{cases} \max -tx_1 + (t-n)x_2 - x_3 \\ -tx_1 + (t-k)x_2 + x_3 + x_4 = 0 \\ (t-1)x_1 + (n-t)x_2 - x_3 + x_5 = -1 \\ tx_1 + (k-t-1)x_2 - x_3 + x_6 = -1 \\ -x_1 + x_2 + x_7 = -1 \\ -x_1 + x_8 = -1 \\ -x_2 + x_9 = -1 \\ -x_3 + x_{10} = -1 \\ x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9 x_{10} \geq 0 \end{cases} \tag{16}$$

Establish the initial simplex table for IPM, as shown in Table 1.

Table 1. The initial simplex table of integer programming models (IPM) I for Case 1.

	$c_j \rightarrow$		-t	$t-n$	-1	0	0	0	0	0	0	0
C_B	X_B	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0	x_4	0	$-t$	$t-k$	1	1	0	0	0	0	0	0
0	x_5	-1	$t-1$	$n-t$	[[-1]	0	1	0	0	0	0	0
0	x_6	-1	t	$k-t-1$	-1	0	0	1	0	0	0	0
0	x_7	-1	-1	1	0	0	0	0	1	0	0	0
0	x_8	-1	-1	0	0	0	0	0	0	1	0	0
0	x_9	-1	0	-1	0	0	0	0	0	0	1	0
0	x_{10}	-1	0	0	-1	0	0	0	0	0	0	1
	$c_j - z_j$		$-t$	$t-n$	-1	0	0	0	0	0	0	0

From Table 1 it can be seen that the solution of the dual problem corresponding to the row of checking number is feasible. Since some numbers in column b is negative, iterative operation is required. Since the values of b_i are equal, the non-basic variable with the smallest subscript in $\boldsymbol{X_B}$ is selected as the leaving variable, i.e. x_5. Check the coefficients $a_{lj}(j = 1, 2, \ldots, 10)$ of the row of a_l in the simplex table, if all $a_{lj} \geq 0$, there is no feasible solution, and the calculation is terminated. If $a_{lj} < 0$, calculate $\theta = \min_j\left(\frac{c_j - z_j}{a_{lj}} \middle| a_{lj} < 0\right) = \frac{c_k - z_k}{a_{lk}}$, and the non-basic variable x_k of the column corresponding to rule of θ is the entering variable. Calculating according to the above steps, we obtain $\theta = \min\left\{-, -, \frac{-1}{-1}\right\} = 1$, so x_3 is the entering variable. "-1" is the pivot element at the intersection of the column and row where the variables are entering and leaving. The iteration is performed according to the calculation steps of dual simplex method, and Table 2 shows the results.

Table 2. Simplex table of IPM I for Case 1 after one iteration.

	$c_j \rightarrow$		$-t$	$t-n$	-1	0	0	0	0	0	0	0
$\boldsymbol{C_B}$	$\boldsymbol{X_B}$	$\boldsymbol{b}$	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0	x_4	-1	[-1]	$n-k$	0	1	1	0	0	0	0	0
-1	x_3	1	$1-t$	$t-n$	1	0	-1	0	0	0	0	0
0	x_6	0	1	$k-n-1$	0	0	-1	1	0	0	0	0
0	x_7	-1	-1	1	0	0	0	0	1	0	0	0
0	x_8	-1	-1	0	0	0	0	0	0	1	0	0
0	x_9	-1	0	-1	0	0	0	0	0	0	1	0
0	x_{10}	0	$1-t$	$t-n$	0	0	-1	0	0	0	0	1
	$c_j - z_j$		$1-2t$	$2(t-n)$	0	0	-1	0	0	0	0	0

From Table 2 it can be seen that the dual problem is still a feasible solution, and there are still negative components in column $\boldsymbol{b}$. Repeat the above iterative steps until the numbers in column $\boldsymbol{b}$ are all non-negative and the test numbers are all non-positive, as shown in Table 3.

Table 3. The final simplex table of IPM I for Case 1.

	$c_j \rightarrow$		$-t$	$T-n$	-1	0	0	0	0	0	0	0
$\boldsymbol{C_B}$	$\boldsymbol{X_B}$	$\boldsymbol{b}$	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$-t$	x_1	$n-k+1$	1	0	0	$k-n-1$	-1	$k-n$	0	0	0	0
-1	x_3	$t(n-k)+k$	0	0	1	$t(k-n)-k+1$	$-t$	$t(k-n+1)-k$	0	0	0	0
$t-n$	x_2	1	0	1	0	-1	0	-1	0	0	0	0
0	x_7	$n-k-1$	0	0	0	$k-n$	-1	$k-n+1$	1	0	0	0
0	x_8	$n-k$	0	0	0	$k-n-1$	-1	$k-n$	0	1	0	0
0	x_9	0	0	0	0	-1	0	-1	0	0	1	0
0	x_{10}	$k+t(n-k)-1$	0	0	0	$t(k-n)-k+1$	$-t$	$t(k-n+1)-k$	0	0	0	1
	$c_j - z_j$		0	0	0	$2t(k-n)+1-k-n$	$-2t$	$2t(k-n+1)-n-k$	0	0	0	0

The numbers in column $\boldsymbol{b}$ are all non-negative and the test numbers are all non-positive, as shown in Table 3. Therefore, the optimal solution of the problem is $X^* = (n-k+1, 1, t(n-k)+k, 0, 0, 0, n-k-1,$

$n-k$,0,$k+t(n-k)-1$). Additionally, since $tx_1+(n-t)x_2=N$, then $t(n-k+1)+n-t=N$. From what has been discussed above, any (t, t, k, n)-EVCS can be constructed by a monotonic (K, N)-VCS = ($t(n-k)+k$, $t(n-k+1)+n-t$)-VCS with $\omega_1=n-k+1$ and $\omega_2=1$.

(Case 2) t ≠ s, s < k.

For this case, we need to solve IPM I with the same method. Table 4 shows the simplex table obtained after two iterations.

Table 4. Simplex table after two iterations for Case 2.

	$c_j\rightarrow$		−s	$s-n$	−1	0	0	0	0	0	0	0
C_B	X_B	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
−s	x_1	1	1	$k+s-t-n$	0	−1	−1	0	0	0	0	0
−1	x_3	t	0	$t-k-t(t+n-k-s)$	1	$1-t$	$-t$	0	0	0	0	0
0	x_6	$t-s-1$	0	$[(t+n-k-s)(s-t)+t-s-1]$	0	$S-t+1$	$s-t$	1	0	0	0	0
0	x_7	0	0	$1-t-n+k+s$	0	−1	−1	0	1	0	0	0
0	x_8	0	0	$k+s-t-n$	0	−1	−1	0	0	1	0	0
0	x_9	−1	0	−1	0	0	0	0	0	0	1	0
0	x_{10}	$t-1$	0	$t-k-t(t+n-k-s)$	0	$1-t$	$-t$	0	0	0	0	1
	c_j-z_j		0	$2(s-n)+(s-t-1)(k+s-t-n)$	0	$1-s-t$	$-s-t$	0	0	0	0	0

Since $t\neq s$, with the relationship between s and t, we have $t-s-1<-1$. Subsequently, the linear programming has a solution if and only if $(t+n-k-s)(s-t)+t-s-1<0$. Continue to iterate. It is calculated that the elements in column $\boldsymbol{b}$ are $1-\frac{(t-s-1)(k+s-t-n)}{(t+n-k-s)(s-t)+t-s-1}$, $t-\frac{(t-s-1)[t-k-t(t+n-k-s)]}{(t+n-k-s)(s-t)+t-s-1}$, $\frac{t-s-1}{(t+n-k-s)(s-t)+t-s-1}$, $-\frac{(t-s-1)(1-t-n+k+s)}{(t+n-k-s)(s-t)+t-s-1}$, $-\frac{(t-s-1)(k+s-t-n)}{(t+n-k-s)(s-t)+t-s-1}$, $\frac{t-s-1}{(t+n-k-s)(s-t)+t-s-1}-1$, $t-1-\frac{(t-s-1)[t-k-t(t+n-k-s)]}{(t+n-k-s)(s-t)+t-s-1}$, respectively. They are obviously all non-negative, except $-\frac{(t-s-1)(1-t-n+k+s)}{(t+n-k-s)(s-t)+t-s-1}$, which needs further discussion. If $1-t-n+k+s\leq 0$, i.e. $-\frac{(t-s-1)(1-t-n+k+s)}{(t+n-k-s)(s-t)+t-s-1}\geq 0$, then the calculation is terminated. It can be known from conditions $(t+n-k-s)(s-t)+t-s-1<0$ and $1-t-n+k+s\leq 0$ that $1\leq t+n-k-s<2$, namely, $t+n-k-s=1$. Thus, the above simplex table can be simplified into the following table.

The elements in column $\boldsymbol{b}$ are all non-negative and the checking numbers are all non-positive, as shown in Table 5. Therefore, the optimal solution of the problem is $X^*=(s-t+2,\ s-t+1,\ t-k(t-s-1)$,0,0,0,0, $s-t+1$, and $t-k(t-s-1)-1$, $s-t)$. From what has been discussed above, (t, s, k, n)-EVCS of Case 2 can be constructed by a monotonic (K, N)-VCS = ($t-k(t-s-1)$, $s(s-t+2)+(n-s)(s-t+1)$)-VCS with ω_1=$s-t+2$ and ω_2=$s-t+1$. If $1-t-n+k+s>0$, we have known that $k+s<n+t$, then $0<t+n-k-s<1$, the absence of t, s, k, and n makes this condition satisfy.

Table 5. The final simplex table of IPM I for Case 2.

		$c_j\rightarrow$	$-t$	$t-n$	−1	0	0	0	0	0	0	0
C_B	X_B	b	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
−s	x_1	$S-t+2$	1	0	0	$t-s-2$	$t-s-1$	−1	0	0	0	0
−1	x_3	$T-k(t-s-1)$	0	0	1	$1-t-k(s-t+1)$	$-t-k(s-t)$	$-k$	0	0	0	0
$s-n$	x_2	$1+s-t$	0	1	0	$t-s-1$	$t-s$	−1	0	0	0	0
0	x_7	0	0	0	0	−1	−1	0	1	0	0	0
0	x_8	$1+s-t$	0	0	0	$t-s-2$	$t-s-1$	−1	0	1	0	0
0	x_9	$t-k(t-s-1)-1$	0	0	0	$t-s-1$	$t-s$	−1	0	0	1	0
0	x_{10}	$s-t$	0	0	0	$1-t-k(s-t+1)$	$-t-k(s-t)$	$-k$	0	0	0	1
		c_j-z_j	0	0	0	$(t-s-1)(k+n)-s-t+1$	$(t-s)(k+n)-s-t$	$-k-n$	0	0	0	0

(Case 3) s − t = 1, s = k.

For this case, we need to solve IPM II with the same method. Table 6 shows the simplex table obtained after three iterations.

Table 6. Simplex table of IPM II for Case 3 after three iterations.

C_B	$c_j \rightarrow$ X_B	b	$-s$ x_1	$s-n$ x_2	-1 x_3	0 x_4	0 x_5	0 x_6	0 x_7	0 x_8	0 x_9	0 x_{10}
$-s$	x_1	$n-s$	1	0	0	$s-n$	-1	$s-n+1$	0	0	0	0
-1	x_3	$t(n-s)+1$	0	0	1	$t(s-n)$	$-t$	$-1-t(n-s-1)$	0	0	0	0
$s-n$	x_2	1	0	1	0	-1	0	-1	0	0	0	0
0	x_7	$n-s-2$	0	0	0	$s-n+1$	[−1]	$s-n+2$	1	0	0	0
0	x_8	$n-s-1$	0	0	0	$s-n$	-1	$s-n+1$	0	1	0	0
0	x_9	0	0	0	0	-1	0	-1	0	0	1	0
0	x_{10}	$t(n-s)$	0	0	0	$t(s-n)$	$-t$	$-1-t(n-s-1)$	0	0	0	1
	$c_j - z_j$		0	0	0	$(s+t+1)(s-n)$	$-s-t$	$s(s-n+2)-t(n-s-1)-1-n$	0	0	0	0

If $n-s \geq 2$, i.e. $n-s-2 \geq 0$, then the calculation is terminated and we have (K, N)-VCS = $(t(n-s)+1, s(n-s)+n-s)$-VCS with $\omega_1 = n-s$ and $\omega_2 = 1$. Otherwise, $n-s-2 < 0$ does not satisfy the conditions that are given in Equation (5).

Similarly, for Case 4, Case 5, and Case 6, the same analysis method is used to solve the corresponding IPM. Finally, we have the solutions of the corresponding IPM with different conditions. as shown in Table 7.

Table 7. The solutions of IPM for all cases.

Conditions			Solution (ω_1, ω_2,K)	Case
$s < k$	$t = s$	$V' \geq 0$ *	$(n-k+1, 1, t(n-k)+k)$	Case1
	$t \neq s$	$V' < 0$	$-$ $(s-t+2, s-t+1, t-k(t-s-1))$	Case2
$s = k$	$s-t=1$	$V'' \geq 0$ *	$(n-s, 1, t(n-s)+1)$	Case3
	$s-t \neq 1$	$V'' < 0$	$-$ $(s-t+1, s-t, t-s(t-s))$	Case4
$s > k$	$k-t=1$	$V''' \geq 0$ *	$(n-s, 1, t(n-s)+1)$	Case5
	$k-t \neq 1$	$V''' < 0$	$-$ $(k-t+1, k-t, t-k(t-k))$	Case6

* Where $V' = (t+n-k-s)(s-t)+t-s-1$, $V'' = (t+n-k-s)(s-t)+s-n$, $V''' = (t+n-k-s)(k-t)+s-n$.

From Table 7, for the most cases, we can find the solutions (ω_1, ω_2, K) of the corresponding IPM. Since N can be calculated by Equation (6), we can construct (t, s, k, n)-EVCS by the corresponding monotonic (K, N)-VCS. For some common cases of (t, s, k, n)-EVCS, we list the solutions of the corresponding IPM, the values of ω_1, ω_2, K, and N in Table 8.

Table 8. The solutions of IPM for some specific essential participants (EVCSs).

(t, s, k, n)-EVCS	(K, N)-VCS	ω_1	ω_2	Case
(1, 1, 2, 3)-EVCS	(3, 4)-VCS	2	1	1
(1, 1, 2, 4)-EVCS	(4, 6)-VCS	3	1	1
(1, 1, 2, 5)-EVCS	(5, 8)-VCS	4	1	1
(1, 1, 3, 5)-EVCS	(5, 7)-VCS	3	1	1
(1, 1, 3, 6)-EVCS	(6, 9)-VCS	4	1	1
(1, 2, 2, 4)-EVCS	(3, 6)-VCS	2	1	3
(1, 2, 2, 5)-EVCS	(4, 9)-VCS	3	1	3
(1, 2, 2, 6)-EVCS	(5, 12)-VCS	4	1	3
(1, 2, 3, 5)-EVCS	(7, 12)-VCS	3	2	2
(1, 2, 3, 6)-EVCS	–	–	–	2
(1, 3, 2, 5)-EVCS	(3, 8)-VCS	2	1	5
(1, 3, 2, 6)-EVCS	(4, 12)-VCS	3	1	5
(1, 3, 3, 6)-EVCS	(7, 15)-VCS	3	2	4
(1, 3, 3, 7)-EVCS	–	–	–	4
(1, 4, 3, 7)-EVCS	(7, 18)-VCS	3	2	6
(1, 4, 3, 8)-EVCS	–	–	–	6
(2, 2, 3, 4)-EVCS	(5, 6)-VCS	2	1	1
(2, 2, 3, 5)-EVCS	(7, 9)-VCS	3	1	1
(2, 2, 3, 6)-EVCS	(9, 12)-VCS	4	1	1
(2, 2, 4, 5)-EVCS	(6, 7)-VCS	2	1	1
(2, 2, 4, 6)-EVCS	(8, 10)-VCS	3	1	1
(2, 3, 3, 5)-EVCS	(5, 8)-VCS	2	1	3
(2, 3, 3, 6)-EVCS	(7, 12)-VCS	3	1	3
(2, 3, 4, 6)-EVCS	(10, 15)-VCS	3	2	2
(2, 4, 3, 6)-EVCS	(5, 10)-VCS	2	1	5

4. Experimental results and Comparison

4.1. Experimental Results

In this subsection, we conduct two experiments to verify the feasibility of the proposed scheme.

Example 3. *The experiment of the proposed (1, 1, 2, 3)-EVCS.*

By Table 8, we can generate our (1, 1, 2, 3)-EVCS by a monotonic (3, 4)-VCS with $\omega_1 = 2$, and $\omega_2 = 1$. First, we share the secret image into four shadows by a monotonic (3, 4)-VCS. Subsequently, the first two shadows are used to generate the essential shadow of (1, 1, 2, 3)-EVCS by OR operation. Additionally, the left two shadows are treated as two non-essential shadows of (1, 1, 2, 3)-EVCS, respectively. Finally, we have three shadows of (1, 1, 2, 3)-EVCS with one essential and two non-essentials.

The chosen (3, 4)-VCS used to construct (1, 1, 2, 3)-EVCS can be any monotonic (3, 4)-VCS proposed by researchers. In this example, we choose two monotonic (3, 4)-VCS separately to implement (1, 1, 2, 3)-EVCS. First, we use monotone (3, 4)-VCS in Example 1 to implement (1, 1, 2, 3)-EVCS. We already show the experiment of (3, 4)-VCS in Example 1. Figure 1 shows the four generated shadows of (3, 4)-VCS. Now, we can generate the essential shadow of (1, 1, 2, 3)-EVCS, as shown in Figure 1a, by performing OR operation on the Figure 1b,c. The rest two shadows Figure 1d,e are treated as two non-essential shadows of (1, 1, 2, 3)-EVCS as shown in Figure 5b,c. For (1, 1, 2, 3)-EVCS, the qualified sets are {1,2}, {1,3}, and {1,2,3}. We show the revealed image by different qualified sets of shadows in Figure 5d,e and g. As we can see, the secret image can only be revealed with at least two shadows, including the essential one. Without the essential shadow, stacking two non-essentials cannot reveal the secret image, as shown in Figure 5f. Since the (3, 4)-VCS has the size expansion 6 and contrast loss of the revealed image. Each shadow of (1, 1, 2, 3)-EVCS has the size six times of the secret image. The visual quality of the revealed image is also degraded.

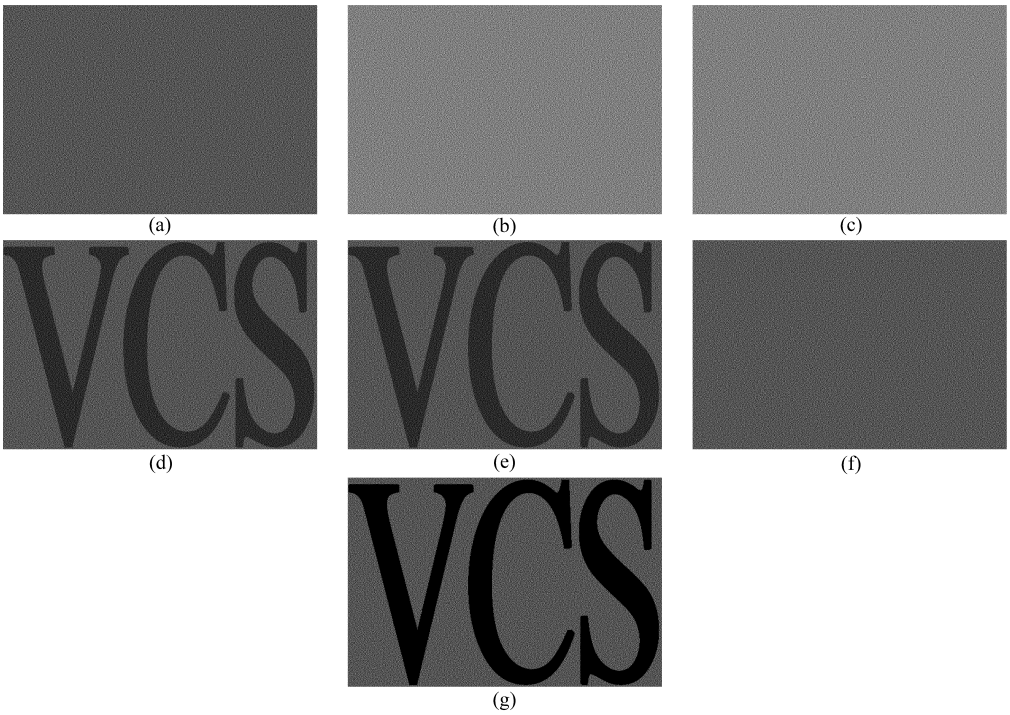

Figure 5. The experimental results of proposed scheme for (1, 1, 2, 3)-EVCS. (**a**) essential shadow; (**b**,**c**) two non-essential shadows; (**d**) revealed image by shadow 1 and 2; (**e**) revealed image by shadow 1 and 3; (**f**) revealed image by shadow 2 and 3; and, (**g**) revealed image by all shadows.

Second, we choose Yan et al.'s (3, 4)-RGVCS to implement (1, 1, 2, 3)-EVCS. First, a secret image (Figure 1a) is shared into four shadows by Yan et al.'s (3, 4)-RGVCS. With the same method, performing OR operation on the first two shadows can obtain the essential shadow of (1, 1, 2, 3)-EVCS. The other two shadows of (3, 4)-RGVCS are treated as two non-essentials, respectively. Since (3, 4)-RGVCS generates shadows without size expansion, all of the shadows of (1, 1, 2, 3)-EVCS have the same size as the secret image. Figure 6a–c show the three generated shadows of (1, 1, 2, 3)-EVCS. In the revealing process, stacking qualified set of shadows can reveal the secret image. Figure 6d–g show the revealed images with different shadows. As we can see, without the essential shadow, we cannot reveal the secret image by the shadows. EVCS based on RGVCS can achieve better performance over that based on traditional VCS since RGVCS has the advantage of no size expansion.

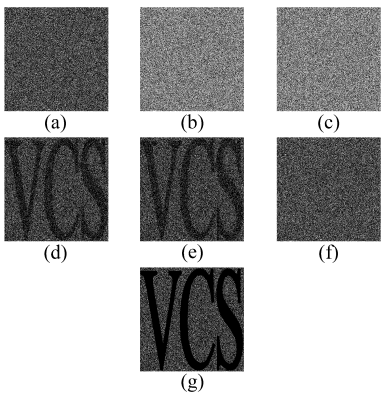

Figure 6. The experiment of the proposed (1, 2, 3)-EVCS. (**a**) essential shadow; (**b**,**c**) two non-essential shadows; (**d**) revealed image by shadow 1 and 2; (**e**) revealed image by shadow 1 and 3; (**f**) revealed image by shadow 2 and 3; (**g**) revealed image by all three shadows.

Example 4. *The experiment of the proposed (1,2,2,4)-EVCS.*

(1, 2, 2, 4)-EVCS can be implemented by (3, 6)-VCS according to the solution of the aforementioned IPM. We still adopt Yan et al. 's (3, 6) -RGVCS to implement (1, 2, 2, 4)-EVCS in order to achieve better performance. Since $s = 2$ and $\omega_1 = 2$, we can get two essential shadows by performing the OR operation twice on any two shadows. It should be noted that the operands of twice OR operations are non-overlapping from each other, i.e. the same shadow of VCS can only participate in OR operation for one time in generating shadow of EVCS and cannot participate in the formation of two shadows of EVCS at the same time. Figure 2 shows the experimental results of Yan et al.'s (3, 6)-RGVCS. Here, we generate the first essential shadow by performing OR operation on Figure 2b,c, and the second essential shadow is generated by performing OR operation on Figure 2d,e. The remaining two shadows of RGVCS are considered as two non-essential shadows of (1, 2, 2, 4)-EVCS.

Figure 7 shows the experimental results. Figure 7a,b are two essential shadows and Figure 7c,d show two non-essential shadows, which have the same size of the secret image. Figure 7e–j illustrate the recovered image by stacking any two shadows. Since none of the essential shadows are included in {3, 4}, Figure 7j is as cluttered as random noise and does not show any information about the secret image. Figure 7k–n show the revealed image recovered by any three shadows and the last one is revealed by all shadows. Stacking two or more shadows that include any one or two essential shadows can reveal the secret image. Reconstruction without the essential shadow cannot get obtain information regarding the secret.

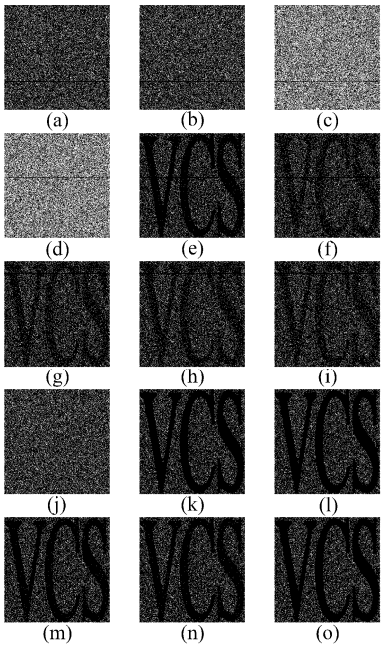

Figure 7. An experiment of the proposed (1,2,2,4)-EVCS. (**a**,**b**) two essential shadows; (**c**,*d*) two non-essential shadows; (**e**) revealed image by shadow 1 and 2; (**f**) revealed image by shadow 1 and 3; (**g**) revealed image by shadow 1 and 4; (**h**) revealed image by shadow 2 and 3; (**i**) revealed image by shadow 2 and 4; (**j**) revealed image by shadow 3 and 4; (**k**) revealed image by shadow 1, 2, and 3; (l) revealed image by shadow 1, 2 and 4; (**m**) revealed image by shadow 1, 3, and 4; (**n**) revealed image by shadow 2, 3, and 4; and, (**o**) revealed image by four shadows.

4.2. Comparison and Discussion

This subsection compares the proposed scheme with some literature schemes in terms of functionalities, as shown in Table 9. Both [24] and [25] are polynomial-based ESIS schemes that can reveal secret image perfectly, while they suffers from the disadvantage of heavy computation that secret information cannot be obtained by superimposing shadow images. In addition, these two schemes have the problem of unequal sizes of essential shadow and non-essential shadow, and the concatenation of sub-shadows. However, scheme [17,18,28] and the proposed scheme do not have

these two problems. Among them, scheme [17,18] and [28] are only applicable to (t, k, n), while the proposed scheme is applicable to a wider range. When compared with the polynomial-based ESIS scheme, the proposed scheme does not require complicated mathematical calculations in the secret reconstruction process. Most importantly, our scheme can select the appropriate (k, n)-VCS according to the actual needs. With the improvement of the (k, n)-VCS scheme, our scheme will also achieve better visual effects. The threshold condition refers to that the number of shadows in a qualified set should be no less than a threshold number. The essentiality condition refers to that a qualified set of shadows should contain at least a certain number of essentials. All of the mentioned schemes in Table 9 satisfy the threshold condition. When compared with the general VCS [32,35], our scheme not only satisfies the threshold condition, but it also satisfies the essentiality condition.

Table 9. The comparison of functionality among the literature schemes and proposed EVCS.

Schemes	Construction Method	Size Expansion	Concatenation of Sub-shadows	Essentiality	Decoding Operation	Stacking-to-see
Scheme [17]	VCS	Large	No	Yes	OR	Yes
Scheme [18]	VCS	Large	No	Yes	OR	Yes
Scheme [24]	PSIS	Small	Yes	Yes	Lagrange's interpolation	No
Scheme [25]	PSIS	Small	Yes	Yes	Lagrange's interpolation	No
Scheme [26]	PSIS	Small	No	Yes	Birkhoff Interpolation	No
Scheme [28]	PSIS	Small	No	Yes	Lagrange's interpolation	No
Scheme [32]	RGVCS	Small	No	No	OR	Yes
Scheme [35]	XVCS	Medium	Yes	No	XOR	Yes
Proposed scheme	VCS	Alternative	No	Yes	OR	Yes

In general, the reconstructed image of VCS is not completely consistent with the secret image. The size expansion and visual quality are commonly used to measure the performance of VCS. The proposed scheme has high flexibility that a (t, s, k, n)-EVCS can be constructed utilizing any monotone (k, n)-VCS. The performance of EVCS is determined by the performance of chosen VCS. For example, if we construct (t, s, k, n)-EVCS by Yan et al.'s (k, n)-RGVCS, we can achieve (t, s, k, n)-EVCS without size expansion. In Example 3, we adopt a traditional (k, n)-VCS and a non-size-expansion (k, n)-RGVCS in order to validate our scheme, respectively. From Example 3, we know that EVCS based on RGVCS has better size expansion than that based on traditional VCS. Arumugam et al. [17] proposed (1, 1, k, n)-EVCS and Guo et al. [18] proposed (t, t, k, n)-EVCS. Both schemes in [17,18] are special cases of (t, s, k, n)-EVCS, and construct the basis matrices of EVCS from the basis matrices of VCS. The main disadvantage of EVCS in [17,18] is the large size expansion, which is not convenient for the storage and transmission. We compare the experiment results among Arumugam et al.'s (1, 1, 3, 4)-VCS, Guo et al.'s (1, 1, 3, 4)-VCS, and the proposed (1, 1, 3, 4)-EVCS. Figure 8 shows the experimental results. The sizes of the revealed secret images of the three schemes are six, six, and one times of the secret image, respectively. Figure 8b,d show the revealed secret images of the three schemes, respectively. As we can see, the revealed images of Arumugam et al.'s (1, 1, 3, 4)-VCS, Guo et al.'s (1, 1, 3, 4)-VCS have large size expansion, while the proposed (1, 1, 3, 4)-EVCS has no size expansion, namely, the size of revealed image is the same as the original secret image. In addition, our proposed scheme can realize general (t, s, k, n)-EVCS.

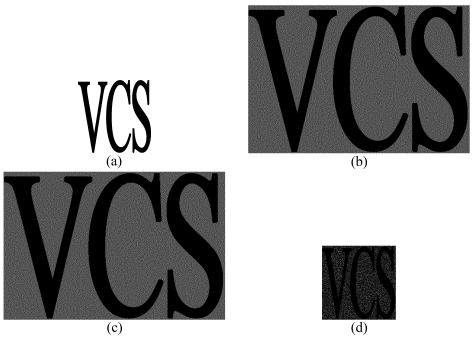

Figure 8. The experimental results of Arumugam et al.'s (1, 1, 3, 4)-VCS, Guo et al.'s (1, 1, 3, 4)-VCS and the proposed (1, 1, 3, 4)-EVCS. (**a**) the secret image; and, (**b**–**d**) the revealed image of the three schemes, respectively.

In this paper, we propose EVCS based on existing monotonic VCS. From the definition of (K, N)-VCS, the contrast condition guarantees that no information about the secret can be revealed with less than K shadows. When we construct (t, s, k, n)-EVCS from a monotonic (K, N)-VCS, the security condition of EVCS is derived from that of VCS. In Section 3.2, we show the construction method of EVCS from a monotonic (K, N)-VCS, where w_1 (resp. w_2) shadows of VCS are stacked together as an essential (resp. non-essential) shadow of EVCS. Combining with the access structures of EVCS and VCS, some constraints should be satisfied when determining the values of w_1 and w_2, as shown in Equations (9)–(11). Therefore, a forbidden set of EVCS cannot contribute at least K shadows of VCS, and then they cannot reveal any information regarding the secret image. In other words, the security level of the proposed EVCS is the same as the VCS.

For the security condition of (K, N)-VCS, no information regarding the secret image can be revealed with less than K shadows. With less than K shadows, the secret pixel is revealed as a black pixel or a white pixel with the same probability. From the view point of information theory, the entropy has the largest value. For the contrast condition of (K, N)-VCS, any K shadows can reveal the secret image by stacking operation. Each shadow can be viewed as the key to decode the secret image. Since all of the shadows have the same size, VCS can be also viewed as a one-time pad system. The Shannon theories have already proven that one-time pad system is a perfect secret system. Therefore, the secret image cannot be revealed with less than K shadows, even with computational resources. To the best of our knowledge, there is no cryptoanalysis scheme for VCS while using machine learning or deep learning algorithms.

4.3. Applications of EVCS

VCS is technique for sharing a binary secret image among the participants. The main advantage of VCS is that the revealing process does not need the computer resources. VCS has potential application when the collective decision making is required and the computer resources is not available. For example, in the battlefield, VCS shares the military instruction from the commander is shared into multiple shadows. Each shadow is delivered to a soldier. Since the environment of the battlefield is not predictable, the soldiers can decode the military instruction by stacking their shadows without any computational resources. VCS with essential participants (EVCS) divide the participants into two groups: essentials with higher status and non-essentials with lower status. In the revealing process, (t, e, k, n)-EVCS requires k participants, including t essentials to stacking their shadows. EVCS has the potential application when some participants are accorded special privileges due to their status or importance, e.g., heads of government, managers of company, high-level corporate officers, major employers, etc.

EVCS also has potential application in key distribution when exchanging message in a public secure network [30,31]. The trapped users may have different social attributes. Hence, they can be divided as essentials or non-essentials according to their attributes. Before message exchange, users

need to obtain the correct password in order to ensure the confidentiality of communication. The password usually consists of letters or numbers. It is suitable to share the password by EVCS. In addition, EVCS has the advantage of easy-decoding without computation. The environment of the various emergency events, e.g., natural disaster, terrorist attacks, etc., is terrible. The computational resources may be also limited. EVCS is a perfect way to decode secret by stacking shadows without computer. The decoding process is simple, and do not need any cryptography knowledge.

In realistic implementation, the access structure of EVCS should be open to public. The values of t, s, k, and n are known to the participants. In addition, the essentials and non-essentials are credible and known to everyone. With the above known information, the participants can confirm whether they can constitute a qualified set. When the participants of a qualified set are gathered and their shadows are collected, the secret information can be easily decoded by stacking their shadows.

5. Conclusions

In this paper, we proposed a construction method for (t, s, k, n)-EVCS with essential participants. The proposed EVCS is constructed from a monotonic VCS that is based on integer programming. When cmpared with literature EVCS, the proposed EVCS might achieve no size expansion if we adopt RGVCS to generate shadows. By solving the corresponding integer programming model, we give the condition and optimal choice of (K, N)-VCS to construct (t, s, k, n)-EVCS. The proposed EVCS also has the advantage of easy decoding since VCS can reveal the secret image by stacking shadows. The experimental results show the feasibility of our scheme. The construction method of general (t, s, k, n)-EVCS scheme with better performance needs further study.

Author Contributions: Conceptualization, P.L. and L.Y.; methodology, L.Y.; formal analysis, J.M. 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).

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

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

References

1. Naor, M.; Shamir, A. Visual Cryptography. *Lect. Notes Comput. Sci.* **1994**, *950*, 1–12.
2. Liu, F.; Guo, T.; Wu, C.K.; Qian, L. Improving the visual quality of size invariant visual cryptography scheme. *J. Vis. Commun. Image Represent.* **2012**, *23*, 331–342. [CrossRef]
3. Fu, Z.; Yu, B. Optimal pixel expansion of deterministic visual cryptography scheme. *Multimed. Tools Appl.* **2014**, *73*, 1177–1193. [CrossRef]
4. Li, P.; Ma, J.; Yin, L.; Ma, Q. A Construction Method of (2, 3) Visual Cryptography Scheme. *IEEE Access* **2020**, *8*, 32840–32849. [CrossRef]
5. Cimato, S.; Prisco, R.D.; Santis, A.D. Optimal Colored Threshold Visual Cryptography Schemes. *Des. Codes Cryptogr.* **2005**, *35*, 311–335. [CrossRef]
6. Liu, F.; Wu, C.K.; Lin, X.J. Colour visual cryptography schemes. *IET Inf. Secur.* **2008**, *2*, 151–165. [CrossRef]
7. Dutta, S.; Adhikari, A.; Ruj, S. Maximal contrast color visual secret sharing schemes. *Des. Codes Cryptogr.* **2019**, *87*, 1699–1711. [CrossRef]
8. Shyu, S.J.; Huang, S.Y.; Lee, Y.K.; Wang, R.Z.; Chen, K. Sharing multiple secrets in visual cryptography. *Pattern Recognit.* **2007**, *40*, 3633–3651. [CrossRef]
9. Chen, C.C.; Wu, W.J. A secure Boolean-based multi-secret image sharing scheme. *J. Syst. Softw.* **2014**, *92*, 107–114. [CrossRef]
10. Tsai, D.S.; Chen, T.; Horng, G. On generating meaningful shares in visual secret sharing scheme. *Imaging Sci. J.* **2008**, *56*, 49–55. [CrossRef]

11. Shyu, S.J. Threshold Visual Cryptographic Scheme with Meaningful Shares. *IEEE Signal Process. Lett.* **2014**, *21*, 1521–1525. [CrossRef]
12. Shyu, S.J. Image encryption by random grids. *Pattern Recognit.* **2007**, *40*, 1014–1031. [CrossRef]
13. Chen, T.H.; Tsao, K.H. User-Friendly Random-Grid-Based Visual Secret Sharing. *IEEE Trans. Circuits Syst. Video Technol.* **2011**, *21*, 1693–1703. [CrossRef]
14. 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]
15. Hu, H.; Shen, G.; Liu, Y.; Fu, Z.; Yu, B. Improved schemes for visual secret sharing based on random grids. *Multimed. Tools Appl.* **2019**, *78*, 12055–12082. [CrossRef]
16. Jin, D.; Yan, W.-Q.; Kankanhalli, M.S. Progressive color visual cryptography. *J. Electron. Imaging* **2005**, *14*, 033019. [CrossRef]
17. Arumugam, S.; Lakshmanan, R.; Nagar, A.K. On (k,n)*-visual cryptography scheme. *Des. Codes Cryptogr.* **2014**, *71*, 153–162. [CrossRef]
18. Guo, T.; Liu, F.; Wu, C.K.; Ren, Y.W.; Wang, W. On (k, n) Visual Cryptography Scheme with t Essential Parties. *Lect. Notes Comput. Sci.* **2013**, *8317*, 56–68.
19. Thien, C.C.; Lin, J.-C. Secret image sharing. *Comput. Graph.* **2002**, *26*, 765–770. [CrossRef]
20. Wu, Z.; Liu, Y.-N.; Wang, D.; Yang, C.-N. An Efficient Essential Secret Image Sharing Scheme Using Derivative Polynomial. *Symmetry* **2019**, *11*, 69. [CrossRef]
21. Liu, Y.; Yang, C.; Wang, Y.; Lei, Z.; Ji, W. Cheating Identifiable Secret Sharing Scheme Using Symmetric Bivariate Polynomial. *Inf. Sci.* **2018**, *453*, 21–29. [CrossRef]
22. Zhou, X.; Lu, Y.; Yan, X.; Wang, Y.; Liu, L. Lossless and Efficient Polynomial-Based Secret Image Sharing with Reduced Shadow Size. *Symmetry* **2018**, *10*, 249. [CrossRef]
23. 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]
24. Li, P.; Yang, C.N.; Wu, C.C.; Kong, Q.; Ma, Y. Essential secret image sharing scheme with different importance of shadows. *J. Vis. Commun. Image Represent.* **2013**, *24*, 1106–1114. [CrossRef]
25. Yang, C.N.; Li, P.; Wu, C.C.; Cai, S.R. Reducing shadow size in essential secret image sharing by conjunctive hierarchical approach. *Signal Process. Image Commun.* **2015**, *31*, 1–9. [CrossRef]
26. Li, P.; Yang, C.N.; Zhou, Z. Essential secret image sharing scheme with the same size of shadows. *Digit. Signal Process.* **2016**, *50*, 51–60. [CrossRef]
27. 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]
28. Peng, L.; Liu, Z.; Yang, C.N. A construction method of (t,k,n)-essential secret image sharing scheme. *Signal Process. Image Commun.* **2018**, *65*, 210–220.
29. Liu, Y.; Yang, C. Scalable secret image sharing scheme with essential shadows. *Signal Process. Image Commun.* **2017**, *58*, 49–55. [CrossRef]
30. Tsiropoulou, E.; Koukas, K.; Papavassiliou, S. A socio-physical and mobility-aware coalition formation mechanism in public safety networks. *EAI Endorsed Trans. Future Internet* **2018**, *4*, 154176. [CrossRef]
31. Thai, M.; Wu, W.; Xiong, H. *Big Data in Complex and Social Networks*; CRC Press Taylor & Francis Group: Boca Raton, FL, USA, 2016; pp. 125–182.
32. 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]
33. Ateniese, G.; Blundo, C.; Santis, A.D.; Stinson, D.R. Visual Cryptography for General Access Structures. *Inf. Comput.* **1996**, *129*, 86–106. [CrossRef]
34. Kafri, O.; Keren, E. Encryption of pictures and shapes by random grids. *Opt. Lett.* **1987**, *12*, 377–379. [CrossRef]
35. Shen, G.; Liu, F.; Fu, Z.; Yu, B. Perfect contrast XOR-based visual cryptography schemes via linear algebra. *Des. Codes Cryptogr.* **2017**, *85*, 15–37. [CrossRef]