symmetry

MDPI

Article

RG-Based Region Incrementing Visual Cryptography with Abilities of OR and XOR Decryption

Yu-Ru Lin and Justie Su-Tzu Juan *

Department of Computer Science and Information Engineering, National Chi Nan University, Puli, Nantou 545, Taiwan; s111321509@mail1.ncnu.edu.tw
* Correspondence: jsjuan@ncnu.edu.tw

Abstract: Visual cryptography (VC) is a cryptographic technique that allows the encryption of a secret image into multiple shares. When the shares of a qualified subset are superimposed, the original secret image can be visually recovered. Region incremental visual cryptography (RIVC) is a class of visual cryptography; it encrypts a single image into a shared image with multiple levels of secrecy, and when decrypted, the secret image of each region can be gradually recovered. Traditional VC encrypts two black-and-white images, and its recovery method is equivalent to a logical OR operation. To obtain a better recognizability of the restored image, the XORoperator becomes a simple and efficient method of encryption and decryption. Because the XOR operation needs extra cost or equipment, if the equipment cannot be obtained, the scheme can be more flexible if the secret can still be restored by using OR decryption (superimpose). In this paper, we propose a novel RIVC that allows encoding multiple secret regions of a secret image into n random grids. Both the OR operation and the XOR operation can be used as operations during decryption. The proposed scheme is evaluated by simulation, and the experimental result shows its correctness, effectiveness and practicability.

Keywords: visual cryptography; region incrementing; secret sharing; XOR operation; random grid

Citation: Lin, Y.-R.; Juan, J.S.-T. RG-Based Region Incrementing Visual Cryptography with Abilities of OR and XOR Decryption. *Symmetry* **2024**, *16*, 153. https://doi.org/10.3390/sym16020153

Academic Editor: Tomohiro Inagaki

Received: 25 December 2023
Revised: 18 January 2024
Accepted: 24 January 2024
Published: 28 January 2024

1. Introduction

Introduced in 1995 [1], the visual cryptography scheme (VCS for short) was a new technique that involves encrypting a secret image into n random grid images (also called "shares") that can be stacked together to disclose the original secret image. To enhance this approach, a (k, n)-threshold visual cryptography scheme ((k, n) VCS for short) was presented. This method entails the encryption of a binary secret image into n shares, with the condition that at least k shares (where $k \leq n$) must be combined to reconstruct the original image. If fewer than k shares are utilized, no clue about the original image can be discerned.

Generally speaking, cryptography systems in information security can be divided into symmetric cryptography systems and asymmetric cryptography systems. For VCS, if shares are regarded as the "key" and "ciphertext" generated by the encryption algorithm, then because at least k shares need to be collected to recover the "plaintext", the visual cipher can be regarded as a type of symmetric cryptosystem. Compared with traditional cryptography systems, VCS usually does not pursue the ability to completely restore the original secret. Instead, it focuses on the ability to use human vision to identify the original secret during restoration.

Looking back to 1987, Kafri and Keren [2] were pioneers in the field of visual secret sharing (VSS for short) through the use of random grids (RG for short). In the field of visual cryptography, random grids are a technique used to divide a secret image into irregular grids or patterns. Each grid or pattern is designed to appear random when viewed individually and does not contain the original secret information, as shown in Figure 1. The original secret image can only be reconstructed when these random grids or

patterns are combined or superimposed. This innovative approach rectified the limitation of secret pixel expansion and eliminated the necessity for codebook designs, which were shortcomings in Naor and Shamir's method [1]. Although Kafri and Keren [2] initially introduced the construction of (2, 2) RG-based VCS, it was not until 2009 that Shyu [3], as well as Chen and Tsao [4] independently proposed the first (n, n) RG-based VCS. In 2011, Chen and Tsao [5] were pioneers in introducing (k, n) RG-based VCS.

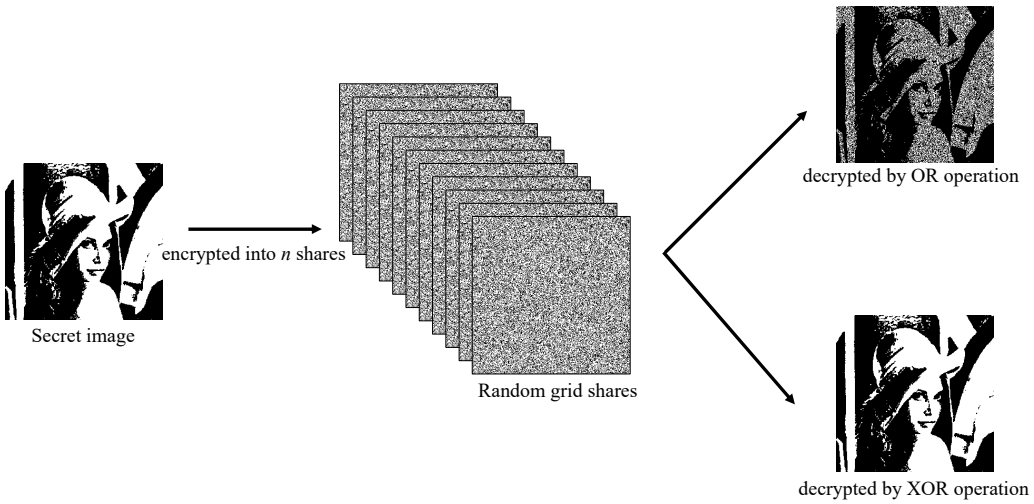


Figure 1. The structure of OR (⊗) and XOR (⊕) decryption.

Traditional VCS involves directly stacking two binary (black-and-white) images, which is equivalent to performing a logical OR operation just like the human visual system (HVS); we treat black pixels as 1 and white pixels as 0. In other words, it treats white pixels as transparent and black pixels as black. When the two images are superimposed, what the human eye perceives is the content of the OR-reconstructed image. In practice, it is equivalent to printing the share on a transparency separately. You can simply overlay those transparencies when restoring. However, this approach results in the image becoming progressively darker and makes it difficult to identify the original secret as more shares are stacked. Furthermore, it is impossible to completely restore the original secret. Therefore, a method for recovering the secret using an XOR operation was developed. By assigning 0 (white) when both pixels are the same and 1 (black) when they differ, as shown in Table 1, this operation makes it possible to fully recover the secret, as shown in Figure 1. In practice, it can be completed by using additional equipment to perform XOR operations. For example, using a copier that can print out black-and-white reverse images (inverting color, that is, a NOT operation logically) and a formula that converts the XOR operation into OR and NOT operations. Because the XOR operation needs extra cost or equipment, if the equipment cannot be obtained, the scheme can be more flexible if the secret can still be restored by using the OR operation (direct stacking). Currently, some scholars have studied a VCS that can restore the original image using both OR and XOR decryptions [6–8].

Table 1. Stacking result by OR (⊗) and XOR (⊕) calculation.

Pixel 1 p_1	**Pixel 2** p_2	$p_1 \otimes p_2$	$p_1 \oplus p_2$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	0

In 2009, Wang proposed a region incrementing visual cryptography scheme (RIVCS for short) [9], which enables a dealer to divide the content of a secret image S into multiple

regions and assign a level of secrecy to each region. In this q-level scheme, the secrecy level of a region ranges from 1 to q, with the first-level secret being the least significant and the q-level secret being the most significant. The dealer has the flexibility to assign each region of S a secrecy level that aligns with the specific needs of their application, reflecting the degree of secrecy required for that region. This level of regulation allows for more precise control over the distribution of secrets, ensuring that sensitive information remains highly protected while still allowing for the sharing of less sensitive information. Wang's scheme [9] utilizes the basic matrix proposed by Naor and Shamir [1] for encryption. However, the encryption process results in pixel expansion and can only be restored using an OR operation. For the purpose of addressing the limitations of pixel expansion and the reliance on a single OR-based decryption, we have made improvements accordingly and propose an RG-based RIVCS with the abilities of OR and XOR decryption. While visual cryptography is considered symmetric due to the division between public and private keys, it is important to note that VC, including RIVCS, often does not rely on complex mathematical operations or key pairs like traditional symmetric encryption (e.g., AES or DES). Instead, it uses visual properties and simple pixel stacking techniques for decryption. After Wang, many scholars have continued research on RIVCS. Wang et al. [10] proposed a (2, 3) RIVCS using random grids in 2010. In 2012, Shyu and Jiang [11] focused on minimizing pixel expansion of basis matrices (2, n) RIVCS, and Yang et al. [12] concentrated on applying (k, n) RIVCS to color images. Kumar and Sharma [13] proposed a (k, n) RIVCS using random grids in 2015. They gave three construction methods using the ideas of Kafri and Keren [2], and the number of the secrecy levels is always equal to $n - k + 1$. Furthermore, in 2019, Yang et al. [14] achieved progressive restoration in a (k, n) RIVCS.

Currently, many scholars are actively engaged in research on visual cryptography schemes. Each researcher brings a unique focus to their exploration of VCS. For example, Huang et al. emphasize generating meaningful encrypted images and utilizing the XOR operator for restoration [15]. Chen and Juan specialize in encrypting grayscale and color images into meaningful encrypted images [16]. Panchbhai and Varade focus on the research of a block-based progressive secret sharing scheme (BPVSS for short) for both grayscale and color images [17]. Zhang et al. concentrate on a group secret sharing scheme [18], where collaborative efforts within a group enable the recovery of secret information.

Therefore, the main contribution of this paper is to propose a new method for RIVCS. Compared with existing RIVCS, the proposed method not only has better visual quality in OR-based recovery in most cases (especially when stacking all shares), but also incorporates the function of XOR-based restoration. It supports OR and XOR operation decryption, making the decryption process flexible. This design allows users to use OR operations for decryption without the need for any specialized equipment, and users with additional equipment can use XOR operations to enhance the visual quality of the recovered images. To the best of our knowledge, this paper is the first to propose an innovative RIVCS that can use both OR and XOR operations for decryption.

The paper follows this structure: the subsequent section covers related work, Section 3 introduces the main result of this paper, which is (k, n) 2D_RIVCS, and Section 4 provides an analysis of this scheme, while Sections 5 and 6 showcase the experimental results and comparison with other similar works, respectively. Finally, Section 7 provides a summary and introduces some innovative ideas for future work.

2. Related Work

In this section, we will provide an overview of the relevant literature that has informed and influenced the research presented in this paper. These foundational references serve as the basis for the exploration conducted in this study. Section 2.1 will introduce the variables and definitions utilized in this paper. Section 2.2 will explore Kafri and Keren's pioneering work on (2, 2) RG-based VCS [2], a method grounded in random grids for encryption. Section 2.3 will present the (k, n) RG-based VCS proposed by Chen and Tsao in 2011 [5]. Section 2.4 will introduce Wang's innovative contribution to the field, known

as region incrementing visual cryptography (RIVCS) [9]. Lastly, Section 2.5 will provide insights into Lin and Juan's RG-based (k, n)-threshold visual cryptography with abilities of OR and XOR decryption (2D_RIVCS) [6]. These foundational studies lay the groundwork for our investigation and contribute to the understanding of the developments in visual cryptography.

2.1. Variables and Definitions

First, the variables used in this paper will be introduced, as shown in Table 2.

Table 2. Notations used in this paper.

Notation	Definition
0	White (transparent) pixel
1	Black (opaque) pixel
$\otimes$	OR operation
$\oplus$	XOR operation
S	The secret image
$S(0)/S(1)$	All of S's transparent/opaque pixels
s	A pixel of S
W, H	The width and height of the image
B	The encrypted image
b	The encrypted image before rearrangement
R	The reconstructed image
(i, j)	The position of the pixel in the image, $1 \leq i \leq W$, $1 \leq j \leq H$
q	The number of secret levels
Sl	Secret level of secret image

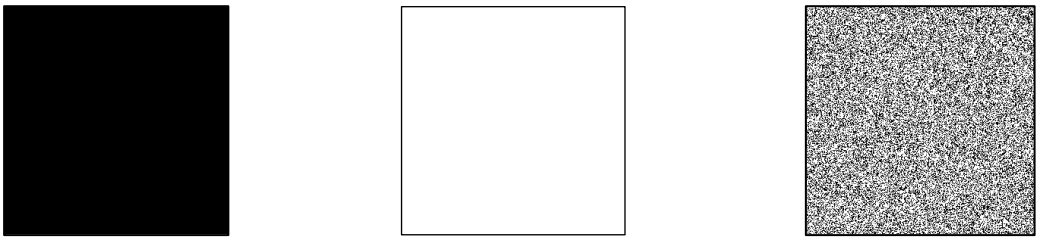
After defining the variables, the next step involves introducing some tools we will employ to assess the quality of the reconstructed images: average light transmission (called T) and contrast (called α). These tools are essential in evaluating the fidelity and visual clarity of the reconstructed images. Average light transmission allows us to compute how well the secret information is revealed, while contrast measures the sharpness and distinction between the secret content and the reconstructed image. These metrics play a crucial role in objectively quantifying the effectiveness of our approach in achieving high-quality reconstructions. Below are the definitions of average light transmission and contrast.

Definition 1. *Average Light Transmission (T) [19].*

The likelihood of a specific pixel, denoted as x, being transparent in the binary image X is represented as Prob(x = 0). This probability is indicated by the symbol $T[x]$, which expresses the light transmission of pixel x. The average light transmission of image X is expressed as

$$T[X] = \frac{\sum_{i=1}^{W} \sum_{j=1}^{H} T\left[x_{i,\ j}\right]}{W \times H}, \tag{1}$$

where $x_{i,\ j}$ is the pixel of X at the position (i, j). Therefore, when $T[X]$ is equal to 0, the entire image will consist of black pixels, as shown in Figure 2a. Conversely, when $T[X]$ is equal to 1, the entire image will be completely white, as shown in Figure 2b. When $T[X]$ is set to $\frac{1}{2}$, the entire image will appear as a half-black and half-white composition, as depicted in Figure 2c. Note that a white pixel is denoted by 0, but the light transmission of a white pixel $T[0] = 1$; a black pixel is denoted by 1, and the light transmission of a black pixel $T[1] = 0$.



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

Figure 2. Examples of different average light transmission (T) (**a**) Average light transmission is 0. (**b**) Average light transmission is 1. (**c**) Average light transmission is 0.5

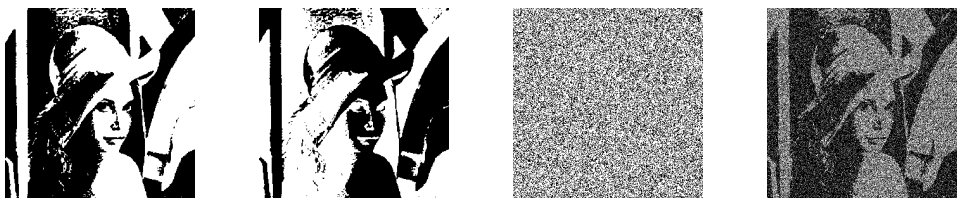
Definition 2. *Contrast (α) [19].*

Contrast serves as a critical criterion for assessing the fidelity of the reconstructed images and evaluating the effectiveness of the method proposed in this paper. The higher the contrast, the closer the visual resemblance between the reconstructed image and the original secret image. Contrast will be calculated using the previously mentioned average light transmission (T) and can be determined using the following formula:

$$\alpha = \frac{T[R[S(0)]] - T[R[S(1)]]}{1 + T[R[S(1)]]} \tag{2}$$

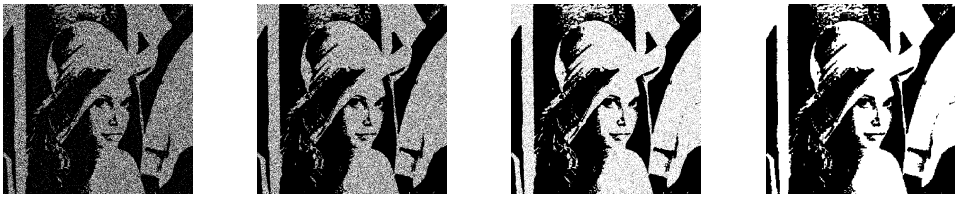
where $S(0)$ represents the white pixels in the secret image S, while $S(1)$ represents the black pixels in the same image. $T[R[S(0)]]$ signifies the average light transmission of the reconstructed image B corresponding to the white area of the secret image S, and $T[R[S(1)]]$ represents the average light transmission of the reconstructed image B corresponding to the black area of the secret image S. By utilizing the difference between $T[R[S(0)]]$ and $T[R[S(1)]]$ and dividing it by $1 + T[R[S(1)]$, the contrast can be calculated. The contrast value falls within the range of -1 to 1.

When the contrast is -1, the secret image and the reconstructed image are exact opposites, with black and white colors inverted, as shown in Figure 3b. When the contrast is 0, the secret image and the reconstructed image lack any discernible correlation, and no traces of the secret image are visible in the reconstructed image, as shown in Figure 3c. When the contrast is greater than 0, details related to the secret will gradually become apparent. As the contrast approaches 1, the content of the secret becomes clearer, as shown in Figure 3d–g. Figure 3d–g are generated based on the contrast formula. Figure 3d,e fix the average light transmission of $S(0)$ at $\frac{1}{2}$ (as in Figure 3c), and the calculated average light transmissions for $S(1)$ are $\frac{1}{4}$ and $\frac{1}{14}$, respectively. In Figure 3f,g, the average light transmission of $S(1)$ is fixed at 0 (as in Figure 3h, because it must be greater than $1/2$ and for ease of comparison with each other), and the calculated average light transmissions for $S(1)$ are $\frac{3}{5}$ and $\frac{4}{5}$, respectively. At a contrast value of 1, the secret image and the reconstructed image match perfectly, representing a perfect reconstruction, as shown in Figure 3h. This relationship is illustrated in Figure 3.



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

Figure 3. *Cont.*



(**e**) (**f**) (**g**) (**h**)

Figure 3. Examples of different contrast (α) (**a**) Secret image S. (**b**) R with $\alpha = -1$. (**c**) R with $\alpha = 0$. (**d**) R with $\alpha = 0.2$. (**e**) R with $\alpha = 0.4$. (**f**) R with $\alpha = 0.6$. (**g**) R with $\alpha = 0.8$. (**h**) R with $\alpha = 1$.

Contrast (α) provides an objective measure of how distinct and well-defined the hidden information appears within the reconstructed image. It allows us to quantitatively gauge the quality of the restoration process and its success in revealing the concealed content.

Definition 3. *Security [20].*

After proposing a scheme, it is essential to validate the security and feasibility of the scheme. In threshold-based visual cryptography, achieving security conditions involves two critical aspects. Firstly, when provided with a single share (encrypted image), no information regarding the secret image should be discernible. Secondly, when collecting fewer than the threshold value k of shares, no details about the secret image should be revealed. Notably, based on the concept of contrast, an image exhibits a contrast of 0 with the secret image S when they are entirely unrelated. Hence, when introducing a novel approach, it is important to demonstrate that both individual shares and collections of fewer than k shares are secure, resulting in a contrast of 0. To achieve a contrast of 0, it is necessary to ensure that

$$T[R[S(0)]] = T[R[S(1)]], \tag{3}$$

as dictated by the contrast formula.

Security is predicated on the concept that examining an individual encrypted image (share) or even a subset of images falling short of the threshold k should yield no information about the secret image. Consequently, ensuring that the contrast is zero in these scenarios is a key indicator of security in threshold-based visual cryptography schemes.

Definition 4. *Visually recognizable [20].*

In addition to verifying its security, the visual recognizability of a method also requires validation. In the context of visual cryptography, usability refers to the ability to reveal information about the secret image S when a sufficient number of encrypted images are collected, specifically, when more than or equal to k images are gathered. The range of contrast, as defined, is between -1 and 1. A contrast of 0 indicates that the two images are entirely unrelated, while a contrast greater than 0 theoretically implies the potential to unveil information about the secret image S. Therefore, based on the contrast formula, it is necessary to achieve

$$T[R[S(0)]] > T[R[S(1)]], \tag{4}$$

to confirm usability.

This approach ensures that when a certain number of encrypted images, equivalent to or exceeding the threshold k, are collected, the contents related to the secret image S can be revealed. This is a fundamental aspect of visual recognizability in the context of visual cryptography. Contrast, a measure that ranges between -1 and 1, plays a crucial role in determining the degree to which the secret image's content can be observed. Achieving a contrast greater than 0 is essential for usability, suggesting the potential for discerning details about the secret image S.

2.2. (2, 2) RG-Based VCS

In 1987, Kafri and Keren [2] made a groundbreaking contribution to the field of visual cryptography by introducing the concept of (2, 2) RG-based VCS. Their innovative approach involved encrypting a binary image, denoted as S, and generating two shares, B_1 and B_2. These shares resemble seemingly random grids when viewed individually. However, when these shares were superimposed, they allowed the secret image S to be seamlessly reconstructed. What set their method apart was its simplicity and accessibility, as it required no specialized computational or cryptographic knowledge; it just needed to stack by OR operation ($\otimes$) like the human visual system (HVS). The Algorithm 1 they devised is depicted below:

Algorithm 1: (2, 2) RG-Based VCS [2]

Input:	A $W \times H$ secret binary image S.
Output:	Two shares B_1, B_2.
Step 1.	Generate the first share B_1 randomly selecting 0 or 1 for each pixel of B_1.
Step 2.	For each pixel $B_2[i, j]$ of share B_2, based on the pixel $S[i, j]$ of S and the pixel $B_1[i, j]$ of B_1, $B_2[i, j]$ is calculated by $B_2[i,j] = \begin{cases} B_1[i,j], & if\ S[i,j] = 0 \\ \overline{B_1[i,j]}, & if\ S[i,j] = 1 \end{cases}$.
Step 3.	Output (B_1, B_2).

The algorithm of (2, 2) RG-based VCS begins by inputting a binary secret image S with dimensions $M \times N$. The ultimate goal of this algorithm is to produce two random encrypted images, denoted as B_1 and B_2, which, when viewed individually reveal no discernible clues about the original secret content. The algorithm operates on a pixel-by-pixel basis and follows these key steps: first, it initiates the process by generating the content of B_1. In this step, each pixel is randomly assigned a value of 0 or 1, creating B_1 as a pattern that appears random and devoid of any identifiable structure; subsequently, we generate B_2, the content of which is determined based on the pixel values of S and B_1. When a pixel in S is 0, corresponding to a white region, B_2 mirrors the content of B_1. Conversely, when a pixel in S is 1, representing a black region, B_2 adopts the opposite value to that of B_1. Each pixel is processed according to the above steps, resulting in the generation of two encrypted images, B_1 and B_2.

These encrypted images form the basis of (2, 2) RG-based VCS, offering a secure and visually decipherable means of sharing confidential data without necessitating complex computations or cryptographic expertise. Kafri and Keren's pioneering work laid the foundation for subsequent advancements in visual cryptography, offering a user-friendly approach to protect and share sensitive information that remains relevant and influential to this day.

2.3. (k, n) RG-Based VCS

In 2009, Chen and Tsao conducted research on Kafri and Keren's (2, 2) RG-based VCS [2]. They expanded the original method, which could only encrypt into two encrypted images, to allow encryption into n encrypted images. These n encrypted images could then be directly superimposed using the OR operation ($\otimes$) to reconstruct the original secret, a scheme referred to as (n, n) RG-based VCS [4]. Two years later, in 2011, Chen and Tsao achieved another significant breakthrough by transforming (n, n) RG-based VCS into (k, n) RG-based VCS [5], introducing a threshold value k, where k must be less than or equal to n. This adaptation made (k, n) RG-based VCS more flexible. Unlike the previous method, which required the collection of all encrypted images for reconstruction, (k, n) RG-based VCS only necessitates gathering k encrypted images to restore the secret, offering users greater convenience. Below is the Algorithm 2 for (k, n) RG-based VCS.

Algorithm 2: (*k*, *n*) RG-Based VCS [5]

Input:	A $W \times H$ secret binary image S, a threshold value k and a total number n.
Output:	n shares $(B_1, B_2, \ldots, B_n)$.
Step 1.	For every pixel $S[i, j] = s$, iterate through Steps 2 to 4.
Step 2.	Randomly select from {0, 1} to generate n random pixels $b_1, b_2, \ldots, b_n$.
Step 3.	$b_k = s \oplus b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1}$.
Step 4.	Randomly rearrange the above n bits $b_1, b_2, \ldots, b_n$ into $B_1[i, j], B_2[i, j], \ldots, B_n[i, j]$.
Step 5.	Output $(B_1, B_2, \ldots, B_n)$.

In the algorithm for (k, n) RG-based VCS [5], we perform encryption on each pixel, employing the (2, 2) RG-based VCS method. Here is how it works: we take a single pixel s from the secret image and encrypt it into two pixels, b_1 and $\widetilde{b_2}$. We then treat $\widetilde{b_2}$ as the secret pixel and encrypt it to obtain b_2 and $\widetilde{b_3}$. We continue this process iteratively until $\widetilde{b_k}$ is generated, as illustrated in the flowchart in Figure 4. Subsequently, b_{k+1} to b_n are randomly assigned values of 0 or 1. At this point, we have obtained $b_1, b_2, \ldots, b_n$. From the flowchart in Figure 4, it is evident that b_1 to b_{k-1} and b_{k+1} to b_n are randomly generated, similar to Step 2 in the (k, n) RG-based VCS algorithm. Only b_k is computed and is the result of repeatedly applying the (2, 2) RG-based VCS process. Consequently, we can generalize b_k as the result of $s \oplus b_1 \oplus b_2 \ldots \oplus b_{k-1}$, just as in Step 3 of the (k, n) RG-based VCS algorithm. Once this process is completed for each pixel $[i, j]$ of the secret image S, these n pixels are randomly distributed across $B_1, B_2, \ldots, B_n$, finalizing the encryption in (k, n) RG-based VCS.

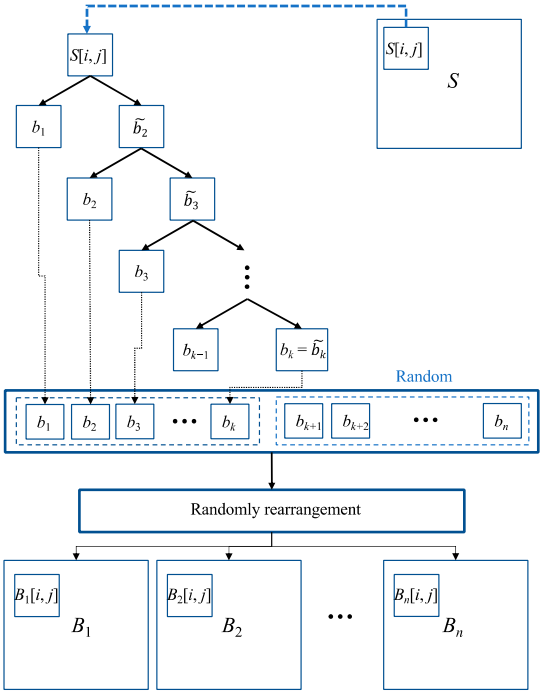


Figure 4. The flowchart of (k, n) RG-VCS.

The decryption process for (k, n) RG-based VCS is straightforward. To visually reconstruct the secret, one can directly stack k or more encrypted images together, achieving visual recovery through an OR operation ($\otimes$).

2.4. Region Incrementing Visual Cryptography (RIVCS)

Wang introduced a region incrementing visual cryptography scheme (RIVCS) in 2009 [9] that marked a significant advancement in the field. This innovative approach empowered the dealer to divide the content of the secret image S into multiple regions, referred to as secret levels. This paper represents the number of secret levels using the variable q. The (k, n)-threshold visual cryptography method further enhanced this concept,

enabling the segmentation of the secret image S into a maximum of $n - k + 1$ secret levels, denoted as Sl_1, Sl_2, ..., Sl_{n-k+1}. So, the range of q is from 1 to $n - k + 1$. These regions follow a hierarchical order where $R_i \subset R_{i+1}$, ensuring that each subsequent level contains increasingly significant secret information. This hierarchical structure allows for the flexible allocation of varying degrees of secrecy to different regions, ensuring that the most sensitive information remains well-protected while allowing for the sharing of less critical data.

The content of lower secret levels is nested within higher secret levels, as illustrated in Figure 5. Figure 5a,e represent the original secret images, Figure 5b–d depict the secret levels when $q = 3$ for Figure 5a, and Figure 5f–i represent the secret levels when $q = 4$ for Figure 5e. It is evident from the above explanation that when referring to the secret level Sl_q, the secret level Sl_q is equal to the secret image S.

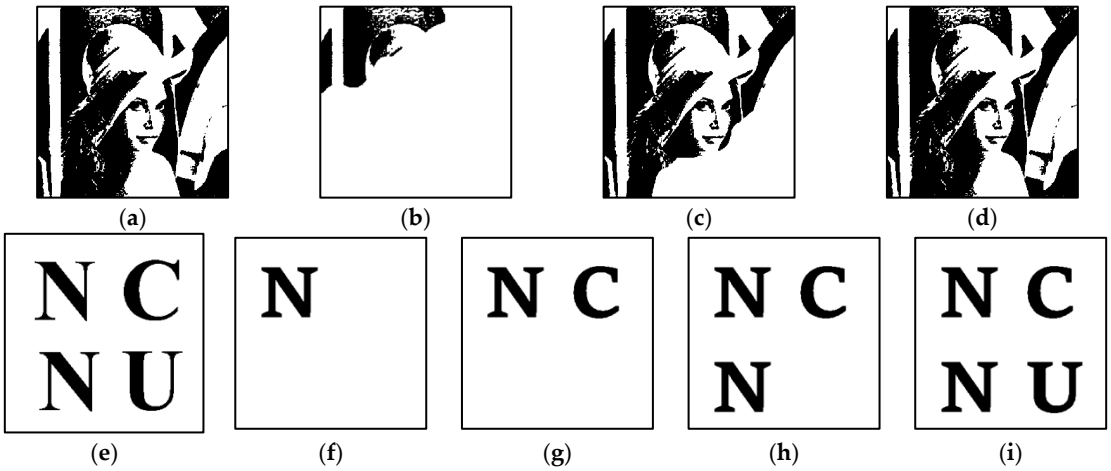


Figure 5. Example of different secret levels (Sl). (**a**) S_1. (**b**) Sl_1 for S_1. (**c**) Sl_2 for S_1. (**d**) Sl_3 for S_1. (**e**) S_2. (**f**) Sl_1 for S_2. (**g**) Sl_2 for S_2. (**h**) Sl_3 for S_2. (**i**) Sl_4 for S_2.

This level structure allows for a hierarchical organization of secret information, with each secret level contributing to the complete reconstruction of the original secret image. Figure 5 visually demonstrates how different secret levels at varying q values can be utilized to reveal different portions of the secret image, with the completeness of the reconstruction being dependent on the specific secret level chosen.

2.5. (k, n) 2D_VCS

In 2023, Lin and Juan introduced a new scheme known as (k, n) 2D_VCS [6]. This scheme presents a novel method for visual cryptography by incorporating random grids and introducing adaptable decryption choices for binary images, employing both OR and XOR operations. Unlike the work by Chen and Tsao [5] discussed in Section 2.3, which solely focused on stacking k shares, our approach broadens the scope to accommodate the stacking of more than k shares, thus offering a higher degree of flexibility. Even it can perfectly recover when stacking n encrypted images by XOR decryption. In addition, to ensure the complete restoration of the original secret image even when all shares are collected, we have incorporated two crucial steps, enhancing the algorithm's efficiency and practicality.

The schematic representation of (k, n) 2D_VCS is illustrated in Figure 2, and the subsequent paragraph outlines the algorithmic details. In the context of Algorithm 3, n denotes the total count of shares, k denotes the threshold value for decryption, b refers to the encrypted pixels before shuffling in the algorithm, and B represents the final output share formed by the shuffled pixels.

Algorithm 3: (*k*, *n*) 2D_VCS [6]

Input:	A $W \times H$ secret binary image S, a threshold value k and a total number n.
Output:	n shares $(B_1, B_2, \ldots, B_n)$.
Step 1.	For each pixel $S[i, j] = s$, repeat Steps 2–6.
Step 2.	Randomly select from {0, 1} to generate n random pixels $b_1, b_2, \ldots, b_n$.
Step 3.	$b_k = s \oplus b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1}$.
Step 4.	Select a number t randomly from $\{k + 1, \ldots, n\}$. Then, $b_t = s \oplus b_1 \oplus b_2 \ldots \oplus b_{t-1}$.
Step 5.	$b_n = s \oplus b_1 \oplus b_2 \oplus \ldots \oplus b_{n-1}$.
Step 6.	Randomly rearrange the above n bits $b_1, \ldots, b_n$ into $B_1[i, j]$, $B_2[i, j]$, $\ldots$, $B_n[i, j]$.
Step 7.	Output n share images $(B_1, B_2, \ldots, B_n)$

Algorithm 3 for (*k*, *n*) 2D_VCS differs only in Steps 4 and 5 from Chen and Tsao's (*k*, *n*) VCS. These two steps lead to important consequences. Step 4 of Algorithm 3 makes the reconstructed image clearer when stacking t ($k < t \leq n$) shares. Step 5 of Algorithm 3 makes it possible to fully recover the secret image after collecting all n shares. Because we do not change the first k shares constructed by (*k*, *n*) VCS, when Algorithm 3 stacks k shares, the quality of the reconstructed image is similar to that yielded by Chen and Tsao's approach. When stacking t shares, the probability of recovering the secret image is $\frac{1}{C_t^n \times (n-k)}$, which makes the proposed scheme more clearly recover the secret using the XOR operator for stacking $k < t \leq n$ shares. Therefore, compared to earlier studies, the method has a higher quality of the recovered image and improved ability to recover the secret image with both XOR and OR operations. Figure 6 depicts the suggested scheme's schematic.

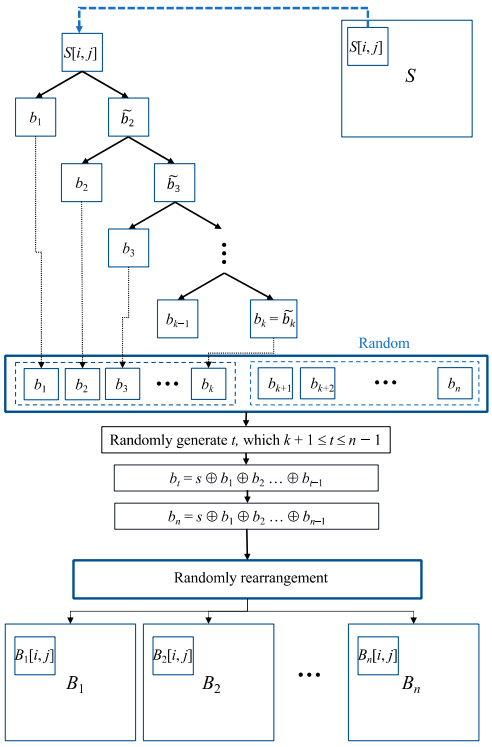


Figure 6. Schematic diagram of Algorithm 3 (*k*, *n*) 2D_VCS.

3. Methods

A new (*k*, *n*) RG-based RIVCS used with two decryption methods (OR and XOR) for binary images is presented in this section, called (*k*, *n*) 2D_RIVCS. The basic idea of this scheme is to encrypt each level of secrets on average, so that when a suitable number of shares are gathered, the original image of the corresponding level can be revealed. The detailed Algorithm 4 is illustrated as follows.

Algorithm 4: (*k*, *n*) 2D_RIVCS

Input:	A $W \times H$ secret image S, a number q, where $1 \leq q \leq n - k + 1$, and q secret level images $Sl_1, Sl_2, \ldots, Sl_q$.
Output:	n shares $(B_1, B_2, \ldots, B_n)$.
Step 1.	For every position $[i, j]$, iterate through Steps 2 to 4.
Step 2.	Randomly pick a numerical value d from $\{1, 2, \ldots, q\}$ with equal probability.
Step 3.	Encrypt secret pixel $Sld[i, j]$ by $(k - 1 + d, n)$ 2D_VCS Steps 2–5.
Step 4.	Randomly rearrange these n bits $b_1, b_2, \ldots, b_n$ into $B_1[i, j], B_2[i, j], \ldots, B_n[i, j]$.
Step 5.	Output n share images $(B_1, B_2, \ldots, B_n)$.

Within Algorithm 4, the (k, n) 2D_VCS encryption method is invoked regularly. To begin, a random value d is selected from the range $\{1, 2, \ldots, q\}$ for each pixel. Subsequently, the $(k + d - 1, n)$ 2D_VCS encryption method is applied to the corresponding Sl_d level. The schematic diagram of Algorithm 4 is shown in Figure 7. It will begin by providing an input of a secret image, denoted as S, and a set of q secret levels $\{Sl_1, Sl_2, \ldots, Sl_q\}$. When it comes to decryption, stacking k shares will allow you to recover Sl_1, while stacking $k + 1$ shares will recover Sl_2, and so forth. This implies that the secret image can be divided into a maximum of $n - k + 1$ levels, where $1 \leq q \leq n - k + 1$. The partitioning of secret levels is determined by the user based on the importance of different portions of the secret image. In addition, due to secret levels $Sl_1, Sl_2, \ldots, Sl_q$ satisfying $Sl_i \subset Sl_{i+1}$, when we recover the secret image, it can achieve a progressive effect.

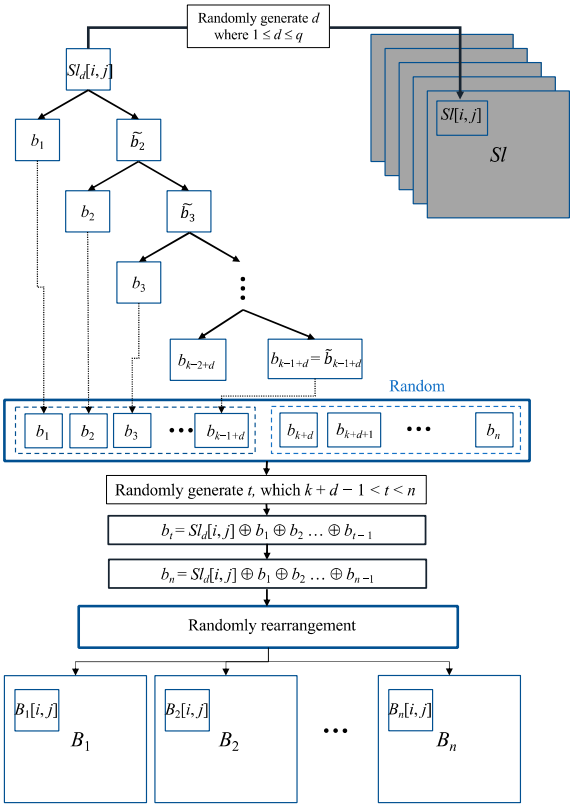


Figure 7. Schematic diagram of Algorithm 4 (k, n) 2D_RIVCS.

We have the option to combine a sufficient quantity of shares ($\geq k$) to restore the original secret image through the OR ($\otimes$) operator, similar to the human visual system (HVS) perception (normal human sight). Leveraging computers provides the capability to achieve more precise secret image reconstruction ($\oplus$). Nevertheless, if you gather fewer than k shares, it becomes infeasible to extract any information about the original secret image.

4. Theoretical Analyses

In this section, we offer a theoretical discussion of the security and visual recognizability of the proposed scheme.

First, this paper will establish the security of Algorithm 3, verifying that each individual encrypted image is secure and that collecting fewer than k encrypted images remains secure. Note that, to facilitate this proof, we will utilize b_i to represent the ith image within the algorithm before rearrangement (Step 4 of Algorithm 4), while B_i will denote the ith image after rearrangement, with $1 \leq i \leq n$. As used in the literature [5], we also define that $b_{1\oplus 2\oplus \ldots \oplus i} = b_1 \oplus b_2 \oplus \ldots \oplus b_i$ and $b_{1\otimes 2\otimes \ldots \otimes i} = b_1 \otimes b_2 \otimes \ldots \otimes b_i$, with $1 \leq i \leq n$.

Lemma 1. *For each grid pixel b_1, b_2, …, b_n generated based on the corresponding secret pixel of S in Algorithm 3,*

$$\text{Prob}(b_i = 0) = \text{Prob}(b_i = 1) = \frac{1}{2}. \tag{5}$$

Proof of Lemma 1. In Step 2 of Algorithm 3, we generate the n grid pixels b_1, b_2, …, b_n with a value of 0 or 1 with equal probability. So, after executing Step 2, we obtain

$$\text{Prob}(b_i = 0) = \text{Prob}(b_i = 1) = \frac{1}{2}, \text{for} \, 1 \leq i \leq n.$$

Then, b_k, b_t and b_n will be changed in Steps 3, 4 and 5 of Algorithm 3 according to conditions for each corresponding secret pixel s. For Step 3, because $\text{Prob}(b_{1\oplus 2\oplus \ldots \oplus k-1} = 0)$ = $\text{Prob}(b_{1\oplus 2\oplus \ldots \oplus k-1} = 1) = \frac{1}{2}$ and $b_k = b_1 \oplus b_2 \ldots \oplus b_{k-1} \oplus s$, we have

$$\text{Prob}(b_k = 0) = \text{Prob}(b_k = 1) = \frac{1}{2}$$

After executing Step 3. For the same reason, we know $\text{Prob}(b_t = 0) = \text{Prob}(b_t = 1) = \frac{1}{2}$, with $k \leq t \leq n$, after executing Steps 4 and 5. So, $\text{Prob}(b_i = 0) = \text{Prob}(b_i = 1) = \frac{1}{2}$ with $1 \leq i \leq n$, in the end. □

In the following, we define $b[S(0)]$ (and $b[S(1)]$) to denote a pixel b in the recovered image corresponding to a transparent (opaque) pixel in the image S.

Lemma 2. *For each grid pixel b_1, b_2, …, b_n generated based on the corresponding secret pixel of S in Algorithm 3, stacking any p grid pixel $\{b_{i1}, b_{i2}, \ldots, b_{ip}\}$ from $\{b_1, b_2, \ldots, b_n\}$, for which $p < k$, we have*

$$T[b_{i1} \otimes b_{i2} \otimes \ldots \otimes b_{ip}[S(0)]] = T[b_{i1} \otimes b_{i2} \otimes \ldots \otimes b_{ip}[S(1)]] \tag{6}$$

$$T[b_{i1} \oplus b_{i2} \oplus \ldots \oplus b_{ip}[S(0)]] = T[b_{i1} \oplus b_{i2} \oplus \ldots \oplus b_{ip}[S(1)]]. \tag{7}$$

Proof of Lemma 2. According to Lemma 1, $\text{Prob}(b_i = 0) = \frac{1}{2} = \text{Prob}(b_i = 1)$, where $1 \leq i \leq n$. However, based on the calculations of the algorithm and the properties of the XOR operation, when $b_1 \oplus b_2 \oplus \ldots \oplus b_k = s$ and $b_1 \oplus b_2 \oplus \ldots \oplus b_t = s$, it results in $b_{k+1} \oplus \ldots \oplus b_t = 0$ for some $k < t < n$. Similarly, when $b_1 \oplus b_2 \oplus \ldots \oplus b_t = s$ and $b_1 \oplus b_2 \oplus \ldots \oplus b_n = s$, it leads to $b_{t+1} \oplus \ldots \oplus b_n = 0$. Additionally, when $b_1 \oplus b_2 \oplus \ldots \oplus b_k = s$ and $b_1 \oplus b_2 \oplus \ldots \oplus b_n = s$, it causes $b_{k+1} \oplus \ldots \oplus b_n = 0$. This situation occurs regardless of whether the secret is white or black, allowing us to determine that no matter the stacking of pixels by OR operation or XOR operation, $T[b_{i1} \otimes b_{i2} \otimes \ldots \otimes b_{ip}[S(0)]] = T[b_{i1} \otimes b_{i2} \otimes \ldots \otimes b_{ip}[S(1)]]$, and $T[b_{i1} \oplus b_{i2} \oplus \ldots \oplus b_{ip}[S(0)]] = T[b_{i1} \oplus b_{i2} \oplus \ldots \oplus b_{ip}[S(1)]]$. □

Theorem 1. *Algorithm 3 (k, n) 2D_VCS is secure.*

Proof of Theorem 1. According to Lemma 1 and Lemma 2, we meet the security condition $T[B[S(0)]] = T[B[S(1)]]$ when stacking any p shares for $1 \leq p < k$. Algorithm 3 guarantees the security of each individual encrypted image and ensures the maintenance of security in the scenario where fewer than k encrypted images are collected. □

Next, we will show the visual recognizability of Algorithm 3, which offers two decryption methods. We will separately prove the visual recognizability based on whether OR or XOR decryption is applied, theoretically ensuring that the contrast is visually recognizable in both cases. First, let us begin with the XOR part.

Lemma 3. *Stacking k grid pixels $\{b_1, b_2, \ldots, b_k\}$, where $b_1, b_2, \ldots, b_k$ are generated after executing Step 2 and Step 3 of Algorithm 3, we have*

$$T[b_1 \oplus b_2 \oplus \ldots \oplus b_k[S(0)]] = 1, \tag{8}$$

$$T[b_1 \oplus b_2 \oplus \ldots \oplus b_k[S(1)]] = 0. \tag{9}$$

Proof of Lemma 3. Based on Step 3 of Algorithm 3, we know that $s = b_1 \oplus b_2 \oplus \ldots \oplus b_k$. So, $b_1 \oplus b_2 \oplus \ldots \oplus b_k = b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1} \oplus (b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1} \oplus s) = s$. Therefore, $T[b_1 \oplus b_2 \oplus \ldots \oplus b_k[S(0)]] = 1$, and $T[b_1 \oplus b_2 \oplus \ldots \oplus b_k[S(1)]] = 0$. □

Lemma 4. *Stacking p ($k < p \leq n$) grid pixels $\{b_1, b_2, \ldots, b_p\}$, where $b_1, b_2, \ldots, b_p$ are generated after executing Step 4 of Algorithm 3, we have*

$$T[b_1 \oplus b_2 \oplus \ \ldots \oplus b_p[S(0)]] = \frac{1}{n-k} + (1 - \ \frac{1}{n-k}) \cdot \frac{1}{2}, \tag{10}$$

$$T[b_1 \oplus b_2 \oplus \ \ldots \oplus b_p[S(1)]] = (1 - \ \frac{1}{n-k}) \cdot \frac{1}{2}. \tag{11}$$

Proof of Lemma 4. For (10) and (11), according to Step 4 of Algorithm 3, we know the probability is $\frac{1}{n-k}$ for choosing b_p from $\{b_{k+1}, \ldots, b_n\}$ to reset $b_1 \oplus b_2 \oplus \ldots \oplus b_p = s$. That is, the probability that the pixel is restored is $\frac{1}{n-k}$; otherwise, the pixel will be randomly turned transparent or opaque. □

Lemma 5. *Stacking p ($k < p \leq n$) grid pixels $\{b_1, b_2, \ldots, b_k, b_{n-p+k+1}, b_{n-p+k+2}, \ldots, b_n\}$, where $b_1, b_2, \ldots, b_n$ are generated after executing Step 5 of Algorithm 3, we have*

$$T[b_1 \oplus \ \ldots \oplus b_k \oplus b_{n-p+k+1} \oplus \ \ldots \ \oplus b_n[S(0)]] = \frac{1}{n-k} + (1 - \frac{1}{n-k}) \cdot \frac{1}{2}, \tag{12}$$

$$T[b_1 \oplus \ \ldots \oplus b_k \oplus b_{n-p+k+1} \oplus \ \ldots \ \oplus b_n[S(1)]] = (1 - \frac{1}{n-k}) \cdot \frac{1}{2}. \tag{13}$$

Proof of Lemma 5. In Step 3 of Algorithm 3, we are aware that $b_k = s \oplus b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1}$. Furthermore, in Step 4 and Step 5 we have $b_t = s \oplus b_1 \oplus b_2 \oplus \ldots \oplus b_{t-1}$ for some $k < t \leq n$, and $b_n = s \oplus b_1 \oplus b_2 \oplus \ldots \oplus b_{n-1}$. It implies $s = b_1 \oplus b_2 \oplus \ldots \oplus b_n = b_1 \oplus b_2 \oplus \ldots \oplus b_t \oplus b_{t+1} \oplus \ldots \oplus b_n = s \oplus b_{t+1} \oplus \ldots \oplus b_n$, so $b_{t+1} \oplus \ldots \oplus b_n = 0$ for some $k < t \leq n$ with probability $\frac{1}{n-k}$. When $t = n - p + k$, $b_1 \oplus b_2 \oplus \ldots \oplus b_k \oplus b_{n-p+k+1} \oplus \ldots \oplus b_n = s$. Hence, $T[b_1 \oplus b_2 \oplus \ldots b_{n-p+k+1} \oplus \ldots \oplus b_n\ [S(0)]] = \frac{1}{n-k} + (1 - \frac{1}{n-k}) \cdot \frac{1}{2}$, and $T[b_1 \oplus b_2 \oplus \ldots b_{n-p+k+1} \oplus \ldots \oplus b_n\ [S(0)]] = (1 - \frac{1}{n-k}) \cdot \frac{1}{2}$. □

Lemma 6. *Stacking p ($1 \leq p < n-k$) grid pixels $\{b_{k+1}, b_{k+2}, \ldots, b_{k+p}\}$, where $b_1, b_2, \ldots, b_n$ are generated after executing Step 5 of Algorithm 3, we have*

$$T[b_{k+1} \oplus b_{k+2} \oplus \ \ldots \oplus b_{k+p}[S(0)]] = T[b_{k+1} \oplus b_{k+2} \oplus \ \ldots \oplus b_{k+p}[S(1)]] = \frac{1}{n-k} + (1 - \frac{1}{n-k}) \cdot \frac{1}{2}. \tag{14}$$

Proof of Lemma 6. We know $b_1 \oplus b_2 \oplus \ldots \oplus b_k = s$, and when $t = k + p$ in Step 4 of Algorithm 3, $b_1 \oplus b_2 \oplus \ldots \oplus b_{k+p} = s$. So, $b_{k+1} \oplus b_{k+2} \ldots \oplus b_{k+p} = 0$ when $t = k + p$. □

Lemma 7. *When $p = n - k$, stacking p grid pixels $\{b_{k+1}, b_{k+2}, \ldots, b_{k+p} = b_n\}$, where $b_1, b_2, \ldots, b_n$ are generated after executing Step 5 of Algorithm 3, we have*

$$T[b_{k+1} \oplus b_{k+2} \oplus \ \ldots \oplus b_n[S(0)]] = T[b_{k+1} \oplus b_{k+2} \oplus \ \ldots \oplus b_n[S(1)]] = 1. \tag{15}$$

Proof of Lemma 7. Because $b_1 \oplus b_2 \oplus \ldots \oplus b_k = s$, $b_1 \oplus b_2 \oplus \ldots \oplus b_n = s$ (by Step 5 of Algorithm 3), and $p = n - k$, $b_{k+1} \oplus b_{k+2} \ldots \oplus b_n = 0$. □

Lemma 8. *Stacking p ($k < p < n$) grid pixels $\{b_{n-p+1}, b_{n-p+2}, \ldots, b_n\}$ where $b_1, b_2, \ldots, b_n$ are generated after executing Step 5 of Algorithm 3, we have*

$$T[b_{n-p+1} \oplus \ldots \oplus b_n[S(0)]] = T[b_{n-p+1} \oplus \ldots \oplus b_n\ [S(1)]] = \frac{1}{n-k} + (1 - \frac{1}{n-k}) \cdot \frac{1}{2}. \tag{16}$$

Proof of Lemma 8. We know $b_1 \oplus b_2 \oplus \ldots \oplus b_n = s$, and when $t = n - p$ in Step 4 of Algorithm 3, $b_1 \oplus b_2 \oplus \ldots \oplus b_{n-p} = s$. Hence, $b_{n-p+1} \oplus b_{\ n-p+2} \ldots \oplus b_n = 0$ when $t = n - p$. □

Lemma 9. *In Algorithm 3, stacking any k shares $\{B_{i1}, B_{i2}, \ldots, B_{ik}\}$ from $\{B_1, B_2, \ldots, B_n\}$, we have*

$$T[B_{i1} \oplus B_{i2} \oplus \ldots \oplus B_{ik}[S(0)]] = \begin{cases} \frac{1+(C_k^n - 1) \cdot \frac{1}{2}}{C_k^n}, & \text{if } k > n-k; \\ \frac{2+(C_k^n - 2) \cdot \frac{1}{2}}{C_k^n}, & \text{if } k = n-k; \\ \frac{1+\frac{2}{n-k}+(C_k^n - 1 - \frac{2}{n-k}) \cdot \frac{1}{2}}{C_k^n}, & \text{if } k < n-k \end{cases} \tag{17}$$

$$T[B_{i1} \oplus B_{i2} \oplus \ldots \oplus B_{ik}[S(1)]] \begin{cases} \frac{(C_k^n - 1) \cdot \frac{1}{2}}{C_k^n}, & \text{if } k > n-k; \\ \frac{(C_k^n - 2) \cdot \frac{1}{2}}{C_k^n}, & \text{if } k = n-k; \\ \frac{\frac{2}{n-k}+(C_k^n - 1 - \frac{2}{n-k}) \cdot \frac{1}{2}}{C_k^n}, & \text{if } k < n-k. \end{cases} \tag{18}$$

Proof of Lemma 9. For $k > n - k$, we use Lemma 3; for $k = n - k$, apply Lemmas 3 and 7; and for $k < n - k$, Lemmas 3, 6 and 8 are corresponding. □

Lemma 10. *In Algorithm 3, stacking any p ($k < p < n$) shares $\{B_{i1}, B_{i2}, \ldots, B_{ip}\}$ from $\{B_1, B_2, \ldots, B_n\}$ we have*

$$T[B_{i1} \oplus B_{i2} \oplus \ldots \oplus B_{\mathrm{ip}}[S(0)]] = \begin{cases} \frac{\frac{2}{n-k}+(C_p^n - \frac{2}{n-k}) \cdot \frac{1}{2}}{C_p^n}, & \text{if } p > n-k; \\ \frac{1+\frac{2}{n-k}+(C_p^n - 1 - \frac{2}{n-k}) \cdot \frac{1}{2}}{C_p^n}, & \text{if } p = n-k; \\ \frac{\frac{4}{n-k}+(C_p^n - \frac{4}{n-k}) \cdot \frac{1}{2}}{C_p^n}, & \text{if } p < n-k. \end{cases} \tag{19}$$

$$T[B_{i1} \oplus B_{i2} \oplus \ldots \oplus B_{ip}[S(1)]] \begin{cases} \frac{\left(C_p^n - \frac{2}{n-k}\right) \cdot \frac{1}{2}}{C_p^n}, \text{ if } p > n-k; \\ \frac{1+\left(C_p^n - 1 - \frac{2}{n-k}\right) \cdot \frac{1}{2}}{C_p^n}, \text{ if } p = n-k; \\ \frac{\frac{2}{n-k}+\left(C_p^n - \frac{4}{n-k}\right) \cdot \frac{1}{2}}{C_p^n}, \text{ if } p < n-k. \end{cases} \tag{20}$$

Proof of Lemma 10. When we stack any p shares, we will meet one of three difference cases: (1) $p > n - k$; (2) $p = n - k$; and (3) $p < n - k$. In the first case we use Lemma 4 and Lemma 5 to prove it. In the second case we meet Lemmas 4, 5 and 7. When the third case happens, we use Lemmas 4, 5, 6 and 8. □

Theorem 2. *Algorithm 3 (k, n) 2D_VCS has visual recognizability when using XOR decryption.*

Proof of Theorem 2. To achieve visual recognizability as defined in Definition 4, it is necessary that $T[B[S(0)]] - T[B[S(1)]]$ be greater than 0. According to Lemma 9 and Lemma 10, we can deduce the light transmission for restoring p encrypted images using XOR, where $k \leq p < n$. Therefore, we can calculate the following:

$$T[B[S(0)]] - T[B[S(1)]] = \begin{cases} \frac{1}{C_p^n}, \; if \; p = k \; and \; p \neq n-k; \\ \frac{2}{C_p^n}, \; if \; p = k \; and \; p = n-k; \\ \frac{2}{C_p^n \times (n-k)}, \; if \; k < p < n; \\ 1, \; if \; p = n. \end{cases} \tag{21}$$

Based on the formula above, it is evident that Algorithm 3 for (k, n) 2D_VCS meets the condition of being visually recognizable when collecting k shares or more. □

The contrast under various cases for (k, n) 2D_VCS for $2 \leq k \leq p \leq n \leq 6$ using XOR decryption can be deduced from Lemma 9 and Lemma 10, as presented in Table 3.

Table 3. The α of the restored image in (k, n) 2D_VCS by using XOR operation.

(*k*, *n*)	***p* = 2**	***p* = 3**	***p* = 4**	***p* = 5**	***p* = 6**
(2, 4)	0.1111	0.1818	1	-	-
(3, 4)	0	0.1818	1	-	-
(4, 4)	0	0	1	-	-
(2, 5)	0.0674	0.0440	0.0930	1	-
(3, 5)	0	0.0690	0.1429	1	-
(4, 5)	0	0	0.1429	1	-
(5, 5)	0	0	0	1	-
(2, 6)	0.0449	0.0167	0.0220	0.0571	1
(3, 6)	0	0.0333	0.0301	0.0769	1
(4, 6)	0	0	0.0455	0.1176	1
(5, 6)	0	0	0	0.1176	1
(6, 6)	0	0	0	0	1

With the XOR decryption ($\oplus$) part thoroughly addressed, this paper will proceed to discuss the visual recognizability when the OR decryption ($\otimes$) method is applied. Before proceeding with the proof, let us introduce a variable s, for which $1 \leq s \leq n - k$, representing the number of encrypted images (out of the total p) that are chosen for restoration and fall within the range of the $(k + 1)$th to the nth encrypted image. For instance, if we are restoring from 3 encrypted images and s is equal to 2, this implies that 1 encrypted image will be selected from $\{b_1, b_2, \ldots, b_k\}$, and 2 encrypted images will be chosen from $\{b_{k+1}, b_{k+2}, \ldots, b_n\}$. The following lemmas will be categorized based on the different cases of these s selected encrypted images.

Lemma 11. *Stacking k grid pixels {b_1, b_2, …, b_k}, where b_1, b_2, …, b_k are generated after executing Step 2 and Step 3 of Algorithm 3, we have*

$$T[b_1 \otimes b_2 \otimes \ \ldots \otimes b_k[S(0)]] = \frac{1}{2^{k-1}}, \tag{22}$$

$$T[b_1 \otimes b_2 \otimes \ldots \otimes b_k[S(1)]] = 0. \tag{23}$$

Proof of Lemma 11. According to Lemma 1, we know that b_1, b_2, …, b_{k-1} are created randomly and independently; thus,

$$T[b_1 \otimes b_2 \otimes \ \ldots \otimes b_{k-1}[S(0)]] = T[b_1 \otimes b_2 \otimes \ \ldots \otimes b_{k-1}[S(1)]] = \frac{1}{2^{k-1.}}$$

The corresponding pixel b_k is generated by $b_k = b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1} \oplus s$. So, we have $b_1 \otimes b_2 \otimes \ldots \otimes b_k = b_1 \otimes b_2 \otimes \ldots \otimes b_{k-1} \otimes (b_1 \oplus b_2 \oplus \ldots \oplus b_{k-1} \oplus s)$. Hence,

$$T[b_1 \otimes b_2 \otimes \ \ldots \otimes b_k[S(0)]] = \frac{1}{2^{k-1}},$$

$$T[b_1 \otimes b_2 \otimes \ldots \otimes b_k[S(1)]] = 0. \ \square$$

When stacking p encrypted images using the OR decryption method, for which $k \leq p < n$, the value of s will gradually increase from 1 to $\min\{n - k - 1, p\}$. These s images will be consecutive encrypted images starting either from b_{k+1} exclusive-or counting backward from b_n. For instance, in the case of (3, 6) 2D_VCS, when p is 3 and s is 1, the set of s images will be $\{b_4\}$ or $\{b_6\}$, and when s is 2, the set of s images will be $\{b_4, b_5\}$ or $\{b_5, b_6\}$. The maximum value for s is $\min\{n - k - 1, p\}$, which is $\min\{2, 3\}$, which is equal to 2. Based on the aforementioned conditions, when decrypting p encrypted images using the OR decryption method, the set of p images containing s consecutive encrypted images from $\{b_{k+1}, b_{k+2}, \ldots, b_n\}$ starting either from b_{k+1} exclusive-or counting backward from b_n, let the recovered image be referred to as $R_{p,1}$, and let the set of all possible $R_{p,1}$ be SR_1.

Lemma 12. *The light transmission of $R_{p,1}$ in SR_1 can be derived as follows:*

$$\begin{aligned}
&\Sigma_{R_{p,1} \in SR_1} T\left[R_{p,1}[S(0)]\right] \\
&\quad = \left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} \frac{s+n-k}{(n-k)\times 2^p} \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2 \right) \\
&\quad + \left(\frac{p+n-2k}{(n-k)\times 2^p} \times \left(C_k^{k+\max\{n-p-2,0\}} + 1 \right) \times 2 \right) \\
&\quad + \left(\sum_{s=1}^{p-k-1} \frac{s+n-k}{(n-k)\times 2^p} \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} + C_{p-s-k}^{\max\{n-k-2-s,0\}} \right) \times 2 \right)
\end{aligned} \tag{24}$$

$$\begin{aligned}
&\Sigma_{R_{p,1} \in SR_1} T\left[R_{p,1}[S(1)]\right] \\
&\quad = \left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} \frac{s+n-k}{(n-k)\times 2^p} \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2 \right) \\
&\quad + \left(\frac{p+n-2k}{(n-k)\times 2^p} \times \left(C_k^{k+\max\{n-p-2,0\}} - 1 \right) \times 2 \right) \\
&\quad + \left(\sum_{s=1}^{p-k-1} \frac{s+n-k}{(n-k)\times 2^p} \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}} \right) \times 2 \right)
\end{aligned} \tag{25}$$

Proof of Lemma 12. In Lemmas 5, 6 and 7, it can be observed that when selecting s consecutive pixels from $\{b_{k+1}, b_{k+2}, \ldots, b_n\}$ starting either from b_{k+1} or counting backward from b_n, there is a $\frac{1}{n-k}$ probability of obtaining white pixels. Next, based on the difference

between p and s, we can divide the possible combination of the remaining $p - s$ pixels of $R_{p,1}$ into three cases:

(1) $p - s < k$: In this case, for each of the remaining $p - s$ pixels, the probability of being black or white is $\frac{1}{2}$, resulting in $C_{p-s}^{k+\max\{n-k-s-2,\,0\}} \times 2$ combinations. Here, $k + \max\{n - k - s - 2,\, 0\}$ counts the number of pixels that can be selected from the initial k and the remaining $n - k$ pixels, except the s consecutive pixels and the 2 pixels that could potentially continue the sequence or be against the definition of the exclusive-or. We multiply by 2 because there are two cases: starting from b_{k+1} and counting backward from b_n. Hence, we can derive:

$$T\left[R_{p,1}[S(0)]\right] = \left(\frac{s}{n-k} \times \frac{1}{2^{p-1}} + \left(1 - \frac{s}{n-k}\right) \times \frac{1}{2^p}\right) \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2$$

This simplifies to:

$$T\left[R_{p,1}[S(0)]\right] = \frac{s+n-k}{(n-k)\times 2^p} \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2$$

(2) $p - s = k$: In this case, for the remaining $p - s$ pixels that exactly match the initial k pixels, there are $C_{p-s}^{k} \times 2 = 2$ combinations. Similarly, there are $(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s}^{k}) \times 2$ combinations for the remaining pixels that do not select the initial k pixels completely. Hence, we can derive:

$$\begin{aligned} T\left[R_{p,1}[S(0)]\right] &= \left(\tfrac{s}{n-k} \times \tfrac{1}{2^{p-2}} + \left(1 - \tfrac{s}{n-k}\right) \times \tfrac{1}{2^{p-1}}\right) \times C_{p-s}^{k} \times 2 \\ &+ \left(\tfrac{s}{n-k} \times \tfrac{1}{2^{p-1}} + \left(1 - \tfrac{s}{n-k}\right) \times \tfrac{1}{2^{p}}\right) \times \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s}^{k}\right) \times 2 \end{aligned}$$

This simplifies to:

$$T\left[R_{p,1}[S(0)]\right] = \frac{p+n-2k}{(n-k)\times 2^p} \times \left(C_{k}^{k+\max\{n-k-2-s,0\}} + 1\right) \times 2$$

(3) $p - s > k$: In this case, for the remaining $p - s$ pixels that are greater than k, there are $C_{p-s-k}^{\max\{n-k-2-s,0\}} \times 2$ combinations which choose all of the first k pixels completely. In addition, for the remaining cases, there are $(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}}) \times 2$ combinations. Hence, we can derive:

$$\begin{aligned} T\left[R_{p,1}[S(0)]\right] &= \left(\tfrac{s}{n-k} \times \tfrac{1}{2^{p-2}} + \left(1 - \tfrac{s}{n-k}\right) \times \tfrac{1}{2^{p-1}}\right) \times C_{p-s-k}^{\max\{n-k-2-s,0\}} \times 2 \\ &+ \left(\tfrac{s}{n-k} \times \tfrac{1}{2^{p-1}} + \left(1 - \tfrac{s}{n-k}\right) \times \tfrac{1}{2^{p}}\right) \times \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) \times 2 \end{aligned}$$

This simplifies to:

$$T\left[R_{p,1}[S(0)]\right] = \frac{s+n-k}{(n-k)\times 2^p} \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} + C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) \times 2$$

Combining the three cases with variations in s, we can obtain:

$$\begin{aligned} &\Sigma_{R_{p,1}\in SR_1} T\left[R_{p,1}[S(0)]\right] \\ &\quad = \left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} \tfrac{s+n-k}{(n-k)\times 2^p} \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2\right) \\ &\quad + \left(\tfrac{p+n-2k}{(n-k)\times 2^p} \times \left(C_{k}^{k+\max\{n-p-2,0\}} + 1\right) \times 2\right) \\ &\quad + \left(\sum_{s=1}^{p-k-1} \tfrac{s+n-k}{(n-k)\times 2^p} \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} + C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) \times 2\right) \end{aligned}$$

In the part of $T[R_{p,1}[S(1)]]$, combinations of light transmission including the first k shares will all be 0. Therefore, we can obtain:

(1) when $p - s < k$, we have $T\left[R_{p,1}[S(1)]\right] = \left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} \frac{s+n-k}{(n-k)\times 2^p} \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2\right)$

(2) when $p - s = k$, we have $T\left[R_{p,1}[S(1)]\right] = \left(\frac{p+n-2k}{(n-k)\times 2^p} \times \left(C_k^{k+\max\{n-k-2-s,0\}} - 1\right) \times 2\right)$

(3) when $p - s > k$, we have $T\left[R_{p,1}[S(1)]\right] = \left(\sum_{s=1}^{p-k-1} \frac{s+n-k}{(n-k)\times 2^p} \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) \times 2\right)$

Combining the three cases with variations in s, we can obtain:

$$\begin{aligned}
&\Sigma_{R_{p,1}\in SR_1} T\left[R_{p,1}[S(1)]\right] \\
&\quad = \left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} \tfrac{s+n-k}{(n-k)\times 2^p} \times C_{p-s}^{k+\max\{n-k-2-s,0\}} \times 2\right) \\
&\quad + \left(\tfrac{p+n-2k}{(n-k)\times 2^p} \times \left(C_k^{k+\max\{n-p-2,0\}} - 1\right) \times 2\right) \\
&\quad + \left(\sum_{s=1}^{p-k-1} \tfrac{s+n-k}{(n-k)\times 2^p} \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) \times 2\right)
\end{aligned}$$

According to the above proof, (24) and (25) are established. □

Based on these conditions, when stacking p encrypted images using the OR decryption method, for which $k \le p < n$, the value of s will gradually increase from 2 to $\min\{n - k - 1, p\}$. These s shares will be the sum of two sets of consecutive encrypted images, one counted from b_{k+1} forward and the other counted from b_n backward. Let the recovered image be referred to $R_{p,2}$, and let the set of all possible $R_{p,2}$ be SR_2. For example, in the (3, 7) 2D_VCS, when p is 3 and s is 2, these s shares will be $\{b_4, b_7\}$, and when s is 3, these s shares will be $\{b_4, b_5, b_7\}$ or $\{b_4, b_6, b_7\}$.

Lemma 13. *The light transmission of $R_{p,2}$ in SR_2 can be derived as follows:*

$$\begin{aligned}
&\Sigma_{R_{p,2}\in SR_2} T\left[R_{p,2}[S(0)]\right] \\
&\quad = \left(\left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} (s-1)\left(\tfrac{s+n-k}{(n-k)\times 2^p}\right) \times C_{p-s}^{k+\max\{n-k-2-s,0\}}\right)\right) \\
&\quad + \left(\left((p-k-1) \times \tfrac{(n+p-2k)}{(n-k)\times 2^p} \times \left(C_k^{k+\max\{n-p-2,0\}} + 1\right)\right)\right) \\
&\quad + \left(\sum_{s=2}^{p-k-1} 2(s-1)\left(\tfrac{s+n-k}{(n-k)\times 2^p}\right)\left(C_{p-s}^{k+\max\{n-k-2-s,0\}} + C_{p-s-k}^{\max\{n-k-2-s,0\}}\right)\right)
\end{aligned} \tag{26}$$

$$\begin{aligned}
&\Sigma_{R_{p,2}\in SR_2} T\left[R_{p,2}[S(1)]\right] \\
&\quad = \left(\sum_{s=p-k+1}^{\min\{n-k-1,p\}} (s-1)\left(\tfrac{s+n-k}{(n-k)\times 2^p}\right) \times C_{p-s}^{k+\max\{n-k-2-s,0\}}\right) \\
&\quad + \left(\left((p-k-1) \times \tfrac{(n+p-2k)}{(n-k)\times 2^p} \times \left(C_k^{k+\max\{n-p-2,0\}} - 1\right)\right)\right) \\
&\quad + \left(\sum_{s=2}^{p-k-1} 2(s-1)\left(\tfrac{s+n-k}{(n-k)\times 2^p}\right)\left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}}\right)\right)
\end{aligned} \tag{27}$$

Proof of Lemma 13. The proof process is similar to Lemma 12, as Lemmas 5, 6 and 7 have shown that there is a $\frac{1}{(n-k)}$ probability of white in $\{b_{k+1}, \ldots, b_n\}$ when counting from b_{k+1} or from b_n backward. Unlike Lemma 12, where one set of s shares is chosen from $\{b_{k+1}, \ldots, b_n\}$, Lemma 13 selects two sets that together add up to s shares. Similar to Lemma 12, it is divided into three cases, although the combination here does not need to be multiplied by 2. Instead, a variable i is used, and its range is from 1 to s. i represents the shares counted

from b_{k+1}, and the remaining $s - i$ shares are counted from b_n backward. Therefore, the three cases are as follows:

(1) when $p - s < k$, we have $T[R_{p,2}[S(0)]] = \left(\frac{s}{n-k} \times \frac{1}{2^{p-1}} + \left(1 - \frac{s}{n-k}\right) \times \frac{1}{2^p}\right) \times C_{p-s}^{k+\max\{n-k-2-s,0\}}$, which simplifies to $\left(\frac{s+n-k}{(n-k)\times 2^p}\right) \times C_{p-s}^{k+\max\{n-k-2-s,0\}}$.

(2) when $p - s = k$, we have $T[R_{p,2}[S(0)]] = \left(\frac{s}{n-k} \times \frac{1}{2^{p-2}} + \left(1 - \frac{s}{n-k}\right) \times \frac{1}{2^{p-1}}\right) \times C_{p-s}^{k} + \left(\frac{s}{n-k} \times \frac{1}{2^{p-1}} + \left(1 - \frac{s}{n-k}\right) \times \frac{1}{2^p}\right) \times \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s}^{k}\right)$, which simplifies to $\left(\frac{p+n-2k}{(n-k)\times 2^p}\right) \times \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} + 1\right)$.

(3) when $p - s > k$, we have $T[R_{p,2}[S(0)]] = \left(\frac{s}{n-k} \times \frac{1}{2^{p-2}} + \left(1 - \frac{s}{n-k}\right) \times \frac{1}{2^{p-1}}\right) \left(C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) + \left(\frac{s}{n-k} \times \frac{1}{2^{p-1}} + \left(1 - \frac{s}{n-k}\right) \times \frac{1}{2^p}\right) \times \left(C_{p-s}^{k+\max\{n-k-2-s,0\}} - C_{p-s-k}^{\max\{n-k-2-s,0\}}\right)$, which simplifies to $\left(\frac{s+n-k}{(n-k)\times 2^p}\right) \times 2\left(C_{p-s}^{k+\max\{n-k-2-s,0\}} + C_{p-s-k}^{\max\{n-k-2-s,0\}}\right)$.

Combining the three cases with variations in s, we can obtain (26). The proof for (27) $\Sigma_{R_{p,2} \in SR_2} T[R_{p,2}[S(1)]]$ is similar to the proof of (26); the results can be concluded. □

There is another possible combination when stacking p encrypted images using the OR decryption method for $n - k \leq p < n$. That is, s equals $n - k$, indicating that $\{b_{k+1}, \ldots, b_n\}$ are all included in the p shares. Let the recovered image be referred to $R_{p,3}$ and the set of all possible $R_{p,3}$ be SR_3.

Lemma 14. *The light transmission of $R_{p,3}$ in SR_3 can be derived as follows:*

$$\Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(0)]\right] = \Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(1)]\right] = \left(\frac{2n-2k-1}{(n-k)\times 2^{p-1}}\right) \times C_{p-n+k}^{k}. \quad (28)$$

Proof of Lemma 14. In the case of $R_{p,3}$, the proof process is similar to Lemmas 12 and 13. When s equals $n - k$, we have $b_{k+1} \oplus b_{k+2} \oplus \ldots \oplus b_n = 0$. In this case, there is a probability of $\frac{1}{n-k}$ that t equals n, and as a result, only b_n has been computed (not random) in b_{k+1} to b_n. The remaining probability of $1 - \frac{1}{n-k}$ may select one pixel b_t from b_{k+1} to b_{n-1}, such that both b_t and b_n have been computed. Furthermore, here $s = n - k$ and $p - s = p - n + k < k$ because $p < n$. Therefore, we only need to consider $p - s < k$, leading to the following:

$$\begin{aligned} \Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(0)]\right] &= \Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(1)]\right] \\ &= \left(\frac{1}{n-k} \times \frac{1}{2^{p-1}} + \left(1 - \frac{1}{n-k}\right) \times \frac{1}{2^{p-2}}\right) \times C_{p-s}^{k} \end{aligned}$$

Substituting $s = n - k$ yields and simplifies to:

$$\Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(0)]\right] = \Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(1)]\right] = \left(\frac{2n-2k-1}{(n-k)\times 2^{p-1}}\right) \times C_{p-n+k}^{k}. \square$$

When stacking p encrypted images using the OR decryption method, for which $k \leq p < n$, and it always includes the first k encrypted images but excludes the condition as $R_{p,1}$ or $R_{p,2}$, let the recovered image be referred to $R_{p,4}$. In addition, let the set of all possible $R_{p,4}$ be SR_4.

Lemma 15. *The light transmission of $R_{p,4}$ in SR_4 can be derived as follows:*

$$\Sigma_{R_{p,4}\in SR_4} T\left[R_{p,4}[S(0)]\right] = \begin{cases} \left(C_{p-k}^{n-k} - 2C_{p-1-k}^{\max\{n-k-3,0\}} - \sum_{s=2}^{p-k}(s+1)C_{p-s-k}^{\max\{n-k-2-s,0\}}\right) \times \frac{1}{2^{p-1}}, \ if\ k < p < n \\ \frac{1}{2^{p-1}}, \ if\ p = k \end{cases} \tag{29}$$

$$\Sigma_{R_{p,4}\in SR_4} T\left[R_{p,4}[S(1)]\right] = 0. \tag{30}$$

Proof of Lemma 15. There will be C_{p-k}^{n-k} combinations when taking p out of n, which are guaranteed to include the first k pixels. However, we need to subtract the combinations already calculated in $R_{p,1}$ and $R_{p,2}$. In the case of $R_{p,1}$ and when $p - s = k$ and $p - s > k$, there are $\sum_{s=1}^{p-k} 2C_{p-s-k}^{\max\{n-k-2-s,0\}}$ combinations that include the first k shares. In the case of $R_{p,2}$ and when $p - s = k$ and $p - s > k$, there are $\sum_{s=2}^{p-k} C_{p-s-k}^{\max\{n-k-2-s,0\}} \times (s+1)$ combinations that include the first k shares. So, the number of possible combinations for $R_{p,4}$ is

$$\begin{aligned} & C_{p-k}^{n-k} - \left(\sum_{s=1}^{p-k} 2C_{p-s-k}^{\max\{n-k-2-s,0\}} + \sum_{s=2}^{p-k} C_{p-s-k}^{\max\{n-k-2-s,0\}} \times (s-1)\right) \\ &= C_{p-k}^{n-k} - \left(\sum_{s=2}^{p-k} C_{p-s-k}^{\max\{n-k-2-s,0\}} \times (2+s-1) + 2C_{p-1-k}^{\max\{n-k-3,0\}}\right. \\ &= C_{p-k}^{n-k} - 2C_{p-1-k}^{\max\{n-k-3,0\}} - \sum_{s=2}^{p-k}(s+1)C_{p-s-k}^{\max\{n-k-2-s,0\}}. \end{aligned}$$

Next, based on Lemma 11, we obtain the result of stacking the first k images, and the results can be concluded. □

At last, when stacking p encrypted images using the OR decryption method, for which $k \le p < n$, these are still some combinations not included in $R_{p,1}$, $R_{p,2}$, $R_{p,3}$ or $R_{p,4}$. Let the recovered image be referred to as $R_{p,5}$ and the set of all possible $R_{p,5}$ be SR_5.

Lemma 16. *The light transmission of $R_{p,5}$ in SR_5 can be derived as follows:*

$$\Sigma_{R_{p,5}\in SR_5} T\left[R_{p,5}[S(0)]\right] = \Sigma_{R_{p,5}\in SR_5} T\left[R_{p,5}[S(1)]\right] = \beta \times \frac{1}{2^p}. \tag{31}$$

where $\beta = C_p^n - \sum_{s=1}^{\min\{n-k-1,p\}} 2C_{p-s}^{\max\{n-k-2-s,0\}} - \sum_{s=2}^{\min\{n-k-1,p\}} \sum_{i=1}^{s-1} C_{p-s}^{\max\{n-k-2-s,0\}} - C_{p-n+k}^{k} - \left(C_{p-k}^{n-k} - \sum_{s=2}^{p-k}(s+1)C_{p-s-k}^{\max\{n-k-2-s,0\}} - 2C_{p-1-k}^{\max\{n-k-3,0\}}\right)$.

Proof of Lemma 16. There will be C_p^n combinations when taking p out of n. However, we need to subtract the combinations already calculated in $R_{p,1}$, $R_{p,2}$, $R_{p,3}$ and $R_{p,4}$. So, we obtain:

$$\begin{aligned} & \beta = C_p^n - \sum_{s=1}^{\min\{n-k-1,p\}} 2C_{p-s}^{\max\{n-k-2-s,0\}} - \sum_{s=2}^{\min\{n-k-1,p\}} \sum_{i=1}^{s-1} C_{p-s}^{\max\{n-k-2-s,0\}} - \\ & C_{p-n+k}^{k} - \left(C_{p-k}^{n-k} - \sum_{s=2}^{p-k}(s+1)C_{p-s-k}^{\max\{n-k-2-s,0\}} - 2C_{p-1-k}^{\max\{n-k-3,0\}}\right) \end{aligned}$$

Then, based on Lemma 1, we obtain the result of stacking all of the p images in $R_{p,5}$, and the results can be concluded. □

Lemma 17. *In Algorithm 3, stacking all n encrypted images from $\{B_1, B_2, \ldots, B_n\}$, we have*

$$T[B_1 \oplus B_2 \oplus \ldots \oplus B_n\ [S(0)]] = \frac{2(n-k)-1}{(n-k)\times 2^{p-2}}, \tag{32}$$

$$T[B_1 \oplus B_2 \oplus \ldots \oplus B_n\ [S(1)]] = 0. \tag{33}$$

Proof of Lemma 17. When all the encrypted images are stacked together using the OR operator, it is known that the first k images will be equal to s, and all n images will also be equal to s. In the sixth step of Algorithm 3, the first t images will also be equal to s, but there is a $\frac{1}{n-k}$ probability that t will be equal to n. Therefore, we can derive the following:

$$T[B_1 \oplus B_2 \oplus \ldots \oplus B_n\ [S(0)]] = \left(\frac{1}{n-k} \times \frac{1}{2^{p-2}} + \left(1 - \frac{1}{n-k}\right) \times \frac{1}{2^{p-3}}\right),$$

$$T[B_1 \oplus B_2 \oplus \ldots \oplus B_n [S(1)]] = 0.$$

Simplifying, we obtain:

$$T[B_1 \oplus B_2 \oplus \ldots \oplus B_n [S(0)]] = \frac{2(n-k)-1}{(n-k) \times 2^{p-2}}$$

$$T[B_1 \oplus B_2 \oplus \ldots \oplus B_n\ [S(1)]] = 0. \square$$

Lemma 18. *In Algorithm 3, stacking any p ($k \leq p \leq n$) shares $\{B_{i1}, B_{i2}, \ldots, B_{ip}\}$ from $\{B_1, B_2, \ldots, B_n\}$, we have*

$$T[B_{i1} \oplus\ B_{i2} \oplus \ldots \oplus\ B_{ip}\ [S(0)]] = \begin{cases} \frac{\Sigma_{R_{p,1} \in SR_1} T\left[R_{p,1}[S(0)]\right] + \Sigma_{R_{p,2} \in SR_2} T\left[R_{p,2}[S(0)]\right] + \Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(0)]\right] + \Sigma_{R_{p,4} \in SR_4} T\left[R_{p,4}[S(0)]\right] + \Sigma_{R_{p,5} \in SR_5} T\left[R_{p,5}[S(0)]\right]}{C_p^n}, & \text{if } k \leq p < n; \\ \frac{2(n-k)-1}{(n-k) \times 2^{p-2}}, \quad if\ p = n \end{cases} \tag{34}$$

$$T[B_{i1} \oplus\ B_{i2} \oplus \ldots \oplus\ B_{ip}\ [S(1)]] = \begin{cases} \frac{\Sigma_{R_{p,1} \in SR_1} T\left[R_{p,1}[S(1)]\right] + \Sigma_{R_{p,2} \in SR_2} T\left[R_{p,2}[S(1)]\right] + \Sigma_{R_{p,3} \in SR_3} T\left[R_{p,3}[S(1)]\right] + \Sigma_{R_{p,4} \in SR_4} T\left[R_{p,4}[S(1)]\right] + \Sigma_{R_{p,5} \in SR_5} T\left[R_{p,5}[S(1)]\right]}{C_p^n}, & \text{if } k \leq p < n; \\ 0, \quad if\ p = n \end{cases} \tag{35}$$

Proof of Lemma 18. Based on Lemma 11 to Lemma 18, we have considered all possible cases when using OR decryption in (k, n) 2D_VCS. Therefore, to calculate the light transmission (T), we simply need to sum them up. $\square$

From Lemma 17 and Lemma 18, we can derive the contrast under different cases for OR decryption in (k, n) 2D_VCS, as illustrated in Table 4.

Table 4. The α of the restored image in (k, n) 2D_VCS by using the OR operation.

(k, n)	**p = 2**	**p = 3**	**p = 4**	**p = 5**	**p = 6**
(2, 4)	0.0606	0.1579	0.375	-	-
(3, 4)	0	0.0526	0.25	-	-
(4, 4)	0	0	0.125	-	-
(2, 5)	0.0382	0.0786	0.1154	0.2083	-
(3, 5)	0	0.0204	0.0674	0.1875	-
(4, 5)	0	0	0.0227	0.125	-
(5, 5)	0	0	0	0.0625	-
(2, 6)	0.0260	0.0492	0.0621	0.0704	0.1094
(3, 6)	0	0.0106	0.0277	0.0495	0.1042
(4, 6)	0	0	0.0074	0.0294	0.0938
(5, 6)	0	0	0	0.0099	0.0625
(6, 6)	0	0	0	0	0.0313

Theorem 3. *Algorithm 3 (*k, n*) 2D_VCS has visual recognizability when using OR decryption.*

Proof of Theorem 3. From Lemma 17, Lemma 18, it can be observed that when collecting k or more encrypted images and using the OR decryption method, the contrast (α) is always greater than 0. A contrast greater than 0 implies that theoretically, the information of the secret image S is visible, and the decryption is successful, meeting the conditions for visual recognizability. Therefore, Theorem 3 is proven. □

Based on the aforementioned Lemmas and Theorems, it becomes evident that Algorithm 3 for (k, n) 2D_VCS is both secure and functional. This conclusion reaffirms that the system not only preserves the confidentiality of the secret image but also offers a practical approach to decrypting it, making it a robust and reliable solution for visual secret sharing.

Now, we will demonstrate the theoretical security and usability of Algorithm 4 presented in this paper for (k, n) 2D_RIVCS.

Theorem 4. *Algorithm 4 for (k, n) 2D_RIVCS is secure and visually recognizable.*

Proof of Theorem 4. According to Algorithm 4, (k, n) 2D_RIVCS is primarily constructed by repetitively employing (k, n) 2D_VCS to encrypt different secret levels Sl_d. Therefore, to establish the security and visual recognizability of Algorithm 4, it is necessary to first demonstrate the security and visual recognizability of Algorithm 3. As shown in Theorems 1, 2 and 3, the security and visual recognizability of Algorithm 3 have already been established. Consequently, it can be inferred that Algorithm 4 is secure and visually recognizable as well. □

As the secret levels Sl of the secret image S are arbitrarily determined by the user, the proportion of these levels can vary with each division. Consequently, the fidelity of the reconstruction may differ when different numbers of encrypted images p are collected. Therefore, it is impractical to establish a fixed contrast between the reconstructed image and the secret image S for each case. According to Theorem 1, it is evident that obtaining a minimum of k or more encrypted images enables the recovery of clues related to the secret image S. With fewer than k images, only a chaotic and indecipherable image can be obtained.

Next, this paper analyzes the time complexity of Algorithm 4. In steps 1~4 of Algorithm 4, it can be observed that, for each pixel (i, j), the steps from 2~4 in Algorithm 4 are repeated. The size of the secret image is $W \times H$, indicating that these steps will be repeated WH times. Note that the time complexity of Steps 2–5 of Algorithm 3 is $O(k + t + n) = O(n)$. Therefore, the time complexity of Algorithm 4 is $O(nWH)$. It is equal to the size of the output, so Algorithm 4 is a linear-time algorithm. Due to the nature of the algorithm, hardware devices and processor efficiency do not significantly impact the execution speed. Users can employ any hardware, software and programming language to implement the algorithm presented in this paper. However, we will provide detailed information about the specific software used in the experiments (in the next section), namely MATLAB R2022b.

Algorithm 4 combines Algorithm 3 and a region incrementing visual cryptography scheme (RIVCS), thus introducing a novel application for Algorithm 3 while affording a broader spectrum of use cases. This innovative approach leverages the inherent strengths of both Algorithm 3 and RIVCS, paving the way for the fusion of these two cryptographic techniques. As a result, Algorithm 4 is capable of addressing diverse scenarios, showcasing its adaptability and extending the realm of possibilities for secure and visually recognizable image encryption and decryption.

5. Experimental Results

In this section, we will present the experimental results of Algorithm 4, the (k, n) 2D_RIVCS. The experiments were conducted using MATLAB R2022b, and all images in the dataset have dimensions of 300 × 300 pixels. This paper will now proceed to discuss and analyze the outcomes of our experiments with the 2D_RIVCS algorithm.

Figures 8 and 9 illustrate the experimental results of (2, 3) 2D_RIVCS. In both sets of experiments, the parameters such as k, n and S remain consistent, and the number of secret

levels is the same. The only variation lies in the user-defined partitioning of secret levels. It is evident from Figures8b,c and 9b,c that in Figure 8, the secret image S is uniformly divided, while Figure 9 depicts an arbitrary division, resulting in varying levels of information on each side. The encrypted results are presented in Figures 8d–f and 9d–f. Upon observation, it is evident that the encrypted image B does not reveal any information related to the secret image S.

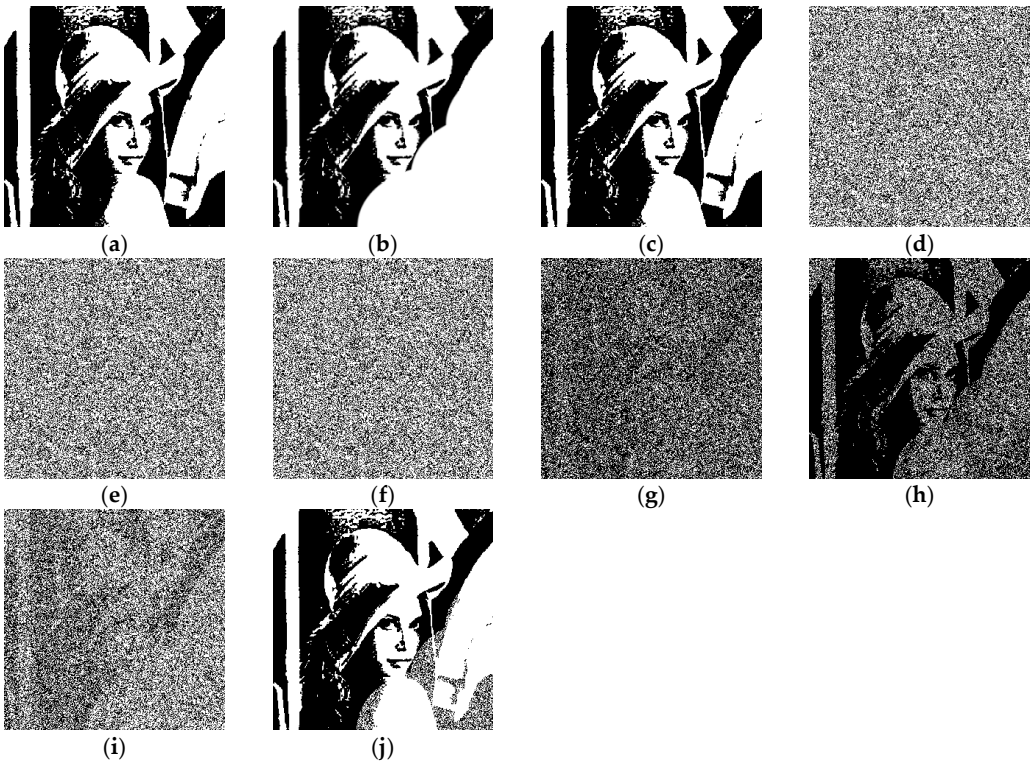


Figure 8. The experimental results of the proposed (2, 3) 2D_RIVCS with two secret levels. (**a**) S. (**b**) Sl_1. (**c**) Sl_2. (**d**) B_1. (**e**) B_2. (**f**) B_3. (**g**) $B_1 \otimes B_2$. (**h**) $B_1 \otimes B_2 \otimes B_3$. (**i**) $B_1 \oplus B_2$. (**j**) $B_1 \oplus B_2 \oplus B_3$.

Furthermore, Figures8g,h and 9g,h show the results of stacking two and three images using the OR decryption method. The contrasts between Figure 8g,h and Sl_1 are 0.06517 and 0.35696, respectively, and the contrasts with Sl_2 are 0.05313 and 0.32209, respectively. In Figure 9g,h, the contrasts with Sl_1 are 0.067229 and 0.31722, while with Sl_2, they are 0.012283 and 0.15064. In the XOR decryption part, it can be observed that there is a significant difference in the contrast values between Figures 8 and 9. This demonstrates that even under identical conditions of k, n, S and the number of secret levels, the achieved contrast upon reconstruction varies. For easier reference and comparison, this paper summarizes the contrast values mentioned above in Table 5.

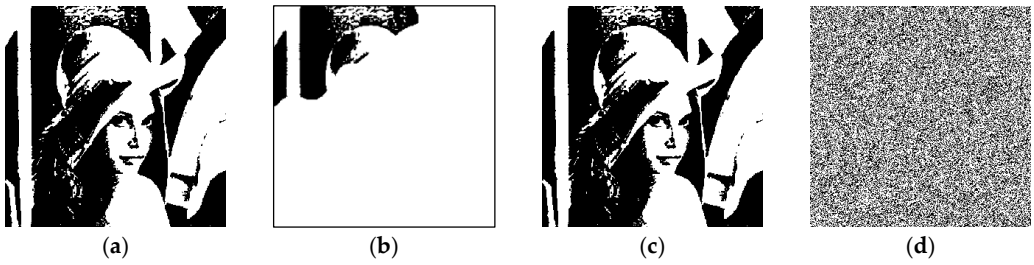


Figure 9. *Cont.*

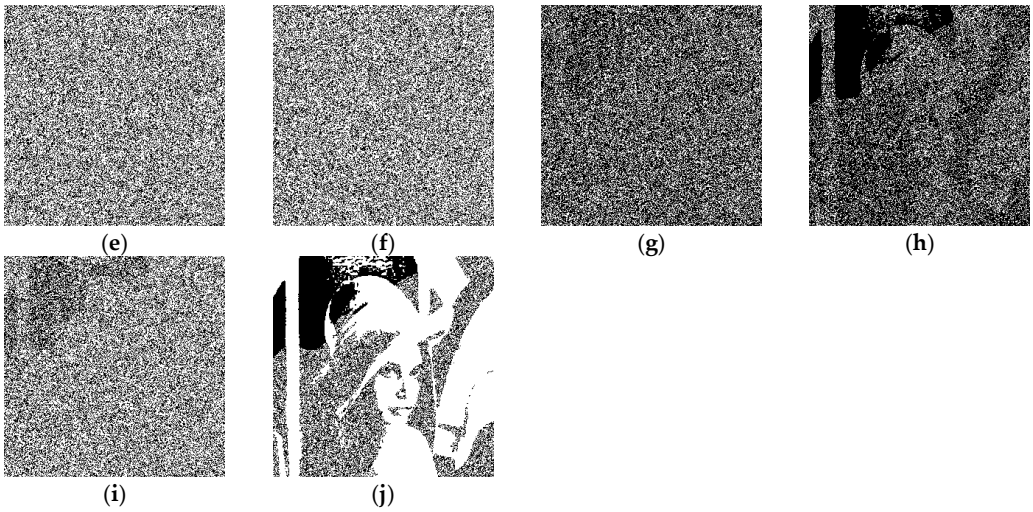


Figure 9. The experimental results of the proposed (2, 3) 2D_RIVCS with two secret levels. (**a**) S. (**b**) Sl_1. (**c**) Sl_2. (**d**) B_1. (**e**) B_2. (**f**) B_3. (**g**) $B_1 \otimes B_2$. (**h**) $B_1 \otimes B_2 \otimes B_3$. (**i**) $B_1 \oplus B_2$. (**j**) $B_1 \oplus B_2 \oplus B_3$.

Table 5. The contrast values of Figures 8 and 9.

		Figure 8		**Figure 9**	
	t	Sl_1	Sl_2	Sl_1	Sl_2
OR Decryption	2	0.06517	-	0.067229	-
	3	0.35696	0.32209	0.31722	0.15064
XOR Decryption	2	0.11629	-	0.11778	-
	3	0.92647	0.85572	0.76527	0.44327

Figure 10 showcases (2, 4) 2D_RIVCS with three secret levels, as depicted in Figure 10b–d. In this set of cases, the image dimensions are all set to 1000. The encrypted results in Figure 10e–h reveal that the entire image appears more grayscale. For this experiment, a textual image was used as the secret image, and the results display remarkably impressive outcomes. By partitioning secret levels, the encryptor can control how much text the decipherer can view when collecting a certain number of encrypted images. The contrast between the reconstructed image and each secret level is presented in Table 6.

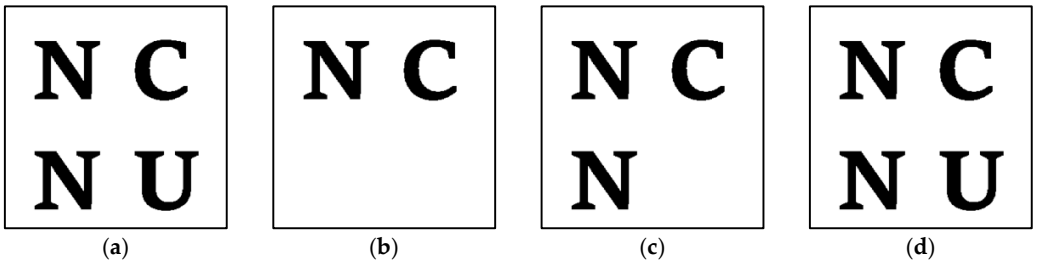


Figure 10. *Cont.*

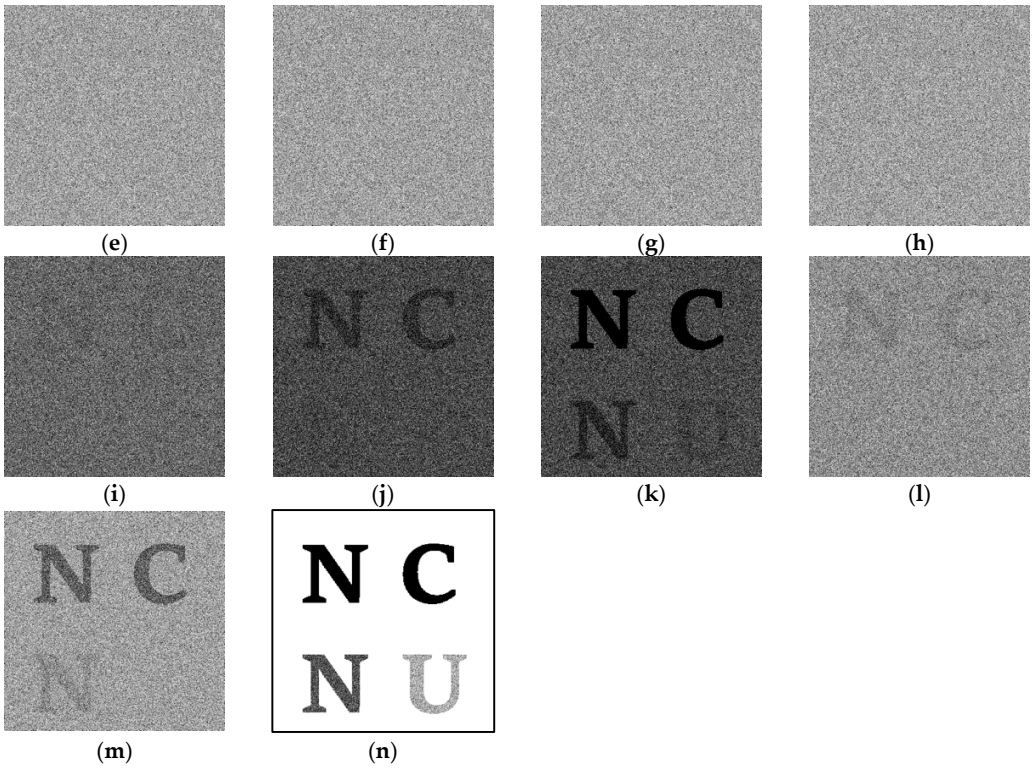


Figure 10. The experimental results of the proposed (2, 4) 2D_RIVCS with three secret levels. (**a**) S. (**b**) Sl_1. (**c**) Sl_2. (**d**) Sl_3. (**e**) B_1. (**f**) B_2. (**g**) B_3. (**h**) B_4. (**i**) $B_1 \otimes B_2$. (**j**) $B_1 \otimes B_2 \otimes B_3$. (**k**) $B_1 \otimes B_2 \otimes B_3 \otimes B_4$. (**l**) $B_1 \oplus B_2$. (**m**) $B_1 \oplus B_2 \oplus B_3$. (**n**) $B_1 \oplus B_2 \oplus B_3 \oplus B_4$.

Table 6. The contrast of Figure 10.

OR Decryption				XOR Decryption			
t	Sl_1	Sl_2	Sl_3	t	Sl_1	Sl_2	Sl_3
2	0.019801	-	-	2	0.03535	-	-
3	0.087095	0.059633	-	3	0.17898	0.13041	-
4	0.28509	0.21552	0.16656	4	0.96108	0.77239	0.5973

In summary, the experimental results presented in Figures 8–10 highlight the versatility of (k, n) 2D_RIVCS with its ability to handle multiple secret levels. The utilization of textual images as secret data yields impressive outcomes, allowing the encryptor to control the visibility of text based on secret level partitioning. Moreover, the contrast values indicate that XOR decryption outperforms OR decryption in these experimental results.

6. Comparison

In this segment, we will conduct a comparative analysis of our proposed approach in relation to other, previously published research. We proceed with a functional comparison. We conducted a comparative analysis among our proposed scheme and several other related RIVCS schemes: Wang [9], Wang et al. [10], Shyu and Jiang [11], Yang et al. [12], Kumar and Sharma [13] and Yang et al. [14], as shown in Table 7. Our comparison focused on various aspects such as encryption method, expansion, threshold, maximum secret level and decryption method. In can be seen that our scheme employs the use of random grids for encryption, which effectively avoids pixel expansion. Furthermore, our decryption process uses two decryption methods. Judging from the experimental results, when the proposed scheme is restored by OR operation, the results will be better when t is closer to n. Especially when $t = n$, the results are almost better than those in the past literature.

When the XOR operation is used to restore the image, the contrast is much larger than the result of the OR operation, and it is even close to perfect recovery (the contrast α is equal to 1). Therefore, we can know that the proposed scheme performs very well in restoring images. Overall, the proposed novel (k, n) 2D_RIVCS is currently the only known RG-based (k, n) threshold region incremental visual cryptography method with XOR and OR decryption capabilities, with better contrast values in most cases. It demonstrates more practical performance compared to other known RIVCS methods.

Table 7. Comparison.

Property	Scheme						
	[9]	[10]	[11]	[12]	[13]	[14]	**Ours**
Encryption	Basis matrices	Random grids	Basis matrices	Basis matrices	Random grids	Basis matrices	Random grids
Expansion	Yes	No	Yes	Yes	No	Yes	No
Threshold	$(2, n)$	$(2, 3)$	$(2, n)$	(k, n)	(k, n)	(k, n)	(k, n)
Max secret level	$n-1$	$n-1$	$n-1$	$n-k+1$	$n-k+1$	$n-k+1$	$n-k+1$
Decryption	OR	OR	OR	OR	OR	OR	OR and XOR

7. Conclusions

This paper introduces 2D_RIVCS, a novel RG-based region incrementing visual cryptography scheme with both OR and XOR decryption capabilities. This approach offers two distinct decryption methods to cater to various user scenarios. In situations where users lack specialized equipment, OR decryption can be employed, requiring no additional machinery; it simply involves stacking the encrypted images for restoration. Alternatively, users equipped with computers can opt for XOR decryption, offering a more advanced method that achieves perfect restoration. The versatility of this proposed method extends its application to a wide range of potential use cases.

Note that if we change the range of the number t from $\{k+1, \ldots, n\}$ to $\{k+1, \ldots, n-1\}$ in Step 4 of Algorithm 3, then Algorithms 3 and 4 are still correct and the performance will be improved.

At present, this method is limited to binary images, significantly constraining its applicability. Therefore, the next phase of our research will focus on a more comprehensive investigation of grayscale and color images, with the aspiration that this approach can extend its utility to a diverse range of visual content. The expansion into color imagery represents a significant stride towards broader applicability, making this method more versatile and accommodating a wider array of practical use cases.

In addition, another avenue for future work is to enhance the camouflage capability of this method to ensure that the encrypted image does not appear as arbitrary noise but displays the camouflaged image. Aiming to embed concealed images within encrypted data elevates the security and covert potential of this approach, bolstering its utility for a multitude of applications across domains.

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

Funding: This work was partially supported by the National Science and Technology Council, R.O.C., under grants MOST 111-2115-M-260-001- and NSTC 112-2115-M-260 -001 -MY2.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to thank the referees for their careful reading of the manuscript and fruitful comments.

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

References

1. Naor, M.; Shamir, A. Visual Cryptography. In *Workshop on the Theory and Application of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 1994; pp. 1–12.
2. Kafri, O.; Keren, E. Encryption of pictures and shapes by random grids. *Opt. Lett.* **1987**, *12*, 377–379. [CrossRef] [PubMed]
3. Shyu, S.J. Image encryption by multiple random grids. *Pattern Recognit.* **2009**, *42*, 1582–1596. [CrossRef]
4. Chen, T.H.; Tsao, K.H. Visual secret sharing by random grids revisited. *Pattern Recognit.* **2009**, *42*, 2203–2217. [CrossRef]
5. Chen, T.H.; Tsao, K.H. Threshold visual secret sharing by random grids. *J. Syst. Softw.* **2011**, *84*, 1197–1208. [CrossRef]
6. Lin, Y.R.; Juan, J.S.T. RG-based (*k*, *n*) threshold visual cryptography with abilities of OR and XOR decryption. *Eng. Proc.* **2023**, *55*, 65.
7. Yan, X.; Wang, S.; EI-Latif, A.A.A.; Niu, X. Visual secret sharing based on random grids with abilities OR and XOR lossless recovery. *Multimed. Tools Appl.* **2015**, *74*, 3231–3252. [CrossRef]
8. 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]
9. Wang, R.Z. Region incrementing visual cryptography. *IEEE Signal Process. Lett.* **2009**, *16*, 659–662. [CrossRef]
10. Wang, R.Z.; Lan, Y.C.; Lee, Y.K.; Huang, S.Y.; Shyu, S.J.; Chia, T.L. Incrementing visual cryptography using random grids. *Opt. Commun.* **2010**, *283*, 4242–4249. [CrossRef]
11. Shyu, S.J.; Jiang, H.W. Efficient construction for region incrementing visual cryptography. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 769–777. [CrossRef]
12. Yang, C.N.; Shih, H.W.; Wu, C.C.; Harn, L. *k* Out of *n* Region Incrementing Scheme in Visual Cryptography. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 799–810. [CrossRef]
13. Kumar, S.; Rajendra Kumar, S. Random-grid based region incrementing visual secret sharing. *Fundam. Informaticae* **2015**, *137*, 369–386. [CrossRef]
14. Yang, C.N.; Wu, C.C.; Lin, Y.C. *k* out of *n* region-based progressive visual cryptography. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *29*, 252–262. [CrossRef]
15. Huang, S.Y.; Lo, A.H.; Juan, J.S.T. XOR-Based Meaningful (*n*, *n*) Visual Multi-Secrets Sharing Schemes. *Appl. Sci.* **2022**, *12*, 10368. [CrossRef]
16. Chen, Y.H.; Juan, J.S.T. XOR-Based (*n*, *n*) Visual Cryptography Schemes for Grayscale or Color Images with Meaningful Shares. *Appl. Sci.* **2022**, *12*, 10096. [CrossRef]
17. Panchbhai, V.V.; Varade, S.W. Enhanced block based progressive visual secret sharing scheme for grayscale and color image. *Multimed. Tools Appl.* **2023**. [CrossRef]
18. Zhang, L.; Zhang, J.; Sun, J.; Chen, Q. A global progressive image secret sharing scheme under multi-group joint management. *Math. Biosci. Eng.* **2024**, *21*, 1286–1304. [CrossRef]
19. Shyu, S.J. Image encryption by random grids. *Pattern Recognit.* **2007**, *40*, 1014–1031. [CrossRef]
20. Guo, T.; Liu, F.; Wu, C. Threshold visual secret sharing by random grids with improved contrast. *J. Syst. Softw.* **2013**, *86*, 2094–2109. [CrossRef]