symmetry

MDPI

Article

Adaptive Partial Image Secret Sharing

Xuehu Yan *, Lei Sun, Yuliang Lu and Guozheng Yang

National University of Defense Technology, Hefei 230037, China; sun19960119@163.com (L.S.); publicLuYL@126.com (Y.L.); publicYangGZ@126.com (G.Y.)

* Correspondence: publictiger@126.com; Tel.:+86-055866927640

Received: 21 February 2020; Accepted: 14 April 2020; Published: 2 May 2020

Abstract: In contrast to encrypting the full secret image in classic image secret sharing (ISS), partial image secret sharing (PISS) only encrypts part of the secret image due to the situation that, in general, only part of the secret image is sensitive or secretive. However, the target part needs to be selected manually in traditional PISS, which is human-exhausted and not suitable for batch processing. In this paper, we introduce an adaptive PISS (APISS) scheme based on salience detection, linear congruence, and image inpainting. First, the salient part is automatically and adaptively detected as the secret target part. Then, the target part is encrypted into n meaningful shares by using linear congruence in the processing of inpainting the target part. The target part is decrypted progressively by only addition operation when more shares are collected. It is losslessly decrypted when all the n shares are collected. Experiments are performed to verify the efficiency of the introduced scheme.

Keywords: symmetry; image secret sharing; partial image secret sharing; image inpainting; linear congruence; adaptive

1. Introduction

An image secret sharing (ISS) scheme encrypts a secret image into n shares and distributes them to n related participants. The secret image is decrypted when collecting any k or more shares. Thus, the ISS technique has been applied to distributed storage in the cloud, block chain, digital watermarking, and access control [1–4].

Now the widely studied principles of ISS techniques include visual secret sharing (VSS), also known as, visual cryptography (VC) [4,5] and polynomial [6–12].

VSS for a (k, n)-threshold [11,13–17], usually outputs n shares printed onto transparent films, which are then also distributed to n participants. The advantage of VSS is that the secret can be recognized by the naked human eye when simply superposing any k or more shares. However, the traditional VSS approaches often have the disadvantages of large pixel expansion and poor image quality.

In order to decrypt a secret image with high resolution, the (k, n) threshold secret sharing scheme based on polynomial was proposed by Shamir [6]. By constructing a random $(k-1)$-degree polynomial, when any k or more shares are obtained, the high-resolution secret image can be decrypted by Lagrange interpolation. Some other enhanced polynomial-based ISS schemes [18–21] with admirable properties have been developed inspired by Shamir's work. The significance of ISS based on polynomial is that the decrypted secret image has no pixel expansion and is of high quality. However, this kind of technique requires large decryption computation, i.e., Lagrange interpolation, and is time-exhausted.

Linear congruence (LC)-based ISS [22] can decrypt the secret image by using only addition operation. It balances decryption complexity and image quality.

However, the above-mentioned ISS schemes encrypt the whole secret image, ignoring the potential situation that in general only part of the secret image is sensitive or secretive. Recently, a partial ISS

(PISS) scheme for a (k, n)-threshold was proposed by Yan et al. [23] based on LC and image inpainting, which only encrypts part of the secret image. However, the target part requires manual selection in their PISS, which is labor-intensive and not conducive to batch processing.

The purpose of this paper is to introduce a PISS scheme that can select the target part automatically and decrypt the secret image progressively by outputting meaningful shares. The key challenge of the work is how to perform the image processing operations and simultaneously realize the ISS procedure, because the encrypting method of an ISS principle in general entails the use of a mathematical function, which is dramatically sensitive to any slight change in the ISS output.

Figure 1 further illustrates the motivation of this work.

$$\text{Secret image} \xrightarrow[\text{ISS and image inpainting}]{\text{Secret part adaptive detection}} \begin{cases} \text{Meaningful share 1} \\ \text{Meaningful share 2} \\ \vdots \\ \text{Meaningful share } n \end{cases} \rightarrow \text{Decrypt secret image}$$

Figure 1. The motivation of this paper.

In this paper, we introduce an adaptive PISS (APISS) scheme based on salience detection, LC, and image inpainting. First, the salient part is automatically and adaptively detected as the secret target part. Then, the target part is encrypted into n meaningful shares by using LC in the processing of inpainting the target part, where each output share looks like a natural image. The target part is decrypted progressively by only addition operation when more shares are collected. It is losslessly decrypted when all the n shares are collected. The validity of the scheme is verified by experiments.

The rest of the paper is organized as follows. Section 2 is devoted to some basic preliminaries for the introduced scheme. In Section 3, the introduced scheme is described in detail. Section 4 gives experiments. Finally, Section 5 concludes this paper.

2. Preliminaries

In this section, we will present some basic preliminaries for the introduced scheme. An original secret image S is encrypted into n shares, and the decrypted secret image S' is recovered when any t ($k \leq t \leq n, t \in \mathrm{Z}^+$) shares are collected.

2.1. Salience Detection

Saliency detection is used to discover the saliency on the target image. It follows the law of the visually salient stimuli in the image, which is thus useful for distinguishing salient object. In general, the salient object (part) is the important part, thus the salient part is severed as the secret target part in this paper. Saliency detection method in [24] will be adopted in this paper, which mainly includes the following steps to detect the saliency on any single image.

1. Cluster the image into K_1 clusters, such as $K_1 = 6$.
2. For each cluster, compute the contrast cue and spatial cue and combine the two saliency cues by multiplication. Herein, the contrast cue can represent the visual feature uniqueness and the contrast operator can simulate the human visual receptive fields; spatial cue is considered because of the "central bias rule" in single image saliency detection, also known as, the regions near the image center draw more attention than the other regions in human visual system.
3. For each pixel, obtain the final saliency map by summing the joint saliency over all clusters.

Please refer to the work in [24] for detail. By using the saliency detection method, the secret target part of the input secret image can be detected automatically and adaptively. Figure 2 presents an example, where Figure 2c shows the automatically selected target part of Figure 2a by applying Otsu's threshold operation [25] to Figure 2b.

(**a**) S (**b**) Image saliency (**c**) C

Figure 2. Experimental result of the image saliency detection. (**a**) The secret image S; (**b**) image saliency; (**c**) automatically selected target part by Otsu's threshold operation.

2.2. Image Inpainting

According to the image inpainting techniques proposed in the literature, the widely used approach of Criminisi et al. [26,27] is adopted in our scheme. We will describe it in detail. As in Figure 3c, a region Ω means the secret target part with arbitrary size and shape, part Φ denotes the untouched part, and $\partial\Omega$ represents the edge of the two parts. The design significance of this method lies in the selection of patch priority in the process of region filling. The patch with the highest priority will be preferentially filled. After each filling, the priorities will be updated until all the whole target is inpainted completely in the same manner. The chief inpainting process is as follows.

1. Select a target part Ω to be inpainted, and $\Phi = S - \Omega$, where S indicates the whole image.
2. Determine the size of the template window by using the image texture feature, denoted by Ψ_p , where any $p \in \partial\Omega$ denotes the center of the template window. In addition, the size of the window should be larger than the largest texture element.
3. Calculate patch priorities by using Equation (1), i.e., the product of the data term and the confidence term.

$$W(p) = C(p)D(p) \tag{1}$$

where $C(p)$ and $D(p)$ mean the data term and the confidence term, respectively, defined as

$$D(p) = \frac{\left|\nabla S_p^{\perp} \cdot n_p\right|}{a} \tag{2}$$

$$C(p) = \frac{\sum_{q \in \psi_p \cap \bar{\Omega}} C(q)}{\left|\psi_p\right|} \tag{3}$$

where $\left|\psi_p\right|$ means the area of ψ_p and a denotes a normalization factor. At point p, $\nabla S_p^{\perp}$ and n_p ,respectively, denote the isophote direction and the normal vector direction.

The data term represents the difference between the direction of isophote and the direction of the normal vector. In the template window the confidence term is used to measure the amount of reliable information. In other words, if the difference between the normal vector direction and the isophote direction is smaller and the information contained in the template window is greater, the priority of patch will be higher.

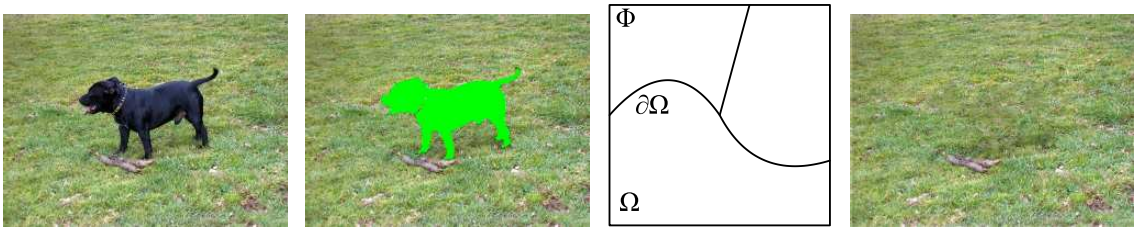
4. Find $\hat{p}$ according to Equation (4), and the block $\psi_{\hat{q}} \in \Phi$ with the highest matching in the source image with the template window as specified in Equation (5), where the sum of squared differences (SSD) is utilized as the evaluation standard. Finally, the highest matching block replaces the patch of the current window.

$$\hat{p} = \arg \max_{p \in \partial\Omega} W(p) \tag{4}$$

$$\hat{q} = \arg\min_{q \in \Phi} d(\psi_{\hat{p}}, \psi_q) \tag{5}$$

5. For any $q \in \psi_{\hat{p}} \cap \Omega$, after each filling process, renew the confidence terms $C(q) = C(\hat{p})$.
6. Repeat the above steps 3–5 until the image is inpainted completely.

As an example, Figure 3 shows the result of a inpainted image using the approach of Criminisi et al. Another visually plausible image is obtained, and therefore image inpainting will be adopted in the introduced scheme to obtain meaningful shares.



(**a**) *S* (**b**) *C* (**c**) Notations (**d**) *DC*

Figure 3. An example of the inpainted image obtained using the approach of Criminisi et al. (**a**) The secret image *S*; (**b**) the same input cover image, denoted by *C*, through selecting and removing the secret target part with green color from *S*; (**c**) the general notations; (**d**) directly inpainted result.

2.3. Linear Congruence-Based Iss

Equations (6) and (7) are the basic equations for LC-based ISS, by which the (k, k) threshold ISS can be easily achieved, where P is a number larger than the biggest pixel value, such as $P = 256$ for grayscale image, and s and x_i represent the secret pixel and the i-th shared pixel, respectively. According to Equation (6), a one-to-many mapping between s and x_i is established; therefore, the secret can be losslessly decrypted when collecting all the k shared values. However, the method is secure, as there is no such mapping with less than k shared value. In such a way, Equation (6) guarantees the security and feasibility of precise decryption, which is thus adopted in the introduced scheme. In addition, Equation (7) ensures that there are no duplicate pixel values in the first k shared pixel values.

According to Figure 4, LC-based ISS can be easily extended to support (k, n) threshold [22].

$$(x_1 + x_2 + \cdots + x_k) \; mod \; P = s \tag{6}$$

$$x_i \neq x_j, when \; i \neq j. \tag{7}$$

In the decryption phase, the remaining shared values $x_{i_1}, x_{i_2}, \cdots, x_{i_t}$ are used to decrypt the secret value s' by using Equation (8) after removing the duplicate shared values.

$$s' = (x_{i_1} + x_{i_2} + \cdots + x_{i_t}) \; mod \; P \tag{8}$$

Figure 5 shows an example of applying the LC-based ISS for $(3, 3)$ threshold to encrypt the secret target directly. Here, $S'_{1,2}$ denotes the secret image decrypted with SC_1 and SC_2. For the sake of saving pages, only the decrypted results by the first tth shares are given. Only when the corresponding target part of each share is collected can the secret target part be losslessly decrypted. As the corresponding target part of each share is noise-like, it will increase the suspicion of the encryption.

In the decrypting phase, to decode the secret image, we only need to iterate t pixels and perform the operations of less than $t - 1$ times addition and one time module. Thus, the time complexity is smaller.

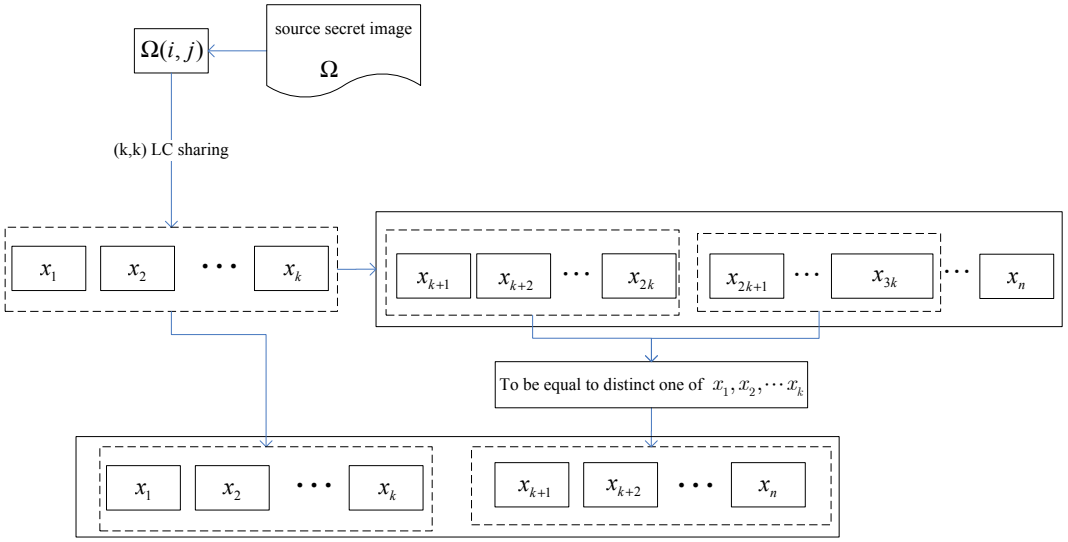


Figure 4. (k, n) threshold extension from LC-based ISS for (k, k) threshold.

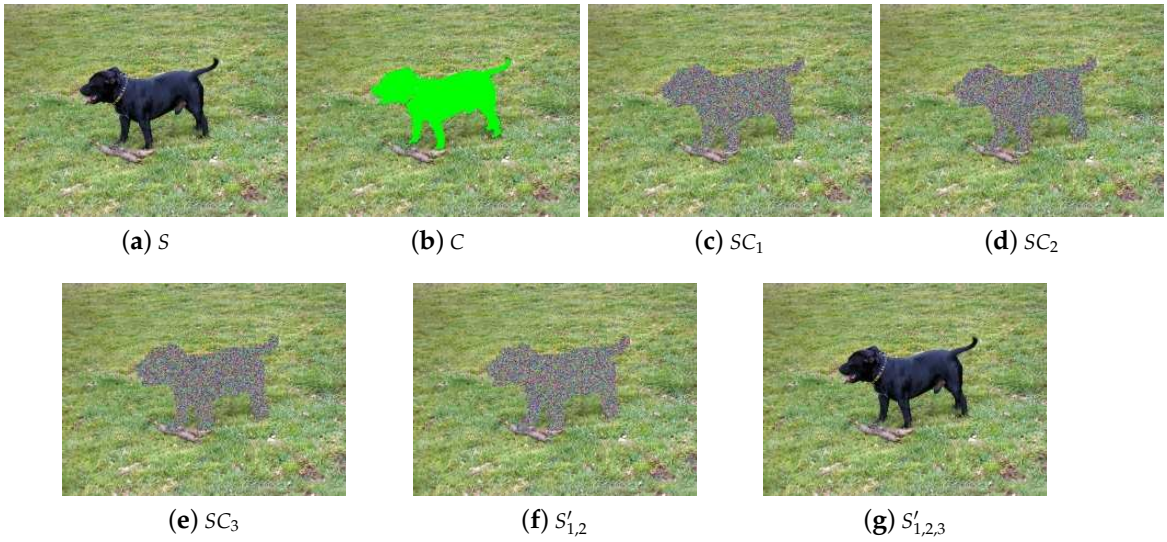


Figure 5. Experimental results when directly applying linear congruence (LC)-based image secret sharing (ISS) for $(3, 3)$-threshold. (**a**) The secret image S; (**b**) the same input cover image, denoted by C, through selecting and removing the secret target part with green color from S; (**c**–**e**) three shares SC_1, SC_2, and SC_3; (**f**–**g**) decrypted results by any two or more shares.

3. The Introduced Apiss Scheme

In this section, we will introduce an APISS scheme based on salience detection, LC, and image inpainting. The original secret image S and the selected target part Ω of the cover image will be encrypted to output n meaningful shares $SC_1, SC_2, \cdots SC_n$.

Our generation steps are described in Algorithm 1.

In Algorithm 1, we note the following.

1. In Step 1, the salient part is adaptively detected and removed by salience detection method and Otsu's threshold operation, thus the introduced scheme can achieve automatic processing. Moreover, the salient target part may include a single object or multiple objects.
2. In Step 3, each share has its own filling order, i.e., $\hat{p}_i$. To inpaint synchronously, the highest priority is selected from the n candidate orders as the the adopted order for all of the n shares.

Algorithm 1: The introduced APISS scheme for the (k, n) threshold.

Input: The threshold parameters (k, n) and a color secret image S with size $H \times W$.
Output: n color shares $SC_1, SC_2, \cdots SC_n$.
Step 1: Utilize salience detection method on S and Otsu's threshold operation to automatically obtain the target part Ω. Remove Ω with green color from S to obtain C_i, for $i = 1, 2, \cdots n$, where $SC_i = C_i$ denotes the input un-inpainted cover image.
Step 2: Use the method in Section 2.2 to determine the size of the template window, denoted by Ψ_{p^*}.
Step 3: For each share, find $\hat{p}_i$ with Equation (4). Find $i^* = \arg \max_{i \in [1,n]} W_i(\hat{p}_i)$, and let $\hat{p}_i = p_{\hat{i}^*}$, $i = 1, 2, \cdots n$.
Step 4: For each cover image, by using $\hat{p}_i$ and Equation (5), search for the most matching block to gain $\psi_{\hat{q}_i}$, and then, replace the patch of the current window by the most matching block, for $i = 1, 2, \cdots n$.
Step 5: For each position $(h, w) \in \{(h, w) | H_1 \leq h \leq H_2, W_1 \leq w \leq W_2\}$, where (H_1, W_1) and (H_2, W_2) denote the coordinates of the current processing template window, repeat Step 6.
Step 6: For the input of $SC_1(h, w), SC_2(h, w), \cdots SC_n(h, w)$, use LC-based ISS for (k, n) threshold to encrypt $S(h, w)$ to output updated $SC_1(h, w), SC_2(h, w), \cdots SC_n(h, w)$, where $SC_1(h, w), SC_2(h, w), \cdots SC_n(h, w)$ are least modified to satisfy the requirement of LC-based ISS.
Step 7: Renew the confidence terms after each filling process $C(q_i) = C(\hat{p}_i)$ for any $q_i \in \psi_{\hat{p}_i} \cap \Omega$, $i = 1, 2, \cdots n$.
Step 8: Repeat Steps 3-7 until each cover image is completely inpainted.
Step 9: Output n shares $SC_1, SC_2, \cdots SC_n$.

3. In Step 6, the values of $SC_1(h, w), SC_2(h, w), \cdots SC_n(h, w)$ are updated in the processing of encrypting $S(h, w)$ by LC-based ISS.
4. After the patch of the current window for each share is replaced by the most matching block in Step 4, the secret block of the current window is encrypted into n corresponding updated blocks with close values based on LC-based ISS to replace the patch of the current window in Step 4. Then, the modified patches of the current window for each share will be the basis for the next subsequent inpainting processing. Although the ISS sharing processing will introduce slight modification into the shares, the already inpainted block with the slight noise will be the input of the next inpainting round. Based on the current input, the next order and the most matching block will be selected. In such a way, meaningful shares can be obtained in a visually plausible way.

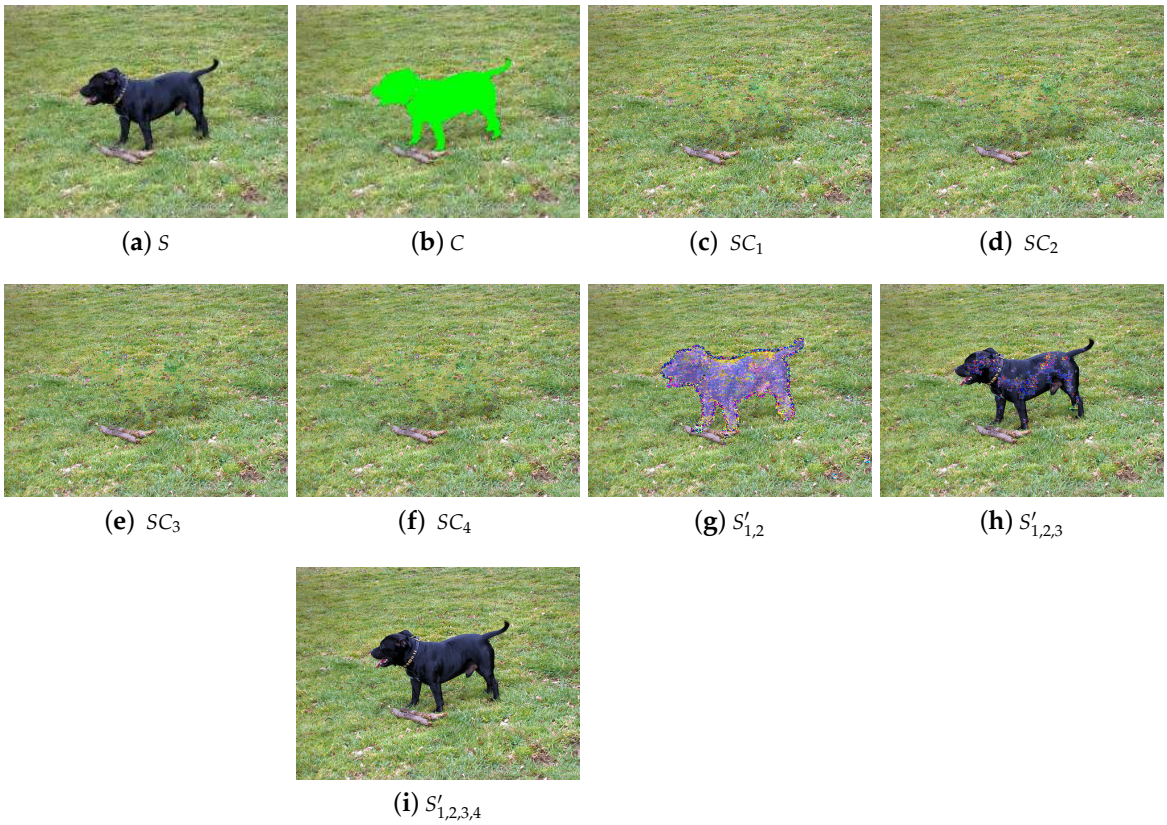
The secret decrypting of the introduced scheme is the same as LC-based ISS based on addition when any k or more shares are collected after removing the duplicate shared values.

4. Experimental Results and Analyses

In this section, experiments are performed to verify the efficiency of the introduced scheme.

4.1. Image Illustration

The simulated results by the introduced APISS scheme for case $(3, 4)$ are shown in Figure 6, where Figure 6c–f presents the outputted four shares SC_1, SC_2, SC_3, and SC_4 and the decrypted results obtained by addition with any two or more shares are illustrated in Figure 6g–i. The shares in Figure 6 corresponding to the target part are meaningful, and thus they look reasonable to the human eyes. When any three or more shares are collected, secret images with high quality are obtained. When all the four shares are collected, the secret image is losslessly decrypted. However, no information on the content of the secret image is decrypted when fewer than three shares are collected but with shape leakage. Decreasing the artifacts will be our future work.



(**a**) S (**b**) C (**c**) SC_1 (**d**) SC_2

(**e**) SC_3 (**f**) SC_4 (**g**) $S'_{1,2}$ (**h**) $S'_{1,2,3}$

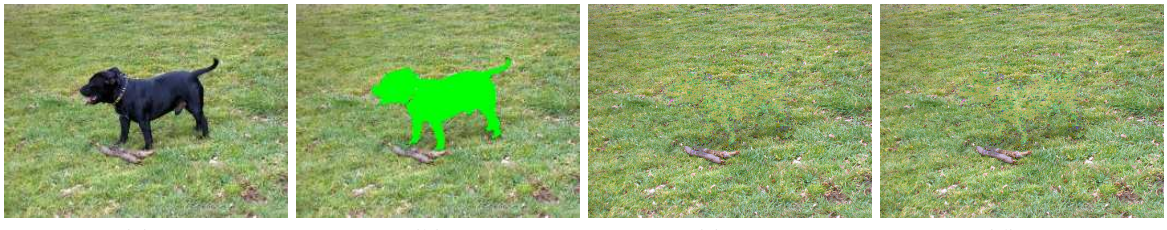
(**i**) $S'_{1,2,3,4}$

Figure 6. Experimental result for the introduced scheme for the threshold $(3, 4)$. (**a**) The secret image S; (**b**) the same input cover image through automatically selecting and removing the secret target part with green color from S using salience detection method on S and Otsu's threshold operation; (**c**–**f**) four shares SC_1, SC_2, SC_3, and SC_4; (**g**–**i**) decrypted results by any two or more shares.

Additionally, the simulated results for $(3, 3)$-threshold are given in Figure 7, and it is observed that the results are similar to the results described above.

Based on the above-observed results, we know the following.

1. The target part is automatically and adaptively detected, and then is successfully inpainted into the visually plausible shares.
2. Each share looks reasonable to the human eye, and thus is meaningful.
3. We cannot decrypt the secret when any $t < k$ shares are collected; when any $t = k$ or more shares are collected, the secret image is progressively decrypted; when all the n shares are collected, the secret image is losslessly decrypted.
4. An APISS for the (k, n) threshold is achieved by the introduced scheme.



(**a**) S (**b**) C (**c**) SC_1 (**d**) SC_2

Figure 7. *Cont.*

(**e**) SC_3 (**f**) $S'_{1,2}$ (**g**) $S'_{1,2,3}$

Figure 7. Experimental result for the introduced scheme for the threshold $(3,3)$. (**a**) The secret image S; (**b**) the same input cover image through automatically selecting and removing the secret target part with green color from S using salience detection method on S and Otsu's threshold operation; (**c**–**e**) three shares SC_1, SC_2, and SC_3; (**f**–**g**) decrypted results by any two or more shares.

4.2. Image Quality

According to Equation (6), when the other factors are fixed, the threshold value of k reflects the ratio of the difference (noise or error) covered by every shared pixel $x_i, i = 1, 2, \cdots k$. Specifically, a larger value of k will lead to a better image quality of the shares.

n plays a less important role than k in image quality of the shares. As n changes, the image quality of the share and that of the decrypted target will change slowly.

The image quality will be evaluated using the peak signal-to-noise-ratio ($PSNR$) defined in Equation (9) and the structural similarity index measure (SSIM) [28] defined in Equation (11). The same secret image in Figure 6a is employed to perform the experiments.

$$PSNR = 10\log_{10}(\frac{255^2}{MSE})\text{dB} \tag{9}$$

where

$$MSE = \frac{1}{W \times H}\sum_{i=1}^{W}\sum_{j=1}^{H}[S'(h,w) - S(h,w)]^2 \tag{10}$$

$$SSIM(x,y) = [l\,(x,y)]^{\alpha} \cdot [c\,(x,y)]^{\beta} \cdot [s\,(x,y)]^{\gamma} \tag{11}$$

where

$$l\,(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}$$
$$c\,(x,y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}$$
$$s\,(x,y) = \frac{2\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$$

$\mu_x, \mu_y, \sigma_x, \sigma_y$, and σ_{xy} denote the local means, standard deviations, and cross-covariance for the two images x and y. In this paper, we set $C_3 = \frac{C_2}{2}, \alpha = \beta = \gamma = 1$.

For the automatically selected target area and (k,n) ($2 \le k \le n, 2 \le n \le 4$) threshold, the average PSNR and SSIM between SC_i and DC_i on the target area, for $i = 1, 2, \cdots, n$, are presented in Table 1.

From Table 1, when the value of n is fixed, a larger value of k results in better image quality of the share because of Equation (6); when k is fixed, as the value of n changes, the image quality of the share keeps steadily.

Table 1. Average peak signal-to-noise-ratio (PSNR) and structural similarity index measure (SSIM) between SC_i and DC_i on the target area.

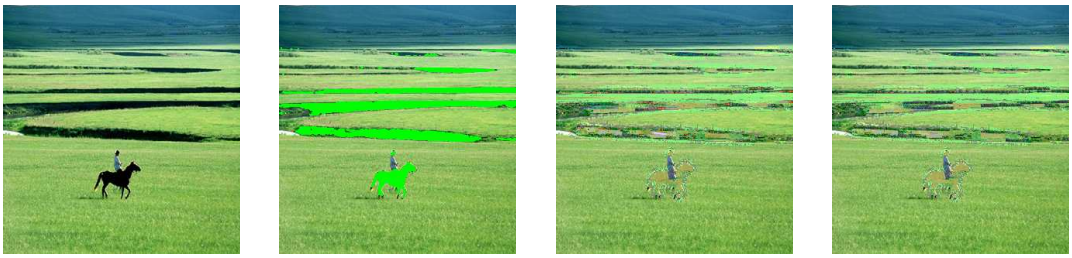
Threshold (*k*,*n*)	PSNR	SSIM
(2,2)	20.7750	0.8396
(2,3)	20.9817	0.8510
(3,3)	22.3658	0.8939
(2,4)	20.8117	0.8376
(3,4)	22.5924	0.8958
(4,4)	24.1274	0.9153

4.3. Comparisons with Related Schemes

We compare our APISS with the scheme of Yan et al. [23], where the same secret image as in Figure 8a and the $(3, 4)$ threshold are used. It is selected for comparison because the scheme of Yan et al. also achieves partial ISS.

Yan et al. [23] proposed a PISS scheme for the (k, n)-threshold, which can recover the full secret image including the secret target part when collecting any k or more shares. We use the same parameters as Yan et al. [23] to perform the comparisons shown in Figures 8 and 9, where $k = 3, n = 4$. Figure 9 shows the experimental results obtained using the scheme of Yan et al. [23]. Figure 8 illustrates the experimental results obtained using our scheme. According to Figures 8 and 9, both the scheme of Yan et al. and our scheme obtain meaningful shares and decrypt the secret image losslessly. However, the differences between the scheme of Yan et al. and ours are analyzed as follows.

1. Yan et al.'s scheme needs to select the target part manually, which is human-exhausted, especially for the target with irregular shape, and is therefore not suitable for batch processing. However, our scheme automatically and adaptively detects and removes the secret target part, which is thus suitable for the processing of large-scale images.
2. The selected target part of our scheme is less precise than that of Yan et al.'s scheme, because they select the target part manually. This weakness of our scheme can be enhanced through combining salience detection and object segmentation.



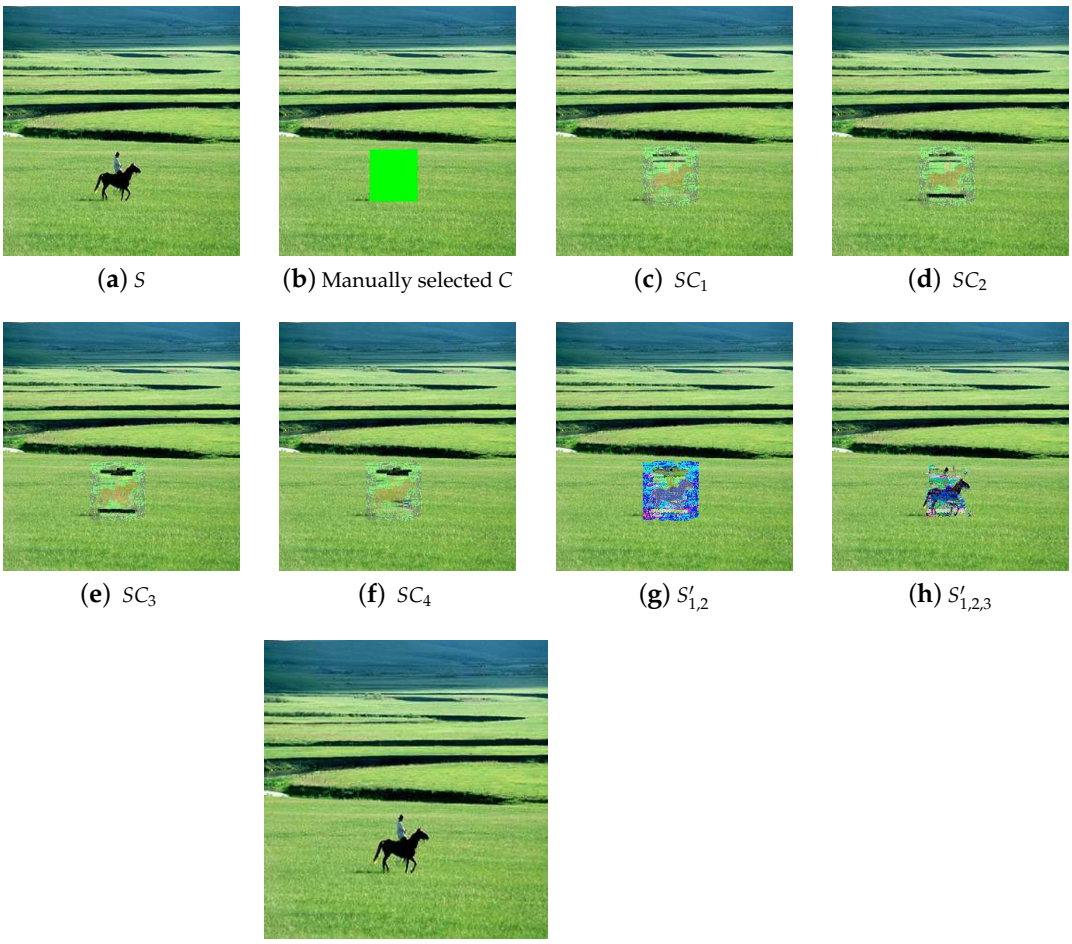
(a) S **(b)** Automatically generated C **(c)** SC_1 **(d)** SC_2

(e) SC_3 **(f)** SC_4 **(g)** $S'_{1,2}$ **(h)** $S'_{1,2,3}$

Figure 8. *Cont.*

(i) $S'_{1,2,3,4}$

Figure 8. Experimental result for the introduced scheme for the threshold $(3, 4)$. (**a**) The secret image S; (**b**) the same input cover image through automatically selecting and removing the secret target part with green color from S using salience detection method on S and Otsu's threshold operation; (**c**–**f**) four shares SC_1, SC_2, SC_3, and SC_4; (**g**–**i**) decrypted results by any two or more shares.



(a) S **(b)** Manually selected C **(c)** SC_1 **(d)** SC_2

(e) SC_3 **(f)** SC_4 **(g)** $S'_{1,2}$ **(h)** $S'_{1,2,3}$

(i) $S'_{1,2,3,4}$

Figure 9. Experimental results of the scheme of Yan et al. [23] for the (k, n) threshold, where $k = 3, n = 4$. (**a**) The secret image S; (**b**) the same input cover image through manually selecting and removing the secret target part with green color from S; (**c**–**f**) four shares SC_1, SC_2, SC_3, and SC_4; (**g**–**i**) decrypted results by any two or more shares.

4.4. Extensions and Discussions

We may improve the performance of the introduced scheme by using the following methods.

1. The important information of the input image can be selected by other techniques according to practical requirements, such as edge detection and object segmentation.
2. Salience detection on multiple secret images with close content can be utilized to improve the salience detection accuracy.
3. Some other inpainting methods, such as the PDE-based method, can also be applied to the introduced scheme.
4. We can adopt different ISS schemes, different filling order selection methods, or different (k, n) threshold extension methods to achieve different features.
5. Our method can be applied to grayscale image. If a binary image inpainting algorithm is employed, our method may be applied to binary image as well.
6. We can use more images to test the scheme. The advantage of adaptive PISS and the effectiveness of saliency detection are dependent on the adopted saliency detection algorithm if the images have multiple objects or more complex background.

5. Conclusions

This paper introduces a method for outputting meaningful partial image secret sharing (PISS) based on salience detection, linear congruence, and image inpainting. Experiments confirm the efficiency of the introduced scheme. The PSNR of the outputted share is more than 20, which means that acceptable image quality of the meaningful share is achieved. Comparisons with related typical scheme show the advantages of the introduced scheme. Our scheme is more suitable for batch processing with medium precision of automatically selecting target part. Decreasing the artifacts; improving the image quality through applying follow-up improved image inpainting, salience detection, and ISS methods; and including more sample images with different contrast distribution and with more objects to demonstrate the generalization of the suggested method will be our future works.

Author Contributions: Data curation, L.S.; Formal analysis, G.Y.; Investigation, Y.L.; Methodology, X.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (Grant Number: 61602491) and the Key Program of the National University of Defense Technology (Grant Number: ZK-17-02-07).

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

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

References

1. Yan, X.; Lu, Y.; Liu, L.; Wan, S.; Ding, W.; Liu, H. Exploiting the Homomorphic Property of Visual Cryptography. *Int. J. Digit. Crime Forensics* **2017**, *9*, 45–56. [CrossRef]
2. Belazi, A.; El-Latif, A.A.A. A simple yet efficient S-box method based on chaotic sine map. *Opt. Int. J. Light Electron Opt.* **2017**, *130*, 1438–1444. [CrossRef]
3. Cheng, Y.; Fu, Z.; Yu, B. Improved Visual Secret Sharing Scheme for QR Code Applications. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2393–2403. [CrossRef]
4. Wang, G.; Liu, F.; Yan, W.Q. Basic Visual Cryptography Using Braille. *Int. J. Digit. Crime Forensics* **2016**, *8*, 85–93. [CrossRef]
5. Naor, M.; Shamir, A. Visual Cryptography. In *Advances in Cryptology-EUROCRYPT'94, Workshop on the Theory and Application of Cryptographic Techniques, May 9–12*; Lecture Notes in Computer Science; Springer: Perugia, Italy, 1995; pp. 1–12.
6. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [CrossRef]
7. Yan, X.; Liu, L.; Lu, Y.; Gong, Q. Security analysis and classification of image secret sharing. *J. Inf. Secur. Appl.* **2019**, *47*, 208–216. [CrossRef]
8. Yan, X.; Li, J.; Lu, Y.; Liu, L.; Yang, G.; Chen, H. *Relations between Secret Sharing and Secret Image Sharing. Security with Intelligent Computing and Big-data Services*; Yang, C.N., Peng, S.L., Jain, L.C., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 79–93.

9. Ding, W.; Liu, K.; Yan, X.; Wang, H.; Liu, L.; Gong, Q. An Image Secret Sharing Method Based on Matrix Theory. *Symmetry* **2018**, *10*, 530. [CrossRef]
10. Zhou, Z.; Arce, G.R.; Di Crescenzo, G. Halftone visual cryptography. *IEEE Trans. Image Process.* **2006**, *15*, 2441–2453. [CrossRef]
11. Wang, Z.; Arce, G.R.; Di Crescenzo, G. Halftone visual cryptography via error diffusion. *IEEE Trans. Inf. Forensics Secur.* **2009**, *4*, 383–396. [CrossRef]
12. Liu, F.; Wu, C. Embedded extended visual cryptography schemes. *Inf. Forensics Secur. IEEE Trans.* **2011**, *6*, 307–322. [CrossRef]
13. Weir, J.; Yan, W. A comprehensive study of visual cryptography. In *Transactions on DHMS V*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6010, pp. 70–105.
14. Yan, X.; Lu, Y.; Liu, L. General Meaningful Shadow Construction in Secret Image Sharing. *IEEE Access* **2018**, *6*, 45246–45255. [CrossRef]
15. Guo, T.; Jiao, J.; Liu, F.; Wang, W. On the Pixel Expansion of Visual Cryptography Scheme. *Int. J. Digit. Crime Forensics* **2017**, *9*, 38–44. [CrossRef]
16. 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]
17. Yan, X.; Wang, S.; Niu, X.; Yang, C.N. Halftone visual cryptography with minimum auxiliary black pixels and uniform image quality. *Digit. Signal Process.* **2015**, *38*, 53–65. [CrossRef]
18. Thien, C.C.; Lin, J.C. Secret image sharing. *Comput. Graph.* **2002**, *26*, 765–770. [CrossRef]
19. Yang, C.N.; Ciou, C.B. Image secret sharing method with two-decoding-options: Lossless recovery and previewing capability. *Image Vis. Comput.* **2010**, *28*, 1600–1610. [CrossRef]
20. Bao, L.; Yi, S.; Zhou, Y. Combination of Sharing Matrix and Image Encryption for Lossless (k, n)-Secret Image Sharing. *IEEE Trans. Image Process.* **2017**, *26*, 5618–5631. [CrossRef]
21. Liu, Y.; Yang, C.; Wang, Y.; Zhu, L.; Ji, W. Cheating identifiable secret sharing scheme using symmetric bivariate polynomial. *Inf. Sci.* **2018**, *453*, 21–29. [CrossRef]
22. Liu, L.; Lu, Y.; Yan, X.; Wang, H. Greyscale-images-oriented progressive secret sharing based on the linear congruence equation. *Multimed. Tools Appl.* **2017**, *77*, 20569–20596. [CrossRef]
23. Yan, X.; Lu, Y.; Liu, L.; Wang, S. Partial secret image sharing for (k,n) threshold based on image inpainting. *J. Vis. Commun. Image Represent.* **2018**, *50*, 135–144. [CrossRef]
24. Fu, H.; Cao, X.; Tu, Z. Cluster-Based Co-Saliency Detection. *IEEE Trans. Image Process.* **2013**, *22*, 3766–3778. [CrossRef] [PubMed]
25. Otsu, N. A threshold selection method from gray-level histograms. *Automatica* **1975**, *11*, 23–27. [CrossRef]
26. Criminisi, A.; Perez, P.; Toyama, K. Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process. A Publ. IEEE Signal Process. Soc.* **2004**, *13*, 1200-1212. [CrossRef] [PubMed]
27. Shen, W.; Song, X.; Niu, X. Hiding Traces of Image Inpainting. *Res. J. Appl. Sci. Eng. Technol.* **2012**, *4*, 4962–4968.
28. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]