A High-Quality (k,n) Visual Cryptography Algorithm Based on Secret Sharing (Postprint)

Authors: Ding Haiyang

Date: 2018-05-24T00:00:00+00:00

Abstract

The (k,n) visual cryptography algorithm is an important information hiding algorithm. By employing the concept of Shamir' s secret sharing, a binary secret image is concealed within n share images. Through the introduction of randomness control during the sharing process, it is ensured that the generated share images are meaningless images that approximate noise images. From the n meaningless share images, any k share images can be selected, and the secret image can be extracted using Lagrange interpolation. This algorithm utilizes Shamir' s secret sharing to implement (k,n) visual cryptography, requires no codebook, and does not result in unrestricted pixel expansion. Experimental results demonstrate that the algorithm can achieve (k,n) visual cryptography, and the extraction accuracy rate of the secret image can be guaranteed at 100%.

Full Text

Preamble

A High-Quality (k,n) Visual Cryptography Algorithm Based on Secret Sharing

Ding Haiyang[1,2] [1]College of Information Engineering, Beijing Institute of Graphic Communication, Beijing 102600, China [2]Information Security Center, Beijing University of Posts & Telecommunications, Beijing 100876, China

Abstract: (k,n) visual cryptography is an important information hiding algorithm. By applying Shamir' s secret sharing concept, a binary secret image is hidden within n share images. Randomness control is added during the sharing process to ensure the generated shares are meaningless images resembling noise. Any k shares selected from these n meaningless shares can be used to extract the secret image via Lagrange interpolation. This algorithm implements (k,n) visual cryptography using Shamir' s secret sharing without requiring a codebook and without causing uncontrolled pixel expansion. Experimental results

demonstrate that the algorithm successfully achieves (k,n) visual cryptography with a guaranteed 100% correct decoding rate for the extracted secret image.

Keywords: Shamir' s secret sharing; (k,n) visual cryptography; Lagrange interpolation

0 Introduction

Visual cryptography (VC) was first introduced by Naor and Shamir in their seminal work [1]. The (k,n) VC concept involves hiding a secret image in n shares, where any k shares suffice to reconstruct the secret. Extended visual cryptography algorithms were proposed in [2,3], while Hou et al. [4] developed a friendly visual cryptography scheme that conceals a secret image within two images. Yan et al. [5] presented an information hiding algorithm based on error-correcting codes. All these approaches [1-5] fall under pixel-expansion-based visual cryptography, which, despite its simplicity and ability to generate meaningless shares, suffers from pixel expansion and codebook requirements. As the number of users n increases, the share images become significantly enlarged.

Random-grid-based visual cryptography generates meaningless shares using random grid techniques without pixel expansion. Kafri and Keren [6] introduced the original concept of random grids and proposed three information hiding methods. Chen et al. [7] developed a (2,n) random-grid-based visual cryptography scheme, later extending it to the (k,n) case [8]. References [9,10] describe general formulations of random-grid-based visual cryptography. Ou et al. [11] proposed an improved tagged visual cryptogram using random grids, while Yan et al. [12] introduced a random-grid-based visual secret sharing algorithm with universal interface structures and multiple decryption capabilities. Rabari et al. [13] proposed an extended (2,m,n) random-grid visual cryptography algorithm applicable to grayscale and color images. While random-grid-based methods [6-13] avoid pixel expansion and codebooks, their primary drawback is extremely low contrast in the extracted secret image—often below 1/10—making it difficult to recover a clear secret image.

Recently, block-based visual cryptography has emerged as a new research direction. Hou et al. [14] proposed a block-based (2,n) visual cryptography algorithm where a secret image is decomposed into n non-overlapping blocks to generate n shares, with any k shares (2 k n) capable of extracting partial image blocks. However, this approach [14] is vulnerable to cheating attacks, which Hou et al. [15] later addressed with a cheating-immune block-based progressive visual cryptography scheme. Roy et al. [16] presented a (3,4) secret image sharing scheme that decomposes each 2×2 block of the secret image into four combinations shared among four shares, enabling reconstruction with any three shares. Block-based visual cryptography [14-16] offers low computational complexity and meaningless shares without codebooks, but its main limitation is that k

shares can only recover a portion of the secret image, requiring re-blocking when n changes.

Table 1 compares various visual cryptography algorithms. The proposed algorithm aims to achieve high-quality (k,n) visual cryptography without codebooks or uncontrolled pixel expansion by leveraging Shamir's secret sharing [17], where a secret data D is shared into n pieces, and any k pieces can reconstruct D.

---

1 Shamir' s Secret Sharing

Shamir's (k,n) threshold secret sharing algorithm [17] works as follows. A secret D is divided into n shares D ,⋯,D , with any k shares sufficient to reconstruct D. The process defines a (k-1)-degree polynomial q(x) as shown in Equation (1), where a = D and coefficients a ,⋯,a are randomly selected. For n input values x ,⋯,x , the shares are computed as D = q(x), producing n share pairs (x ,D), ⋯,(x ,D). With any k share pairs, Lagrange interpolation recovers the secret data D.

Four key considerations apply: a) All values must be modulo a prime p, with D, a ,⋯,a , and x in [0,p); b) Share data x and y must also be in [0,p), so y is computed modulo p; c) Large secret data D can be split into multiple m-bit blocks; d) Different secret data D can use different polynomials for share generation.

---

2 Implementing Secret Image Sharing Using Shamir' s Secret Sharing

A secret image cannot be treated as a single secret value; it must first be decomposed into m-bit blocks. Each m-bit block serves as secret data D, shared using a (k-1)-degree polynomial f(x) with a = D. Computing y = f(x) yields n share pairs (x ,y),⋯,(x ,y), which are stored in corresponding positions across n share images. Repeating this process for the entire secret image generates n share images.

Three critical issues must be addressed: a) Selection of data bit length m; b) Range of primary data values; c) Addition of randomness control.

2.1 Selection of Data Bit Length m

Treating m-bit data as secret D gives D a range of [0,2 -1], requiring n 2 -1. From a security perspective, larger m is better. However, halftone information hiding introduces errors, and excessively large m would increase the overall bit error rate. Conversely, too small m limits the number of users n. This work selects m = 4, meaning each 4-bit block constitutes a secret data D with range [0,15].

Given m = 4: a) a ranges in [0,15]; b) Prime p > 2 -1, so p = 17; c) x ranges in [0,15], but x = 0 yields y = a , meaning a is not hidden, so x [1,15]; d) y ranges in [0,15]; e) Number of users n 15, with k < n.

2.3 Secret Image Sharing Process

2.3.1 Process Without Randomness Control The sharing flow without randomness control is shown in Figure 1 [Figure 1: see original paper]. A secret image S is shared into n meaningless shares through these steps: a) Generate a (k-1)-degree polynomial f(x); b) For each a value, generate and select usable share data; c) Decompose secret image S (resolution W×H) into W/4×H groups of 4-bit data, where each group B , serves as secret data D, and compute its secret value S , using Equation (5); d) Set a = S , , retrieve n share pairs (x ,y) for this a from the array(a ,x ,y), and store them in corresponding positions across n share images by converting each (x ,y) into 8-bit binary values; e) This process yields n meaningless share images, each with resolution 2W×H.

The data structure array(a ,x ,y) stores n share pairs for each a value.

2.3.2 Implementation Results Figure 2 [Figure 2: see original paper] shows secret image S (256×256). Using the method from Section 2.3.1 with Figure 2(a) as S, a (3,4) sharing model is implemented, distributing S across 4 share images where any 3 can reconstruct S. The generated shares are shown in Figure 3 [Figure 3: see original paper]. The results are unsatisfactory—the secret image is discernible, failing to meet secrecy requirements.

2.3.3 Cause Analysis Three regularities cause this problem: a) Limited variation in secret image S creates many consecutive identical a values (e.g., repeated a =15 or a =0), with a changes reflecting secret image patterns; b) Using the same polynomial f(x) and a value always generates identical share data; c) The storage order of 4 share groups across 4 images is fixed.

These regularities, particularly (b) and (c) which stem from algorithm design, can be mitigated through randomness control.

2.4 Adding Randomness Control

2.4.1 Randomness Control Strategies Three randomness controls can be added: a) Use multiple generating polynomials, with different secret data using different polynomials; b) For each a , generate more than n usable share pairs and randomly select n as shares; c) Randomize share data storage order across images.

2.4.2 Implementation of Randomness Control The enhanced sharing process includes: a) Initialize pm (k-1)-degree polynomials f (x),···,f (x). For each 4-bit secret data D, randomly select polynomial f (x) and set a = D; b) Generate n+expnum share pairs using f (x), then select n pairs from these,

offering C(n+expnum, n) possible combinations; c) Before storing n share pairs, generate a random integer cshift [0,n-1], compute c = (i + cshift) mod n, and store the i-th share pair (x ,y) in share image c at the corresponding position.

The resulting data structure is array(pnum,a ,x ,y), where pnum indexes the polynomial and stores n+expnum share pairs for each a .

2.4.3 Experimental Results Using Figure 2(a) as secret image S, a (3,4) sharing model is implemented with varying randomness controls.

Figure 4 [Figure 4: see original paper] uses one polynomial with n+expnum = 6 share pairs (n=4, expnum=2), selecting n pairs. While improved, secret image contours remain visible.

Figure 5 [Figure 5: see original paper] uses pm = 6 polynomials with n+expnum = 6 pairs (pm=6, n=4, expnum=2). The secret image is no longer visible, but the four shares show uneven distribution.

Figure 6 [Figure 6: see original paper] combines all three randomness controls (pm=6, n=4, expnum=2) with randomized storage order. The shares appear uniformly distributed and reveal no secret image information.

These experiments demonstrate that combining all three randomness controls yields optimal results. Section 3 presents the complete algorithm for generating n shares from a secret image.

---

3 Proposed Algorithm

This paper proposes a high-quality (k,n) visual cryptography algorithm based on Shamir' s secret sharing. A binary secret image is shared into n meaningless shares; any k shares can extract the secret image via Lagrange interpolation.

3.1 Sharing a Secret Image into n Meaningless Shares

Based on Section 2, Shamir' s secret sharing distributes secret image S into n meaningless shares. The process, shown in Figure 7 [Figure 7: see original paper], includes: a) Generate pm (k-1)-degree polynomials f (x),⋯,f (x); b) For each polynomial f (x) (r=1:pm), generate and select usable share data for each a value; c) Decompose secret image S (resolution W×H) into W/4×H groups of 4-bit data. Each group B , serves as secret data D, and its secret value S , is computed using Equation (5); d) Set a = S , , select n share pairs (x ,y) for this a from array(pnum,a ,x ,y) under a chosen polynomial, and store them across n share images with randomness control: - Random integer r [1,pm] selects the polynomial from array(r,a ,x ,y) where i [1,n+expnum]; - Randomly select n pairs from n+expnum candidates, with i [1,n]; - Random integer cshift [0,n-1] computes c = (i + cshift) mod n, storing the i-th pair (x ,y) in share

image c ; e) This process yields n meaningless share images (Share 1-n), each with resolution 2W×H.

3.2 Extracting C(n,k) Secret Images from n Shares

From n share images, any k shares can extract the secret image via Lagrange interpolation. Since C(n,k) combinations exist, C(n,k) secret images can be extracted (Extracted 1-C(n,k)). The extraction process, shown in Figure 8 [Figure 8: see original paper], includes: a) Select k share images (Selected Shares 1-k); b) Decompose each 2W×H share image into W/4×H groups of 8-bit data, generating share pairs (x ,y). From k selected shares, obtain k share pairs (x ,y , i=1-k) at corresponding positions; c) Compute Lagrange interpolation using Equation (6) on the k share pairs to recover a ; d) Set S , = a and compute 4-bit B , using Equation (7) as the extracted secret data; e) After processing all positions across k shares, the extracted secret image R has resolution W×H; f) Use Equation (8) to compute the correct decoding rate (CDR) between extracted secret image R (m=1-C(n,k)) and original secret image S, both of resolution W×H.

The key to correctness: ensuring the k share pairs originate from the same polynomial and identical a value.

---

4 Algorithm Analysis and Experimental Results

4.1 Correctness

4.1.1 Generating n Shares from a Secret Image

a) Generate array(pnum,a ,x ,y) as described in Section 3.1(a)-(b), where pm polynomials each produce n+expnum share pairs for every a value;
b) Form secret data values S , from each 4-bit block of S and set a = S , ;
c) Through randomness control, obtain n share pairs array(r,a ,x ,y);
d) Store the n share pairs in share image c at corresponding positions.

The critical aspect is ensuring n share pairs come from polynomial r and the determined a value.

4.1.2 Extracting Secret Images from Shares

a) Arbitrarily select k shares from n share images;
b) Extract 8-bit data from corresponding positions in each share to generate k share pairs (x ,y , i=1-k);
c) Compute Lagrange interpolation on k share pairs using Equation (6) to obtain a ;
d) Convert a to 4-bit data as the extracted secret value.

Extraction correctness: k share pairs from the same polynomial and a value guarantee correct a recovery via Lagrange interpolation, ensuring a perfectly

reconstructed secret image.

4.2 Experimental Results

Using Figure 2(a) as secret image S, Section 3.1 generates 4 share images (Figure 6). Selecting any 3 of 4 shares yields C(4,3)=4 combinations, each extracting one secret image (Figure 9 [Figure 9: see original paper]). The CDR values for the four extracted images are 1, 1, 1, 1 (100% accuracy).

Similarly, using Figure 2(b) as secret image S produces 4 share images, with any 3 shares extracting 4 secret images (Figure 10 [Figure 10: see original paper]). All CDR values are 1, confirming 100% extraction accuracy.

Figures 9 and 10 demonstrate that Lagrange interpolation on generated shares produces zero errors and 100% correct decoding rates.

4.3 Comparison with Existing Algorithms

Table 2 compares the proposed algorithm with existing visual cryptography schemes. The proposed method achieves arbitrary (k,n) VC without pixel expansion, maintaining fixed share resolution of 2W×H regardless of user count n, while guaranteeing 100% CDR for all extracted images. This represents superior performance compared to pixel-expansion-based methods [1-5], random-grid-based methods [6-13], and block-based methods [14-16].

---

5 Conclusion

This paper proposes a high-quality (k,n) visual cryptography algorithm based on Shamir' s secret sharing. A binary secret image is concealed within n meaningless shares, and any k shares can extract the secret image via Lagrange interpolation. The algorithm requires no codebook and avoids uncontrolled pixel expansion. Experimental results confirm successful (k,n) visual cryptography implementation with guaranteed 100% correct decoding rates for extracted secret images.

References

[1] Naor M, Shamir A. Visual cryptography [C]// Proc of Workshop on the Theory and Application of Cryptographic Techniques. Italy: Springer, 1995: 1-12.

[2] Ateniese G, Blundo C, Santis A, et al. Extended capabilities for visual cryptography [J]. Theoretical Computer Science, 2001, 250 (1-2): 143–161.

[3] Liu Feng, Wu Chuankun, Lin Xijun. Step construction of visual cryptography schemes [J]. IEEE Trans on Information Forensics and Security, 2010, 5 (1): 27-38.

[4] Hou Y C, Quan Zenyu, Liao H Y. New designs for friendly visual cryptography scheme [J]. International Journal of Information and Electronics Engineering, 2015, 5 (1): 15-20.

[5] Yan Xuehu, Lu Yuliang, Chen Yuxin, et al. Secret image sharing based on error-correcting codes [C]// Proc of the 3rd IEEE International Conference on Big Data Security on Cloud. [S. l.] : Institute of Electrical and Electronics Engineers Inc, 2017: 86-89.

[6] Kafri O, Keren E. Encryption of pictures and shapes by random grids [J]. Optics Letters. 1987, 12 (6): 377-379.

[7] Chen T H, Tsao K H. Visual secret sharing by random grids revisited [J]. Pattern Recognition. 2009, 42 (9): 2203-2217.

[8] Chen T H, Tsao K H. Threshold visual secret sharing by random grids [J]. Journal of Systems and Software, 2011, 84 (7): 1197-1208.

[9] Wu Xiaotian, Sun Wei. Random grid-based visual secret sharing for general access structures with cheat-preventing ability [J]. Journal of Systems and Software, 2011, 85 (5): 1119-1134.

[10] Wu X, Sun W. Visual secret sharing for general access structures by random grids [J]. Iet Information Security, 2012, 6 (4): 299-309.

[11] Ou Duanhao, Wu Xiaotian, Dai Lu, et al. Improved tagged visual cryptograms by using random grids [J]. Lecture Notes in Computer Science, 2014, 8389: 79-94.

[12] Yan Xuehu, Lu Yuliang, Liu Lintao, et al. Progressive visual secret sharing for general access structure with multiple decryptions [C]// Proc of the 8th International Conference on Information Technology in Medicine and Education. [S. l.] : Institute of Electrical and Electronics Engineers Inc, 2016: 668-673.

[13] Rabari D K, Meghrajani Y K. Lock and key share-based random grid visual secret sharing scheme for grayscale and color images with two decoding options [C]// Proc of ISEA Asia Security and Privacy Conference. India: Institute of Electrical and Electronics Engineers Inc, 2017: 1-5.

[14] Hou Y C, Quan Zenyu, Tsai C F, et al. Block-based progressive visual secret sharing [J]. Information Sciences, 2013, 233 (2): 290-304.

[15] Hou Y C, Quan Zenyu, Tsai C F, et al. Cheating immune block-based progressive visual cryptography [J]. Lecture Notes in Computer Science, 2014, 8389: 95-108.

[16] Roy R, Bandyopadhyay S, Kandar S, et al. A novel 3-4 image secret sharing scheme [C]// Proc of International Conference on Advances in Computing,

Communications and Informatics. India: Institute of Electrical and Electronics Engineers Inc, 2015: 2072-2075.

[17] Shamir A. How to share a secret [J]. Communications of the ACM, 1979, 22 (11): 612-613.

Note: Figure translations are in progress. See original paper for figures.