

Key Sharing Schemes Using Visual Cryptography

Neelam Yadav

Lecturer in DAVCET Kanina

Dhiraj Kumar

Lecturer in RPSGOI

Ravi Yadav

Student in SEC Dundlod

Abstract - Visual Cryptography is a technique which allows visual information (pictures, text, etc.) to be encrypted in such a way that the decryption can be performed by the human visual system, with the aid of computers. The message consists of a collection of black and white pixels and each pixel is handled separately. Now suppose the secret image is to be distributed as shares among a set of two participants in such a way that from a single share it is not possible to decode the secret image, but if both of the participants come together and stack their shares one above the other, the secret image will be visible. But there will be loss of contrast. Here we will be converting the image into n matrix and during the decryption we will be selecting an k share among the given n share so that we can recreate the original image i.e. $(k;n)$. To provide more security we will be applying Hybrid encryption methods to it.

Keywords- Visual Cryptography, key Sharing Scheme.

I. INTRODUCTION

With the rapid advancement of network technology, multimedia information is transmitted over the Internet conveniently. Various confidential data such as military maps and commercial identifications are transmitted over the Internet. While using secret images, security issues should be taken into consideration because hackers may utilize weak link over communication network to steal information that they want. To deal with the security problems of secret images, various image secret sharing schemes have been developed.

In 1994, Naor and Shamir described a new $(k; n)$ visual cryptographic scheme using black and white images, where the dealer encodes a secret into n participants. In this scheme, shared secret information (printed text, handwritten notes, pictures, etc.) can be revealed without any cryptographic computations. For example, in a $(k; n)$ visual cryptography scheme, a dealer encodes a secret into n shares and gives each participant a share, where each share is a transparency.

The secret is visible if any k (or more) of participants stack their transparencies together, but none can see the shared secret if fewer than k transparencies are stacked together. By identifying that the result of stacking the transparencies are the same as Boolean- OR operation denoted by XOR on the binary digits involved, it using Visual Cryptography Introduction is possible to extend the Visual Cryptography schemes to any 2 binary string. For example, the following scheme describes how one could implement Visual cryptography scheme for a single 4 binary digit. In order to share a binary string, each binary digit in it could be shared independently, one after the other using the same scheme.

II. BLACK AND WHITE VISUAL CRYPTOGRAPHY SCHEMES

2.1 Sharing Single Secret

Naor and Shamir's [3] proposed encoding scheme to share a binary image into two shares Share1 and Share2. If pixel is white one of the above two rows of Table 1 is chosen to generate Share1 and Share2. Similarly If pixel is black one of the below two rows of Table 1 is chosen to generate Share1 and Share2. Here each share pixel p is encoded into two white and two black pixels each share alone gives no clue about the pixel p whether it is white or black. Secret image is shown only when both shares are superimposed.

To hide a binary image into two meaningful shares Chin-Chen Chang et al [4] suggested spatial-domain image hiding schemes. These two secret shares are embedded into two gray level cover images. To decode the hidden embedding images can be superimposed. Balancing the performance between pixel expansion and contrast Ligu Fang recommend a (2, n) scheme based on combination. Threshold visual secret sharing schemes mixed XOR and OR operation with reversing and based on binary linear error correcting code was suggested by Xiao-qing and Tan.

The disadvantage of the above schemes is that only one set of confidential messages can be embedded, so to share large amounts of confidential messages several shares have to be generated.

















To hide a binary image into two meaningful shares Chin-Chen Chang et al [4] suggested spatial-domain image hiding schemes. These two secret shares are embedded into two gray level cover images. To decode the hidden messages, embedding images can be superimposed. Balancing the performance between pixel expansion and contrast Ligu Fang recommend a (2, n) scheme based on combination. Threshold visual secret sharing schemes mixed XOR and OR operation with reversing and based on binary linear error correcting code was suggested by Xiao-qing and Tan.

The disadvantage of the above schemes is that only one set of confidential messages can be embedded, so to share large amounts of confidential messages several shares have to be generated.

2.2 Visual Cryptography:

There are various connections between combinatorial structures and secret sharing. For threshold scheme can be implemented based on a small Latin square. In 1994, Naor and Shamir invented a new type of secret sharing scheme, called Visual Cryptography scheme. In secret sharing schemes using Visual Cryptography, shared secret information (printed text, handwritten notes, pictures, etc.) can be revealed without any cryptographic computations. For example, in a (k; n) visual cryptography scheme, a dealer encodes a secret into n shares and gives each participant a share, where each share is a transparency. The secret is visible if any k (or more) of participants stack their transparencies together, but none can see the shared secret if fewer than k transparencies are stacked together.

Table1. Naor and Shamir’s scheme for encoding a binary pixel into two shares

Pixel	Probability	Share 1	Share 2	Superposition of Share 1 & Share2
	50%			
	50%			
	50%			
	50%			

III. PRINCIPLE OF SECRET SPLITTING

The simplest sharing scheme splits a message between two people. Consider the case where Daniel has a message M , represented as an integer that he would like to split between two people Alice, and Bob, in such a way that neither of them alone can reconstruct the message. A solution to the problem readily lends itself: Choose a random number r . Then r and $M - r$ are independently random. He gives $M - r$ to Alice and r to Bob as their shares. Each share by itself means nothing in relation to the message, but together, they carry the message M . To recover the message, Alice and Bob have to simply add their shares together.

IV. THRESHOLD SCHEME

In 1979 Shamir [2] and Blakley [1] introduced the concept of sharing of the secret message as a means and a method of making the message secure. Under this scheme, the message M is divided into n pieces $M_1; M_2; M_3; \dots; M_n$, with or without transformation of the message, in such a way that, for a specified k , ($2 < k < n$),

1. knowledge of any k or more pieces- M_i makes M computable;
2. Knowledge of any $k - 1$ or fewer M_i pieces leaves M completely undetermined (in the sense that all its possible values are equally likely).

Such a scheme is called a $(k; n)$ -threshold scheme. The parameter $k < n$ is called the threshold value.

4.1 A method of solution

Now a key is shared by computing points on a random polynomial in $(Z=pZ)[X]$. So first we must find a way of representing the "plaintext" secret as a set of classes modulo p . This is not really part of secret sharing process; it is merely a way to prepare the secret so that it can be shared. To keep the things as simple as possible, we will assume that the "plaintext" secret contains only words written in uppercase letters. Thus the secret is ultimately a sequence of letters and blank spaces. The first step consists of replacing each letter of the secret by a number, using the following correspondence:

"A SMALL LEAK WILL SINK A GREAT SHIP" is

1099282210212199211410209932182121992818232099109916271410299928171825

The blank space between words is replaced by 99. Having done that, we obtain a number, possibly a very large one, if the secret is large. However it is not a number we want, but rather classes modulo p . Therefore, we must break the numerical representation of the secret into a sequence of positive integers, each smaller than p . These are called the blocks of the secret.

If we choose the prime $p = 9973$, the numerical representation of the proverb above must be broken into blocks smaller than 9973. One way to do this is as follows:

1099-2822-1021-2199-2114-1020-9932-1821-2199- 2818-2320-9910-9916-2714-1029-9928-1718-25

When secret is reconstructed, one obtains a sequence of blocks. The blocks are then joined together to give the numerical representation of the secret. It is only after replacing the numbers by letters, according to the table above, that one obtains the original secret.

Note that we have made each letter correspond to a two-digit number in order to avoid ambiguities. For example, if we had numbered the letters so that A corresponds to 1, B to 2, and so on, then we wouldn't be able to tell whether 12 stood for AB or for the letter L, which is the twelfth letter of the alphabet.

Of course, any convention that is unambiguous can be used instead of the one above. For example, one might prefer to use ASCII code, since the conversion of characters is automatically done by the computer.

Algorithm to recover the shared byte

Input: Two shares S_1 and S_2 of length 7 bits each and the random 4 integer r .

Output: The secret information $K = K_1K_2K_3 \dots K_8$.

Step 1. Let A and B be the POB-numbers corresponding to S_1 and S_2 respectively.

Step 2. For $i = 1$ to 8 do

if $(i - r) \leq j = i + 1$;

else $j = i$;

$K_i = A_j - B_j$.

Step 3. The recovered secret is $K = K_1K_2K_3 : : : K_8$

The above scheme is a 2 out of 2 secret sharing scheme.

Chang's et al. Algorithm

Chang et al. proposed in 2002 a new secret color image sharing scheme based on modified visual cryptography. The proposed approach uses meaningful shares (cover images) to hide the colored secret image and the recovery process is lossless. The scheme defines a new stacking operation (XOR) and requires a sequence of random bits to be generated for each pixel. Chang's scheme can be generalized to an n out of n approach as opposed to Chang Tsai's scheme presented previously.

Method description

Assume that a gray image with 256 colors constitute a secret to be hidden. Each color can be represented as an 8-bit binary vector. The main idea is to expand each colored pixel into m sub pixels and embed them into n shares. This scheme uses $m=9$ as an expansion factor. The resulting structure of a pixel can be represented by an $n \times 9$ Boolean matrix $S=[S_{ij}]$ where $(1 \leq i \leq n, 1 \leq j \leq 9)$ and $S_{ij}=1$, if and only if, the j th sub pixel in the i th share has a non-white color. To recover the color of the original secret pixel, an "XOR" operation on the stacked rows of the n shares is performed.

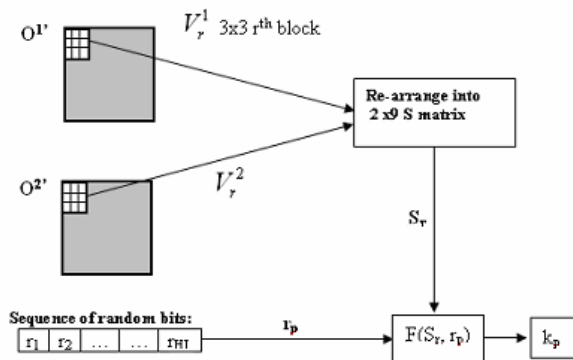


Figure 1: Chang Tsai secret sharing recovering algorithm

Improved image generation scheme

In this section, we introduce a modification of Chang's algorithm to generate better quality camouflage images. Most of the modifications are applied to the sub pixel expansion block described in the next section.

Hiding Algorithm

Before sub pixel expansion, add one to all pixels in the cover images and limit their maximum value to 255. This ensures that no "0" valued pixels exist in the images. When the images are expanded, replace all the 0's in S_0, S_1 by values corresponding to k_1-1 in B_1 and k_2-1 in B_2 instead of leaving them transparent. Also, adjust all pixel values to be between 0-255.

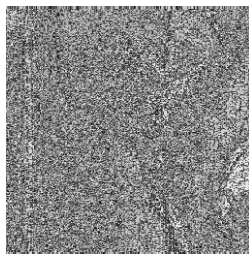
Steps of the Algorithm
1 Take all regions of size $t \times t$ in the camouflage images
2 Re-structure the square matrices as $1 \times m$ vectors
3 Scan through the 9 sub pixels in the vector and note coordinates of the k_1 and the k_1-1 colors previously
Encrypted.
4 Count the number of k and $k-1$ pixels in the processed vector, denoted as $count_{k-1}$, $count_k$, respectively.
5 If $count_{k-1} < count_k$, the transparent pixel is color $k-1$, otherwise, set it to k
6 Use the k_1 and k_2 colors to find the secret pixel using the $F(.,.)$ function and the random number previously transmitted

7 Repeat for all txt block pixels in the camouflage images
--

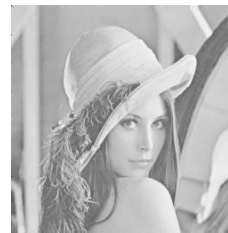
In this paper image processing software package (MATLAB) is used as the engine for the image processing experiments. An RGB image is stored in MATLAB as an M -by- N -by-3 data array that defines red, green, and blue color components for each individual pixel. The color of each pixel is determined by the combination of the red, green, and blue intensities stored in each color plane at the pixel's location. Images that are used during these experiments are uncompressed images.



Lena Original



Encrypted



Text inserted and decrypted

Process for generating share B and share C

To split the share Temp into two share B and share C, the coding rules are designed carefully. The idea is to consider the problem reversed. Assume to do logic exclusive-or share B and share C will generate the share Temp Then the code book for the second encoding process can be designed. For example, if the extended block of the share Temp is, then Table 2 shows all the possible cases of share B and share C. So to split the share Temp into two shares can be selected one of the ten solutions randomly. This encoding process also increases the security.

	1	2	3	4	5	6	7	8	9	10
Share B										
Share C										
Shares Temp										

Table2: The partial rules of the second encoding process.

Process for generating share A and share Temp

For the extended block of share A, one of the patterns, is selected randomly from the patterns shown in Figure 2. Put the selected pattern in the pixel located at (i, j) , $(j, N-i-1)$, $(N-j-1, i)$, $(N-i-1, N-j-1)$, where i , j , and N , represent the pixel coordinates and the size of the original image, respectively. After the extension, the size of share A is become four times of the original one. Share Temp can be generated according to the three secret data and the generated share A. The generating rules are shown in Table 1. For example, if the extended block of the share A is, and the first, second and third secrets are white, black, and white, respectively, then the extended block of share Temp is corresponding to according to the rule shown in Table.1. According the rules designed in Table 1, it is shown that if share A and share Temp are



Stacked together, and assume the extended block is filled with two white pixels and two black pixels, and then it means there is a white pixel in the secret image. On the other hand, if the stacked extended block is filled with one white pixel and three black pixels, it means that the confidential image is a black pixel. In Table 1, the share A' and share A'' represent the results of share A rotated 90o clockwise and 90o counterclockwise, respectively, rotating the share A clockwise 90o and stacks it with the share Temp to identify the second secret image. Finally, rotating the share A counterclockwise 90o and stacks it with the share Temp to obtain the third secret image.

V. CONCLUSION

I have now presented an $(n - 1; n)$ -threshold secret sharing scheme, in which the size of a share is $\lceil n/2 \rceil$ times the size of the secret. In this I have classified three types of balanced strings, and established a very strong theorem related to balanced string. As per the theorem, any string can be written as the ring sum (xor) of two balanced strings. I have used this property and presented a secret sharing scheme, in which the size of a share is just one bit more than the size of the original secret. In this paper I have described about how the secret images or text has been sent to the other party in such a way that if any Third person or hacker gets the message then he/she cannot find out the original message. This method is possible only when I use visual cryptography i.e. in this method I divide the image in different shares such that seeing single piece of share no-one can understand what the secret text is about. I can only get the final image when I stack all the shares or the threshold that I have set for the shares to get the final image. But during dividing the share processes their will be loss of contrast i.e. loss of some pixels due to which the final image will not be as clear as the original image. But this method is useful for many applications such as in Banks, Military etc.

REFERENCES

- [1] G. R. Blakley: Safeguarding Cryptographic Keys, Proceeding of AFIPS 1979 National Computer Conference, vol. 48, New York, NY, June 1979, pp. 313-317.
- [2] A. Shamir: How to Share a Secret, Communications of the ACM, vol. 22, no. 11, Nov. 1979, pp. 612-613.
- [3] M. Naor and A. Shamir: Visual Cryptography, Advances in cryptology- EUROCRYPT94, Lecture Notes in Computer Science, vol. 950, 1995, pp. 1-12.
- [4] Chin-Chen Chang and Tai-Xing Yu: Sharing Secret Gray Image in Multiple Images, National Chung Cheng University, Taiwan, 2002.
- [5] D. Chen and D. R. Stinson: Recent Results on Combinatorial Constructions for Threshold Schemes, Australasian Journal of Combinatory, vol 1, 1990, pp. 29-48.
- [6] Wen-Pinn Fang, "Visual Cryptography In Reversible Style", IEEE Proceeding on the Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP2007), Kaohsiung, Taiwan, R.O.C, 2007.
- [7] Jen-Bang Feng, Hsien-Chu Wu, Chwei-Shyong Tsai, Ya-Fen Chang, Yen-Ping Chu, "Visual Secret Sharing For Multiple Secrets", Pattern Recognition 41 .pp. 3572 – 3581, 2008.
- [8] Tzung-Her Chen, Kai-Hsiang Tsao, and Kuo-Chen Wei, "Multiple-Image Encryption By Rotating Random Grids", Eighth International Conference on Intelligent Systems Design and Applications, pp. 252-256 , 2008.
- [9] D. R. Stinson and S. A. Vanstone: A combinatorial approach to threshold schemes. SIAMJ. on Discrete Mathematics, 1(2):230-236, 1988.
- [10] M. Tompa and H. Woll: How to share a Secret with Cheaters, Journal of Cryptography, vol. 1, no.2, 1988,pp 133-138 .