# IMPLEMENTATION OF VISUAL CRYPTOGRAPHY TECHNIQUE ON SQUARE BW SECRET IMAGES

Stephanie I. Pella [1], M. J. Pella [2]

1. Electrical Engineering Department, Faculty of Science and Engineering, University of Nusa Cendana, Jl Adisucito, Penfui, Kupang, 85000, Indonesia
2. Mathematics Department, Faculty of Science and Engineering, University of Nusa Cendana, Jl Adisucito, Penfui, Kupang, 85000, Indonesia

E-mail : s.i.pella@gmail.com, mj_pella@yahoo.com

## Abstract

The advance of computer security technology has led to the invention of many cryptography algorithms. Visual Cryptography is a cryptography technique that needs no cryptographic computation in decoding. To encode a secret image, the image is encrypted to several transparent shares. In order to retrieve the secret image shares are stacked on top of each other. This paper describes a technical implementation of this algorithm on a square black and white image for VC2,2 Scheme and VC3,3 Scheme. The result shows that the model works well with an image that contains text.

*Keywords : Visual Cryptography, Computer Security, Secret Sharing, BW Image.*

## 1. Introduction

Most currently implemented cryptography technique relies on the availability of computer for the encryption and description processes. The use of complex mathematics function makes it nearly impossible to do the calculation by hand. Both encryption and description processes need a quite expensive calculation resulting in additional processing time when the cryptography method is implemented in data security schemes.

Cryptography technique usually uses a secret key in concealing the secret information, so only the ones that have the key can decrypt the message. The strength of encryption relies on the strength of the key, generally proportional to the length of the key. Traditionally the key used on encryption process is much smaller than the information itself. So in theory, by trying at most $2^k$ times, where k is the length of the key in bits, an intruder can break every cryptography technique. This method of trying every possible combination of a key is generally known as brute force [1].

In [2][3], Visual Cryptography, a cryptography technique relying only on human visual sight in the description process, is proposed. This technique focused on the encryption of a written material such as printed text, picture, handwritten notes or map. The original image is encrypted into several new images, called shared image, by adding a random noise. The principle is, from any individual new images, one cannot determine the original image due to the additional random noise. When a set of shared images is printed on transparence sheets and stacked together, the original will be revealed.

In visual cryptography, each shared image can act as either the encrypted information (chipper) or the key, hence the length of the key equal to the length of the cipher. So other than reduce the complexity of description process, this cryptography technique also provides a better resistance to the brute-force method. By having a key that have the same length with the chipper (n bits), an intruder have to tries at most $2^n$ to reveal the message. Since n is much longer than k, visual cryptography provides much better security to the brute force technique, given the same n, than other symmetric cryptography schemes.

Many paper has discussed this technique in mathematical level [2][3][4][5]. This paper will address the technical implementation of the visual cryptography technique on a square BW image, using MATLAB 7.

## 2. Methods

The development of this implementation contains three phases. In the first phase we investigate the mathematical models of visual cryptography and adjust some properties to meet the need of our implementation. Here we inspect the models for encoding the secret images into two special cases VC2,2 and VC3,3 as described in [2][3][4]. The second phase deals with the software development base on the model in phase 1. In phase three, we test our implementation by running security check to ensure that stacking less than n shares cannot reveal the secret image.

### 2.1 Mathematics Model

In the basic model every pixel in the original image is encrypted into *n* shares. Each shares contains *m* black and

white sub-pixels, with

$$m = 2^{n-1} \quad (1)$$

A matrix *S* (size *n* x *m*) is used to describe the structure using the rule described in equation (2)

$$S_{ij} = \begin{cases} 1, & subpixel\ j\ in\ share\ i\ is\ black \\ 0, & otherwise \end{cases} \quad (2)$$

The solution of this model is stored in two matrices ($C_0$, $C_1$). To construct $C_0$ and $C_1$ in this implementation, we simplify the rules in [2], such that:

1. For any matrix *S* in $C_0$ the "or" all required shares will satisfy $H(V)<m$
2. For any *S* in $C_1$ the "or" *V* of any *k* of then rows of *S* will satisfy $H(V) = m$
3. For any subset of required shares, the "or(s)" of some or all the subset will contain same matrices with same frequency.

Thus, if for a given white pixel from the original image every share is picked up from $C_0$, condition (1) ensures that the pixel will be greater than the threshold *d* and recognized as white once *k* shares stacked. For black pixel opposite condition holds if for every share *S* is chosen from $C_1$. Those properties are called *contrast*, whereas property (3) is called *security*, because it is not possible to decide whether the pixel is black or white when less that *n* shares are available.

In a VC2,2 case, the secret image is encrypted into two shared image and both shares are required in decryption process. While according to equation (1), using 2 subpixels per pixel in the original image is enough, in this implementation, each pixels in the original image will be encrypted into 4 subpixels to maintain the aspect ratio of the image. $C_0$ and $C_1$ are generated by randomly picking the shares in Figure 1. When the pixel in the original image is white the same subpixels from the same shares will be used for $C_0$ and $C_1$. Opposite condition is applied for a black pixel.

horizontal shares vertical shares diagonal shares

**Figure 1 Shares of VC2,2 [2]**

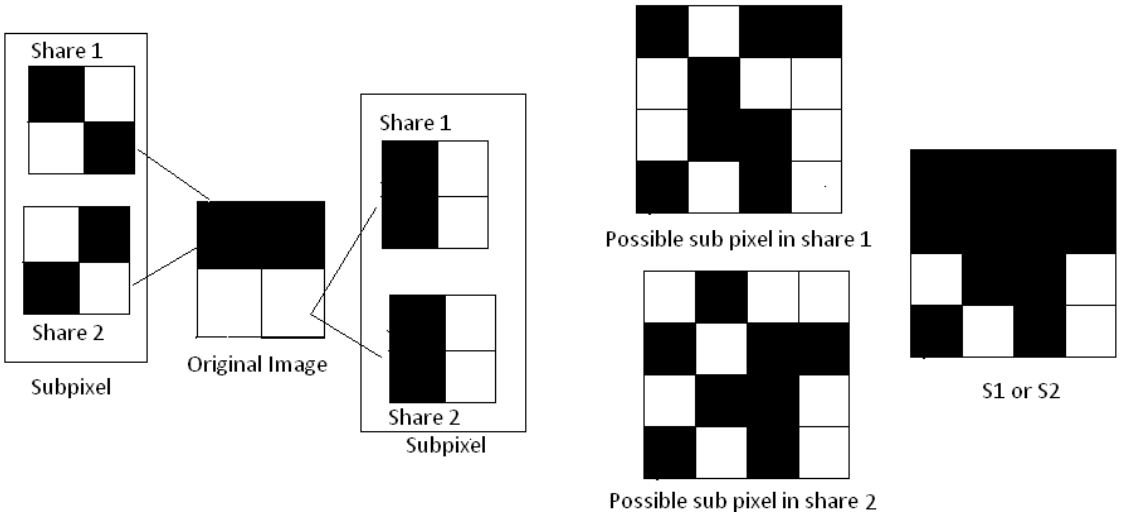


**Figure 2 Model for VC2,2**

The more detailed explanation of the model can be viewed in Figure 2, where we described the example of 2x2 image, with the possible subpixel in share 1 and share 2. Here we can see that where the pixel in original image Im(i,,j) is white, we have the same subpixels for share 1 and share 2. On the other hand, when pixel in original image is black, the subpixel in share1 is the negation of the subpixel in share2. To map the subpixel into the shared images $S_1$ and $S_2$ we use the rule described in equation 3.

$$S_n(2i-1:2i, 2j-1:2j) = \begin{cases} \text{perm } C_0, \text{Im}(i,j) = \text{White} \\ \text{perm } C_1, \text{ Im}(i,j) = \text{Black} \end{cases} \quad (3)$$

In the description process, human eyes will interpret the gray area (H(v) <m ) as white pixel and the black area as black pixel.
Extending the model of VC2,2 we can generate $C_0$ and $C_1$ matrices for VC3,3 where the secret images is encrypted into 3 shares. Each pixel will be encrypted into m =$2^{3-1}$=4.

$$C_0 = column\ permuting\ of \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$C_1 = column\ permuting\ of \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The or-ed of the three rows in $C_1$ will have H(v) = 4 which is equal to m, while in $C_0$ will have H(v) = 3 which is less than m. These properties satisfy condition (1) and (2). Furthermore the or-ed of less than three rows in $C_0$ and $C_1$ will have same H(v) which satisfy condition (3).

### 2.2 Software Design

Software design in this implementation is divided into four stages, namely binarization, subpixel construction, subpixel mapping and shared image creation as described in Figure3.

The binarization process convert the 8 bit grayscale image (256 level intensity) into 1 bit black and white (BW) image using 128 as the threshold as shown in the flowchart in Figure 4.

In constructing the subpixel for each pixel in the original image, a set of share subpixel is randomly selected from figure 1 for VC2,2 case and equation (4) for VC3,3 case. $C_0$ matrix is used for a white pixel abd $C_1$ matrix for the white one.

The subpixels is mapped into the shared images $S_n$ (where n=1,2,3) using the formula in equation (3). It is important to be noted that the formula is only applicable for 4 subpixels per pixel mapping as the case in VC2,2 and VC3,3. When more shares are used, the formula needs to be alterd to meet the condition.

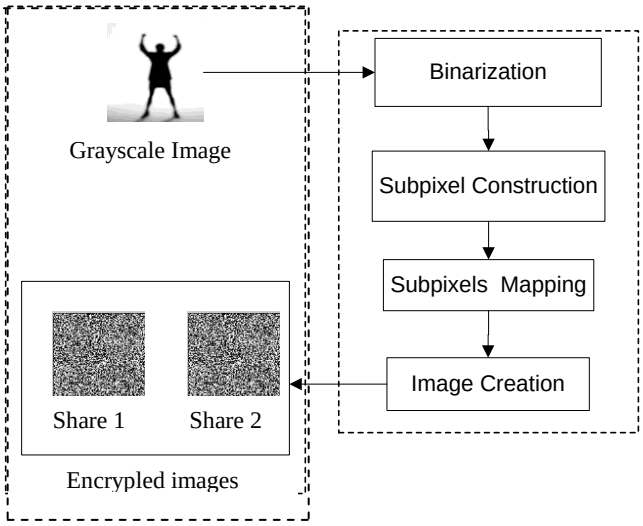


**Figure 3 Encryption Process**

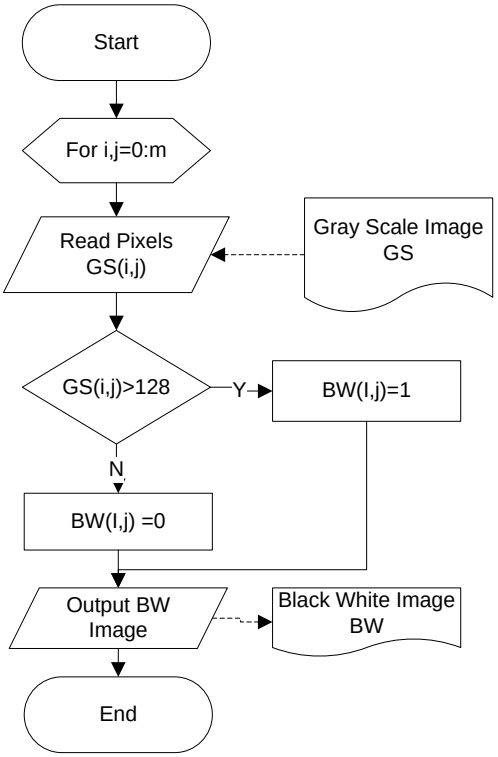


**Figure 4 Binarization**

After mapping the all set subpixel into share $S_n$ the next stage is to create an image to be printed in a transparent sheet. The image is created by using the data in matrix $S_n$, where n is equal to 1 and 2 for the VC2,2 case and 1, 2 and 3 for the VC3,3.

For decoding process, in real execution, to reveal the original image all the transparent sheets are aligned together. In this report, the aligned process is done mathematically using software by using "or" to all matrices that contain the subpixels from original images

## 3. Result

We implement the model discussed in section 2 in MATLAB 7 environment. The encryption and decryption result of VC2,2 for an image containing text can be shown in Figure 5. Here we can see that share 1 and share 2 reveal nothing of the original image. The share images contain the black and white pixels that appear random to the bare eyes. Both the areas that contain the message (black pixel in the original image) and the space (white pixel in the original image) have same frequency of black and white dots. The decrypted image is quite clear so that a human user can distinguish the original image from the decrypted image.

Figure 6 shows the result of the same model when deals with a non-text image. The result shows that even though the decrypted image is still recognizable, the clarity of the image is not as good as the text image from human sight point of view.

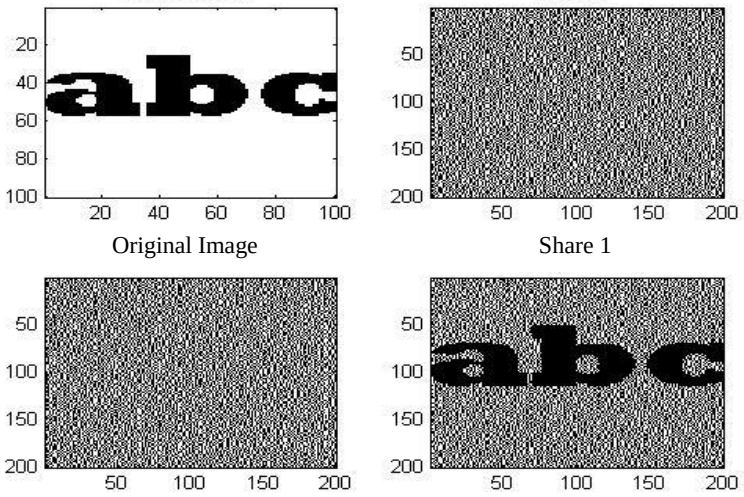


**Figure 5 VC2,2 of Text Image**

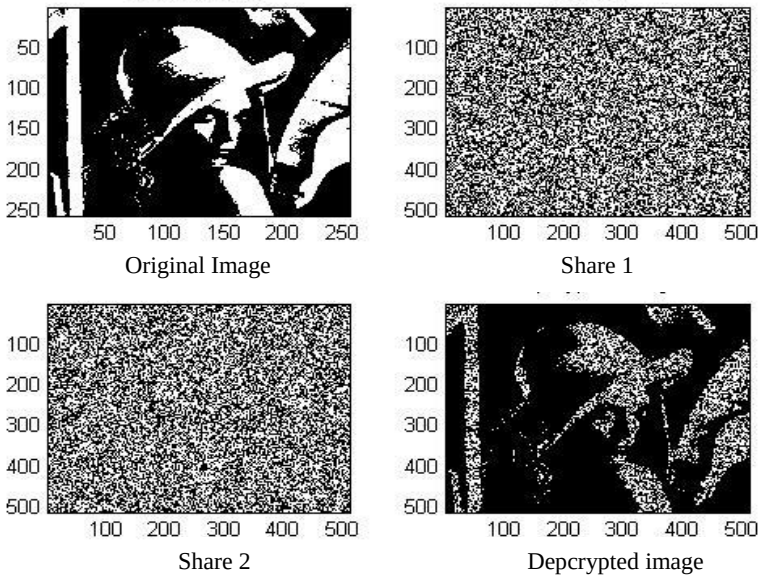


**Figure 6 VC2,2 of Non-Text Image**

Figure 7 and figure 8 shows the result of encryption and decryption in VC3,3 schemes. Share 1, share 2 and share 3 in Figure 7 look random for human sight and by having only one share, one can not reveal the original image.

In Figure 8, it is shown that only by stacking 3 shares together, one can reveal the original image. The result of stacking only 2 shares in any combination still have the same H(v) for the white and black area.

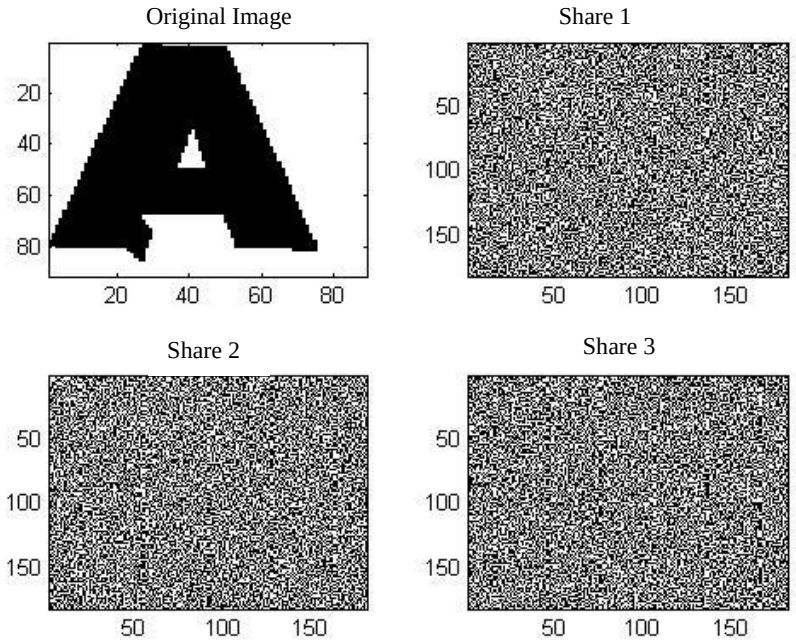


**Figure 7 Encryption of VC3,3**

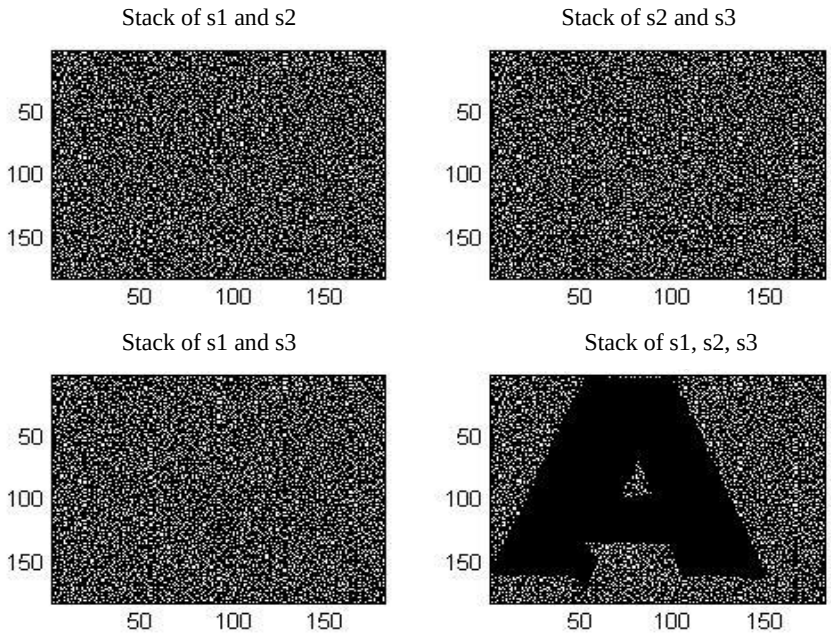


**Figure 8 Descryption of VC3,3**

## 4. Discussion

The result shows that the model works perfectly in images that contain text. The contrast between the white pixel and black pixel in the decrypted image is easily distinguishable by human eyes. In non-text images, the contrast between black and white pixels in the decrypted is not very clear for human sight.

The low quality of the original BW image is the main factor of the problem in non-text images. Converting a grayscale pixel into a BW image decreases the quality of the image to $1/2^8$ of the original image. Adding noise to the white pixel is damage the quality even more.

The result shown in section 3 also confirms that brute force attack for visual cryptography cost more than doing so in other cryptography technique. Most technique uses the key that much smaller than the message. Here the attackers have to deal with the key that actually bigger than the original message.

Figure 5 and Figure 6 shows that the 250x250 pixel original image is encrypted into two 500x500 pixel shares. When an attacker only have one share, to revealed the original image he need $2^{(500)(500)}$ tries, or have the key strength of 250000 bit key. Moreover for general k x k pixel secret image in VC2,2 scheme the attacker need $2^{4(k)(k)}$ tries. The number will expand exponentially when we add more shares and or use larger image.

Here we can see that brute-forcing a visual cryptography schemes cost more expensive than try guessing the pixel original image. So it is safe to say that this scheme is secure from brute force attack.

## 5. Summary

In this paper we have discussed the implementation of Visual Cryptography technique in a BW image. We construct two implementations for VC2,2 and VC3,3 schemes.

The model works properly for images that contain text. For non-text images, the contrast of black and white pixels is not very obvious for human sight due to the low quality of BW image.

## REFERENCES

[1] William Stalling. 2011 ***Network Security Essential: Application and Standards***, Prentice Hall.

[2] M. Naor and A. Shamir. 1994. ***Visual Cryptography***. in Proc. Advances in Cryptology. vol. 950, LNCS, pp. 1-12.

[3] G. Ateniese, C. Blundo, A. De Santis, and D.R. Stinson. 1996. ***Visual Cryptography for General Access Structures***. Inf. Comput. vol. 129, no. 2, pp. 86-106.

[4] C.-N. Yang and C.-S. Laih. 1999. ***Some New Types of Visual Secret Sharing Schemes***. in Proc. Nat. Computer Symp. vol. 3, pp. 260-268.

[5] W.-G. Tzeng and C.-M. Hu. 2002. ***A New Approach for Visual Cryptography***. Design, Codes and Cryptography. 27,207-227.