Variable Length Key based Visual Cryptography Scheme for Color Image using Random Number

Shyamalendu Kandar
Assistant Professor
Computer Sc. & Engineering
Haldia Institute of Technology
Haldia, West Bengal, India

Arnab Maiti
Scholar, M.Tech.
Computer Sc. & Engineering
Haldia Institute of Technology
Haldia, West Bengal, India

ABSTRACT

Visual Cryptography is a special type of encryption technique to obscure image-based secret information which can be decrypted by Human Visual System (HVS). It is a kind of secret-sharing scheme that encrypts the secret image into n number of shares. It is imperceptible to reveal the secret information unless a certain number of shares(k) or more are superimposed. As the decryption process is done by human visual system, secret information can be retrieved by anyone if the person gets at least k number of shares. For this, simple visual cryptography is very insecure.

In this current work we have proposed a variable length Symmetric Key based Visual Cryptographic Scheme for color images where a secret key is used to encrypt the image and division of the encrypted image is done using Random Number. Unless the secret key, the original image will not be decrypted. Here secret key ensures the security of the scheme and visual cryptography is used to break the image into number of shares.

Keywords: Visual Cryptography, Secret Sharing, Random Number, Symmetric Key.

1. INTRODUCTION

Visual cryptography is a cryptographic technique where visual information (Image, text, etc) gets encrypted in such a way that the decryption can be performed by the human visual system without aid of computers. [1]

Image is a multimedia component sensed by human perception. A color digital image is composed of a finite number of elements called pixels. In a 32 bit digital image each pixel consists of 32 bits, which includes four parts, namely Alpha, Red, Green and Blue; each with 8 bits. Alpha part represents degree of transparency. If all bits of Alpha part are ‘0’, then the image is fully transparent. A 32 bit sample pixel is represented in the following figure. [2] [3]

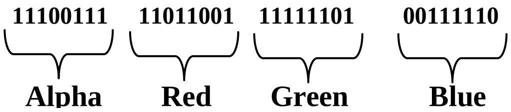


Figure 1: Structure of a 32 bit pixel

Human visual system acts as an OR function. If two transparent objects are stacked together, the final stack of objects will be transparent. But if any of them is non-transparent, then the final stack of objects will be non-transparent.

A Key is used to make information secure so that attacker can not retrieve the secret information without the key. Original data is encrypted using key and produces cipher. Cipher is decrypted using key and the original data is retrieved. Symmetric encryption is a type of encryption technique where same key is used for both encryption and decryption.

Fixed length key can be easily computed by combination of characters by the attacker. For variable length key, it is difficult to find the key as the length can be 0 to any number.

In this paper Section 2 describes the Overall process of Operation, Section 3 describes the encryption process of the image using key, Section 4 describes the process of k-n secret sharing Visual Cryptography scheme on the encrypted image, Section 5 describes decryption process, Section 6 describes the experimental result, Section 7 describes the future scope and Section 8 draws the conclusion.

2. OVERALL PROCESS

Step I: Any combination of characters [Characters, Numbers and Special Symbol] of any length is taken as KEY, which is XOR ed with the pixel array computed from the original image. This makes the image blur to some extent.

Step II: The encrypted image is divided into n number of shares using k-n secret sharing visual cryptography scheme such that k number of shares is sufficient to reconstruct the encrypted image.

Step III: k number of shares produced in Step II is stacked together to reconstruct the encrypted image.

Step IV: The KEY taken in Step I is XOR ed with the image produced in Step III, to generate the original image.

This is described by the following figure:

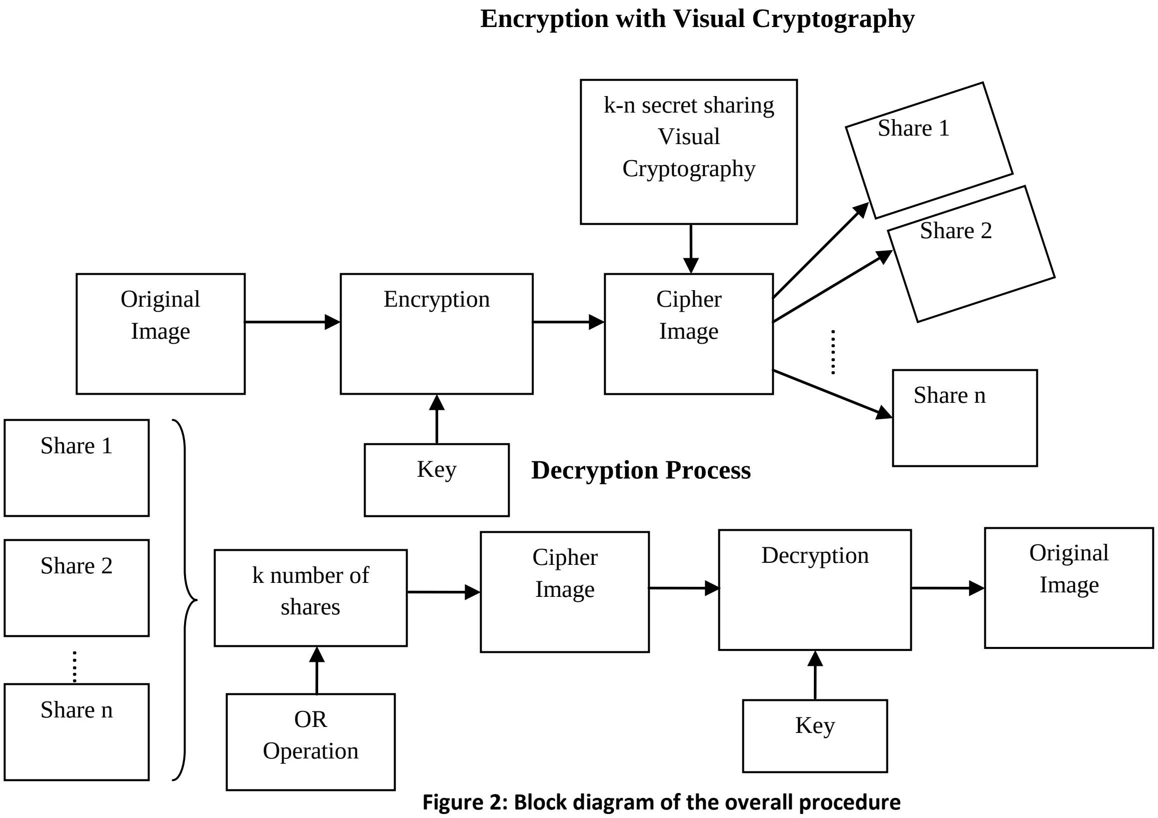


Figure 2: Block diagram of the overall procedure

3. VARIABLE LENGTH KEY BASED ENCRYPTION

An image is taken as input. A key of any length is taken as input from keyboard. An XOR operation is done on the image using the key to generate the cipher image. Following algorithm is used for encryption.

Step I: Take an image as input. Calculate Height (h) and Width (w) of the image.

Step II: Create an array STORE of size w*h*32 to store the binary pixel values of the image using the loop

```
for i = 0 to (w*h-1)
{  Scan each pixel value of the image and convert it into 32 bit binary string let PIX
     for j = 0 to 31
      { STORE[i*32+j] = PIX.charAt(j)
      }
}
```

Step III: Enter a key of any length from keyboard. Calculate the length (len) of the key. Convert the key into binary string let CONVERTED_KEY.

[String is broken into character. Characters are converted into binary of length 7, then merged again to produce CONVERTED_KEY]

Step IV: Create an array KEY of size len*7 to store the binary values of the key by the following process.

```
for i = 0 to (len-1) {
   KEY[i] = CONVERTED_KEY. charAt(i)
}
```

Step V: Calculate ITERATION = (w*h*32)/(len*7). KEY array is XOR ed with the STORE array by the following process.

```
for i = 0 to (ITERATION-1)
{  for j = 0 to (len*7-1)
   {
     STORE[i*len*7+j] = STORE[i*len*7+j]^KEY[j]
    }  //XOR operation
}
```

Step VI: Create a one dimensional array IMG_CONS[w*h] to store constructed pixels.

Construct Alpha, Red, Green and Blue part of each pixel by taking consecutive 8 bit substring starting from 0th bit. Construct pixel from these part and store it into IMG_CONS [w*h].[1] [4]

Step VII: Generate image ENCRPT_IMG from IMG_CONS[w*h]. [3] [6]

4. k-n SECRET SHARING VISUAL CRYPTOGRAPHY SCHEME ON THE ENCRYPTED IMAGE

The encrypted image obtained from Section 3, number of shares it will be divided (n) and number of shares to be taken to reconstruct the encrypted image (k) are taken as input. The division is done by the following algorithm.

Step: I: Take encrypted image ENCRPT_IMG produced in Section3 as input and calculate its width (w) and height (h).

Step II: Take the number of shares (n) and minimum number of shares (k) to be taken to reconstruct the image where k must be less than or equal to n. Calculate RECONS = (n-k)+1.

Step III: Create a three dimensional array IMG_SHARE[n][w*h][32] to store the pixels of n number of shares. k-n secret sharing visual cryptographic division is done by the following process.

```
for i = 0 to (w*h-1)
{
  Scan each pixel value of ENCRPT_IMG and     convert it
into 32 bit binary string let PIX_ST.
      for j = 0 to 31
        {      if (PIX_ST.charAt(i) =1){
               call Random_Place(n, RECONS)
               }
               for k = 0 to (RECONS−1)
               {
                  Set IMG_SHARE [RAND[k]][i][j] = 1
               }
         }
}
```

Step IV: Create a one dimensional array IMG_CONS[n] to store constructed pixels of each n number of shares by the following process.

```
for k1 = 0 to (n-1)
{  for k2 = 0 to (w*h-1)
  {  String value= “”
    for k3 = 0 to 31
     {
        value = value+IMG_SHARE[k1][k2][k3]
     }
      Construct alpha, red, green and blue part of each pixel
      by taking consecutive 8 bit substring starting from 0.
      Construct pixel from these part and store it into
      IMG_CONS[k1].[1] [4]
  }
 Generate image from IMG_CONS[k1].[3] [6]
}
```

subroutine int Random_Place(n, RECONS)

```
{  Create an array RAND[RECONS] to store the random
number generated.
 for i = 0 to (recons-1)
   {
    Generate a random number within n, let rand_int.[2] [5]
    if (rand_int is not in RAND[RECONS])
       RAND[i] = rand_int
   }
  return RAND[RECONS]
}
```

5. DECRYPTION PROCESS

The decryption process consists of into two steps. First step is done by human visual system where atleast k number of shares out of n number of shares is superimposed, and the second step is decryption by the key taken in section 3. It is already discussed that human visual system acts as an OR function. For computer generated process; OR function can be used for the case of stacking k number of shares out of n.

A. Stacking k number of shares out of n:

k number of shares out of n is taken as input from user. As the shares are created from a single image in Section 4, each share is of equal height and width as the source image. Bitwise OR operation is performed among pixels of the shares, and final pixel values are stored in an array. This is done by the following algorithm.

Step I: Input the number of shares to be taken (k); height (h) and width (w) of each share.

Step II: Create a two dimensional array SHARE [k][w*h] to store the pixel values of each share.
Create an one dimensional array FINAL[w*h] to store the final pixel values of the image which will be produced by performing bitwise OR operation. The OR operation is done by the following process.

```
for i = 0 to k-1
{
   Input the name of the i th image share to be taken.
   for j = 0 to (w*h-1)
    {
      Scan each pixel value of the i th image share and store
the value in SHARE[i][j].
   }
 }
```

Step III:

```
for i = 0  to (k-1)
 {
   for j = 0 to (w*h - 1)
    {
     FINAL[j] = FINAL[j] | SHARE[i][j];
    }          // [ | is bitwise OR]
}
```

Step IV: Generate image VC_IMAGE from FINAL[w*h].^{3} [8]

B. Decryption using Key:

Step I: The image VC_IMAGE is taken as input.

Step II: The key is taken input.

Step III: Follow Section II for XOR operation

[If A XOR B = C the C XOR B = A]

6. EXPERIMENTAL RESULT

Encryption Process:
Source Image: Lena.png
Source image is

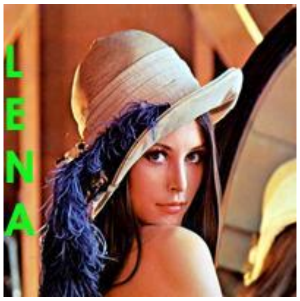


Figure 3: Source Image

Secret Key is: @HALDIA+INSTITUTE

The encrypted image becomes:

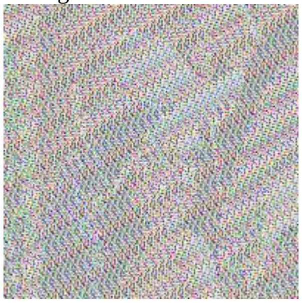


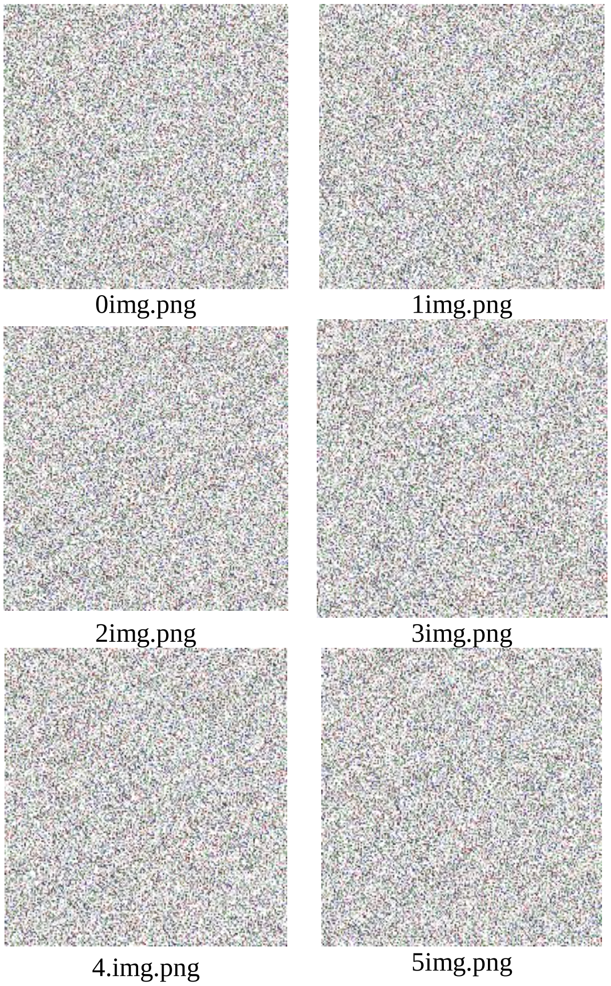
Figure 4: Encrypted Image

k-n Secret Sharing Visual Cryptography:

Number of Shares (n): 6

Numbers of shares to be taken (k): 4

Image shares produced after applying Visual Cryptography are:



0img.png

1img.png

2img.png

3img.png

4.img.png

5img.png

Figure 5: Image shares produced after applying k-n Visual Cryptography

Decryption Process:

Number of shares: 5

Height and Width of each share: 200, 200

Shares inputted: 0img.png, 1img.png, 3img.png, 5img.png

The reconstructed image is:

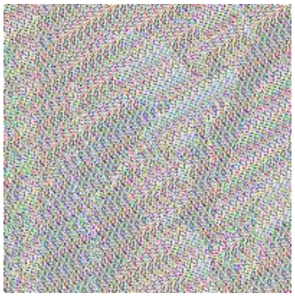


Figure 6: Reconstructed Image

Secret Key is: **@HALDIA+INSTITUTE**

Key is applied on the image produced in Figure 6. The Final image is.

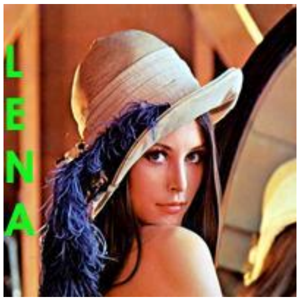


Figure 7: Final Reconstructed Image after applying key

If less than k number of shares suppose 0img.png and 4img.png is stacked with applying right key, then the final image will be distorted. If k number of shares is taken but wrong key let SHYAMALENDU is entered, then the final image also will be distorted. The images obtained in above mentioned two cases are shown below.

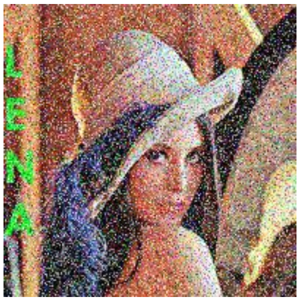


Figure 8.1: Reconstructed image with less than k number of shares but right key

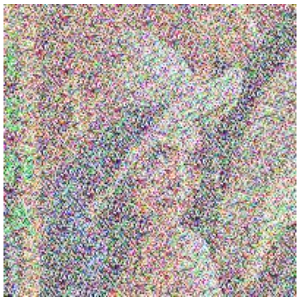


Figure 8.2: Reconstructed image with k number of shares but Wrong key

7. FUTURE SCOPE

Visual Cryptography technique is used to protect image-based secret information. In this proposed scheme we have used a secret key which makes the technique more robust. Visual Cryptography technique makes an illusion to the hacker's mind to protect secret information encoded in an image. Here the shares and the key are sent through different communication channels from sender to receiver so that the probability of getting sufficient shares by the intruder minimizes. In this proposed scheme, the main part is using a key. Key in the text format may arise suspicion to the hacker's mind that some secret information is passed.

If a small image let size w1 X h1 is taken as a key, where w1<w and h1<h; then sending the image with the shares through different channels; will avoid malicious eye.

Steganography can be used by enveloping the secret shares within apparently innocent covers of digital picture. This technique will be more secure from illicit attacks. [7]

8. CONCLUSION

In this paper we have proposed a technique of well known k-n secret sharing on color images using a variable length key with share division using random number. As we know - Decryption part of visual cryptography is based on OR operation, so if a person gets sufficient k number of shares; the image can be easily decrypted. Key adds robustness to the visual cryptography techniques and variable length of the key makes it more secure. At the time of dividing an image into n number of shares we have used random number generator, which is a new technique not available till date. This technique needs very less mathematical calculation compare with other existing techniques of visual cryptography on color images [8][9][10]. This technique only checks '1' at the bit position and divide that '1' into (n-k+1) shares using random numbers. A comparison is made with the proposed scheme with some other schemes to prove the novelty of the scheme.

Table1: Comparison of Existing Techniques with Proposed Scheme

Other Processes	Proposed Scheme
1. k-n secret sharing process is applied directly on original image. [10][11]	1. k-n secret sharing process is applied on encrypted image by key. Key encryption makes the Original image blur.
2. k-n secret sharing process is Complex.[8][9][10]	2. k-n secret sharing process is simple, as random number is used.
3. Decryption is done by OR operation [1][10][11]	3. Decryption is done by OR as well as XOR with the key.

9. REFERENCES

[1]. M. Naor and A. Shamir, "Visual cryptography," Advances in Cryptology-Eurocrypt'94, pp. 1–12, 1995.

[2]. Ranjan Parekh, "Principles of Multimedia", Tata McGraw Hill, 2006

[3]. John F Koegel Buford, Multimedia Systems, Addison Wesley, 2000

[4]. Schildt, H. The Complete Reference Java 2, Fifth Ed. TMH, Pp 799-839

[5]. Krishmoorthy R, Prabhu S, Internet & Java Programming, New Age International, pp 234.

[6]. How to Split an Image into Chunks - Java ImageIO, http://kalanir.blogspot.com, Feb 2010

[7]. Naskar P.,Chaudhuri A, Chaudhuri Atal, Image Secret Sharing using a Novel Secret Sharing Technique with Steganography, IEEE CASCOM 2010, Jadavpur University pp 62-65

[8]. F. Liu1, C.K. Wu1, X.J. Lin, Colour visual cryptography schemes, IET Information Security, July 2008

[9]. Kang InKoo el. at., Color Extended Visual Cryptography using Error Diffusion, IEEE 2010

[10]. SaiChandana B., Anuradha S., A New Visual Cryptography Scheme for Color Images, International Journal of Engineering Science and Technology, Vol 2 (6), 2010

[11]. Li Bai , A Reliable (k,n) Image Secret Sharing Scheme by, IEEE,2006

APPENDIX

[1] Java Language implementation is

```
int c=0;
int a=(Integer.parseInt(value.substring(0,8),2))&0xff;
int r=(Integer.parseInt(value.substring(8,16),2))&0xff;
int g=(Integer.parseInt(value.substring(16,24),2))&0xff;
int b=(Integer.parseInt(value.substring(24,32),2))&0xff;
img_cons[c++]=(a << 24) | (r<< 16) | (g << 8) | b;
```

[2] The source code written in Java is

```
int rand_int = randomGenerator.nextInt(n);
```

[3] Java language implementation for saving an image is

```
public static Image getImageFromArray1(int[ ] source,
int width, int height, int share) throws IOException
{ BufferedImage image = new BufferedImage(width,
height, BufferedImage.TYPE_INT_ARGB);
image.getRaster().setDataElements(0,0,width,height,
source);
Graphics2D gr = image.createGraphics();
gr.drawImage(image,width,height,null);
gr.dispose();
ImageIO.write(image, "png", new File(share+"<file
name>"));
return image; }
```