IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, February 2022
Impact Factor: 6.252

Visual Cryptography with Enveloping by Digital Watermarking

Mayuri Avhad, Smita Chinchole, Muskan Varma, Nikita Shinde
Department of Computer Engineering
Guru Gobind Singh Polytechnic, Nashik, Maharashtra, India

Abstract: *Visual Cryptography is a special type of encryption technique to obscure image-based secret information. This cryptographic system encrypts the secret image by dividing it into n number of shares. In this Cryptographic Scheme for color images where the divided shares are enveloped in other images using invisible digital watermarking. Decryption is done by a certain number of shares to generate the original images and it perform the OR operation.*

Keywords: Visual Cryptography, Digital Watermarking, Random Number

I. INTRODUCTION

Traditional visual cryptography technique which is used to share our information over network by dividing them in n parts and then share the that part our network no extra security added over them. So this is not a secure method so share secret information over network. Hence, to provide security and privacy to the information we propose new visual cryptographic technique called Visual Cryptography with Enveloping by Digital Watermarking. In this current work we have proposed Visual Cryptographic Scheme for color images where the divided shares are enveloped in other images using invisible digital watermarking. The shares are generated using Random Number. So this is secure method to share our information over network.

II. PROCESS

2.1 Encryption

The images are randomly divided into n number of shares and k shares are sufficient to restore the encrypted image. And this process is done using k-n secret sharing algorithm. The images are taken by user as input and k value is also taken by user as an input. k is need to restore the image. The division is done by the following algorithm:

Step I: Take an image IMG 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 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
                      }
           }
}
```

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, February 2022
Impact Factor: 6.252

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] [4].
                }
        Generate image from IMG_CONS [k1]1[8].
}
Subroutine int Random_Place(n, RECONS)
{       Create an array RAND[RECONS] to store the generated random number.
for i = 0 to (recons-1)
        {       Generate a random number within n, let rand_int. [9]
                if (rand_int is not in RAND [RECONS]) RAND [i] = rand_int
        }
        return RAND [RECONS]
}
```

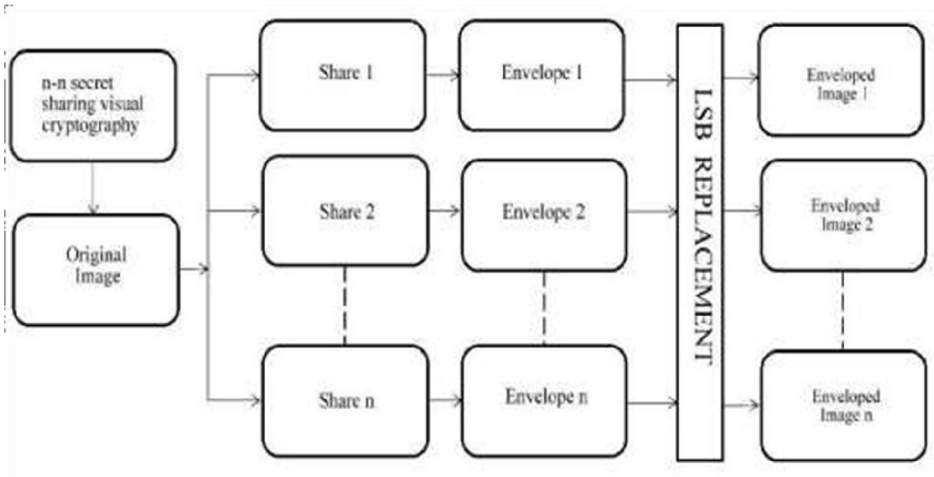

Fig.1 ENCRYPTION PROCESS

2.2 Enveloping

The shares which are divided in encryption by k-n secret algorithm that are enveloping in other images using Least Significant Bit (LSB) replacement digital watermarking. Then this enveloped shares are share over network. This process is nothing but an extended version of invisible digital watermarking technique. This add the security over the information. The enveloping is done using the following algorithm:

Step I: Take number of shares (n) as input. for share = 0 to n-1 follow Step II to Step IV.

Step II: Take the name of the share, let SHARE_NO (NO is from 0 to n-1) and name of the envelope, let ENVELOPE_NO (NO is from 0 to n-1) as input. Let the width and height of each share are w and h. The width of the envelope must be 4 times than that of SHARE_NO.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, February 2022
Impact Factor: 6.252

Step III: Create an array ORG of size w*h*32 to store the binary pixel values of the SHARE_NO 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
                        {       ORG [i*32+j] = PIX.charAt(j)
                        }
        }
```

Create an array ENV of size 4*w*h*32 to store the binary pixel values of the ENVELOPE_NO using the previous loop but from i = 0 to 4*w*h*32 −1.

Step IV: Take a marker M= −1. Using the following process the SHARE_NO is embedded within ENVELOPE_NO.

```
for i = 0 to 4*w*h −1
        {       ENV [i*32+6] = ORG [++M];
                ENV [i*32+7 ] = ORG [++M];
                ENV [i*32+14] = ORG [++M];
                ENV [i*32+15] = ORG [++M];
                ENV [i*32+22] = ORG [++M];
                ENV [i*32+23] = ORG [++M];
                ENV [i*32+30] = ORG [++M];
                ENV [i*32+31] = ORG [++M];
        }
```

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 a one dimensional array let IMG_CONS of size 4*w*h

}

Generate image from IMG_CONS []

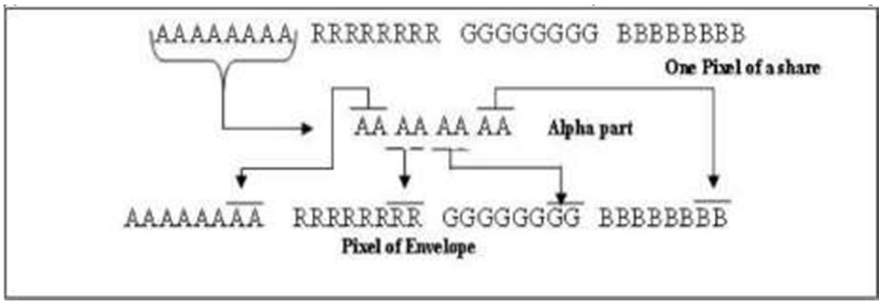

Fig.2 ENVELOPING PROCESS

2.3 Decryption

Using k parts of the enveloped parts which are shares over network the can be restore. On that k parts LSB retrieving with OR operation is done to restore the original image. From each of these images for each pixel, the last two bits of alpha, red, green and blue are retrieved and OR operation is performed to generate the original image.

The decryption process is performed by the following algorithm:

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

Step II: Create a two dimensional array STORE[k][w*h*32] to store the pixel values of k number of enveloped images. Create a one dimensional array FINAL[(w/4)*h*32] to store the final pixel values of the image which will be produced by performing bitwise OR operation of the retrieved LSB of each enveloped images.

Step III:

```
for share_no = 0 to k-1
```

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, February 2022
Impact Factor: 6.252

```
{       Take the name of the enveloped image to be taken and store the pixel values in STORE
        {       [share_no][w*h*32] using the following loop :
                for i = 0 to (w*h-1)
                  { Scan each pixel value of the Enveloped image and convert it into 32 bit binary string let
                  PIX.
                for j = 0 to 31
                  { STORE[share_no][i*32+j] = PIX.charAt(j) }
        }
}
}
```

Step IV: Take a marker M= −1. Using the following process the last two bits of alpha, red, green and blue of each pixel of each k number of enveloped images are OR ed to produce the pixels of the original image.

```
for i = 0 to w*h
{ Consider 8 integer values from C0 to C7 and set all of them to 0.
        for SH_NO = 0 to k-1
{       c0 = c0 | STORE [SH_NO] [i*32+6];
        // | is bitwise OR c1 = c1 | STORE [SH_NO] [i*32+7];
        c2 = c2 | STORE [SH_NO] [i*32+14];
        c3 = c3 | STORE [SH_NO] [i*32+15];
        c4 = c4 | STORE [SH_NO] [i*32+22];
        c5 = c5 | STORE [SH_NO] [i*32+23];
        c6 = c6 | STORE [SH_NO] [i*32+30];
        c7 = c7 | STORE [SH_NO] [i*32+31];
}
FINAL [++M] =c0;
FINAL [++M] = c1;
FINAL [++M] = c2;
FINAL [++M] = c3;
FINAL [++M] = c4;
FINAL [++M] = c5;
FINAL [++M] = c6;
FINAL [++M] = c7;
}
```

Create a one dimensional array IMG_CONS[ ] of size (w/4)*h to store constructed pixels. Construct alpha, red, green and blue part of each pixel by taking consecutive 8 bit substring from FINAL[ ] starting from 0. Construct pixel from these part and store it into IMG_CONS[(w/4)*h] Generate image from IMG_CONS[]

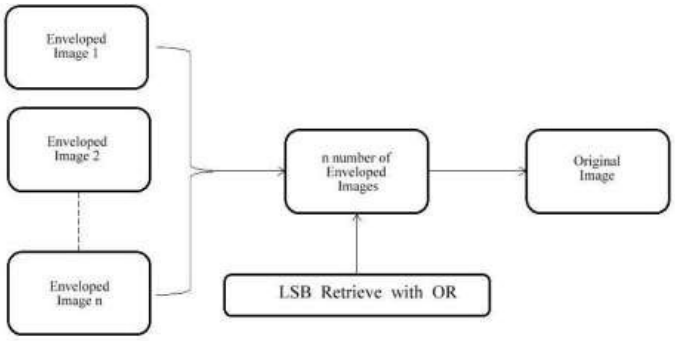

Fig.3 DECRYPTION PROCESS

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 2, February 2022
Impact Factor: 6.252

III. Literature Survey

In 1994, Visual Cryptography scheme have been proposed by Naor and Shamir. In visual cryptography the key image is spilt into 2 shares. The 2 shares area unit stacked along, it produces the first image. This theme is in black and white pictures. In 1996, An extension of Visual Cryptography was proposed by Ateniese, Blundo and Stinson. This scheme contains significant shares. The (2,2)EVC theme projected throughout tis required enlargement of 1 element at interval initial image to four sub pixels which can be chosen to produce the required pictures for each picture.

In 1997, Visual Cryptography first colored scheme was proposed by Verheul and Tilborg[18]. The shares generated this scheme were meaningless. Secret Colored images can be represented with the concept of arcs to build a colored Visual Cryptography scheme. In colorful scheme, there are many pixels. One pixel has changed into n sub pixels. n is considered as sub pixels. And each sub pixels are separated from c color region. There is only one color region each pixel and rest of the color regions are black.

Yang and Laih enhanced the pixel expansion to c *2 of Verheul and Van Tilborg. But both scheme generated shares were meaningless. Thein and Lin [4] proposed a secret image sharing scheme based on shammir-Lagreange technique which is used to share image secretly. Researchers have proposed secret image sharing schemes based on meaningless shares.

IV. Conclusion

In this paper various visual cryptography studied and their performances evaluated on four criteria no of secret images, pixel expansion, image format and type of share generated. The decryption part is based on OR operation. The visual cryptography is best way to transfer images securely in internet and we can secretly share that images or text to other person. At the time image is dividing into n no. of shared and shares are generated using random no. In this technique we need less mathematical calculation.

References

[1]. M. Naor and A. Shamir, "Visual cryptography," Advances in Cryptology-Eurocrypt'94, 1995, pp. 1–12.
[2]. P. Ranjan, "Principles of Multimedia", Tata McGraw Hill, 2006.
[3]. John F Koegel Buford, Multimedia Systems, Addison Wesley, 2000.
[4]. Kandar Shyamalendu, Maiti Arnab, "K-N Secret Sharing Visual Cryptography Scheme For Color Image Using Random Number" International Journal of Engineering Science and Technology, Vol 3, No. 3, 2011, pp.1851-1857.
[5]. Naskar P., Chaudhuri A, Chaudhuri Atal, Image Secret Sharing using a Novel Secret Sharing Technique with Steganography, IEEE CASCOM, Jadavpur University, 2010, pp 62-65.
[6]. Hartung F., Kuttter M., "Multimedia Watermarking Techniques", IEEE, 1999.