ISSN : 2278 – 1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 6, August 2012

Extended Visual Cryptography for Color Shares using Random Number Generators

Savita Patil[1], Jyoti Rao[2]

Assistant Professor, Dept. of CSE, D. Y. Patil Institute of Engineering and Technology, Pimpri, Pune, India[1]

Assistant Professor, Dept. of CSE, D. Y. Patil Institute of Engineering and Technology, Pimpri, Pune, India[2]

ABSTRACT— *Visual* ***cryptography is a special image encryption technique. It is different from traditional cryptography, because it does not need complex computation to decrypt. The benefit of the visual secret sharing scheme is in its decryption process where without any complex cryptographic computation encrypted data is decrypted using Human Visual System (HVS). But the encryption technique proposed here uses a new k-n secret sharing scheme for color image where encryption (Division) of the image is done using Random Number generator and to decrypt the image at least a group of k shares out of n shares reveals the secret information, less of it will reveal no information. The concept of visual information pixel (VIP) synchronization and error diffusion is used to attain a color visual cryptography encryption method that produces meaningful color shares with high visual quality. VIP synchronization retains the positions of pixels carrying visual information of original images throughout the color channels and error diffusion generates shares pleasant to human eyes. Comparisons with previous approaches show the superior performance of the new method.***

Keywords— **Color meaningful shares, digital halftoning, error diffusion, random number, secret sharing, visual cryptography (VC).**

I. INTRODUCTION

Visual Cryptography (VC) is a type of secret sharing scheme introduced by Naor and Shamir. In a k-out-of-n scheme of VC, a secret binary image is cryptographically encoded into shares of random binary patterns. The shares are xeroxed onto transparencies, in order, and distributed amongst participants; one for each participant. No participant knows the share given to another participant. Any or more participants can visually reveal the secret image by superimposing any transparencies together. The secret cannot be decoded by any or fewer participants, even if infinite computational power is available to them. VC scheme proposed by Naor and Shamir serves as a basic model and has been applied to many applications. Aside from the obvious applications to information hiding, there are many applications of VC, which include general access structures, copyright protection, watermarking, visual authentication and identification, print and scan applications, etc.   To illustrate basic principles of VC scheme, consider a simple (2, 2)-VC scheme in Fig.1 Each pixel from a secret binary image is encoded into black and white subpixels in each share. If is a white (black) pixel, one of the six columns is selected randomly with equal probability, replacing. Regardless of the value of the pixel, it is replaced by a set of four subpixels, two of them black and two white.

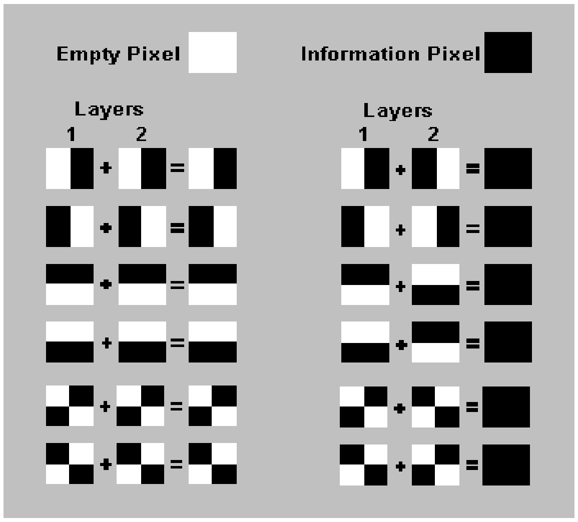

Figure 1. Construction of (2,2) VC scheme

Thus, the subpixel set gives no clue as to the original value of. When two subpixels originating from two white are superimposed, the decrypted subpixels have two white and two black pixels. On the other hand, a decrypted subpixel having four black pixels indicates that the subpixel came from two black pixels.

Fig. 2 shows an example of a simple (2, 2)-VC scheme with a set of subpixels shown in Fig.1 Superimposing these two shares leads to the output secret message as shown in Fig.2. The decoded image is clearly identified, although some contrast loss is observed. Several new methods for VC have been introduced recently in the literature.

Blundo proposed an optimal contrast k-out-of-n scheme to alleviate the contrast loss problem in the reconstructed images.

Ateniese proposed a more general method for VC scheme based upon general access structure. The access structure is a specification of qualified and forbidden subsets of shares. The participants in a qualified subset can recover the secret image while the participants in a forbidden subset cannot.

The VC scheme concept has been extended to grayscale share images rather than binary image shares.

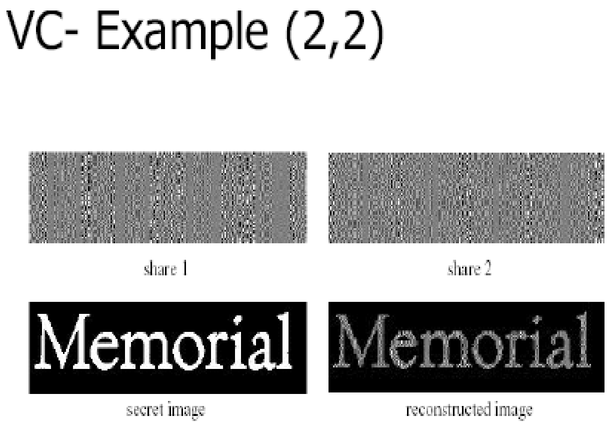

Figure 2 Example of 2-out-of-2 scheme

Blundo proposed VC schemes with general access structures for grayscale share images.

Hou transformed a gray-level image into halftone images and then applied binary VC schemes to generate grayscale shares. Although the secret image is grayscale, shares are still constructed by random binary patterns carrying visual information which may lead to suspicion of secret encryption.

Ateniese developed a method of extended visual cryptography (EVC) in which shares contain not only the secret information but are also meaningful images. Hypergraph colorings are used in constructing meaningful binary shares. Since hypergraph colorings are constructed by random distributed pixels, the resultant binary shares contain strong white noise leading to inadequate results. Wang generalized the Ateniese’s scheme using concatenation of basis matrices and the extended matrices collection to achieve simpler deviation of basis matrices.

Nakajima extended EVC to a scheme with natural grayscale images to improve the image quality.

Zhou et al. used halftoning methods to produce good quality halftone shares in VC.

Fu generated halftone shares that carry visual information by using VC and watermarking methods.

Myodo proposed a method to generate meaningful halftone images using threshold arrays. Wang et. al. produced halftone shares showing meaningful images by using error diffusion techniques. This scheme generates more pleasing halftone shares owing to errors diffused to neighbor pixels. Visual secret sharing for color images was introduced by Naor and Shamir based upon cover semigroups.

Rijimen presented a 2-out-of-2 VC scheme by applying the idea of color mixture. Stacking two transparencies with different colors rises a third mixed color. Hou devised schemes for color shares by applying halftone methods and color decomposition. Hou decomposed the secret color image into three (yellow, magenta and cyan) halftone images. He then devised three colored 2-out-of-2 VC schemes which follow the subtractive model for color mixture by exploiting some of the existing binary VC schemes.

All of the above mentioned methods, however, discuss color schemes for 2-out-of-2 , or 2-out-of- n secret sharing where the reconstructed colors are interpreted by some mixing rules of colors. The general construction of a k-out-of-n VC scheme for the color shares was first introduced by Verheul. He proposed a k-out-of-n VC scheme for a c-colored image with pixel expansion q^{k-1} , where q>=c.

Koga and Yamamoto used a lattice structure to define the mixing result of arbitrary two colors.

All of these VC schemes for color images produce random pattern shares. Even though the decrypted messages show messages with various colors, it is more desirable to generate meaningful shares which are less suspicious of encryption. Other approaches to color VC attempting to generate meaningful color shares include. These methods, however, produce shares with low visibility due to color inconsistency across color channels as discussed in the experiment section. Ching-Nung Yand and Tse-Shih Chen proposed a VCS for color images based upon an additive color mixing method. In this scheme, each pixel is expanded by a factor of three. It is found that this scheme suffers from the problem of pixel expansion in the size of encrypted shares. In order to reduce the size of encrypted shares we propose the VC for color image using visual information pixel (VIP) synchronization with error diffusion technique.

A color VC encryption method which leads to meaningful shares and is free of the previously mentioned limitations is introduced here. The method is simple and efficient. It relies on two fundamental principles used in the generation of shares namely, error diffusion and VIP synchronization.

Error diffusion is a simple but efficient algorithm for image halftone generation. The quantization error at each pixel is filtered and fed to future inputs. The error filter is designed in a way that the low frequency differences between the input and output images are minimized and consequently it produces pleasing halftone images to human vision.

Synchronization of the VIPs across the color channels improves visual contrast of shares. In color VC schemes, the colors of encrypted pixels and the contrast can be degraded due to random matrix permutation. Random matrix permutations are key security features in VC schemes. In grayscale VC schemes, it does not affect the visual quality; however, in color schemes, independent execution of random matrix permutation for each color channel can cause color distortion by placing VIPs at random positions in subpixels which finally degrades the visual quality. VIP synchronization prevents the color and contrast of original images from degradation even with matrix permutation.

The rest of the paper is organized as follows: Section II provides preliminaries about standard VC, the extended VC scheme, and the fundamentals of halftone techniques for easy understanding of the proposed VC method. Section III describes the proposed encryption method using random number generator and then including the VC matrix derivation method to generate final shares. Section IV shows expected experimental results of the new method and comparisons with previous approaches to prove its effectiveness, followed by the conclusion in Section V.

II. Preliminaries

In this section, a brief description of VC, extended VC, color models in VC and an error diffusion quantization is given.

A Fundamentals of VC

Generally, a (k,n)-VC scheme encrypts a secret message into shares to be distributed to n participants. Each share shows noise-like random black and white patterns and does not reveal any information of the secret image by itself. In a k-out of-n scheme, access to more than k shares allows one to recover the secret image by stacking them together, but access to less than k shares is not sufficient for decryption. A black and white (k,n)-VC scheme consists of two collections of nxm binary matrices S_0 and S_1 , having elements denoted by 1 for a black pixel and 0 for a white pixel. To encrypt a white (black) pixel, a dealer randomly chooses one of the matrices in $S_0(S_1)$ and distributes its rows to the n participants. More precisely, a formal definition of the black and white (k,n)-VC scheme is given next.

Definition 1: Let k , n, m and h be nonnegative integers satisfying 2<=k<=n and 0<=h<=m. The two collections of nxm binary matrices (S_0 , S_1) constitute a black and white (k,n) –VC scheme if there exists a value α(>0) satisfying the following.

1) Contrast: for any s € S_0, the “OR” operation of any k out of n rows of s is a vector v, that satisfies w(v) <= h-αm where w(v) is the Hamming weight of the vector v, m is the pixel expansion of the scheme and α is the contrast of the scheme.

2) Contrast: for any s € S_1 , the “OR” operation of any k out of n rows of s is a vector v that satisfies w(v)>=h.

3) Security: for any $i_1 < i_{2,\dots,} < i_t$ in {1,2,…n} with t < k, the two collections of txm matrices D_j, j=0 ,1, obtained by restricting each nxm matrix in S_j, j=0 ,1, to rows $i_{1\ ,}$ $i_{2,\dots,}$ i_t , are indistinguishable in the sense that they contain the same matrices.

In the previously mentioned definitions, the first two contrast conditions ensure that the stacking of k out of n shares can recover the secret image. The security condition ensures that any less than shares cannot get any information of the secret image other than the size of the secret image. That means no matter what the secret message pixel is 0 or 1, the expected appearances of a restricted matrix D_j is same, and i.e., D_0 and D_1 are equal to a column permutation of the other in all possible ways.

Based upon the principle of VC, extended VC has been proposed whose shares take meaningful images rather than random noise-like patterns to avoid suspicion.

B. Extended VC

Generally, a (k,n)-EVC scheme takes a secret image and n original images as input and produces n encrypted shares with approximation of original images that satisfy the following three conditions:

- any k out of n shares can recover the secret image;
- any less than k shares cannot obtain any information of the secret image;
- all the shares are meaningful images; encrypted shares and the recovered secret image are colored.

Denote $S_c^{c_1,c_2,\dots,c_n}$ as the collection of matrices from which the dealer chooses a matrix to encrypt, where $c,c_{1,\dots}c_n$ € {0,1} . For i=1,…n,c_i , is the bit of the pixel on the ^{ith} original image and c is the bit of the secret message. For a black and white (k,n)-EVC scheme, we have to construct 2^n pairs of such collection , one for each possible combination of white and black pixels in the original images. Here we give a definition of the black and white EVC scheme.

Definition 2: A family of 2^n pairs of collection of nxm' binary matrices, constitute a black and white (k,n)-EVC scheme if there exist values , and satisfying the following.

1) Contrast: for any M € $S_0^{c_1,\dots,c_n}$ the "OR" operation of any k out of n rows of M is a vector v that satisfies w(v) >= h.

2) Security: for any $i_1 < i_2 < \dots < i_t$ in {1,2,…n} with t < k , the two collections of matrices ,txm' , obtained by restricting each nxm'matrix in to rows are indistinguishable in the sense that they contain the same matrices.

3) Contrast: after the original images are encrypted they are still meaningful. Formally for Where m' is the pixel expansion of the black and white (k,n) -EVC scheme. α_F and α_S are the contrast of the recovered secret image and the contrast of the shares, respectively. The first and second conditions correspond to the contrast and security conditions of

Definition 1. The third condition implies that after we encrypt the n original images by using 2^n the pairs of collections {$S_0^{c_{1,\dots}c_n}$, $S_1^{c_{1,\dots}c_n}$} , the encrypted shares are still meaningful.

C. Color Models

The additive and subtractive color models are widely used to describe the constitutions of colors. In the additive color model, the three primary colors are red, green, and blue (RGB), with desired colors being obtained by mixing different RGB channels. By controlling the intensity of red, green, blue channels, we can modulate the amount of red, green, blue in the compound light. The more the colors are mixed, the more the brightness of the light. When mixing all red, green and blue channels with equal intensity, white color will result. The computer screen is a good example of the additive color model. In the subtractive model, color is represented by applying the combination of colored- lights reflected from the surface of an object. By mixing cyan, magenta and yellow pigments, we can produce a wide range of colors. The more the pigments are added, the lower the intensity of the light is and, thus, the darker the light is. This is the reason it is called the subtractive model. Cyan, magenta, and yellow are the three primitive colors of pigment which cannot be composed from other colors. The color printer is a typical application of the subtractive model and, hence, the VC model of Naor and Shamir is also of such kind.

A natural color image can be divided into three color channels red, green and blue (cyan, magenta, and yellow, respectively) and each channel constitutes a grey-level image, where each pixel is represented by a 8-bit binary value. Denote $x_{(p,q)} = \{x_{(p,q)1},x_{(p,q)2},x_{(p,q)3}\}$ as the color of a pixel located at the position (p,q) of a color image of size K_1 x K_2 , for p=1,2…K_1 and q= 1,2,…K_2 . Let t describe the color channel and the color component $x_{(p,q)t}$ is coded with 8-b binary value allowing it to be an integer value between 0 and 255. Hence, the color of the pixel can be expressed in a binary form as of such kind.

$$x_{(p,q)} = \sum_{i=1}^{8} x_{(p,q)}^{i} 2^{8-i}$$

where this denotes the binary vector at the ith bit-level with i=1 denoting the most significant bit.

D. Error Diffusion

Error diffusion is a simple yet efficient way to halftone a grayscale image. The quantization error at each pixel is filtered and fed into a set of future inputs. Fig. 3 shows a binary error diffusion diagram where f(m,n) represents the pixel at (m,n) position of the input image,d(m,n) is the sum of the input pixel value and the diffused errors, g(m,n) is the output quantized pixel value. Error diffusion consists of two main components. The first component is the thresholding block where the output is given by g(m,n)=1 if d(m,n) >= t(m,n) else 0.The threshold t(m,n) can be position dependant. The second component is the error filter h(k,l) where the input e(m,n) is the difference between d(m,n)and g(m,n). Finally, we compute d(m,n) as

$$d(m,n) = f(m,n) - \sum_{k,l} h(k,l)e(m-k,n-l)$$

where h(k,l) belongs to H and H is a 2-D error filter. A widely used filter is the error weight originally proposed by Floyd and Steinberg where is the current processing pixel.

$$h(k,l) = \frac{1}{16} \times \begin{bmatrix} & \bullet & 7 \\ 3 & 5 & 1 \end{bmatrix}$$

Where ● is the current processing pixel.

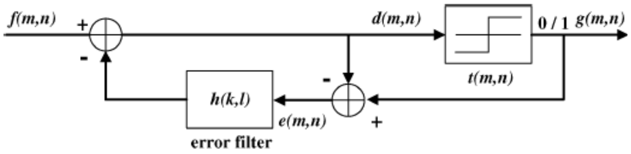

Figure 3 Error diffusion block diagram

The recursive structure of the block diagram indicates that the quantization error depends upon not only the current input and output but also the entire past history. The error filter is designed in such a way that the low frequency difference between the input and output image is minimized. The error that is diffused away by the error filter is high frequency or “blue noise”. These features of error diffusion produce halftone images that are pleasant to human eyes with high visual quality.

III. Color VC Encryption Based Upon Pixel Synchronization and Error Diffusion

In this section, the algorithm for extended color visual cryptography is described.

Step I: Take a secret color image as input.

Step II: Encrypt it into ‘n’ number of shares using Encryption Algorithm.

Step III: Take ‘n’ other meaningful images.

Step IV: Embed individual secret image share into the Meaningful image using VIP synchronization and Error Diffusion Technique.

Step V: Distribute the meaningful images among ‘n’ participants.

Step VI: Take minimum of ‘k’ shares out of ‘n’.

Step VII: XOR them to get the original secret image.

Then encryption, i.e. division of the image into n number of shares such that k numbers of shares are sufficient to reconstruct the image; is done using an algorithm.

Encryption Algorithm

An image is taken as input. The number of shares the image would be divided (n) and number of shares to reconstruct the image (k) is also taken as input from user. The encryption, i.e. division of the image into n number of shares such that k numbers of shares are sufficient to reconstruct the image; is done by the following algorithm.

Step: I: Take an image 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.

k must be less than or equal to n.

Step III: Calculate recons=(n-k)+1.

Step IV: Create a three dimensional array img_share[n][w*h][32] to store the pixels of n number of shares.

Step V:

```
for i=0 to (w*h-1)
{
Scan each pixel value of the image and convert it into32 bit binary string let PIX.
for j=0 to 31
{ if ith position of PIX contains ‘1’
call Random_Place(n, recons)
for k=0 to (recons-1) {
Set img_share[rand[k]][i][j] = 1
}
}
}
```

Step VI: Create a one dimensional array img_cons[n] to store constructed pixels of each share.

Step VII:

```
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].
}
generate image from img_cons[k1].
}
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.
if(rand_int is not in rand[recons])
rand[i] = rand_int.
}
return rand[recons]
}
```

The method for color meaningful shares with a VIP synchronization and error diffusion is described. First, the VC matrix derivation method for VIP synchronization from a set of standard VC matrices is described. Then an error diffusion process to produce the final shares is introduced.

B Matrix Derivation with VIP Synchronization

Our encryption method focuses on VIP synchronization across color channels. VIPs are pixels on the encrypted shares that have color values of the original images, which make the encrypted shares meaningful. In each of the subpixels of the encrypted share, there are number of VIPs, denoted as and the remaining pixels deliver the message information of the secret message image. Thus, in our method, each subpixel carries visual information as well as message information, while for other methods extra pixels are needed in addition to the pixel expansion to produce meaningful shares. Since each VIP is placed at the same bit position in subpixels across the three color channels, VIP represents accurate colors of the original image. These features are further elaborated in the next Chapter.

First, derive the basis matrices from a given set of matrices used in standard VC scheme. Algorithm 1 generates a set of basis matrices where is a bit pixel from the message image and indicate the corresponding pixel bits from the original images. In each row of, there are numbers of and the values are unknown in the matrix derivation stage. Halftoning then defines actual bit values of by referring the pixel values of original images and errors diffused away. The in the algorithm is a hamming weight of a "OR"-ed row vector up to th rows in . It should be noted that the "OR"-ed row vector should not have any s as elements. Since the s are undefined values which can be defined as 0 or 1 in halftone stage, we cannot ensure the contrast difference between matrices $S_0{}^{c}{}_{1,}{}^{\dots c}{}_{n}$ and $S_1{}^{c}{}_{1}{}^{,\dots c}{}_{n}$.

Algorithm 1 Construction of Matrices with VIP Synchronization.

Given the matrices S_0 and S_1 of size nxm , let $S_c[i,j]$ be a jth bit of ith row in S_c ,c€{0,1}.Let γ be the number of 1's in each row of S_c and let λ indicate the number of c_i in each row.

The algorithm produces a set of matrices

```
procedure MATRICES CONSTRUCTION ($S_0$ ,$S_1$ ,$\lambda$)
for i=1,…,n do
  for j=1,…m do
    (a): set count=0
      (b): if $S_0$ [i,j] = $S_1$[i,j] = 0 found, then $S_0$[i,j]=$c_i$ and $S_1$[i,j]=$c_i$ and count =count+1.
     goto (d) if i<k or goto (e) if i>=k.
       (c): if $S_0$ [i,j] = $S_1$[i,j] = 0 is not found, then switch element $S_1$[i,$j_1$] and $S_1$[i,$j_2$] or
     switch element $S_1$[i,$j_1$] and $S_1$[i,$j_2$] and ,goto (b).
        (d): if count = and i<k, then goto (a) with i increased by 1.
    (e): if count = and i>=k, then check if there exists an satisfying:
                W($S_1$[i]) – W($S_0$[i]) >= α.m
            if α exists, goto (a) with increased by 1 until reaches at n.
            if α doest not exists, undo all changes of ith row and goto (c).
end for
end for
end procedure
```

Example 1 : An example with given (2, 2)-VC scheme matrices is follows.

((2, 2)-Color EVC Matrices Derivation): Consider the basis matrices S_0 and S_1 of (2, 2)-VC scheme with m=4,λ=1 such that

$$S_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \quad S_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

Let us assume λ to be 1, and then the example given in the following generate the EVCS matrices VIP synchronized. The first row in each of the matrices and are (1100) and (1100).We begin by inserting the s in the first row of each matrix as and ; the 0s at third position in each row is replaced with . Check the step (d) and go back to step (a) with i=2. For the second rows, the condition is not found. Switch the second and the third bits of by the step (c) leading (0101). for S_1 . The condition is satisfied at third position and replace them with c_2 resulting in ($01c_21$) for S_1 and ($11c_20$) for S_0 by (b). So far, we have matrices as

$$S_1^{c_1c_2} = \begin{bmatrix} 1 & 1 & c_1 & 0 \\ 0 & 1 & c_2 & 1 \end{bmatrix}, \quad S_0^{c_1c_2} = \begin{bmatrix} 1 & 1 & c_1 & 0 \\ 1 & 1 & c_2 & 0 \end{bmatrix}$$

Go to the step (e) and check the condition, however, there is no α satisfying the condition. Un-do the changes of the second row and go back to the step (c). This time let the second bit and the third bit of be switched by (c), leading (1010). Then, we find the second bits of both matrices that meet the condition. Replace them within both matrices by (b), and then we then have matrices as

$$S_1^{c_1c_2} = \begin{bmatrix} 1 & 1 & c_1 & 0 \\ 0 & c_2 & 1 & 1 \end{bmatrix}, \quad S_0^{c_1c_2} = \begin{bmatrix} 1 & 1 & c_1 & 0 \\ 1 & c_2 & 1 & 0 \end{bmatrix}$$

By (e), the “OR”-ed vectors are (1111) for S_1 , (1110) for S_0 and there exists the $\alpha = ¼$ satisfying the contrast difference.

The algorithm guarantees the placement of c_i at the same positions in ith row of S_c and the corresponding ith rows of S_c are used to encrypt an ith share. Furthermore, each row in the matrix is used to encrypt bit 0 and 1 on each color channel of original images, respectively. Thus, each encrypted subpixel has the same VIP positions across three channels, which means that these subpixels carry accurate visual information of the original images. In the example, subpixels on three color channels of the first share have VIPs at the third pixel and those of the second share have VIPs at the second pixel throughout all channels. Consequently, VIP positions are synchronized across channels regardless of pixel colors and this results in high visual quality of the encrypted shares.

Distribution of Matrices across Color Channels

The encryption process starts with basis matrices distribution by referring secret message pixels. The encryption shares should be in a form of 3-b per pixel because they will be the results of the halftoned shares. Furthermore, the secret message of size K_1 x K_2 should be halftoned ahead of the encryption stage as

$$X_{(p,q)} = \left[x^C_{(p,q)}, x^M_{(p,q)}, x^Y_{(p,q)}\right] \in \{0,1\}^3$$

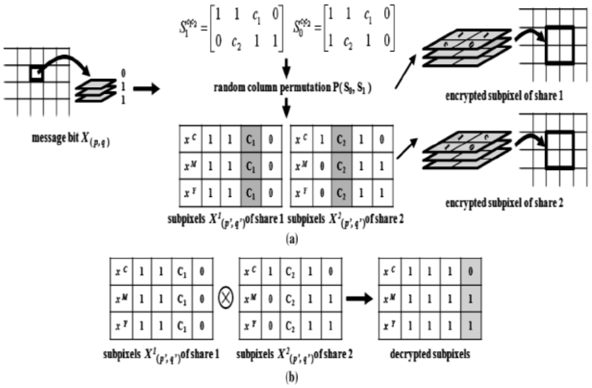
$S_1^{c_1c_2} = \begin{bmatrix} 1 & 1 & c_1 & 0 \\ 0 & c_2 & 1 & 1 \end{bmatrix}$ $S_0^{c_1c_2} = \begin{bmatrix} 1 & 1 & c_1 & 0 \\ 1 & c_2 & 1 & 0 \end{bmatrix}$
0
1
1
random column permutation P(S_0, S_1)
message bit $X_{(p,q)}$
encrypted subpixel of share 1
encrypted subpixel of share 2
subpixels $X^1_{(p',q')}$ of share 1
subpixels $X^2_{(p',q')}$ of share 2
(a)
decrypted subpixels
(b)

Figure 4 General illustration of matrices distribution of (2,2)-color EVC

where $1<=p<=K_1$,$1<=q<=K_2$. $X_{(p,q)}$ is a pixel of the message image at location composed of three binary bits $x^C_{(p,q)}, x^M_{(p,q)}, x^Y_{(p,q)}$ representing values for Cyan, Magenta and Yellow color channels, respectively. Each message pixel composed of 3b is encoded and expanded to subpixels of length m in the encrypted shares as

$$X^i_{(p',q')} = \left[x^C_{(p',q')}, x^M_{(p',q')}, x^Y_{(p',q')}\right]^i \in \left\{S_0^{c_1,\cdots c_n}[i], S_1^{c_1,\cdots c_n}[i]\right\}^3$$

Where

$$\begin{aligned} & 1 \le i \le n \\ & p' = p \cdot m_x - (m_x - 1) \\ & q' = q \cdot m_y - (m_y - 1) \\ & m = m_x \cdot m_y \end{aligned}$$

Where m_x and m_y are nonnegative integers and decide the aspect ratio of encryption shares. The $S_c[i]$ is the ith row of the matrix. Each $X^i_{(p^i,q^i)}$ corresponds to subpixels on three channels starting at that position and each subpixel takes one of the rows in or according to the bit value of the corresponding color channel of the message pixel. A Pseudo-code of the general algorithm for matrices distribution is described in Algorithm 2. This algorithm produces encryption shares. An example of the matrices distribution for (2, 2)-color EVC scheme is depicted in Fig.4. Fig. 4 shows the matrices distribution along with each message pixel. Each binary bit on three color channels of message pixel is expanded into four subpixels on corresponding color channels throughout the encryption shares by taking the matrix or according to its bit value. Since the VIPs are placed at the same spot on the mth row in matrices and, each encrypted subpixels has the VIPs at the same positions throughout the color channels, where colored in gray in the figure. This feature makes the shares carry accurate colors of the original image after encryption. It also depicts a decryption mechanism by the unit of subpixels showing how they present the desired color of the original message pixel. Regardless of the VIP values which will be decided in the error diffusion stage, the decrypted subpixels reveal the color of the message pixel with 1/4 contrast loss. Since the matrices and are derived in a way that the contrast difference is, the decrypted subpixels show the intended color of the message pixel with probability.

Algorithm 2 Matrices Distribution

For the basis matrices and of size nxm , the secret image of size and encryption shares, the algorithm produces matrix distributed shares .

```
1: procedure MATRICES DISTRIBUTION($X, S_0^{c1,\dots cn}, S_1^{c1,\dots cn}$)
2: for $p = 1,\dots K_1$ and $q = 1,\dots K_2$ do
3:     find the starting pixel position on share $X^i$ ,$p^i = p.m_x-(m_x - 1), q^i = q.m_y-(m_y-1)$
4:     conduct random column permutation,$P(S_0^{c1,\dots cn}, S_1^{c1,\dots cn})$
5:     for the color channel C of the secret message, $x_{(p,q)}^C$ do
6:         if the bit $x_{(p,q)}^C = 1$, then
               place $i^{th}$ row of the $S_1^{c1,\dots cn}$ to $[x^C_{(p^i,q^i)}]^i$ of size $m_x$ x $m_y$
               $[x^C_{(p^i,q^i)}]^i$ goes to the channel C of the $i^{th}$ share
7:         else if the bit $x^C_{(p,q)}=0$ , then
               place $i^{th}$ row of the $S_0^{c1,\dots cn}$ to $[x^C_{(p^i,q^i)}]^i$ of size $m_x$ x $m_y$
               $[x^C_{(p^i,q^i)}]^i$ goes to the channel C of the $i^{th}$ share
8:         end if
9:     end for
10:    Repeat 5 to 9 for the channel M and Y.
11: end for
12: end procedure
```

The random permutation for S_0 and S_1 is done independently in standard VC schemes having one color channel. On the contrary, the random permutation of our scheme should be executed for $S_0^{c1,\dots cn}$ and $S_1^{c1,\dots cn}$ at the same time, denoted as P, since each row in the matrices has VIPs and their positions are correlated between S_0 and S_1. This feature should be reflected on the permutation process so as to preserve the VIP structure.

Share Generation via Error Diffusion

Once the distribution of the basis matrices is completed, a halftoning algorithm is applied to produce the final encrypted shares. Error diffusion is used in our scheme as it is simple and effective. The quantization error at each pixel is filtered and fed back to future inputs. Fig. 5(a) shows a binary error diffusion diagram designed for our scheme. To produce the halftone share, each of the three color layers is fed into the input.

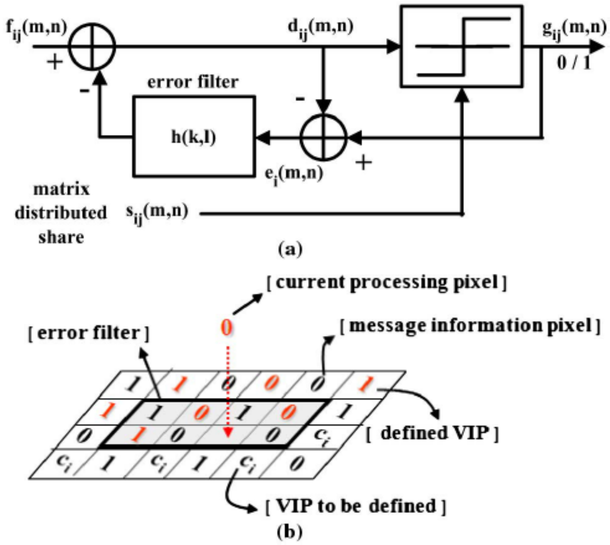

Figure 5.Error diffusion block diagram with share encryption

The process of generating halftone shares via error diffusion is similar to that shown in Fig.5(a) except that is the pixel on the input channel of share. The other difference between our scheme from standard error diffusion is that the message information components, non , are predefined on the input shares such that they are not modified during the halftone process, i.e., the process is applied when the input is c_i. Fig. 5(b) depicts this process. 1s and 0s in black are message information pixels that should not be modified and those are in red are VIPs that are already defined by the error diffusion. They are also VIPs whose values are to be decided by referring the corresponding pixel values of original images and errors from neighboring pixels when the error filter window comes. Non elements, however, still affect and the quantization error when they are calculated in the filter

ISSN : 2278 – 1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 6, August 2012

window. The non elements may increase quantization errors added to the shares, but in turn, these errors are diffused away to neighboring pixels.

The visual quality of shares via error diffusion can be improved through edge enhancement methods. The measure of a particular halftoning algorithm is its performance in DCregions and its performance near edges or in areas of high frequency image content can be manipulated through prefiltering the image prior to halftoning. So the remedy for the apparent blurring of edges caused by the error diffusion algorithm is to apply an edge sharpening filter prior to halftoning such that

$$X^i_{sharp}[n] = X[n] - \beta(\psi[n] * X[n])$$

where stands for the original image, is a digital Laplacian filter, denotes convolution and is a scalar constant regulating the amount of sharpening with larger leading to a sharper image . Consequently, error diffusion produces high quality halftone images. The effectiveness of error diffusion can be confirmed in the simulation result section.

IV. Expected Experimental Results

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

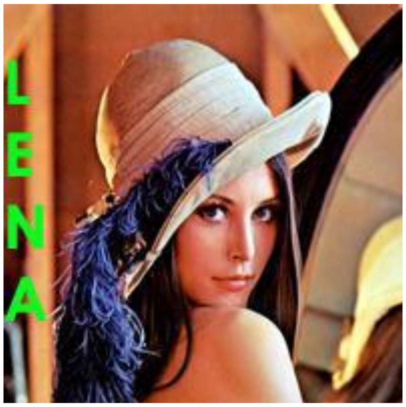

Figure 6. Source Image

Number of Shares: 6
Numbers of shares to be taken: 5

The experimental result after encryption by the encryption algorithm is given below.

0img.png

1img.png

2img.png

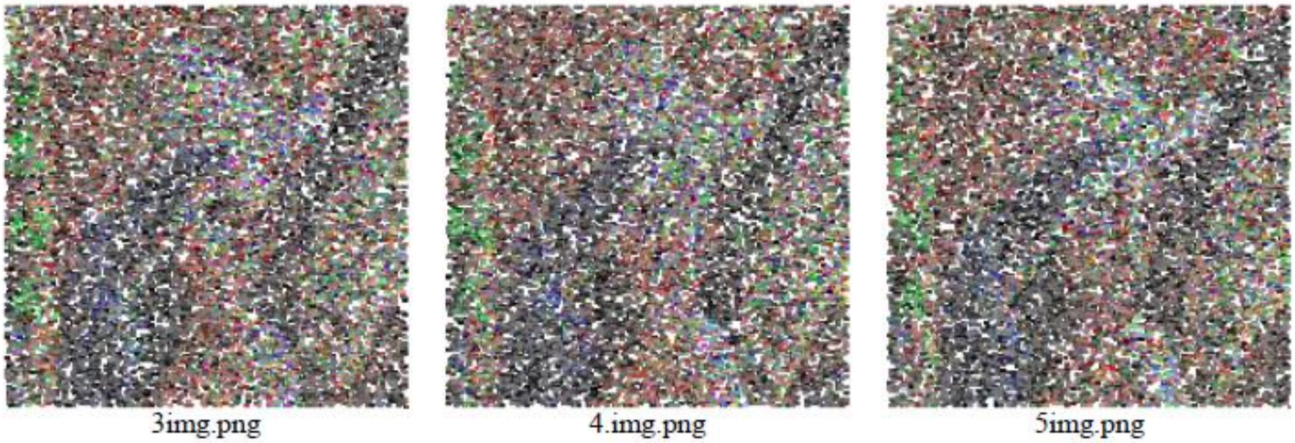

Figure 7. Encrypted Shares

Decryption Process:

Number of shares: 5

Height and Width of each share: 200, 200

Shares inputted

0img.png, 1img.png, 3img.png, 4img.png, 5img.png

Final image reconstructed:

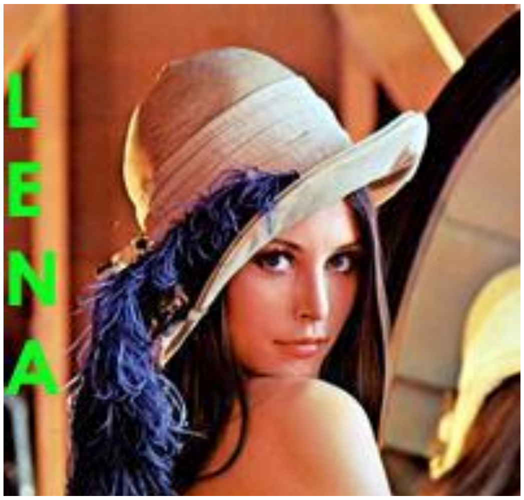

Figure 8. Reconstructed Image

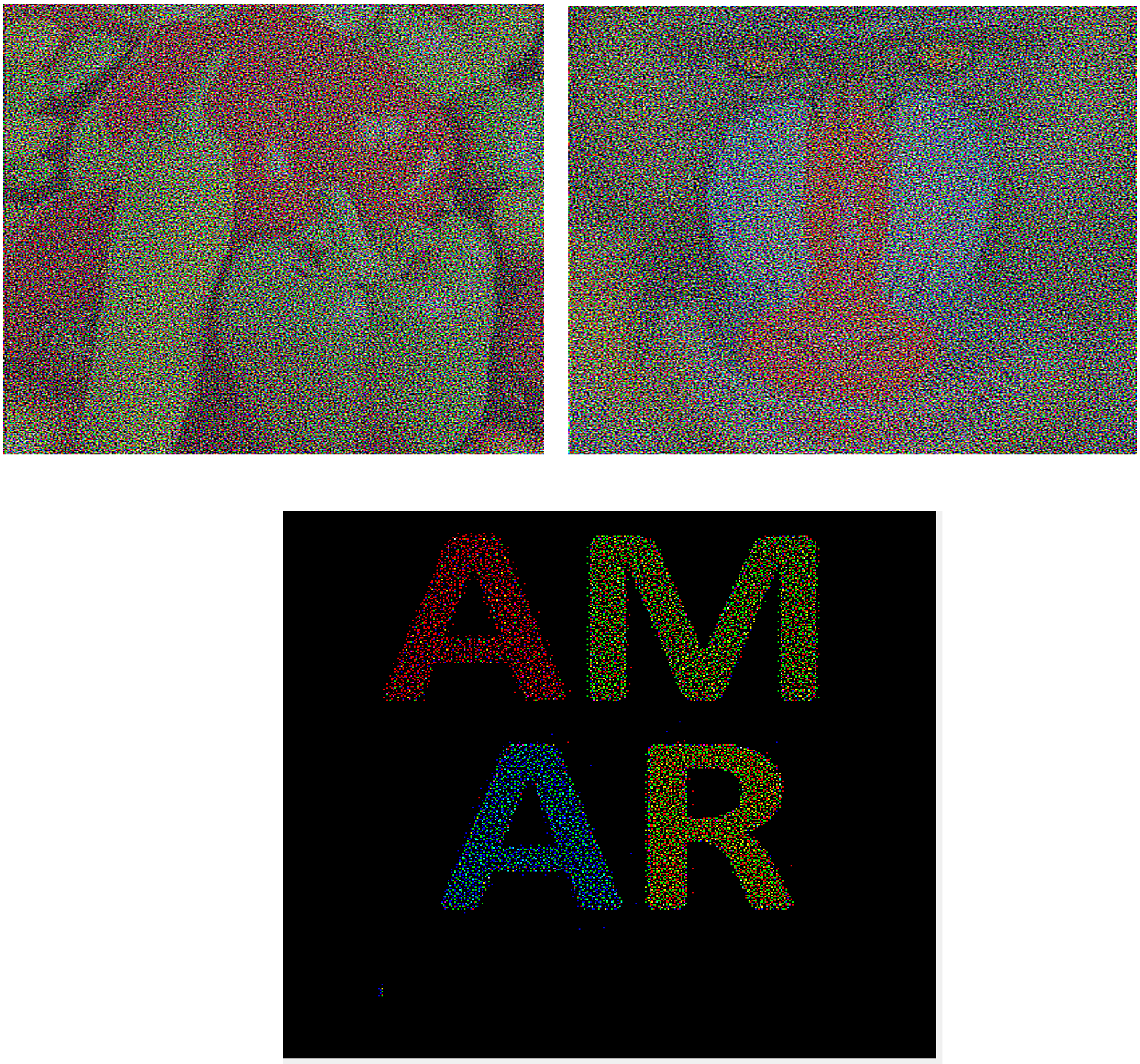

Figure 9. Encrypted Shares and Decoded Secret Image

V. CONCLUSION

An encryption method to construct color EVC scheme with VIP synchronization and error diffusion for visual quality improvement. VIPs synchronize the positions of pixels that carry visual information of original images across the color channels so as to retain the original pixel values the same before and after encryption. The VIPs are pixels that carry pixel values of original images to make shares meaningful. When these VIPs are not assigned during the halftone stage, the resultant shares are the same as that of standard VC schemes except the colorful decrypted messages. Other schemes deal with EVC schemes in color, however, they do not consider relationship throughout color channels. Unlike standard EVCS, the robustness of our proposed scheme to cheating comes from that fact that it is impossible to differentiate VIPs and other pixels in the encrypted shares and it is hard to know the actual VIP values which were decided during the error diffusion. Error diffusion is used to construct the shares such that the noise introduced by the preset pixels are diffused away to neighbors when encrypted shares are generated. It is obvious that there is a tradeoff between contrast of encryption shares and the decryption share, however, we can recognize the colorful secret messages having even low contrast. Either VIP synchronization or error diffusion can be broadly used in many VC schemes for color images

ACKNOWLEDGMENT

I express my gratitude towards Prof. Jyoti Rao, of Department of Computer Engineering, Padmashree Dr. D. Y. Patil Institute of Engineering and Technology, Pimpri, Pune who guided and encouraged me.

No words are sufficient to express my gratitude to Prof. Pramod Patil and Prof. Rajesh Bharati for their unwavering encouragement.

REFERENCES

[1] InKoo Kang, *Member, IEEE,* Gonzalo R. Arce, *Fellow, IEEE,* and Heung-Kyu Lee, *Member, IEEE,* "Color Extended Visual Cryptography Using Error Diffusion", IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 20, NO. 1, JANUARY 2011.

[2] Shyamalendu Kandar , Arnab Maiti," K-N secret sharing visual cryptography scheme for color image using Random number",vol 3,no.3,Mar 2011.

[3] M. Naor and A. Shamir, "Visual cryptography," in Proc. EUROCRYPT, 1994, pp. 1–12.

[4] G. Ateniese, C. Blundo, A. D. Santis, and D. R. Stinson, "Visual cryptography for general access structures," Inf. Comput., vol. 129, no. 2, pp. 86–106, 1996.

[5] A. Houmansadr and S. Ghaemmaghami, "A novel video watermarking method using visual cryptography," in Proc. IEEE Int. Conf. Eng. Intell. Syst., 2006, pp. 1–5.

[6] M. S. Fu and O. C. Au, "Joint visual cryptography and watermarking," in Proc. IEEE Int. Conf. Multimedia Expo, 2004, pp. 975–978.

[7] C. S. Hsu and Y. C. Hou, "Copyright protection scheme for digital images using visual cryptography and sampling methods," Opt. Eng., vol. 44, p. 077003, 2005.

[8] M. Naor and B. Pinkas, "Visual authentication and identification," Adv.Cryptol., vol. 1294, pp. 322–336, 1997.

[9] W. Q. Y, J. Duo, and M. Kankanhalli, "Visual cryptography for print and scan applications," in Proc. IEEE Int. Symp. Circuits Syst., 2004, pp. 572–575.Fröhlich, B. and Plate, J. 2000. The cubic mouse: a new device for three-dimensional input. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems

[10] C. Blundo, P. D'Arco, A. D. S. , and D. R. Stinson, "Contrast optimal threshold visual cryptography schemes," SIAM J. Discrete Math., vol.16, no. 2, pp. 224–261, 2003.

[11] L. A. MacPherson, "Gray level visual cryptography for general access structrue," M. Eng. thesis, Univ. Waterloo, Ontario, Canada, 2000.

[12] C. Blundo, A. D. Santis, and M. Naor, "Visual cryptography for grey level images," Inf. Process. Lett., vol. 75, no. 6, pp. 255–259, 2000.

[13] Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd Y. T. Hsu and L. W. Chang, "A new construction algorithm of visual crytography for gray level images," in Proc. IEEE Int. Symp. Circuits Syst., 2006, pp. 1430–1433.