GES: A Group based Encoding of Shares for Visual Cryptography without Pixel Expansion

Mradula Sharma
JECRC University, Jaipur

Kunal Sain
MNNIT, Allahabad

Suneeta Agarwal
MNNIT, Allahabad

ABSTRACT

Pixel Expansion has been one of the problems of Visual Cryptography that is yet to be properly addressed. Existing methods to deal with this problem either have security vulnerabilities and/or produce results of poor quality. In this paper we propose a grouping based approach to encoding shares in Visual Cryptography without pixel expansion. In our approach we try to find groups of $\gamma_x \times \gamma_y$ (where γ_x and γ_y are number of sub-pixels along width and height respectively in the pixel expansion structure of traditional visual cryptography) of the same type where ever possible to encode them. Pixels that do not fall into such group are collected and encoded separately. In our technique, we are able to avoid security vulnerabilities (i.e. shares showing patterns resembling secret image) present in existing techniques of pixel expansion free visual cryptography. Also the resultant image produced by overlapping the shares are of much better visual quality compared to existing schemes.

General Terms:

Secret Sharing, Security

Keywords:

Visual Cryptography, Pixel Expansion, Group based Encoding of Shares

1. INTRODUCTION

Visual Cryptography was first proposed by Shamir in 1997[5]. In this technique message is encoded in the form of shares and decoded by human visual system. So it does not require any cryptographic computation to decode the secret image. The encoded shares are distributed to different users. User can not obtain any information from his share alone each because share shows only random noise. This is a simple and perfectly secure technique for secret sharing.

In [5] Naor Shamir presented k out-of-n threshold visual cryptography scheme where in-order to reveal the secret image, at least k (which is threshold) number of shares, out of the n shares have to be stacked together (by printing the share images onto transparencies). No information about the secret image can be revealed by stacking any i number of shares,where $i \leq (k-1)$. That means subset is qualified if and only if it consists of at least k participants. Visual cryptography was extended to grey scale images in [1] [4]. In [1] author analysed Visual Cryptography with general access structure for grey level images whose pixels have g grey levels ranging from 0 to g-1. In [4] author suggested Visual Cryptography for grey level images by dithering techniques instead of taking grey levels or grey sub pixels. A dithering technique is used first to convert a grey level image into an approximate binary image then applied existing visual cryptography to create the shares. In [2] the author transformed a gray-level image into a half-tone image and then applied simple binary visual cryptography scheme to generate a grey-level visual cryptography scheme. In [2] the author also proposed visual cryptographic schemes for color images where, before encoding shares using visual cryptography, images are decomposed into three channels, i.e., yellow, magenta and cyan by applying color decomposition, followed by half-toning.

Visual cryptography has two parameters one is contrast β, another one pixel expansion γ. Contrast is difference of stacked sub pixels representing a white or a black pixel, also called the relative difference. Pixel expansion is number of sub pixel which is required to encode a black or white pixel. In a visual cryptographic scheme, if one pixel in the secret image is encoded to γ sub-pixels in the shares, then γ is the pixel expansion, and the shares are γ times as large as the secret image. So, ideally, we require pixel expansion to be as small as possible while making the relative difference (contrast) as large as possible. In this paper we propose a scheme of encoding shares for visual cryptography without pixel expansion.

In [6] author proposed Visual cryptography with out pixel expansion. Their scheme encodes a single pixel based upon the probability of it being a black or white sub-pixel. The drawback of this scheme is that the quality of stacked image is very poor. In Hou et al.[3] author proposed Multi pixel encoding method for visual cryptography without pixel expansion. In their technique visual quality of stacked image is good when compared to Ito et al [6] but also have some shortcomings. Their scheme encodes γ successive white or black pixels but they did not mention about the case when the number of successive pixels of same type is less than γ. Their scheme also has a security issue. The shares created by their scheme show some patterns resembling those in the secret image. So the secret message may be recognized or guessed by viewing a single share. In Zhang et al.[7] Haibo zhang proposed pixel-block aware encoding method, in which they applied zig-zag scan mode and in a single run collects and encodes consecutive pixels of same type, till meeting a pixel of different type. They are encoding variable number of consecutive pixels of same type in each run. The drawback of this scheme is that quality of stacked image quality is poor (even compared to Hou et al.).

In this paper, we propose group based scheme for encoding shares for Visual cryptography without pixel expansion. The rest of the paper is organised as follows: Section 2 explains our approach in details. In Section 3 we compare the results obtained by our proposed approach to two of the existing state-of-the-art methods for pixel expansion free visual cryptography while Section 4 concludes the paper.

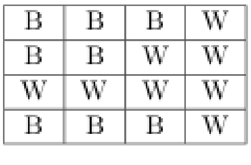
2. PROPOSED METHOD

In k-out-of-n visual cryptography, there are two collections of $n\times\gamma$ boolean matrices C_0/C_1. C_0/C_1 are formed by all possible column permutations on the basis matrices for encoding white/ black pixels respectively. In order to encoding a white pixel, one of the matrices in C_0 is randomly chosen; whereas to encode a black pixel, one of the matrices in C_1 is randomly chosen. In our approach we use a linear scanning technique to scan the pixels in the secret image. We look at an area of $\gamma_x \times \gamma_y$ pixels starting from the current pixel and check if all the pixels in the area are of the same type. If they are, we encode them in one go using one of the matrices in C_0/C_1. If they are not, we take the current pixel and store its position in list$_{(0)}$/list$_{(1)}$. If the size of either list$_{(0)}$ or list$_{(1)}$ reaches γ, we use a type of dispersion technique to encode the pixel into the shares. For this we pick one of the shares. In this share, we check the density of white pixels around the positions of the pixels in list$_{(0)}$/list$_{(1)}$. The positions are sorted based on the density of white pixel around them. Then we start by filling black sub-pixels into positions that have higher density of white around the pixel position that is to be filled. The rest of the shares are filled based on the filling of sub-pixels in the share under consideration. At the end of the scanning, if less than γ pixels are left in list$_{(0)}$ or list$_{(1)}$, these pixels are filled by picking one of the matrices in C_0/C_1 and truncating it to the size of the list.

Given below is our approach in algorithmic form. Let **n** be the number of shares and γ be the number of columns in the basis matrices, γ = $\gamma_x \times \gamma_y$ where γ_x = no. of sub-pixels along x-axis, γ_y = no. of sub-pixel expansion along y-axis. **Img** = secret image, **r×c** = size of the secret image, and **list$_{(0)}$/list$_{(1)}$** are the arrays to store the location of black or white pixel which is not in the group of $\gamma_x \times \gamma_y$. **getWhiteDensity(A , i , j)** returns the density of white pixels in a 3×3 neighbourhood around pixel (i , j) in the image A. **REVERSE_SORT(B)** return the positions of elements of array B if we were to sort the array B in reverse order. M_0/M_1 are the basis matrices for encoding a white/black pixel respectively. The matrices M_0/M_1 are arranged in such a manner that the columns containing **'1'** in the first row of M_0/M_1, are at the begining, while the columns containing **'0'** in the first row are at the end.

2.1 How algorithm works :

We are taking an example of 2 out of 2 Visual Cryptography to show the working of our algorithm step by step. Figure 1 shows the sample input image containing black and white pixels that we will use to demonstrate the working of our proposed algorithm. Our algorithm scans the image to check if image contain γ same type of pixels in a group form of $\gamma_x \times \gamma_y$. (Here γ = 4 i.e. ($\gamma_x \times \gamma_y$), γ_x = 2, and γ_y = 2) then encode the white/black pixels using basis matrices (one of the matrices in C_0/C_1 is randomly selected for the white/black pixels). Otherwise store the location of the pixel in the array PoS$_w$ or PoS$_b$ for white and black pixel respectively. if length of the array = γ then sort the pixel position based on density of white sub-pixel around the pixel. Assign black sub-pixels to positions with high white sub-pixel density and vice-versa.



B	B	B	W
B	B	W	W
W	W	W	W
B	B	B	W

Fig. 1: Sample binary secret image

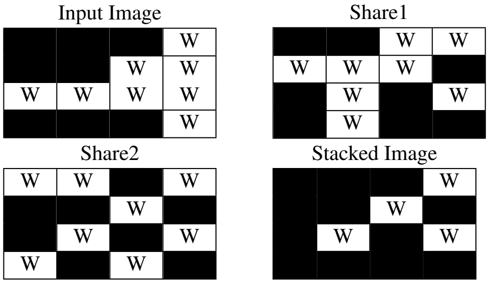
let M_0 and M_1 be two $n \times \gamma$ basis matrices for encoding white and black pixel.(Here n=2 γ = 4 and 1 → black , 0 → white)

$$M_0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}$$

$$M_1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$C_0 = \{$ *all the matrices obtained by permuting the column of* $M_0\}$,
$C_1 = \{$ *all the matrices obtained by permuting the column of* $M_1\}$

Figure 2 shows the how our algorithm encodes the sample binary secret image shown in figure 1. The final result is :



Input Image

■	■	■	W
■	■	W	W
W	W	W	W
■	■	■	W

Share1

■	■	W	W
W	W	W	■
■	W	■	W
■	W	■	■

Share2

W	W	■	W
■	■	W	■
■	W	■	W
W	■	W	■

Stacked Image

■	■	■	W
■	■	W	■
■	W	■	W
■	■	■	■

As can be seen from above example stacked image showing 50% loss in contrast, same as 2-out-of-2 traditional visual cryptography. Each group of black pixels gets encoded into 100% black. Each group of white pixels gets encoded into 50% black and 50% white. Secret image contains number of black pixels = 8 and number of white pixels = 8. When shares are overlapped then number of black pixels of the stacked image would be 12 and number of white pixels of the stacked image would be 4. So contrast would be 50% loss in the stacked image. Our algorithm is satisfy the contrast and security condition of visual cryptography [5]. Our algorithm is applicable for k out of n visual cryptography. Our algorithm is also applicable for grey-level and color images. Dithering technique is applied in the grey-level images while color decomposition into C(yan), M(agenta) and Y(ellow) follow dithering method is applied in the color images. Then using our algorithm we can create the shares. Figure 3 shows the experimental result of color image using our method.

3. EXPERIMENTAL RESULTS

In this section we demonstrate the effectiveness of our approach with experimental results and discuss the performance of our proposed algorithm by comparing the results of our method with those obtained for two existing state-of-the-art methods Hou et al.[3] and Zhang et al. [7]. Figure 4 shows the images that we have taken for the performance analysisi of our proposed method. All grayscale images taken for our test purpose are first half-toned to convert then into binary images.

Algorithm 1 The algorithm for encoding Visual Cryptographic Shares using our proposed approach

```
rndP ← random_permutation([1...n])
ind ← 1
for $i = 1, 2, ..., r$ do
    for $j = 1, 2, ..., c$ do
        $p = Img_{(i,j)}$
        if $(Img_{(i+x,j+y)} == p \qquad \{\forall x \in [0, \gamma_x], \forall y \in [0, \gamma_y]\})$ then
            if (p == 0) then
                Create shares by choosing a Matrix from $C_0$.
            else
                Create shares by choosing a Matrix from $C_1$.
            end if
        else
            list$_{(p)}$ ← list$_{(p)}$ $\cup \{i, j\}$
            if (length(list$_{(p)}$) $== \gamma$) then
                for $k = 1, 2, ..., \gamma$ do
                    denW$_{(k)}$ ← getWhiteDensity(SHARE[rndP(ind)] , list$_{(p)}$[k].$i$ , list$_{(p)}$[k].$j$ )
                end for
                posP ← REVERSE_SORT(denW)
                indx ← ind
                for $m = 1, 2, ..., n$ do
                    for $w = 1, 2, ..., \gamma$ do
                        SHARE[rndP(indx)]$_{(\text{list}_{(p)}[\text{posP(w)}].i \text{ , list}_{(p)}[\text{posP(w)}].j)}$ ← $M_p(m, w)$
                    end for
                    indx ← indx + 1
                    if indx $> n$ then
                        indx ← 1
                    end if
                end for
                list$_{(p)}$ ← $\phi$
                ind ← ind + 1
                if (ind > n) then
                    ind ← 1
                end if
            end if
        end if
    end for
end for
```

As can be seen from figure 5a and 5b, the first as well as second shares generated by Hou et al. show some structures or patterns that have very close resemblance with the secret image. It is a security flaw which is un-acceptable in visual cryptography because one can obtain an idea of the secret image from a single share only. No such security flaw is observed in the resultant images produced by Zhang et al.(figure 6) or our proposed method (figure 7). However the quality of the resultant image produced by our proposed method (figure 7) is visually better compared to that produced by Zhang et al.(figure 6).

As can be seen from figures 8, 9, 10, 11, 12, the resultant images (created by stacking of the shares) produced by our proposed algorithm is visually better looking and clearer when compared those produced by Zhang et al. as well as Hou et al.. In figure 8d (the resultant image produced by our proposed method), the suspension cables, cars and the railing in the middle of the bridge is far clearly visible when compared to figure 8b and figure 8c. In figure 10, the writing within the logo is readable in the stacked image produced by our proposed method. The stacked images produced by our proposed method shown in figures 9, 11 and 12 are significantly clearer and visually better looking compared to those produced by the two other state-of-the-art methods Hou et al. and Zhang et al.

3.1 Analysis of Resultant Images

In order to analyse the quality of the resultant images produced by stacking of shares, we divide the resultant image into blocks of 8×8. For each block we count the number of black/white pixels that are present in the resultant image and calculate the number of black/white pixels that should ideally be present in the resultant image. We calculate the standard deviation between these two counts for each resultant image. The ideal numbar of black/white pixels per block are calculated using the equations 1 and 2.

$$IBc_i = nB_i \times nbB + nW_i \times nbW \tag{1}$$

$$IWc_i = nB_i \times nwB + nW_i \times nwW \tag{2}$$

where nB_i = Number of black pixels in i^{th} block of secret image, nW_i = Number of white pixel in i^{th} block of secret image, nbB = number of black sub-pixels per black pixel in the resultant image, nbW = number of black sub-pixels per white pixel in the resultant

	Hou et al. [3]		Zhang et al. [7]		PROPOSED METHOD	
	Deviation of Black (B_{div})	Deviation of White (W_{div})	Deviation of Black (B_{div})	Deviation of White (W_{div})	Deviation of Black (B_{div})	Deviation of White (W_{div})
SecretImage1(E)	1.921	1.921	2.361	2.361	1.232	1.232
SecretImage2(Bridge)	2.205	2.205	2.376	2.376	1.651	1.651
SecretImage3(Lena)	2.105	2.105	2.098	2.098	1.473	1.473
SecretImage4(Baboon)	2.265	2.265	2.324	2.324	1.537	1.537
SecretImage5(F16)	2.235	2.235	2.471	2.471	1.665	1.665
SecretImage6(Logo)	2.031	2.031	2.335	2.335	1.483	1.483

Table 1. : Analysis of Resultant images using 8×8 Blocks

image, nwB = number of white sub-pixels per black pixel in the resultant image, nwW = number of white sub-pixels per white pixel in the resultant image. Thus IBc_i gives the number of black sub-pixels that should ideally be present in the i^{th} block of the resultant image, IWc_i gives the number of white sub-pixels that should ideally be present in the i^{th} block of the resultant image.
The standard deviation for black/white pixels in the resultant images are calculated using the equations 3 and 4

$$B_{div} = \sqrt{\frac{\sum_1^N (IBc_i - Bc_i)^2}{N_{Blocks}}} \tag{3}$$

$$W_{div} = \sqrt{\frac{\sum_1^N (IWc_i - Wc_i)^2}{N_{Blocks}}} \tag{4}$$

where B_{div} = deviation of black pixel, W_{div} = deviation of white pixel, Bc_i = Number of black pixels in i^{th} block of the resultant image, Wc_i = Number of white pixels in i^{th} block of the resultant image, N_{Blocks} = Number of Blocks.

As can be seen from the table 1, our resultant images are closer to the ideal results (i.e. standard deviation 0) compared to the two state-of-art algorithms.

4. CONCLUSION

In this paper we have proposed a novel scheme for encoding shares for visual cryptography without pixel expansion. We have also shown that our scheme performs better than two of the state of the art scheme for encoding shares for visual cryptography without pixel expansion. Our scheme does not have the security flaw that is observed in Hou et al.[3]. We have also shown that the resultant images produced by our images are of much better quality (visually) compared to the shares produced by both the existing schemes considered. Using standard deviation we have shown that our resultant images are closer to the ideal results than the two state of that art schemes considered. We have also shown that our scheme is applicable for grayscale images as well as chromatic images.

5. REFERENCES

[1] Carlo Blundo, Alfredo De Santis, and Moni Naor. Visual cryptography for grey level images. *Information Processing Letters*, 75(6):255 – 259, 2000.

[2] Young-Chang Hou. Visual cryptography for color images. *Pattern Recognition*, 36(7):1619 – 1629, 2003.

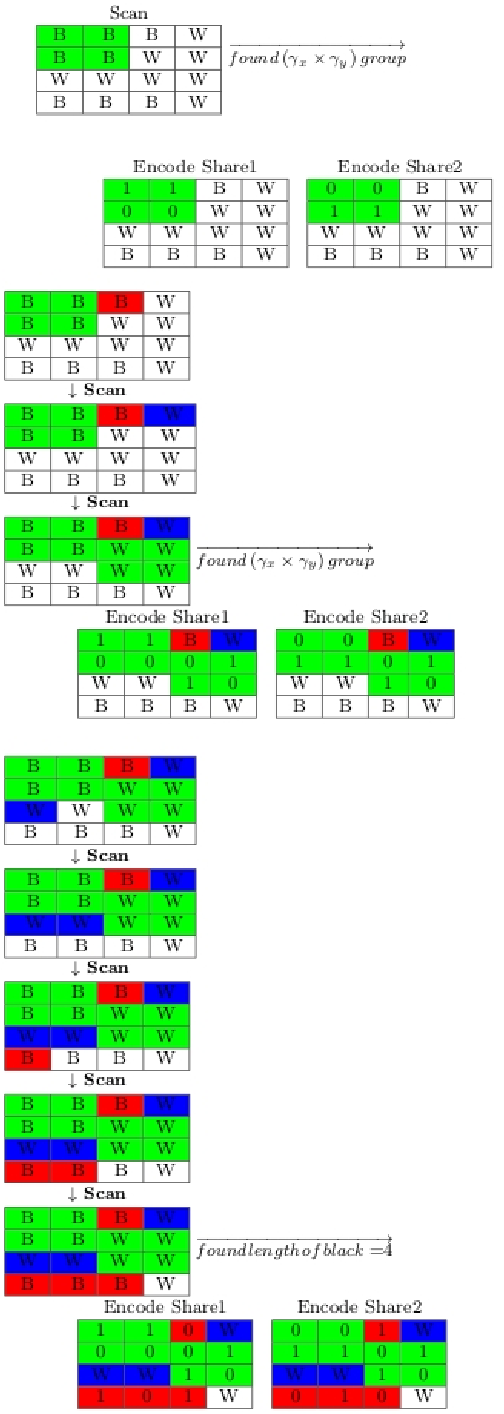
[3] Young-Chang Hou and Shu-Fen Tu. A visual cryptographic technique for chromatic images using multi-pixel encoding method. *Journal of Research and Practice in Information*, 37(2):179–191, 2005.

[4] Chang-Chou Lin and Wen-Hsiang Tsai. Visual cryptography for gray-level images by dithering techniques. *Pattern Recognition Letters*, 24(13):349 – 358, 2003.

[5] Moni Naor and Adi Shamir. Visual cryptography. 950:1–12, 1995. 10.1007/BFb0053419.

[6] ITO Ryo, KUWAKADO Hidenori, and TANAKA Hatsukazu. Image size invariant visual cryptography (special section on information theory and its applications). *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 82(10):2172–2177, 1999-10-25.

[7] Haibo Zhang, Xiaofei Wang, Wanhua Cao, and Youpeng Huang. Visual cryptography for general access structure using pixel-block aware encoding. *Journal of Computers,(JCP, ISSN 1796-203X)*, 3(12):68–75, 2008.



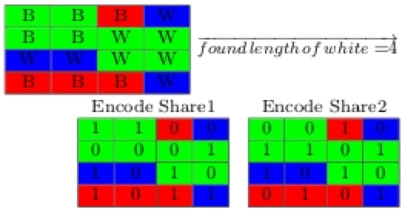
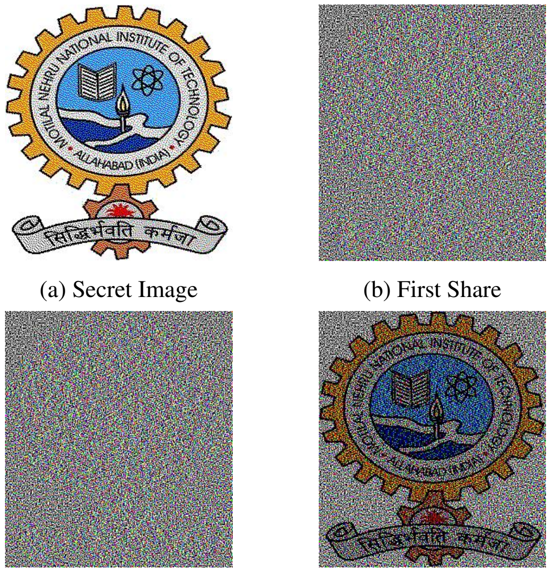


Fig. 2: A small example showing the working of our algorithm

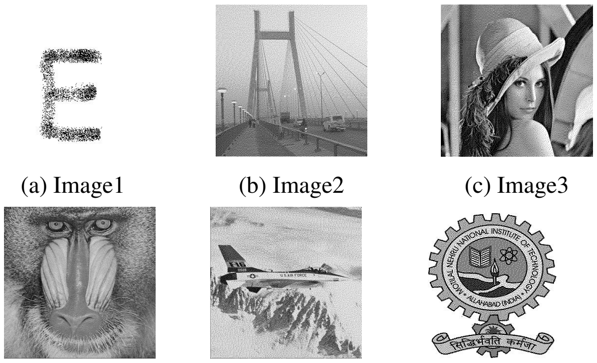

(a) Secret Image

(b) First Share

(c) Second Share

(d) Resultant Image

Fig. 3: Using Proposed Scheme


(a) Image1

(b) Image2

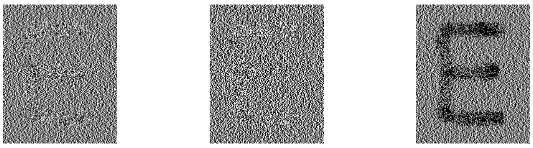
(c) Image3

(d) Image4

(e) Image5

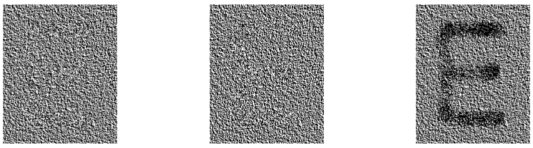
(f) Image6

Fig. 4: Our Test Images



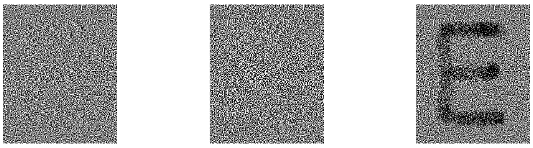
(a) 1st Share (b) 2nd Share (c) Result Image

Fig. 5: Using MPEM Scheme



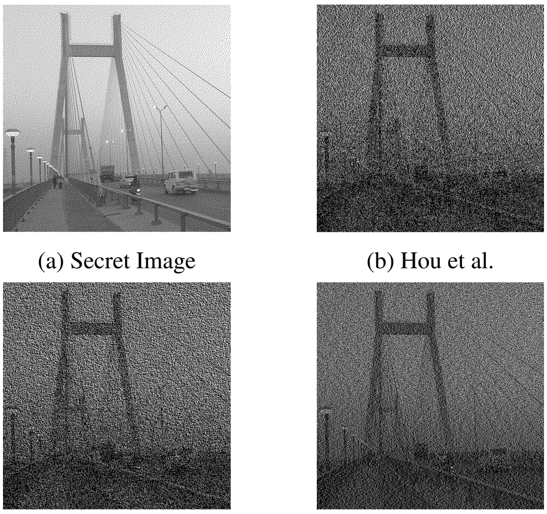
(a) 1st Share (b) 2nd Share (c) Result Image

Fig. 6: Using ZIG-ZAG Scheme



(a) 1st Share (b) 2nd Share (c) Result Image

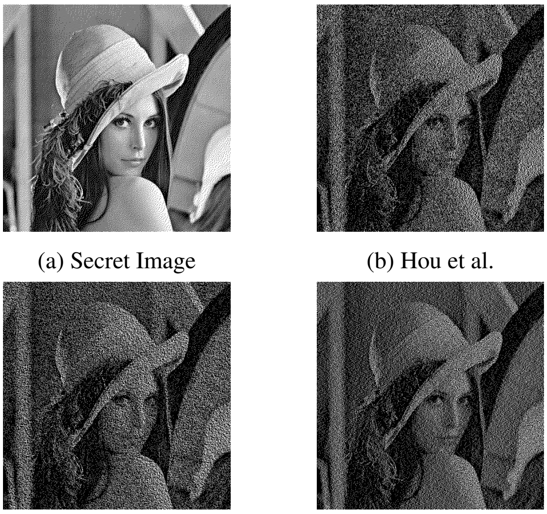
Fig. 7: Using Our Proposed Scheme



(a) Secret Image (b) Hou et al.

(c) Zhang et al. (d) Proposed Method

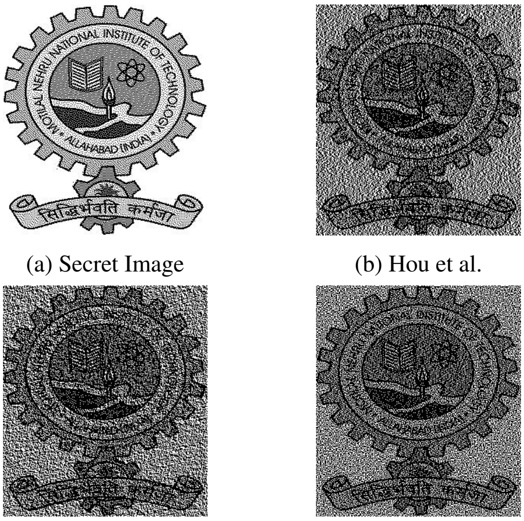
Fig. 8: Resultant Images of Bridge



(a) Secret Image (b) Hou et al.

(c) Zhang et al. (d) Proposed Method

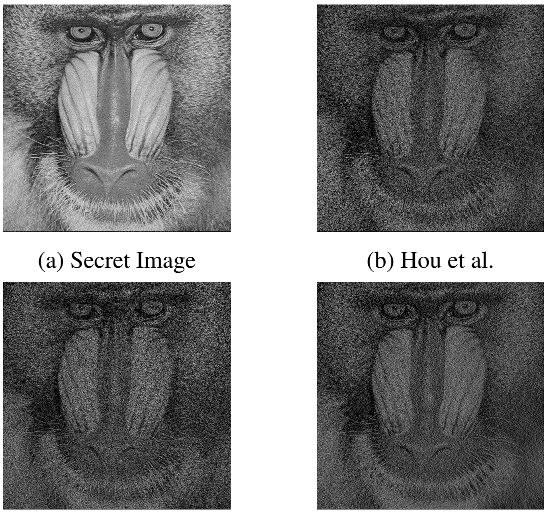
Fig. 9: Resultant Image of Lena



(a) Secret Image (b) Hou et al.

(c) Zhang et al. (d) Proposed Method

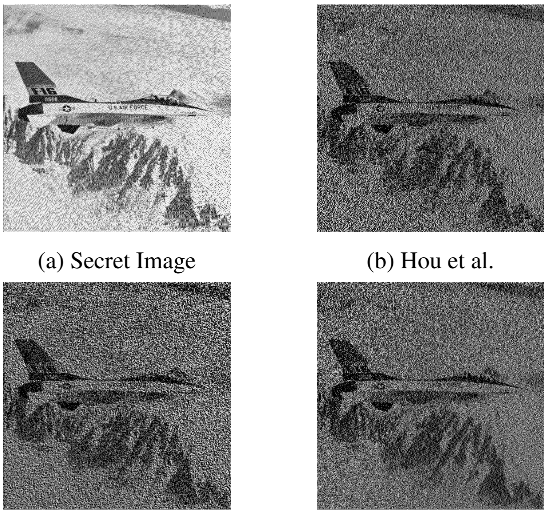
Fig. 10: Resultant Image of logo



(a) Secret Image (b) Hou et al.

(c) Zhang et al. (d) Proposed Method

Fig. 11: Resultant Image of Baboon



(a) Secret Image (b) Hou et al.

(c) Zhang et al. (d) Proposed Method

Fig. 12: Resultant Image of F16