

HYBRIDIZATION OF DCT BASED STEGANOGRAPHY AND RANDOM GRIDS

Pratarshi Saha¹, Sandeep Gurung² and Kunal Krishanu Ghose³

^{1,2}Department of Computer Science & Engineering, Sikkim Manipal Institute of Technology, Majhitar, Sikkim, India

¹pratarshisaha@gmail.com, ²gurung_sandeep@yahoo.co.in

³QualComm, Sandiego, CA, USA

³kunal.sghose@gmail.com

ABSTRACT

With the increasing popularity of information technology in communication network, security has become an inseparable but vital issue for providing for confidentiality, data security, entity authentication and data origin authentication. Steganography is the scheme of hiding data into a cover media to provide confidentiality and secrecy without risking suspicion of an intruder. Visual cryptography is a new technique which provides information security using simple algorithm unlike the complex, computationally intensive algorithms used in other techniques like traditional cryptography. This technique allows visual information to be encrypted in such a way that their decryption can be performed by the Human Visual System (HVS), without any complex cryptographic algorithms. To provide a better secured system that ensures high data capacity and information security, a multilevel security system can be thought for which can be built by incorporating the principles of steganography and visual cryptography.

KEYWORDS

Data Security, DCT based Steganography, Random Grids, Visual Cryptography, Hybrid

1. INTRODUCTION

In the advent of booming communication technology, the needs for information sharing and transfer have increased exponentially. The threat of an intruder accessing secret information has been an ever existing concern for the data communication in the public domain. Cryptography and Steganography are the most widely used techniques to overcome these threats.

Cryptography involves converting a message text into an unreadable cipher. On the other hand, Steganography embeds message into a cover media and hides its existence. A digital image is considered as the carrier in these techniques. Both these techniques provide some level of security of data. However, neither of them alone is secure enough for sharing information over an unsecure communication channel and is vulnerable to intruder attacks. Although these techniques are often combined together to achieve higher levels of security there still is a need of a highly secured system to transfer information over any communication media minimizing the threat of intrusion.

Visual cryptography (VC) is a powerful technique that combines the notions of ciphers and secret sharing in cryptography with that of graphics. VC takes a binary image (the secret) and divides it into two or more pieces known as shares. When the shares are printed on transparencies and then superimposed, the secret can be recovered. No computer participation is required, thus

demonstrating one of the distinguishing features of VC. VC is a unique technique in the sense that the encrypted message can be decrypted directly by the human visual system (HVS). It focuses on solving the problem of secret sharing. A secret sharing scheme suggested by Naor and Shamir's [2] enables distribution of a secret amongst n parties, such that only predefined authorized sets will be able to reconstruct the secret. In a k out of n secret sharing problem n transparencies are generated and it requires a minimum of k shares to retrieve the original image (message). The image remains hidden if fewer than k transparencies are stacked together. Each pixel appears within n modified versions known as shares. The shares are a collection of m black and white sub-pixels arranged closely together. An example of the traditional visual cryptography [4] is given in Figure 1.

Random Grids extends the solution to the secret sharing problem by implementing a collection of 2-D transparent and opaque pixels arranged randomly which reveals the secret to the Human Visual System (HVS) when being superimposed. Unlike other visual cryptography approaches, random grid does not need the basis matrices to encode the shares. Pixel expansion is disallowed which is therefore a great advantage of using Random Grids. Also, the sizes of secret image and the shares are identical to each other.

Steganography [3] is an important sub division of information hiding. In the frequency domain group the frequency coefficients of the images is derived and is used to embed the messages with them. These hiding methods overcome the robustness and imperceptibility problem found in the spatial domain.

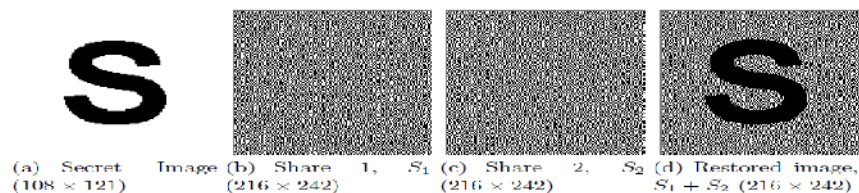


Figure 1: The result of traditional visual cryptography scheme.

Thus to increase the security of the information system a hybrid idea of combining steganography and visual cryptography together is suggested. It would offer us a multilevel security trying to incorporate the best characteristics features of each of the techniques.

2. RANDOM GRIDS

Random grid consists of a transparency comprising of transparent and opaque pixels arranged randomly which is designed that when being superimposed, it reveals the secret to the Human Visual System (HVS) without the help of any computational parameters. A random grid [6] can also be defined as a transparency comprising a two-dimensional array of pixels. Every pixel is either transparent or opaque. Transmission of light through these chosen pixels is random. Opaque pixels block out light whereas transparent pixels allow light to pass through.

The number of opaque pixels where O denotes opaque is equal to $P(O)=1/2$; similarly the number of transparent pixels where Tr denotes transparent is equal to $P(Tr) = 1/2$. Thus the average light transmission of a random grid is also $1/2$. If we assume ' R ' to be the random grid then $T(R) = 1/2$. For a certain pixel ' r ' in random grid R the probability of r to be transparent is equal to that of r being opaque therefore:

$P(r=0) = P(r=1) = 1/2$; where 0 denotes opaque and 1 denotes transparent.

Probability of light transmission of a random pixel ' r ' in random grid ' R ' i.e equal to $t(r) = 1/2$.

Superimposing of two random grids pixel by pixel is denoted by the generalized OR operation “ \otimes ”, therefore it is quite clear that $R \otimes R$ is same as R therefore $T (R \otimes R) = t (r \otimes r) = 1/2$; for each pixel r in R . Table 1 below for gives us a brief idea of the stacking of random pixels r_1 and r_2 .

Table 1 : Stacking of Random Pixels r_1 and r_2

r_1	r_2	$r_1 \otimes r_2$
0	0	0
0	1	1
1	0	1
1	1	1

From the above probability table we can deduce that the probability of transparency of the random pixel is 1/4 i.e., the average light transmission of the random grids (R_1 and R_2) or (r_1 and r_2) is 1/4.

2.1 Random Grid Encryption Algorithms for Binary Images

Given are three different algorithms proposed by Kafri and Keren [5] to accomplish the encryption for the binary images. Given a secret binary image B , these algorithms as follows produce two random grids R_1 and R_2 such that they leak no information of B individually, yet they reveal B in our visual system when superimposed.

Note that random_pixels (0, 1) is a function that returns a binary 0 or 1 to represent a transparent or opaque pixel, respectively, by a coin coin-flip procedure and $R_1[i, j]$ denotes the inverse of $R_1[i, j]$. Here the initial grid (first) is a combination of random collections of ones and zeros. The second grid is created using the original image (secret) as a reference using the algorithms given below.

Algorithm 1: Generation of random grids by inverting the pixels in the corresponding (second) grid for an occurrence of a black pixel in the original image.

Algorithm 2: Encryption of binary image using random grids by inserting random pixel in the second random grid for an occurrence of a black pixel in original image.

Algorithm 3: Generation of grids by inverting pixels in each of the grids for an occurrence of a black pixel in the original image and also substituting random pixels in one of the grids for an occurrence of a white pixel in the original image.

Algorithm 4: The gray-scale images are converted into its half toned version and then any one of the algorithms discussed above is used to generate the random grids.

Algorithm 1 – 3: Encryption of Binary images by Random Grids.

Function name: Encryption (Image)

Input: A $w \times h$ binary image B where $B[i, j] \in \{0, 1\}$ (white or black), $1 \leq i \leq w$ and $1 \leq j \leq h$

Output: Two shares of random grids R_1 and R_2 which reveal G when superimposed where $R_k[i, j] \in B$, $1 \leq i \leq w$ and $1 \leq j \leq h$ and $k \in \{1, 2\}$

Algorithm 1

```

1: Generate  $R_1$  as a random grid
// for (each pixel  $R_1[i, j]$ ,  $1 \leq i \leq w$  and  $1 \leq j \leq h$ ) do
//  $R_1[i, j] = \text{random\_pixel}(0, 1)$ 
2: for (each pixel  $B[i, j]$ ,  $1 \leq i \leq w$  and  $1 \leq j \leq h$ ) do
2.1: { if ( $B[i, j] = 0$ )  $R_2[i, j] = R_1[i, j]$ 
      else  $R_2[i, j] = R_1[i, j]$ 
      }
3: output ( $R_1, R_2$ )

```

Algorithm 2

```

1: Generate  $R_1$  as a random grid
2: for (each pixel  $B[i, j]$ ,  $1 \leq i \leq w$  and  $1 \leq j \leq h$ ) do
2.1: { if ( $B[i, j] = 0$ )  $R_2[i, j] = R_1[i, j]$ 
      else  $R_2[i, j] = \text{random\_pixel}(0, 1)$ 
      }
3: output ( $R_1, R_2$ )

```

Algorithm 3

```

1. Generate  $R_1$  as a random grid
2. for (each pixel  $B[i, j]$ ,  $1 \leq i \leq w$  and  $1 \leq j \leq h$ ) do
2.1 { if ( $B[i, j] = 0$ )  $R_2[i, j] = \text{random\_pixel}(0, 1)$ 
      else  $R_2[i, j] = R_1[i, j]$ 
      }
3. output ( $R_1, R_2$ )

```

Algorithm 4: Random Grids for Gray Level Image

```

Input: A  $w \times h$  grey-level image  $G$  where  $G[i, j] \in G$ ,
 $1 \leq i \leq w$  and  $1 \leq j \leq h$ 
Output: Two shares of random grids  $R_1$  and  $R_2$  which
reveal  $G$  when superimposed where  $R_k[i, j] \in B$ ,  $1 \leq i \leq w$ 
and  $1 \leq j \leq h$  and  $k \in \{1, 2\}$ 
1:  $H = H(G)$ 
2:  $(R_1, R_2) = \text{Encryption}(H)$ 
3: output ( $R_1, R_2$ )

```

The contrast achieved by Algorithms 1, 2 and 3 are $\frac{1}{2}$, $\frac{1}{5}$ and $\frac{1}{4}$, respectively. Thus Algorithm 1 achieves the largest contrast among the three.

For gray and color images half toning is initially performed on the images. We represent the half toning procedure that transforms a gray-level image G into its halftone version H by $H = H(G)$, where $g \in G$ for each pixel g in G and $h \in B$ for each pixel h in H . Since H is simply a binary image, we can apply any of the algorithms as mentioned earlier. Figure 2-5 shows the implementation of the algorithms mentioned before.

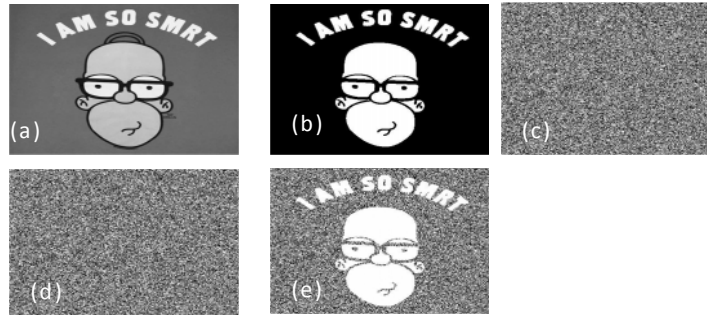


Figure 2: Implementation details of Algorithm 1 for encrypting image B: (a) Input image B; (b) Threshold image; (c) and (d) Encrypted Shares using random grids; (e) Final output image with PSNR value 4.2661.

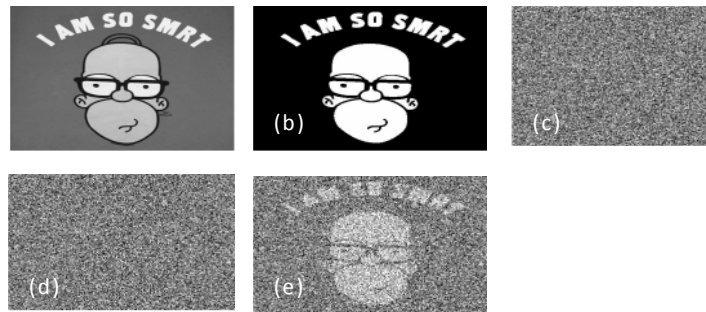


Figure 3: Implementation details of Algorithm 2 for encrypting image B: (a) Input image B; (b) Threshold image; (c) and (d) Encrypted Shares using random grids; (e) Final output image with PSNR value 3.7061

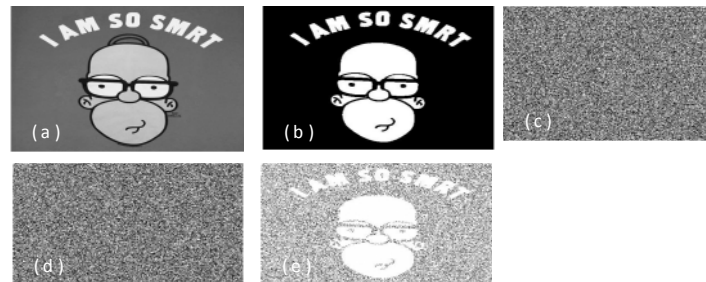


Figure 4: Implementation details of Algorithm 3 for encrypting image B: (a) Input greyscale image B; (b) Threshold image; (c) and (d) Encrypted Shares using random grids; (e) Final output image with PSNR value 2.5189

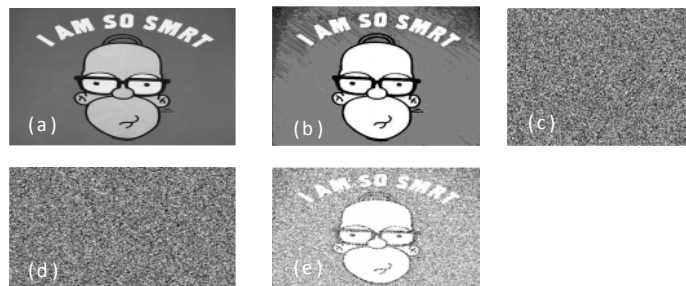


Figure 5 : Implementation details of Algorithm 4 for encrypting image B: (a) Input greyscale image B; (b) Image after error diffusion; (c) and (d) Encrypted Shares using random grids; (e) Final output image with PSNR value 1.2863

2.2 Encryption of Color Image by Color Random Grids

The natural (continuous-tone) images must be first converted into half tone images by using the density of the net dots to simulate the original gray of color levels in the target binary representation. The Floyd-Steinberg Error Diffusion [1] technique is used to convert the original colored image to gray scale.

The image is segmented into CMY channels after half toning and then the algorithm as mentioned earlier are implemented for each of the channels.

Let $E = \{c, m, y\}$ be a set of primary colors in the subtractive model and $C = \{0, c, m, y, r, g, b, 1\}$ denote the set of colors mixed by all subsets of E . Consider a secret color image $p \in C$ for each color pixel p in P . We can decompose each color pixel p into three monochromatic pixels, namely p^c, p^m and p^y in terms of the three primary colors c, m and y respectively where $p^x \in E^x$ for $x \in E$. The monochromatic images composing of all p^c 's, p^m 's and p^y 's are referred to as P^c, P^m and P^y , respectively. The color decomposition of p is therefore denoted as $d(p) = (p^c, p^m, p^y)$. The procedure of decomposing P into P^c, P^m and P^y is denoted as $D(P) = (P^y, P^m, P^c)$. Similarly the composition of mixing of these mono chromes into p can be denoted by $p = m(p^c, p^m, p^y)$ and the procedure for combining P 's as $P = M(R_1^c, R_1^m, R_1^y)$.

Consider a secret color image $P, D(P) = (P^y, P^m, P^c)$ and $P^x = H^x(P^x)$ for $x \in E$. Since $p^x \in E^x$ in P^x can be regarded as binary (0 or x), we can encrypt P^x into two shares, namely R_1^x and R_2^x , by using the ideas of binary image encryption out of Algorithms 1, 2 or 3. With regard to P^x , an x -colored halftone (0 or x) image, we hence generate an x -color random grid R_1^x with $T(R_1^x) = 1/2$ in which each pixel r_1^x in R_1^x is either 0 (transparent) or x , i.e., $r_1^x \in R_1^x$, and $\text{Prob}(r_1^x = 0) = \text{Prob}(r_1^x = 1) = 1/2$ for $x \in E$. We refer to r_1^x as an x -colored random pixel in R_1^x . Due to the independence of the three primary colours, it is reasonable to represent the average light transmission and average colour intensity of $R = M(R^c, R^m, R^y)$ as 3-tuple vectors in terms of the transmissions and colour intensities of R^c, R^m and R^y , respectively: $T(R) = (T(R^c), T(R^m), T(R^y))$ and $I(R) = (I(R^c), I(R^m), I(R^y))$. Moreover, we define $R = M(R^c, R^m, R^y)$ to be a color random grid if and only if $T(R^c) = T(R^m) = T(R^y) = 1/2$ ($T(R) = (1/2, 1/2, 1/2)$), or equivalently, $I(R^c) = I(R^m) = I(R^y) = 1/2$ ($I(R) = (1/2, 1/2, 1/2)$). Each pixel $r = m(r^c, r^m, r^y)$, referred to as a color random pixel, in R satisfies $\text{Prob}(r^x = 0) = \text{Prob}(r^x = x) = 1/2$, i.e. $t(r^x) = 1/2$; or equivalently, $i(r^x) = 1/2$, for $x \in E$. In the same way, we denote the average light transmission and average colour intensity of r in R as follows, respectively:

$$t(r) = (t(r^c), t(r^m), t(r^y)) = (\text{Prob}(r^c = 0), \text{Prob}(r^m = 0), \text{Prob}(r^y = 0)) \text{ and } i(r) = (i(r^c), i(r^m), i(r^y)) = (\text{Prob}(r^c = c), \text{Prob}(r^m = m), \text{Prob}(r^y = y)).$$

Encrypt_Color uses any of the algorithms discussed for binary image. An example after applying Algorithm 5 is shown in the Figure 6. The output is expressed using two mechanisms to give the outputs (o) and (r).

Thus each of the channels can be implemented by any of the three algorithms that work on binary images. A combination of results can be generated from the idea.

Algorithm 5: Random Grids for Color Images

Input: A $w \times h$ colour image P where $P[i, j] \in C, 1 \leq i \leq w$ and $1 \leq j \leq h$

Output: Two shares of color random grids $R1$ and $R2$ which reveal P when superimposed where $R_k[i, j] \in C, 1 \leq i \leq w$ and $1 \leq j \leq h$

- 1: Decompose P into P^y, P^m and P^c , that is, $D(P) = (P^y, P^m, P^c)$
 - 2: for (each $x \in E$) do $P^x = H^x(P^x)$
 - // Transform P^x into x -coloured halftone image P^x
 - 3: for (each $x \in E$) do $(R_1^x, R_2^x) = \text{Encryption_color}(x, P^x)$
 - 4: $R_1 = M(R_1^c, R_1^m, R_1^y)$
 - 5: $R_2 = M(R_2^c, R_2^m, R_2^y)$
 - 6: Output($R1, R2$)
-

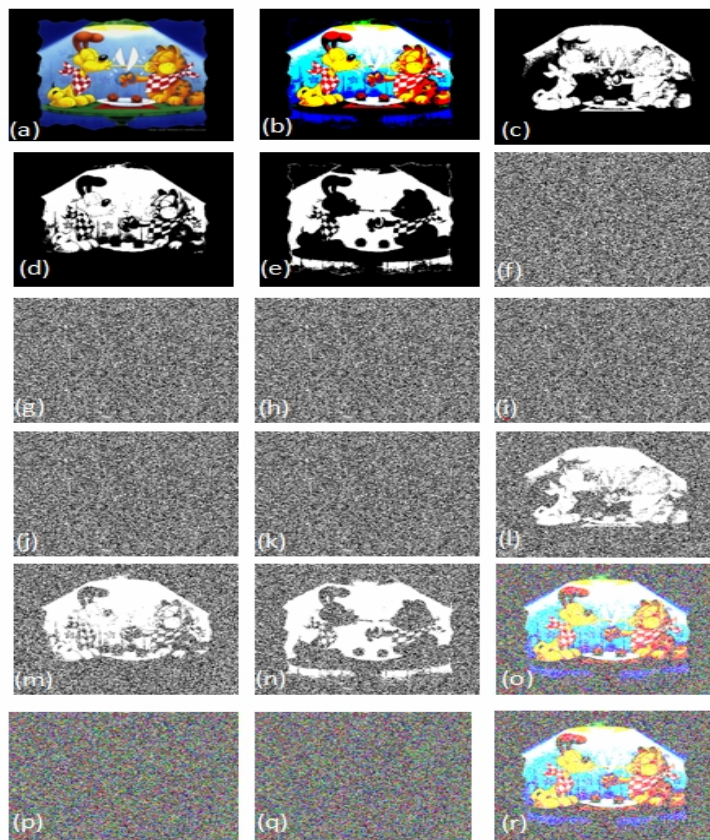


Figure 6: Implementation details of Algorithm 1 for encrypting colour image C: (a) Input color image P; (b) Image after half toning; (c),(d) and (e) Image segmented into CMY channels separately; (f) and (g) C_a and C_b : Random grids 1 and 2 of Cyan channel; (h) and (i) M_a and M_b : Random grids 1 and 2 of Magenta channel; (j) and (k) Y_a and Y_b : Random grids 1 and 2 of Yellow channel; (l) (m) and (n) C_{out} , M_{out} and Y_{out} ; (o) Final output image combining C_{out} , M_{out} and Y_{out} giving PSNR value 4.2889 ; (p) and (q) Combining CMY channels of grid 1 and 2 to form A_{out} and B_{out} respectively; (r) Final output image combining A_{out} and B_{out} giving PSNR value 4.2889.

3. STEGANOGRAPHY USING DCT COEFFICIENTS

Steganography in the DCT Domain is one popular method of encoding secret information in the frequency domain by modulating the relative size of two (or more) DCT coefficients within one image block. The Discrete Cosine Transform (DCT) transforms the image from spatial domain to frequency domain. It separates the image into spectral sub-bands with respect to its visual quality, i.e. high, middle and low frequency components. This approach lessens DCT's effect on enforcing the image data contained in this specific block without severely damaging it. Each block DCT coefficients are quantized using a specific quantization table (QT). The logic behind choosing a table with such values is based on extensive experimentation that tried to balance the trade-off between image compression and quality factors. The HVS dictates the ratios between values in the QT. The aim of quantization is to loosen up the tightened precision produced by DCT while retaining the valuable information descriptors.

In this [3] DCT based technique, the DCT coefficients are obtained for the given carrier image. The secret data is embedded using the LSB substitution technique in the carrier image for DCT coefficients lower than the threshold value. Thus all the coefficients lesser than the threshold are all potential carriers. To avoid visual distortion, embedding of secret information is avoided for DCT coefficient value 0. The Key matrix is used to identify the potential pixels in the stego image. It reflects the position of the coefficients which carry the secret information. Once the potential pixels are identified the hidden bits of secret image can be extracted. Thus it suitably combines both the ideas of the spatial (LSB) and the frequency domain (DCT) techniques. The algorithm below explains the embedding and the retrieval procedure. An implementation of DCT based steganography is shown in Figure 7.

Algorithm : DCT based Steganography Embedding Process

Step 1: Select Carrier Image from the set.

Step 2: Find DCT coefficients of Carrier Image.

Step 3: Traverse through each pixel in Carrier Image till end of Secret Image.

Step 3.1: If DCT coefficient value is below threshold then replace LSB(s) with MSB(s) of pixels in Secret Image.

Step 3.2: Insert 1 at that location in the key matrix.

Step 4: Evaluate the Stego Image

Algorithm : DCT based Steganography Retrieval Process

Step 1: Get the Stego Image.

Step 2: Traverse through each pixel in Stego Image till end.

Step 2.1: Check the key matrix for that location.

Step 2.2: If it is 1, then extract LSB(s) from Stego Image.

Step 2.3: Otherwise move on to next pixel.

Step 3: Get Estimate of Secret Image.



Figure 7: Implementation of Steganography: (a) Secret Image; (b) Stego Image; (c) Retrieved Secret

4. PROPOSED SOLUTION

Steganography and visual cryptography have many similarities and differences, and thus have various uses in the digital and real worlds. Both of them have their different advantages, as well as disadvantages; therefore a hybrid model can be used. Thus a proper exploitation of their pros can help us model a stronger, secure and a robust system.

In the proposed methodology the information to be transmitted is encrypted using visual cryptography and then the cipher is embedded into natural or artificial image/images by using steganography. The solution aims at using both the ideas in one compact form. For the encryption process the secret image is degenerated into its CMY components and the Random grids of the respective components are generated. These grids are then embedded into the carrier image by using a steganographic algorithm in the spatial domain with LSB replacement based on DCT coefficients of the pixels. Firstly the DCT coefficients of the carrier image is obtained and then based on a proper threshold value random locations are selected. LSBs of these potential locations in carrier image are replaced with MSBs of the secret image. Pixels having DCT coefficient value below threshold are considered as potential pixels. Here the threshold value is taken as zero and hence the pixels with DCT coefficient value below zero are used for data hiding. With the guidance of the key matrix the grids are generated back from the stegos. The superpositions of these grids then finally reveal the secret information.

5. DESIGN STRATEGY

Under this section the overall stratagem of approaching the solution is reviewed. The solution to the problem of sending the secret without risking security and theft is done by embedding a secret image in an input carrier image. The segment discusses the generation of random grids from the CMY channels and then the creation of stegos as carriers.

The secret image is decomposed into CYM channels and the various random grid algorithms are implemented for each of them. Each of the generated grids can be hidden in carrier images to form stegos. By generating the DCT coefficients for each of the channels of the carrier image, the key matrix for each of the channels can be obtained. The keys for each channel signify which pixels of the carrier image channel can hide value for the secret pixels. For example, the cyan channel of the secret image, C_a can be hidden in the cyan channel of the carrier image; the magenta channel of the secret image in the magenta channel of the carrier image and similarly for the yellow channel. We can embed more grids by using an additional carrier image or hide the grids using recursion as mentioned in [7]. Thus in a simplified case, we can anticipate that the total numbers of the stegos generated will the same as the number of random grids generated for each of the channels of the colour image. If a similar carrier image is used, there will only be 3 keys generated which in turn can be hidden in the 3 channels of a carrier image simply by a single LSB substitution. A diagrammatic representation of the proposed backbone of how the two processes take place is shown in Fig 7 and Fig 8.

The encryption process takes an input image and then channelizes the image into Cyan, Magenta and Yellow. The Random Grid Generator then starts the actual encryption process wherein the channelized images have to be implemented using the Algorithms 1 to 3. The highlight of this process is that the permutation of the algorithms being used on the channels is upon the user i.e., any algorithm can be used on any channel depending on the users choice. The number of grids to be generated is also based upon the users' choice. The second segment of the encryption phase is to pass the grids of every channel on to the stego generator. The key grids are generated by the stego generator which reveals potential positions for LSB substitution in the carrier image to give the key grid stego. Three key grids are generated for the respective channels. The grids are embedded into the respective carrier image channels guided by key grids (Matrix). Thus the carrier image marked with the key matrix acts as a basis to generate the various stegos.

In the decryption phase the stego decrypter takes the stegos generated before as inputs and then uses the key grid matrix for retrieving the random grids. The key grids for the respective channels are used to regenerate the various random grid patterns for each of the channels. The process is repeated for each of the stegos and finally all the random grids are recovered. These grids are then combined i.e., all the cyan grids are overlapped to get the cyan channel, the magenta grids are overlapped to get magenta channel and similarly for yellow. Upon recombination of the final cyan, magenta and yellow grids the output image is obtained and decryption is completed. Thus the decryption phase emphasizes how the stegos are decrypted by the stego decrypter to give the random grids which are combined to give the CMY channels which in turn are combined to give the output image. The key grid stego is also decrypted to give the key grids which are used to retrieve the random grids from the stegos received. On obtaining the random grids, a simple overlap of the transparencies gives the cyan, magenta and the yellow channels on overlapping which the secret image is revealed. An example of the hybrid system is given in Figure 9.

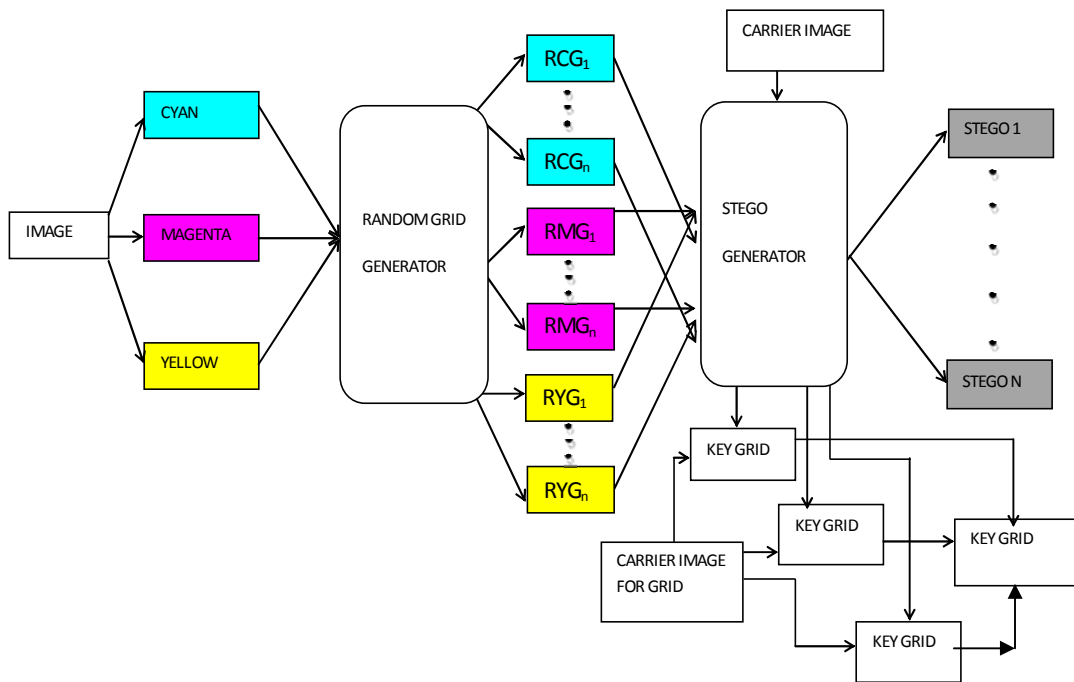


Figure 7: Encryption Phase of the Hybrid Model

The idea can be extended to another scheme where each of the components now undergo a simple conversion such as the C component is altered into M, the Y component is mapped with C and the

M factor is now changed to Y. There can be a total of 6 such combinations of the channels. Though initially, the conversions will be assumed predetermined and known to the sender and the receiver, attempts can be made for randomized conversion techniques, which can be decoded back at the receivers' end through a scholastic and/or soft computing techniques. After this procedure the shares will be generated using visual cryptography schemes.

By using [7] we can generate a n:n Random Grid scheme. However the performance degrades as the value of n increases. The data hiding is done in a recursive manner where the generation of shares is done in a hierarchical tree structure. The second grid of a level embeds the two grids of the next lower level.

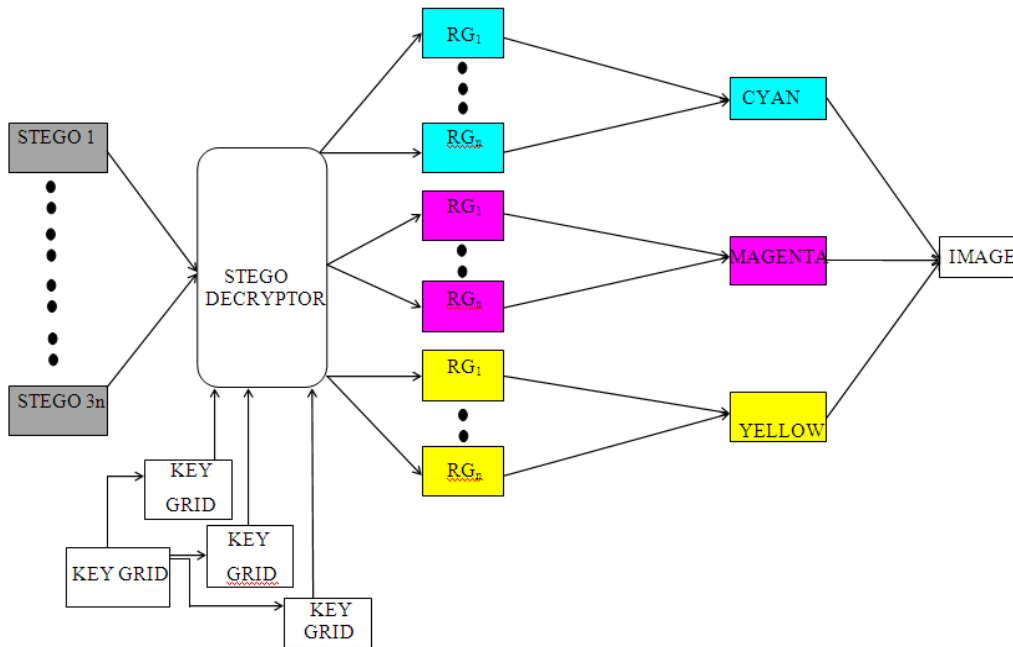


Figure 8: Decryption Phase of the Hybrid Model.

6. EXPERIMENTAL RESULTS AND DISCUSSIONS

6.1 Contrast Calculation

To evaluate the relative difference of the light transmissions between the transparent and opaque pixels in reconstructed image S by these random grid-based based algorithms, we define the contrast of S with respect to B by Algorithm A as:

$$T(S[B(0)]) - T(S[B(1)]) / (1 + T(S[B(1)])),$$

where $S = R1 \otimes R2$, and R1 and R2 are the random grids of B encrypted by Algorithm A, where 'A' denotes either of Algorithms 1,2 or 3 specified above. Thus the contrasts achieved by Algorithms 1, 2 and 3 are 1/2, 1/5 and 1/4 respectively as denoted by Table 2 [6].

We may say that the reconstructed image obtained by Algorithm1, which achieves the largest contrast among the three, can be recognized easier in our visual system than those by the other

two algorithms. Thus we usually prefer Algorithm 1 over the other two algorithms for optimized results in terms of contrast, clarity, brightness. The difference in the results of the performance of the algorithms can already be seen from the previous examples.

6.2. Average Color Intensity

$\text{Prob}(s^x = 0)$ ($=t(s^x)$) describes the average light transmission of s^x , $\text{Prob}(s^x = x)$ ($=1 - \text{Prob}(s^x = 0)$) exposes the average colour intensity of x on s^x with respect to p^x . The possible mixing of colors $m(s^c, s^m, s^y)$ is calculated in terms of the average color intensity due to the reason that it would be more appropriate to describe the mixing colors in the subtractive model by the color intensity instead of the light transmission. Hence, we define the average color intensity of x -colored pixel s^x as: $i(s^x) = \text{Prob}(s^x = x) = 1 - \text{Prob}(s^x = 0) = 1 - t(s^x)$ and the average colour intensity of x -colored grid S^x as: $I(S^x) = 1 - T(S^x)$, for $x \in E$.

Table 2: Encoding b into r_1 and r_2 and results of $s = r_1 \otimes r_2$ by Algorithms 1, 2 and 3.

	b	Probability	r_1	r_2	$s = r_1 \otimes r_2$	$\text{Prob}(s=0)$ ($t(s)$)
Algorithm 1	□	$\frac{1}{2}$	□	□	□	$\frac{1}{2}$
	■	$\frac{1}{2}$	■	■	■	$\frac{1}{2}$
Algorithm 2	□	$\frac{1}{2}$	□	□	□	$\frac{1}{2}$
	■	$\frac{1}{2}$	■	■	■	$\frac{1}{2}$
		$\frac{1}{4}$	□	■	■	$\frac{1}{4}$
		$\frac{1}{4}$	■	□	■	$\frac{1}{4}$
Algorithm 3	□	$\frac{1}{4}$	□	□	□	$\frac{1}{4}$
		$\frac{1}{4}$	□	■	■	$\frac{1}{4}$
		$\frac{1}{4}$	■	□	■	$\frac{1}{4}$
		$\frac{1}{4}$	■	■	■	$\frac{1}{4}$
	■	$\frac{1}{2}$	□	■	■	$\frac{1}{2}$
	■	$\frac{1}{2}$	■	□	■	$\frac{1}{2}$

Table 3[6] summarizes the color intensity for each of the colors encrypted by Algorithm 1, Algorithm 2 and Algorithm 3 respectively. The pixel table for each of the colors encrypted i.e., algorithm 1 for cyan channel, algorithm 2 for magenta channel and algorithm 3 for yellow channel is given in Table 4[6].

Table 3: Color Intensities for each of the colors using Algorithm1 – 3

p	i(S)		
	Algorithm 1	Algorithm 2	Algorithm 3
0	(1/2, 1/2, 1/2)	(1/2, 1/2, 1/2)	(3/4, 3/4, 3/4)
C	(1, 1/2, 1/2)	(3/4, 1/2, 1/2)	(1, 3/4, 3/4)

Y	(1/2, 1, 1/2)	(1/2, 3/4, 1/2)	(3/4, 1, 3/4)
M	(1/2, 1/2, 1)	(1/2, 1/2, 1/4)	(3/4, 3/4, 1)
R	(1/2, 1, 1)	(1/2, 3/4, 3/4)	(3/4, 1, 1)
G	(1, 1/2, 1)	(3/4, 1/2, 3/4)	(1, 3/4, 1)
B	(1, 1, 1/2)	(3/4, 3/4, 1/2)	(1, 1, 3/4)
1	(1,1,1)	(3/4, 3/4, 3/4)	(1, 1, 1)

6.3 Color Recovery Ratio (CCR)

Let N be the number of the total pixels in P (the halftone image of secret image P). Let n denote the number of pixels in S which have the same colors as their corresponding pixels in P (or the number of the colors pixels that are exactly recovered) where S is reconstructed by Algorithm A. The color recovery ratio $r_A(P, S)$ for P and S by Algorithm A is computed as $r_A(P, S) = n/N$. Thus the value is a good measure to evaluate the quality of the reconstructed image.

6.4 CMY model with Swapping and Varying Algorithms

The various combinations of the channels decomposed are taken and the random grid algorithm is applied to them. On implementing the various combinations of the algorithms the following gave the best results as shown in Table 5. The PSNR value is given to show which recovered image better resembles the original secret image.





























Pixel Colour	Random Grid 1 R_1	Random Grid 2 R_2	Superimposed Image
			
			
			
			
			
			
			

Table 4: Pixel distribution table


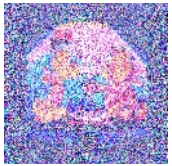




7. CONCLUSION

It is apparent that a lot of time and effort have been dedicated to visual secret sharing using visual cryptography. Many of the schemes presented work extremely well and the current state of the art techniques have proven to be very useful for many applications, such as verification and authentication.

The following trends have been identified within visual cryptography:

1. Contrast improvement.
2. Share size improvement.
3. Wider range of suitable image types (binary to color images).
4. Efficiency of VC schemes.
5. Ability to share multiple secrets.

Table 5: Comparison of Various Combinations of Algorithms.

Procedure (Algorithm)	Input	Output	PSNR	CCR
C Algorithm 1 M Algorithm 3 Y Algorithm 2			4.0324	0.7054
C Algorithm 1 M Algorithm 2 Y Algorithm 3			4.0566	0.7545
C Algorithm 3 M Algorithm 2 Y Algorithm 1			4.0715	0.7573

Essentially the most important part of any VC scheme is the contrast of the recovered secret from a particular set of shares. Ideal schemes provide a high contrast when the secret has been recovered. However, a trade off is required in some schemes depending on the size of the shares along with the number of secrets which may be concealed. Especially within extended visual cryptography schemes, contrast is of major importance. Making sure the base images completely disappear and a clear secret is recovered which could be another high quality image is vitally important.

Some schemes present methods which do not work with printed transparencies and these rely on computation in order to recover the secret. In this respect, high quality secret recovery is

possible, however it is preferred if the scheme works with printed transparencies. After all, this is the idea behind VC. Conversely, if an application requires digital recovery of the secrets, then perfect recovery can be achieved via the XOR operation.

Visual cryptography has had extensive development for monochrome images but the area for colored images remains vacant for considerable research and progress. Generation of random grids for colored images has sparsely been explored

This project aims at implementing a blend of the steganography and visual cryptography. The algorithm developed can be used depending on the situation and application. The proposed system is aimed to simplify the complex and redundant process with the flexibility of a simple process.

The following are the merits of the proposed system.

- It provides two levels of security to the information being transmitted. The intruders cannot easily break the system even if they realize the existence of a secret data as they cannot easily recognize the data, since data is hidden in two ways.
- This system overcomes the demerits of using single level of hiding. That is either using cryptography or steganography.
- It requires only the computation time of single level hiding since visual cryptography uses the HVS to decrypt the information.

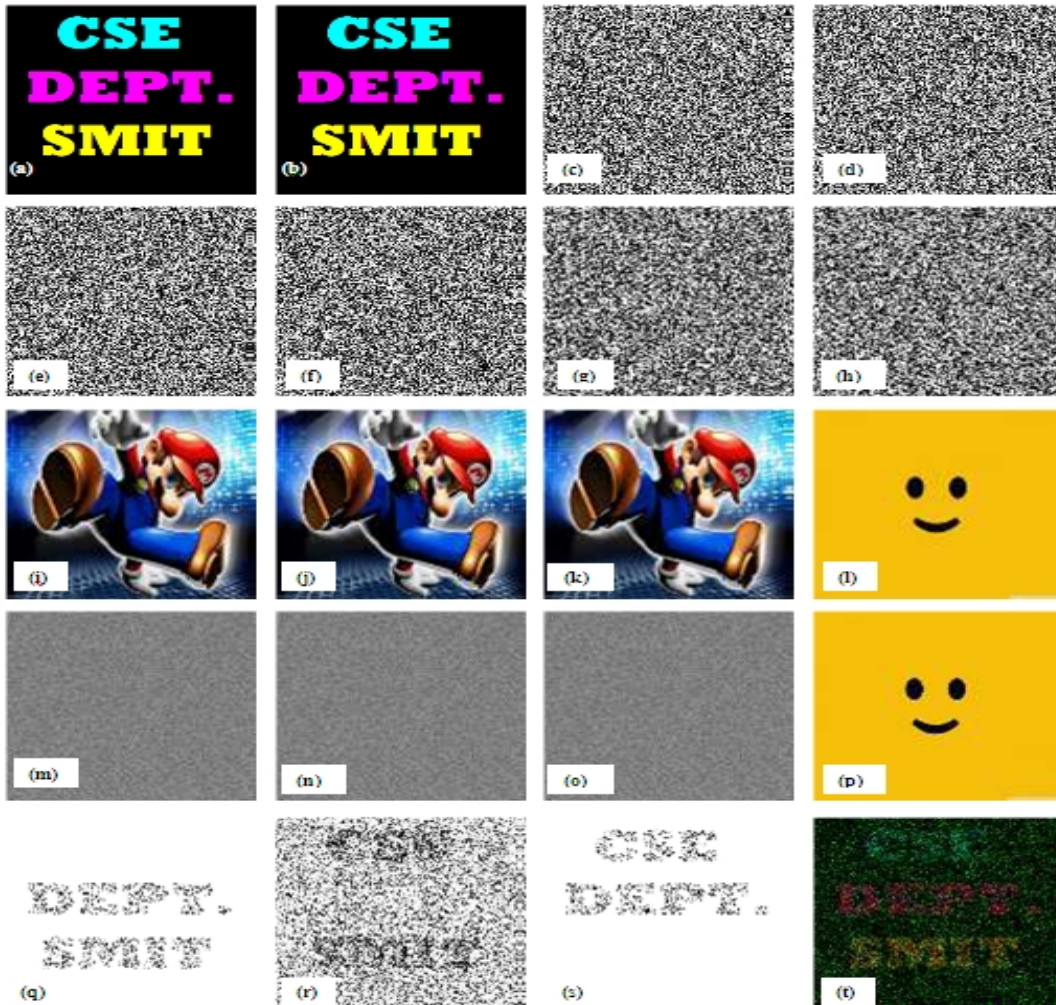


Figure 9: Implementation of steganography using DCT with (n,n) random grids : (a)Input Image; (b) Half toned Image; (c) and (d) Cyan Grids 1 and 2 ; (e) and (f)Magenta Grids 1 and 2; (g) and (h) Yellow Grids 1 and 2; (i) Carrier Image for grids;(j) Stego image 1; (k) Stego Image 2; (l) Key Image; (m), (n) and (o) Key Grids 1, 2and 3; (p) Stego Image for Key; (q) Cyan overlap 1 with PSNR 0.4015 and CRR 0.0786; (r) Magenta overlap 2 with PSNR 1.4854 and CRR 0.2823; (s) Yellow overlap 3 with PSNR 0.5142 and CRR 0.1030 ; (t) Final output image overlapping (q), (r) and (s) with PSNR 0.8004 and CRR 0.8454

Acknowledgement

The corresponding author deeply acknowledges the guidance and inspiration by his Ph.D. guide Prof. (Dr.) M K Ghose, Head, Computer Science and Engineering Department, SMIT, Sikkim, India.

REFERENCES

- [1] Ahmad Movahedian Attar, Isfahan University of Technology, Omid Taheri, Isfahan University of Technology, Saeid Sadri, Isfahan University of Technology, Mohammad Javad Omidi, Isfahan University of Technology, "Data Hiding in Halftone Images Using Error Diffusion Half toning Method with Adaptive Thresholding", 2006,pp. 2.

- [2] Adi Shamir and Moni Naor, "Visual Cryptography", 1964, pp. 1-2, 3-5.
- [3] Hardik Patel and Preeti Dave, "Steganography Technique based on DCT Coefficients", Jan – Feb 2012, International Journal of Engineering Research and Applications, Vol 2, Issue 1, pp 713-717, www.ijera.com.
- [4] Jonathan Weir and Wei Qi Yan Queen's University Belfast, Belfast, BT7 INN,UK,A, 2010, "Comprehensive Study of Visual Cryptography", pp. 70.
- [5] Kafri, O., Keren, E., "Encryption of pictures and shapes by Random Grids." Optics,Letters, 1987, 377–379.
- [6] Shyong Jian Shyu , Department of Computer Science and Information Engineering, Ming Chuan University, 5 Der Ming Rd, Gawi Shan, Taoyuan 333,Taiwan, ROC. "Image Encryption by Random Grids", 2006, The Journal of Pattern Recognition Society, www.sciencedirect.com.
- [7] Tzung-Her Chen, Kai-Hsiang Tsao Department of Computer Science and Information Engineering, National Chiayi University, 300 University Rd., Chiayi City60004, Taiwan, "Threshold Visual Secret Sharing using Random Grids",2011, pp. 1198.

First Author:-

Pratarshi Saha is a Final year student in the Department of Computer Science and Engineering at Sikkim Manipal Institute of Technology, Mazitar, Sikkim, India. He subject of interests are Computer and Information Security, Design and Analysis of Algorithms and Computer Networks.

Second Author:-

Sandeep Gurung received his M. Tech degree in Computer Science and Engineering from the Sikkim Manipal University in 2009 and is currently pursuing his Ph.D. degree in Computer Science and Engineering. He is a Assistant Professor in the Department of Computer Science at Sikkim Manipal Institute of Technology, Mazitar, Sikkim, India. His research interests include Computer Networks, Cryptography, Distributed Systems and Soft Computing.

Third Author:-

Kunal Krishanu Ghose did his MS (Engg.) in Electrical and Communication Engineering with specialization Wireless Sensor Network from University at Buffalo, NY, USA in 2009 and B. Tech (ECE) from NIT Durgapur, INDIA in 2006. After completion of B. Tech, he joined as a System Engineer in Aricent (Hughes Software System), Chennai for a year in 2007. Presently, he is working in Qualcomm Inc., Sandiego, CA, USA as a Sr. Engineer in Architecture Performance Department, looking after the Quad core processor technology. His areas of research interest are Mobile Network, Communications, and Cryptography.