SECRET SHARING IN VISUAL CRYPTOGRAPHY

By

SANDEEP KATTA

Bachelor of Engineering in Electronics and

Communication Engineering

Anna University

Chennai, India

2008

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2011

SECRET SHARING IN VISUAL CRYPTOGRAPHY

Thesis Approved:

Dr. Subhash Kak

Thesis Adviser

Dr. Johnson Thomas

Dr. John Chandler

Dr. Mark E. Payton

Dean of the Graduate College

ACKNOWLEDGMENTS

It gives me immense pleasure in acknowledging all those people, without whom, this thesis would have been impossible.

I would like to express my deepest gratitude and respect to my Thesis Advisor Dr. Subhash Kak for providing me with invaluable guidance, inspiring talks and enthusiastic support throughout the course of my work. His guidance helped me in all the time of research and writing of the thesis. I could not have imagined having a better advisor and mentor for my Masters degree.

I would like to take this opportunity to express my gratitude and thanks to Dr. Johnson Thomas and Dr. John Chandler for accepting my request to be on the advisory committee and giving me invaluable advice especially at the proposal stage.

I would like to thank Brenda L. Johnson and Laura Gann for providing me Graduate assistant position and also like to thank Oklahoma State University for supporting me financially throughout my Masters program.

I would like to thank all my colleagues and my friends, Sandeep Gutta, Vamsi Krishna, Renuka, Suresh Babu Thippireddy, Abhishek Ghale, Ashwin Ravi, Rajesh Hariharan, Prasanth Maddipati, Sravan Bodepudi, Prathyusha Bodepudi, Mounika Paidipally for always being there for me.

Also, I would like to thank my family: my parents Katta Gopala Rao, Katta Rajya Laxmi and my brother Katta Karthik. Without their support I would not have been able to embark on this journey of academic excellence.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

CHAPTER I

1. INTRODUCTION

1.1 Cryptography

Cryptography is derived from Greek word ‘Krypto’ which means hidden and ‘Grafo’, which means written. It is the study and implementation of techniques to hide information, or simply to protect a message or text from being read. The information that is protected can be written text, electronic signals, e-mail messages or data transmissions. The process of making the information unreadable is encryption or enciphering and the result of encryption is a ciphertext or cryptogram. Reversing this process and retrieving the original readable information is called decryption or deciphering. To encrypt or decrypt information, an algorithm or so called cipher is used.

Ever since mankind has existed, people have had secrets, and other people have wanted to know these secrets. The earliest forms of cryptography were performed by pencil and paper, and were available only to those who had access to proper education. Today our lives are completely digitized and cryptography has become an integral part of nearly everyone’s daily life, and it’s used to protect confidential information from hackers. Nearly all our private information is stored in one of the many databases from the government, banks, health care services and so on. Cryptography protects the right to privacy and the right to communicate confidentially.

Secure communications can protect one's intimate private life, business relations, and social or political activities.

1.2 Background on Visual Cryptography

Cryptography has a long and fascinating history [22], and it is one of the most important fields within the security profession. Visual cryptography uses the characteristics of human vision to decrypt encrypted images and in it the secret image is split into two or more separate random images called shares. To decrypt the encrypted information, the shares are stacked one on top of the other, and the hidden secret image appears. Due to its simplicity, anyone can physically manipulate the elements of the system, and visually see the decryption process in action without any knowledge of cryptography and without performing any cryptographic computations.

With the near universal use of the Internet in every field, the need to share important documents from one office to other via this medium becomes increasingly more necessary. With the coming era of the electronic commerce, there is an immediate need to solve the problem of ensuring information safety in today's increasingly open network environment. To protect the security of information, various encrypting technologies of traditional cryptography are usually used. With such technologies, the data can become disordered after being encrypted and later it can be recovered by a correct key. Without the correct key, the encrypted source content can hardly be detected even though unauthorized persons steal the data.

1.3 Traditional Secret Sharing

When important secret information is managed by individuals, secrets may leak. Suppose there is a vault that must be opened every day in a bank. Although the bank employs three senior tellers, management may not trust any individual teller. Therefore, it is necessary to find a possible solution to design a system whereby any two of the three senior tellers can gain access to the vault, but no individual teller can do so. This problem can be easily solved using a secret sharing scheme.

In a more general situation one may need to specify exactly which subsets of participants should be able to determine the key and which should not. Secret sharing schemes are useful in many situations that require the concurrence of several chosen people as in launching a missile or entering an area of restricted access (e.g., a bank vault).

1.4 Secret Sharing Scheme

A secret sharing scheme divides (sharing) a secret key K among a finite set of n participants in such a way that only certain specified subsets (qualified subsets) of participants can compute the secret key K by gathering their information. It was discovered independently by G.R. Blakley and Adi Shamir [3, 13]. Shamir's secret sharing scheme is an interpolating scheme based on polynomial interpolation while Blakley's secret sharing scheme is geometric in nature. Both of the schemes are k-out-of n schemes but they represent two different ways of constructing such schemes, based on which more advanced secret sharing schemes can be designed. The biggest motivation for secret sharing is secure key management. In particular situations, there will be only one secret key that provides access to many important files. If such a key is lost, then all the important files become inaccessible.

The basic model for secret sharing is called a k-out-of-n scheme (or sometimes referred as (k, n) threshold scheme). In this scheme, there is a sender and n participants. The secret information is divided into n parts by the sender, and gives each participant one part so that any k parts can be put together to recover the secret, but any $k - 1$ parts reveal no information about the secret. The pieces are usually called shares or shadows. Different choices for the values of k and n reflect the tradeoff between security and reliability. A secret sharing scheme is perfect if any group of at most $k - 1$ participants (insiders) has no advantage in guessing the secret over the outsiders.

1.5 Visual-Threshold Scheme

In a threshold scheme the secret can be any type of data. For example, it might be an image I, consists of black and white pixels. The secret image I could be encoded as a binary string K=K (I), where 1 represents a black pixel and 0 represents a white pixel. By using any convenient secret sharing scheme, shares for K could be constructed. K would later be reconstructed using the appropriate algorithm for the secret sharing scheme. The image I is converted back using the resulting binary string. In this basic secret sharing scheme, however, cryptographic computations using computer are necessary to share a secret and decode the secret from shared data. In all the secret sharing schemes, a great deal of complexity is necessary to encrypt and decode a secret, and therefore computers are essential.

The following question was asked by Kafri and Keren [1]: Is it possible to create a secret sharing scheme in which the secret image *I* that can be reconstructed visually by superimposing random grids? Each grid would consist of a transparency, made up of black and white pixels. Later Naor and Shamir [2] introduced a specific implementation that was named visual secret sharing (VSS). This method can securely share image information (printed text, handwritten notes, pictures etc.), and it is possible to decode shared secrets by the human visual system. Based on the secret message (the original image) the VSS scheme generates *n* images (known as shares) which can be printed on *n* transparencies. In a *k*-out-of-*n* scheme, there would be *n* transparencies, and if any *k* or more than *k* transparencies are superimposed, the original secret image *I* should appear, but no information about the original image can be gained if fewer than the threshold number of *k* transparencies are stacked ($k - 1$ shares).

The main difference between a traditional threshold scheme and a visual threshold scheme is how the secret is recovered. A traditional threshold scheme typically involves computations in a finite field; in a visual threshold scheme, the computation is performed by the human visual system [8]. In both types of schemes the security conditions is the same.

1.6 Recursive Hiding of Secrets

Recursive hiding of secrets was first introduced by M. Gnanaguruparan and Subhash Kak [5], with applications to both images and printed text, to increase the efficiency of visual cryptography and to make it possible to incorporate additional secret information that serves as a steganographic channel [6]. The idea involved in recursive hiding of secrets is that several multiple messages can be hidden in one of the shares of the original secret image. The secret images that are to be hidden are taken according to their sizes from the smallest to the largest (i.e., secret size doubling at every step). The smallest secret image is divided into *n* shares using the basic idea of visual cryptography. These *n* shares are placed below each other, and they now represent the first share of the secret image. The second share is accomplished in such a manner that if the *n* shares are overlaid, then the secret image is revealed under consideration. This process is repeated recursively. Important thing to be noticed is that the share of the original secret image that contains the recursively-hidden information must also contain both the shares of the last hidden secret image [5]. With respect to the original secret image this enforces a compulsion on the size of the secret images.

1.7 Grayscale Images

In a grayscale image the value of each single pixel carries intensity information. Images of this kind are also known as black-and-white. These are exclusively composed of gray shades and are thus distinct from one-bit black and white images. The darkest possible shade is black, which is the total absence of transmitted or reflected light and the lightest possible shade is white.

The thesis is organized as follows: Chapter II outlines the previous work that has been done in this field. Chapter III explains the proposed approach in detail, i.e., how the secret information break down into smaller secrets in shares of larger secrets by doubling the secret size at every step and also explains secret sharing scheme for gray scale images. The summary and conclusions are presented in Chapter IV.

CHAPTER II

2. REVIEW OF LITERATURE

2.1 Visual Cryptography

Visual cryptography is a powerful encryption technique to hide information in images in such a way that it can be decrypted by the human vision if the correct key image is used. It uses two or more transient images (called *shares*). One image contains random pixels and the other image contains the secret information. It is impossible to retrieve the secret information from one of the images. Either transparent images or layers are required to reveal the secret information. The easiest way to implement visual cryptography is to print the two layers onto a transparent sheet. When the random image contains truly random pixels it can be seen as one-time pad system and will offer unbreakable encryption. In the overlay animation it can be observed by sliding the two layers over each other until they are correctly aligned and the hidden information appears.

In visual secret sharing, the message bit consists of a collection of black and white pixels i.e. it is assumed to be a binary image and each pixel is handled separately. Each original pixel appears in *n* modified versions (called *shares*) of the image, one for each transparency. Each share consists of *m* black and white subpixels. Each share of the subpixels is printed on the transparency in close proximity (to best aid the human perception, they are typically arranged together to form a square with *m* selected as a number). The resulting structure can be described by a [$n \times m$] Boolean matrix S = $(S_{ij})_{n \times m}$ where $s_{ij} = 1$ if and only if the j_{th} subpixel in the i_{th} share (transparency) is black and $s_{ij} = 0$ if and only if the j_{th} subpixel in the i_{th} share (transparency) is white. When transparencies $i_1, i_2, \ldots i_r$ are stacked together in a way which properly aligns the subpixels, we see a combined share whose black subpixels are represented by the Boolean "or" of rows $i_1, i_2, \ldots i_r$ in S. The grey level of this combined share is proportional to the Hamming weight $H(V)$ of the "or" ed m-vector V. This grey level is interpreted by the visual system of the users as white if $H(V) < d- \alpha m$ and as black if $H(V) \geq d$ for some fixed threshold $1 \leq d \leq m$ and relative difference $\alpha > 0$.

Definition 1. A solution to the k out of n visual secret sharing scheme consists of two collections of $n \times m$ Boolean matrices C_0 and C_1. To share a white pixel, the associate randomly chooses one of the matrices in C_0, and to share a black pixel, the associate randomly chooses one of the matrices in C_1. The chosen matrix defines the color of the m sub pixels in each one of the n transparencies. The solution is considered valid if the following three conditions are met [2]:

Contrast

1) For S in C_0 (WHITE): $H(V) < d- \alpha m$
2) For S in C_1 (BLACK): $H(V) \geq d$

Security

1) For any subset { $i_1, i_2, \ldots i_q$} of {1, 2, . . . n} with $q < k$, the two collections of $q \times m$ ($1 \leq q \leq k$) matrices, formed by restricting $n \times m$ matrices in C_0 and C_1 to any q rows, are indistinguishable.

2.2 Preliminary Notation

1) ***n*** → Group Size

2) ***k*** → Threshold

3) ***m***, the number of pixels in a share. This parameter represents the loss in resolution from the original image to the recovered one.

4) ***α***, the relative difference in the weight between the combined shares that come from a white pixel and a black pixel in the original image. This parameter represents the loss in contrast.

5) ***γ***, the size of the collection of C_0 and C_1. C_0 refers to the subpixel patterns in the shares for a white pixel and black refers to the subpixel patterns in the shares for the C_1 pixel.

6) $\mathbf{C_0}$ → Collection of $n \times m$ Boolean matrices for shares of white pixel

7) $\mathbf{C_1}$ → Collection of $n \times m$ Boolean matrices for shares of black pixel

8) ***V*** → OR_{ed} *k* rows

9) ***H*(*V*)** → Hamming weight (*H*) of a string is the number of symbols that are different from the zero-symbol. Figure-1: explains the hamming weight of the pixels when they are stacked.

A small example gives the clear idea of *H*(*V*).

String	Hamming weight
11101	4
11101000	4

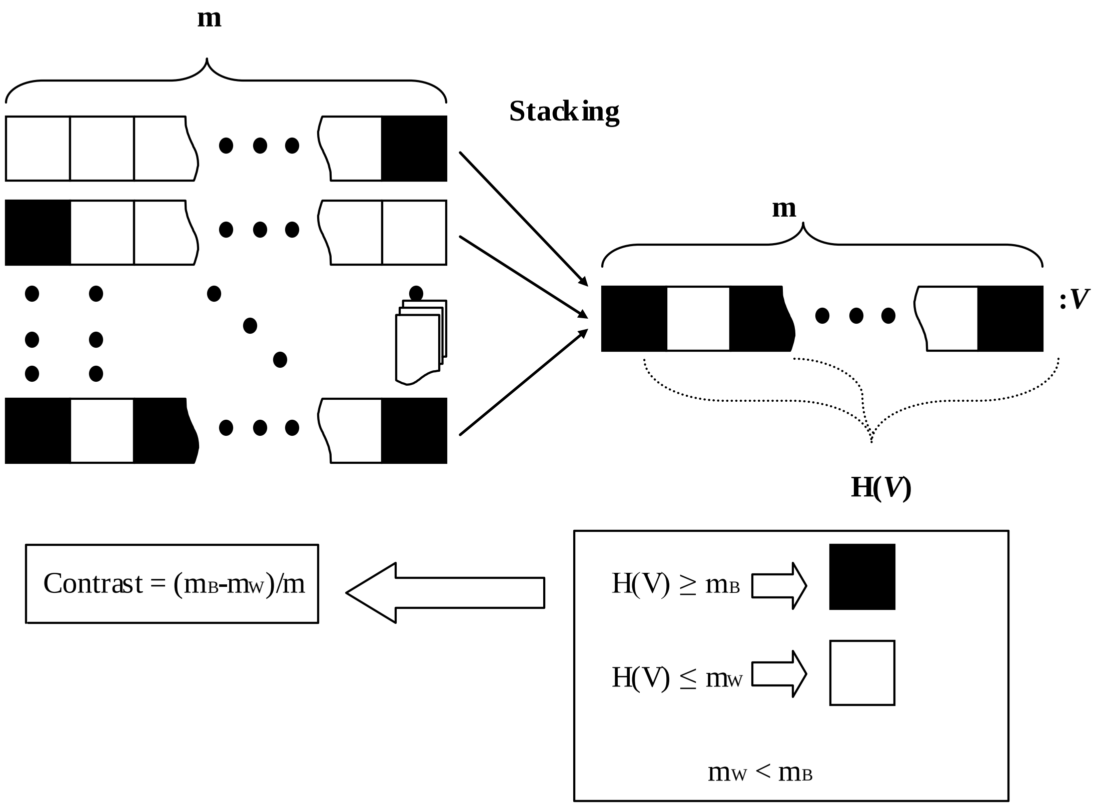


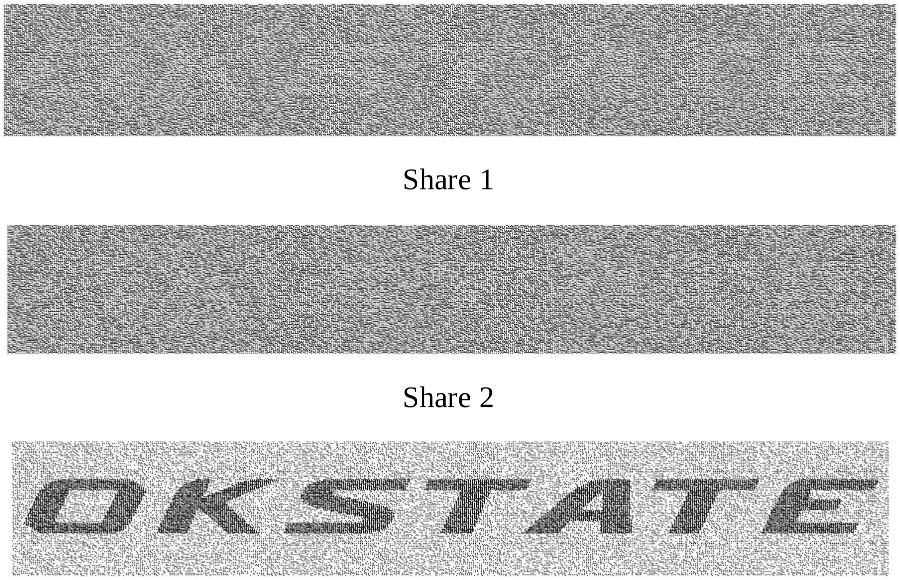
Figure-1: Stacking and finding the pixel contrast using hamming weight (*H*)

10) ***d*** → number in $[1,m]$

11) ***r*** → Size of collections C_0 and C_1

2.3 How Visual Cryptography Works

Each pixel of the image is divided into smaller blocks and always has the same number of black and white (transparent) blocks. For example if a pixel is divided into two parts (2 subpixels), there will be one white and one black blocks. Similarly if the same pixel is divided into four equal parts (4 subpixels), there will be two white and two black blocks. Here is a simple example that explains the idea of how visual cryptography works.



Share 1

Share 2

Overlay (Share 1 + Share 2)

Figure-2: Example to show how VC works

From Figure-2 we can observe that the original image is broken up into two parts which are its shares. Separately these shares look like random noise but when combining reveals an image. Every single pixel is split into subpixels and the human vision still perceives them as one pixel. To try this, one can copy the share 1 and 2 and print them onto a transparent sheet or thin paper. Always use a program that displays both the black and white pixels aligned correctly and set the printer so that all pixels are printed accurate.

2.3.1 Two-out-of-Two Scheme (2 subpixels)

The encoding scheme is to share a binary image into two different shares Share 1 and Share 2. Each pixel is divided into a black and white subpixel placed next to each other. For the case of white pixel, one of the two combinations of subpixels will be chosen with a probability of 0.5 to represent the pixel in each of the shares. When these shares are placed one on top of the other, the pixel are visually OR$_{ed}$ and hence a white pixel looks gray (half black and half white) to the human eye. The pixels are chosen in a similar manner for the case of a black pixel. But when the subpixels are visually OR$_{ed}$, the two black subpixels placed next to each other appear as a single black pixel. This idea can be applied to images to develop a basic Two-out-of-Two scheme by using 2 subpixels.

The 2 out of 2 visual secret sharing problem can be solved by the following collection of $n \times n$ matrices:

C_0 = { all the matrices obtained by permuting the columns of $\begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$ }

C_1 = { all the matrices obtained by permuting the columns of $\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$ }

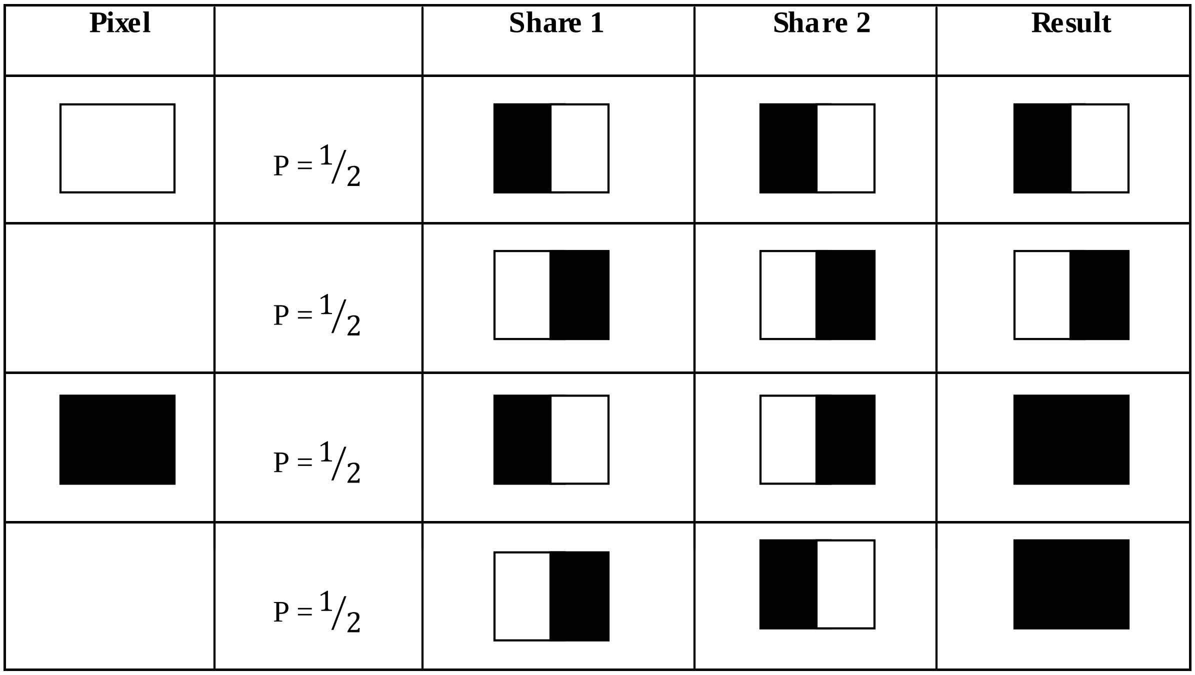


Figure-3: Partitions for black and white pixels for 2-out-of-2 scheme (2 subpixels)

2.3.2 Two-out-of-Two Scheme (4 subpixels)

The original problem of visual cryptography is the special case of a Two-out-of-Two visual secret sharing problem. It can be solved with 2 subpixels per pixel, but in practice this can distort the aspect ratio of the original image. It is thus recommended to use 4 subpixels arranged in a 2 × 2 array where each share has one of the visual forms in Figure-4. A white pixel is shared into two identical arrays from the list, and a black pixel is shared into two complementary arrays from the list. Any single share is a random choice of two black and two white subpixels, which looks medium grey. When two shares are stacked together, the result is either medium grey (which represent white) or completely black.

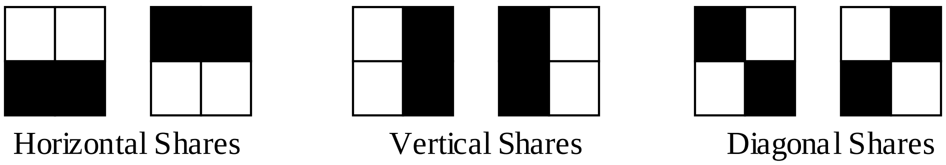


Figure-4: Partitions for black and white pixels for 2-out-of-2 scheme (4 subpixels)

Note that the horizontal, vertical and diagonal shares described in Figure-4, is used to solve the following 3-out-of-3 scheme. The six shares described below by the rows of C_0 and C_1 are exactly the six 2 × 2 arrays of subpixels. By stacking all the three transparencies from C_0 and C_1 we can observe C_0 is only ¾ black whereas a C_1 is completely black.

C_0 = { all the matrices obtained by permuting the columns of $\begin{bmatrix} 0011 \\ 0101 \\ 0110 \end{bmatrix}$ }

C_0 = { all the matrices obtained by permuting the columns of $\begin{bmatrix} 1100 \\ 1010 \\ 1001 \end{bmatrix}$ }

In Figure-5: it can be seen that a pixel, divided into four parts, can have six different states. If a pixel on layer 1 has a given state, the pixel on layer 2 may have one of two states: identical or inverted to the pixel of layer 1. If the pixel of layer 2 is identical to layer 1, the overlaid pixel will be half black and half white. Such overlaid pixel is called grey or empty. If the pixels of layer 1 and 2 are inverted or opposite, the overlaid version will be completely black. This is an information pixel. If the pixel states of layer 1 are truly (crypto secure) random, both empty and information pixels of layer 2 will also have completely random states. One cannot know if a pixel in layer 2 is used to create a grey or black pixel, since we need the state of that pixel in layer 1 (which is random) to know the overlay result.

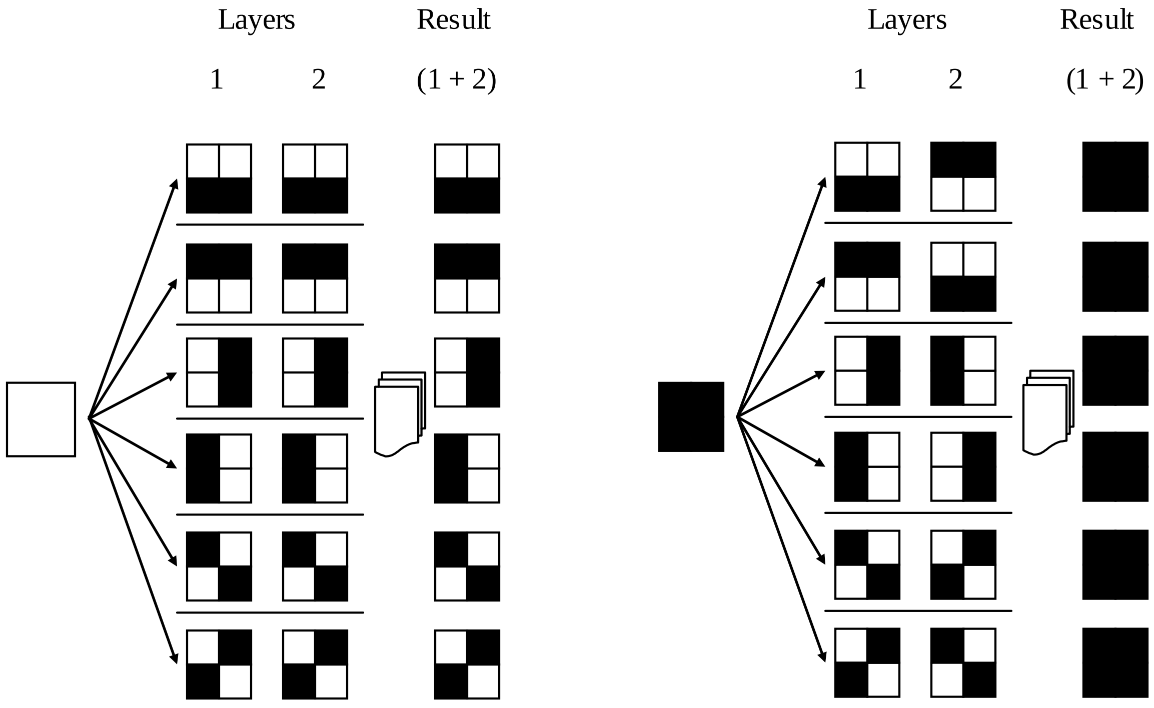


Figure-5: Superimposition of black and white subpixels for 2-out-of-2 scheme using 4 subpixels

If all the requirements for true randomness are fulfilled, Visual cryptography offers absolute secrecy according to the information. The sender will distribute one or more random layers 1 in advance to the receiver. If the sender has a secret message he creates a layer 2 for a particular distributed layer 1 and sends it to the receiver. The receiver aligns the two layers and the secret information is revealed, this without the need for an encryption device, a computer or performing any mathematical calculations by hand. The whole system is unbreakable, as long as both layers don't fall in the wrong hands. When one of both layers is intercepted it's impossible to retrieve the encrypted information.

In general there are four criteria's, used to evaluate the performance of a (k, n) Visual secret sharing scheme. The first criterion is security: fewer k shadows offer no information about the secret image, where $k \leq n$. The second criterion is accuracy: it is similarity between the reconstructed image and the original one. The next criterion is computational complexity: the number of operations is required to produce shadows and to generate the reconstructed image. The last criterion is the size of a shadow, which is also called the pixel expansion problem.

2.4 Proposed Approach

I propose a new recursive hiding of secrets, with applications to both images and text, to increase the efficiency of visual cryptography and to make it possible to incorporate additional secret information that serves as a steganographic channel [6]. I extend the idea of recursive hiding of secretes to 3 out of 5 threshold scheme and apply it to both images and text. However we deal with only binary images and regard each pixel as one bit of information, denoting black or white pixel.

The idea used is to hide smaller secrets in the shares of larger secrets without an expansion in the size of the later. While the scheme proposed in [5] is a non-threshold scheme, here the size of the secret increases by a factor of two as one goes from the smallest to the largest. The smallest secret would be a single bit. At the next level it will be an image of size 2 × 1 the next will be an image of size 2 × 2 and so on, until the full image size has been reached. Schemes in [6, 17-20] are threshold schemes. In [17], a tree data structure is used for recursive encoding of text such that the internal nodes of the tree also carry information in addition to leaves of the tree. Repeated application of Shamir's scheme is used in [18] to share k-1 secrets in n shares. However, the general threshold recursive schemes in [17-20] are not visual cryptography schemes. The scheme in [6], although a recursive visual cryptography scheme, is restricted to 2-out-of-3 scheme. For text represented as a binary sequence, a 3 out of 5 secret sharing scheme can be developed on a comparison based algorithm as follows: I divide the secret bit into 5 pieces p_1, p_2, p_3, p_4 and p_5 such that $p_1 = p_2 = p_3 = p_4 = p_5$ or any 3 combinations of p_1, p_2, p_3, p_4 and p_5 must be equal, if we wish to encode bit 0 and $p_1 \neq p_2 \neq p_3 \neq p_4 \neq p_5$ if we wish to encode bit 1.

To satisfy the above conditions there would be at least 5 symbols, say 0, 1, 2, 3, 4. Therefore to encode bit 0 we could create pieces $p_1p_2p_3p_4\ p_5$ as 00000, 11111, 22222, 33333, or 44444. The candidate to encode bit 1 is 01234 and all possible permutations of it, i.e., 04123, 12304, 21340, etc. In all, to encode secret bit 0 and secret bit 1, we have 5 and 120 possibilities, respectively, out of which any one can be chosen to satisfy our requirement based on the secret encoded.

Secret	**Shares**	
M_1: 1	0	S_{M11}
	2	S_{M12}
	4	S_{M13}
	3	S_{M14}
	1	S_{M15}
M_2: 01011	**0**1423	S_{M21}
	0**2**442	S_{M22}
	03**4**00	S_{M23}
	004**3**4	S_{M24}
	044**1**1	S_{M25}
M_3: 1101001111000101100011100	**01423**01234034101343413211	S_{M31}
	42413**02442**034304243442311	S_{M32}
	1343304310**03400**2443431011	S_{M33}
	204430304303420**00434**20111	S_{M34}
	34403001010344031434**04411**	S_{M35}
M_4: 125 bits	**0142301234034101343413211**	S_{M41}
	4241302442034304243442311	S_{M42}
	…	S_{M43}
	…	S_{M44}
	..	S_{M45}
M_5: 625 bits	625 bits (combination of 0, 1, 2, 3 and 4)	S_{M51}

	”	S_{M52}
	”	S_{M53}
	”	S_{M54}
	”	S_{M55}
M: 3125 bits	3125 bits (combination of 0, 1, 2, 3 and 4)	S_1
	”	S_2
	”	S_3
	”	S_4
	”	S_5

Table-1: Recursive hiding of smaller messages in the shares of larger messages

From Table-1 it can be observed that at each step I have used the shares of the previous smaller messages, these smaller shares are denoted in bold. Also, I have distributed the shares at each step so that no player has access to all the shares of the smaller message and hence, every message remains secure until at least three players come together. Since each bit is mapped into 5 shares, in order to take advantage of the recursive technique, the secrets at each step must increase by a factor of 5. We can then hide the following secrets M_1, M_2, M_3, M_4 and M_5 in S_1, S_2, S_3, S_4, and S_5. This approach is different from that discussed in [4], where the shares of smaller messages were all accumulated into one of the larger shares instead of distributing them among all the possible players. As a result, any individual having that share which encodes the smaller images could in principle recreate these smaller images without the help of the other individual, which in some cases might not be totally secured. Therefore, this new approach seems to be more secure for certain applications.

This idea can be further organized to a 3-out-of-*n* threshold scheme. Here I consider monochrome images, and each pixel (or subpixel) is considered to be one bit of information.

CHAPTER III

3. METHODOLOGY

The idea described in previous section can be applied to images to develop a recursive 3-out-of-5 visual cryptography scheme. For this purpose we need to construct black and white pixels using few properties.

3.1 Properties of 3-out-of-*n* Scheme ($n \geq 3$)

The 3 out of *n* visual secret sharing scheme can be solved by the following collections of n × n matrices:

C_0 = { all the matrices obtained by permuting the columns of $\begin{bmatrix} 100 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 100 & \cdots & 0 \end{bmatrix}$ }

C_1 = { all the matrices obtained by permuting the columns of $\begin{bmatrix} 100 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 000 & \cdots & 1 \end{bmatrix}$ }

The 3-out-of -3 scheme proposed in [2], can be generalized to 3 out of *n* scheme for an arbitrary *n* ≥ 3 using these properties.

1) Pixel Expansion, $m = 2n - 2$.

2) Relative Contrast, $\alpha = \frac{1}{2n} - 2$.

3) Let B be the black $n \times (n\text{-}2)$ matrix which contains only 1's.

4) Let I be the Identity $n \times n$ matrix which contains 1's on the diagonal and 0's elsewhere.

5) BI is an $n \times (2n\text{-}2)$ concatenated matrix.

6) c(BI) is the complement of BI.

7) C_0 contains matrices obtained permuting columns of c(BI).

8) C_1 contains matrices obtained permuting columns of BI.

9) Any single share contains an arbitrary collection of $(n\text{-}1)$ black & $(n\text{-}1)$ white sub-pixels.

10) Any pair of shares has $(n\text{-}2)$ common black & two Individual black sub-pixels.

11) Any stacked triplet of shares from C_0 has n black sub-pixels.

12) Any stacked triplet of shares C_1 has $(n+1)$ black sub-pixels.

Based upon the following properties we can design the matrix for 3(k)-out-of-5(n) scheme.

Let B be the black $n \times (n\text{-}2)$ matrix which contains only 1's.

B = $n \times (n\text{-}2) \rightarrow 5 \times (5\text{-}2) = 5 \times 3$

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}_{5 \times 3}$$

Let I be the Identity $n \times n$ matrix which contains 1's on the diagonal and 0's elsewhere.

I = $n \times n = 5 \times 5$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}_{5 \times 5}$$

Two collections of $n \times m$ Boolean matrices C_0 and C_1 are obtained by permuting the columns of c(BI) and BI

$m = 2n - 2 \rightarrow 2(5) - 2 = 8$

i.e, 5 × *8* Boolean matrices.

$$c(\mathrm{BI}) = \begin{array}{cc} \left[\begin{array}{cccccccc} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{array}\right]_{5 \times 8} & \begin{array}{cc} \rightarrow & \text{Share1} \\ \rightarrow & \text{Share2} \\ \rightarrow & \text{Share3} \\ \rightarrow & \text{Share4} \\ \rightarrow & \text{Share5} \end{array} \\ \begin{array}{cccccccc} [1] & [2] & [3] & [4] & [5] & [6] & [7] & [8] \end{array} & \end{array}$$

Hamming weight of c(BI) i.e. is White *H(V)* = 5

$$\mathrm{BI} = \begin{array}{cc} \left[\begin{array}{cccccccc} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right]_{5 \times 8} & \begin{array}{cc} \rightarrow & \text{Share1} \\ \rightarrow & \text{Share2} \\ \rightarrow & \text{Share3} \\ \rightarrow & \text{Share4} \\ \rightarrow & \text{Share5} \end{array} \\ \begin{array}{cccccccc} [1] & [2] & [3] & [4] & [5] & [6] & [7] & [8] \end{array} & \end{array}$$

Hamming weight of BI i.e. is Black *H(V)* = 8

C_0 = {all the matrices obtained by permuting the columns of c(BI)}

C_1 = {all the matrices obtained by permuting the columns of BI}

If the columns are not permuted then there is a possibility to reveal the secret information in any single share and therefore the process fails.

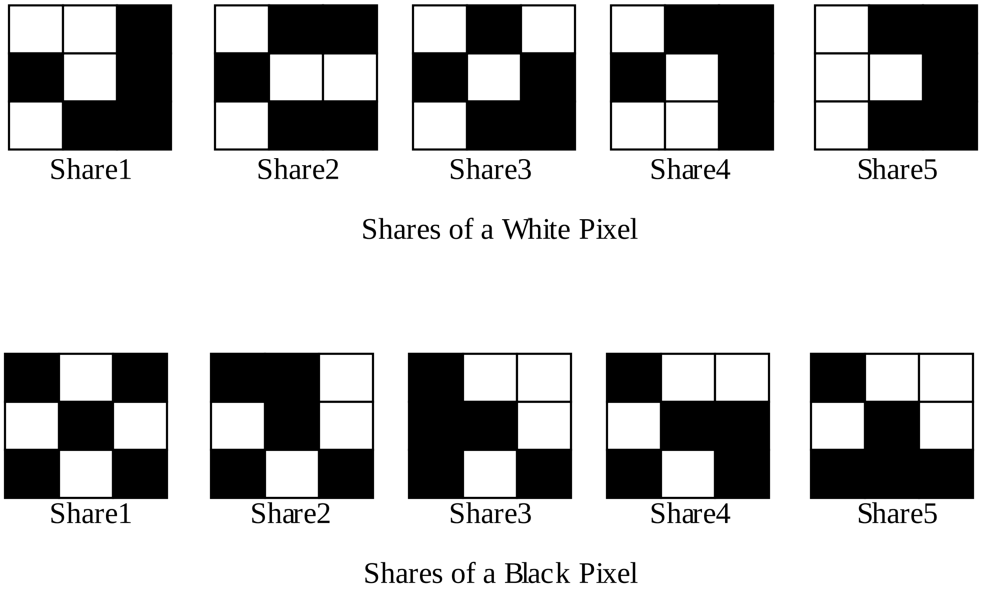
For example here are few different permutations.

{[1] [8] [2] [7] [3] [6] [4] [5]}

{[2] [4] [6] [8] [1] [3] [5] [7]}

{[3] [2] [1] [8] [7] [6] [4] [5]}

{[5] [8] [1] [6] [2] [3] [7] [4]}



Shares of a White Pixel

Shares of a Black Pixel

Figure-6: Possible partitions for black and white pixels

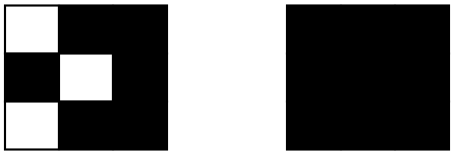
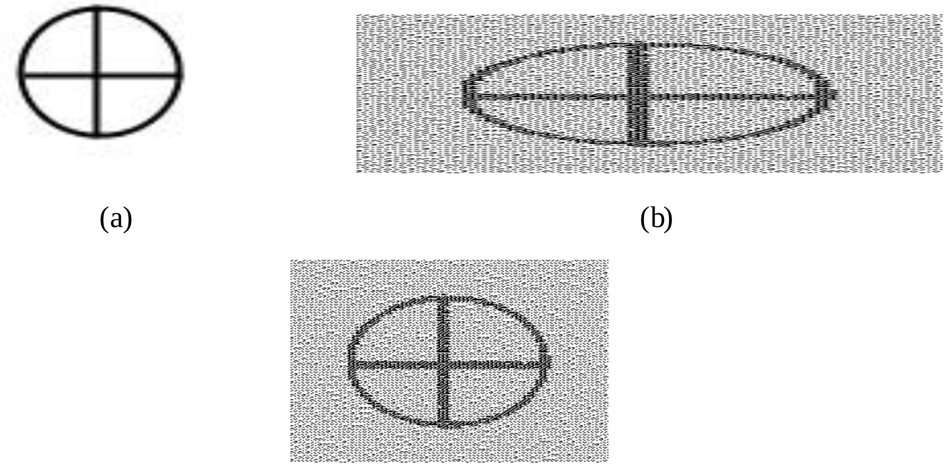


Figure-7: Superimposition of white and black subpixels

3.2 Importance of Aspect Ratio in Visual Cryptography

Pixel expansion plays a vital role in visual secret sharing schemes. Here I used *m*, the pixel expansion, subpixels to represent a pixel. Suppose the secret image is a solar system symbol earth and *m* is not a square value, i.e., the aspect ratio is changed. After performing visual secret sharing scheme, the original symbol will be changed to an ellipse and consequently lead to loss of information.



(a) (b)

(c)

Figure-8: Distortion due to the variant aspect ratio: (a) The original secret image; (b) The aspect ratio is changed; (c) The aspect ratio is unchanged

To avoid distorting the image, the dummy subpixels are added to keep the aspect ratio unchanged. In the 3-out-of-5 scheme any single share contains 4 black and 4 white pixels, so to make it a complete square array without distorting their aspect ratio, we need to add one more pixel. It should be either black or white.

$$\text{White Pixel} \rightarrow [0] \rightarrow \begin{bmatrix} 000 \\ 000 \\ 000 \end{bmatrix}_{3 \times 3} \qquad \text{Black Pixel} \rightarrow [1] \rightarrow \begin{bmatrix} 111 \\ 111 \\ 111 \end{bmatrix}_{3 \times 3}$$

For each white and black pixel we need to transform these matrices; by this way we can maintain the aspect ratio of the image.

3.3 Recursive Information hiding in 3-out-of-5 Scheme

Recursive information hiding is a technique where certain additional secret information can be hidden in one of the shares of the original secret image [5]. Here the secret information which we are going to hide is taken according to size, i.e. small images to larger. The first small secret image is divided into five different shares using visual cryptography. These shares are placed in the next level to create the shares of larger secret information. We distribute the shares at each consecutive level so that no one has access to all the shares of the smaller images, unless until at least three participants come together to reveal the secret information, resulting in *3* out of *5* scheme.

Figure-9 is an example to help readers to understand the concept visually. The original secret image under consideration is of size 5 × 5 and the first secret image is of size 1 × 1. There are totally five shares obtained for the first secret image based on the basic idea of visual cryptography. The second secret image is of size 5 × 1. To obtain the second secret image we are going to use the shares of first secret image and they are placed in different levels (level 1-5) one after the other in five different shares. Now the shares of second secret image are designed by seeing the shares of first secret image and the original second secret image.

From Figure-9, we can see share 2 of the first secret image is placed on share 2 of the second secret image (level 2), and this share is a part of complete black pixel. The original second secret image has a complete white pixel under level 2, so we need to design all the shares of level 2 of second secret image by comparing both the shares of first secret image and the original second secret image to get a partial white pixel when we combine all the 5 shares or at least 3 shares in our case. This process is recursively repeated for all the shares. By doing this we can obtain the information of original second secret image by combining any 3 shares. This is the process for recursive information hiding of images. From Figure-7, we can see that the partitions of white pixel are stacked upon each other three fifth of the pixel is white and hence appears light gray to human eye. However, the sub-pixels of the black pixel are not complete black when 3 shares are stacked together but it is completely black when all the 5 shares are stacked.

Figure-10 shows an illustration of recursive information hiding. The original secret image considered is the Lena image of size 380 × 390. The first secret image is stick figure of size 78 ×

78 and the second secret image is a text of size 380 × 78 and these both are hidden recursively. Shares of original secret image are constructed by using the shares of second secret image and placed in different levels and the process gets repeated and finally when we combine any 3 shares the original secret information is revealed.

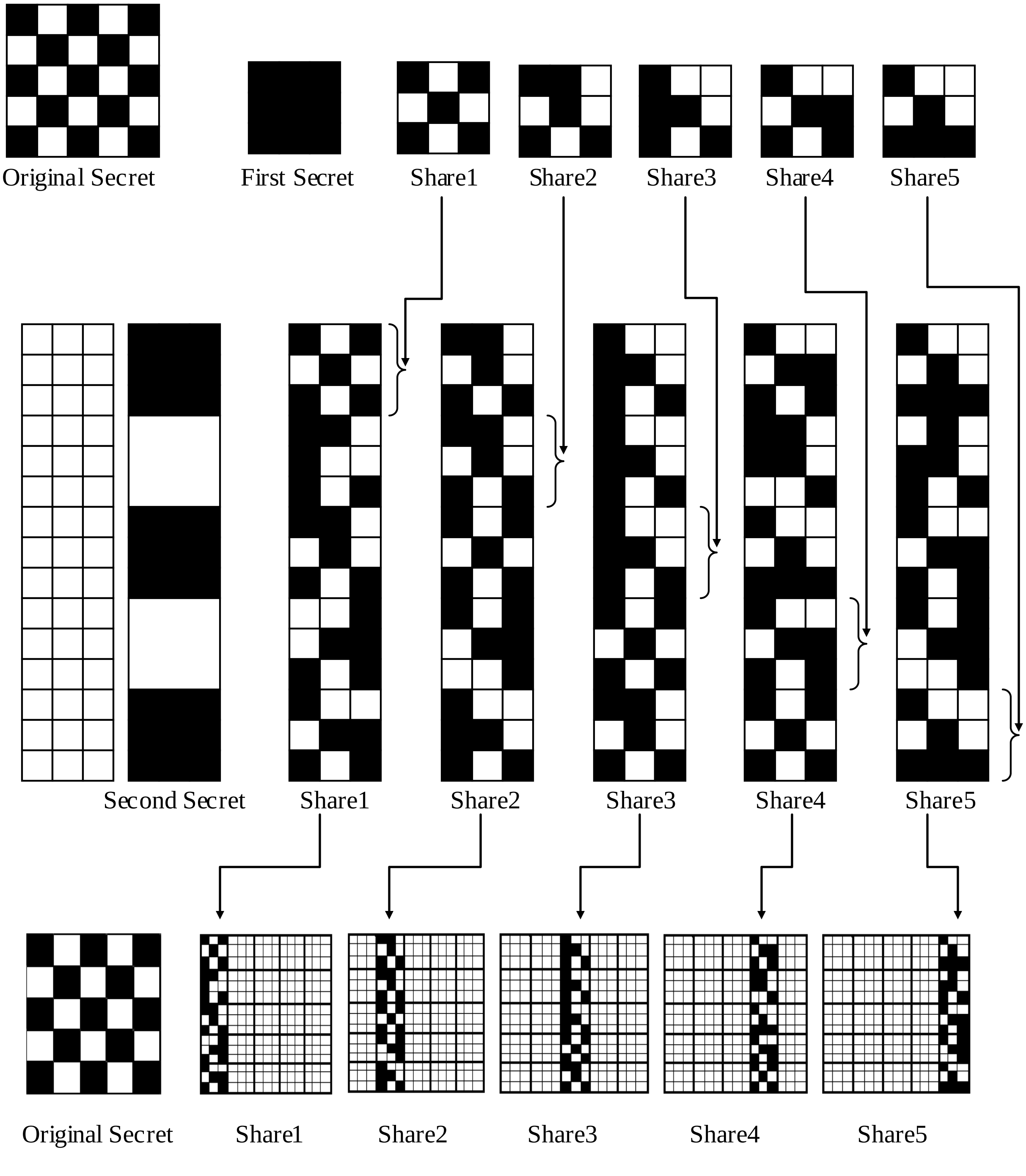


Figure-9: Representation of Recursive Information Hiding of secret images in the shares of larger original image using a 3-out-of-5 threshold scheme

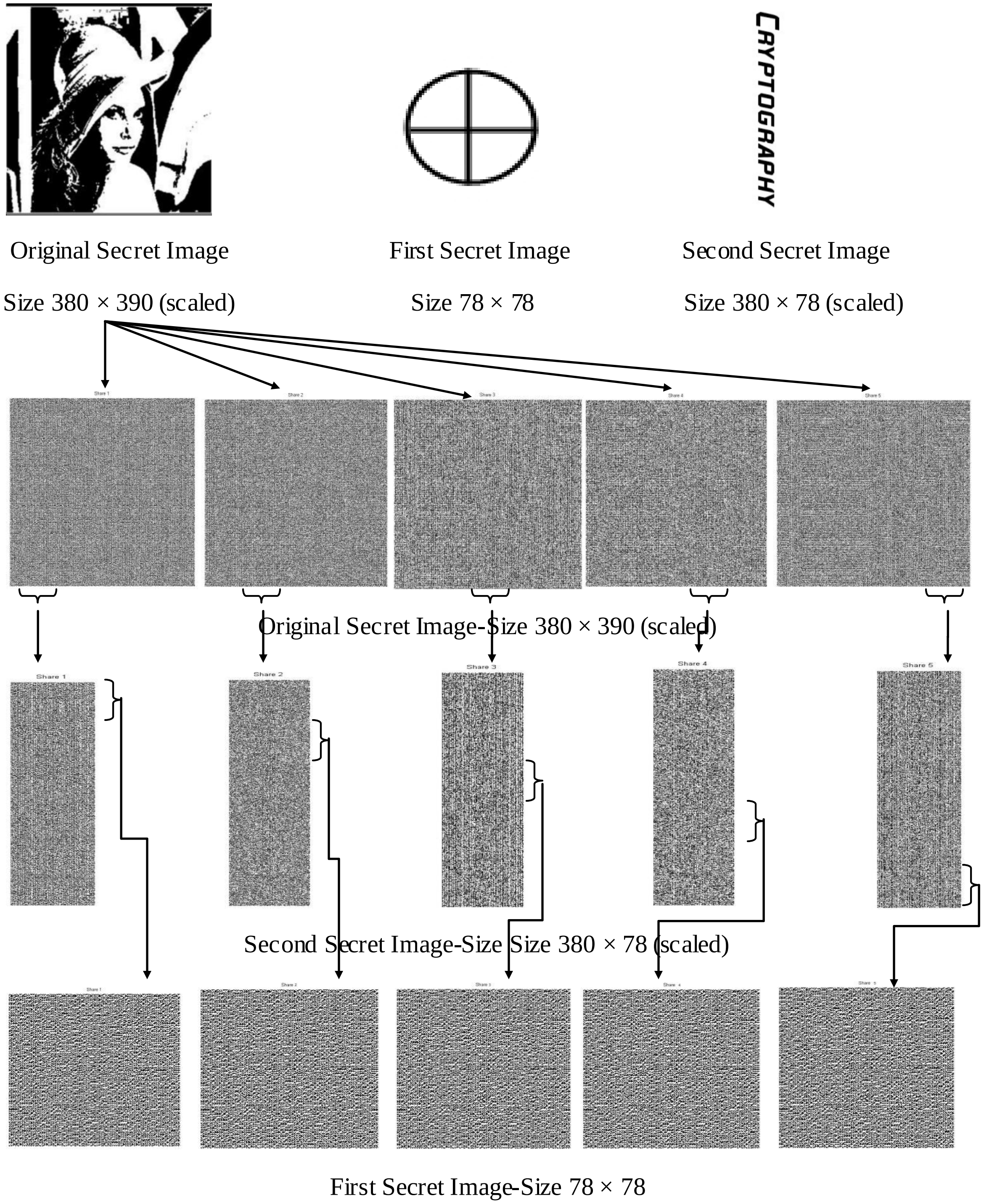
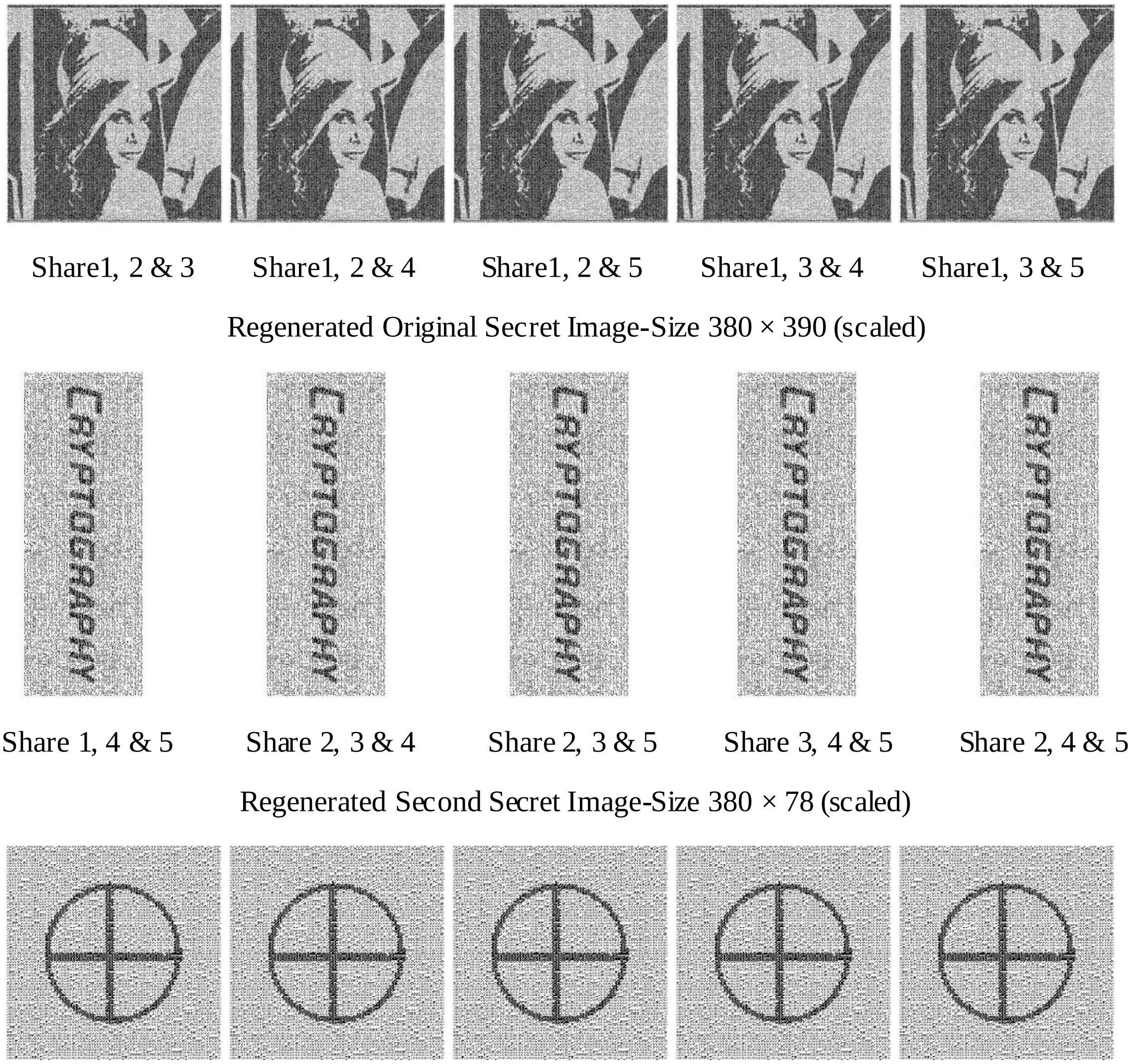


Figure-10: Interpretation of the process of recursive information hiding of secrets in shares of larger original image

3.4 Simulation Results for Recursive Information hiding in 3-out-of-5 Scheme

After simulation these are few images than can reveal the secret information by combining any 3 shares out of 5 shares.



Share1, 2 & 3 Share1, 2 & 4 Share1, 2 & 5 Share1, 3 & 4 Share1, 3 & 5

Regenerated Original Secret Image-Size 380 × 390 (scaled)

Share 1, 4 & 5 Share 2, 3 & 4 Share 2, 3 & 5 Share 3, 4 & 5 Share 2, 4 & 5

Regenerated Second Secret Image-Size 380 × 78 (scaled)

Share 2, 4 & 5 Share 1, 4 & 5 Share 1, 2 & 4 Share 2, 3 & 4 Share 1, 3 & 5

Regenerated First Secret Image-Size 78 × 78

Figure-11: Regenerated smaller images from the shares hidden inside the shares of the original larger image

3.5 A General k-out-of-k Scheme

For all k there exists a general construction of k-out-of-k visual secret sharing scheme, the pixel expansion must use at least 2^{k-1} pixels, and the relative contrast should be $\frac{1}{2^{k-1}}$. There is a need to construct two collections of $k \times 2^{k-1}$ Boolean matrices i.e. S^0 and S^1.

1) S^0 Handles the white pixels.
2) S^1 Handles the black pixels.

All 2^{k-1} columns have an even number of 1's in S^0 and odd number of 1's in S^1 and no two k rows are same in both S^0 & S^1. C_0 and C_1 contains all permutations of columns in S^0 & S^1.

3.5.1 Properties of k-out-of-k Scheme

1) Pixel Expansion, $m = 2^{k-1}$ (m should be as small as possible)
2) Relative Contrast, $\alpha = \frac{1}{2^{k-1}}$ (α should be as large as possible)
3) r, the size of the collections. C_0 and C_1 (they need not be the same size, but in all of our constructions they are). Here $r = 2^{k-1}!$
4) $\log r$ is number of random bits needed to generate share.

By Naor and Shamir, any k out of k scheme as $\alpha \leq \frac{1}{2^{k-1}}$ and $m \geq 2^{k-1}$

Based upon the following properties we can design the matrix for $k = 3$ and $k = 4$

k = 3, therefore m = 4, α = ¼ and r = 24

$$S^0 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}_{3 \times 4} \qquad S^1 = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{bmatrix}_{3 \times 4}$$

k = 4, therefore m = 8, $\alpha = \frac{1}{8}$ and r = 40320

W = {1, 2, 3, 4}

Even cardinality subsets {{}, {3, 4}, {2, 4}, {2, 3}, {1, 4}, {1, 3}, {1, 2}, {1, 2, 3, 4}}

Odd cardinality subsets {{4}, {3}, {2}, {2, 3, 4}, {1}, {1, 3, 4}, 1, 2, 4}, {1, 2, 3}}

$$S^0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}_{4 \times 8} \begin{array}{ll} \rightarrow & \text{Share1} \\ \rightarrow & \text{Share2} \\ \rightarrow & \text{Share3} \\ \rightarrow & \text{Share4} \end{array}$$

Hamming weight of S^0 i.e. is White $H(V)$ = 7

$$S^1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}_{4 \times 8} \begin{array}{ll} \rightarrow & \text{Share1} \\ \rightarrow & \text{Share2} \\ \rightarrow & \text{Share3} \\ \rightarrow & \text{Share4} \end{array}$$

Hamming weight of S^1 i.e. is Black $H(V)$ = 8

We need two collections of 4 × 8 Boolean matrices C_0 and C_1, contains all permutations of columns in S^0 and S^1, by these two matrices we can design the shares of black and white pixels.

Based upon the following properties we can also design the matrix for $k = 5$ and $k = 5$

k = 5, therefore m = 16, $\alpha = \frac{1}{16}$ and r = 16!

W = {1, 2, 3, 4, 5}

S^0 Contain the 2^{k-1} ➔ 2^4 (16) Vectors with even no: of 1's

S^1 Contain the 2^{k-1} ➔ 2^4 (16) Vectors with odd no: of 1's

$$S^0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}_{5 \times 16} \begin{array}{ll} \rightarrow & \text{Share1} \\ \rightarrow & \text{Share2} \\ \rightarrow & \text{Share3} \\ \rightarrow & \text{Share4} \\ \rightarrow & \text{Share5} \end{array}$$

Hamming weight of S^0 i.e. is White $H(V)$ = 15

$$S^1 \quad = \quad \left[\begin{array}{cccccccccccccccc} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right]_{5 \times 16} \begin{array}{cc} \rightarrow & \text{Share1} \\ \rightarrow & \text{Share2} \\ \rightarrow & \text{Share3} \\ \rightarrow & \text{Share4} \\ \rightarrow & \text{Share5} \end{array}$$

Hamming weight of S^1 i.e. is Black $H(V) = 16$

We need two collections of 5 × 16 Boolean matrices C_0 and C_1, contains all permutations of columns in S^0 and S^1, by these two matrices we can design the shares of black and white pixels. Similarly as above, we can apply recursive scheme for any k-out-of-k scheme.

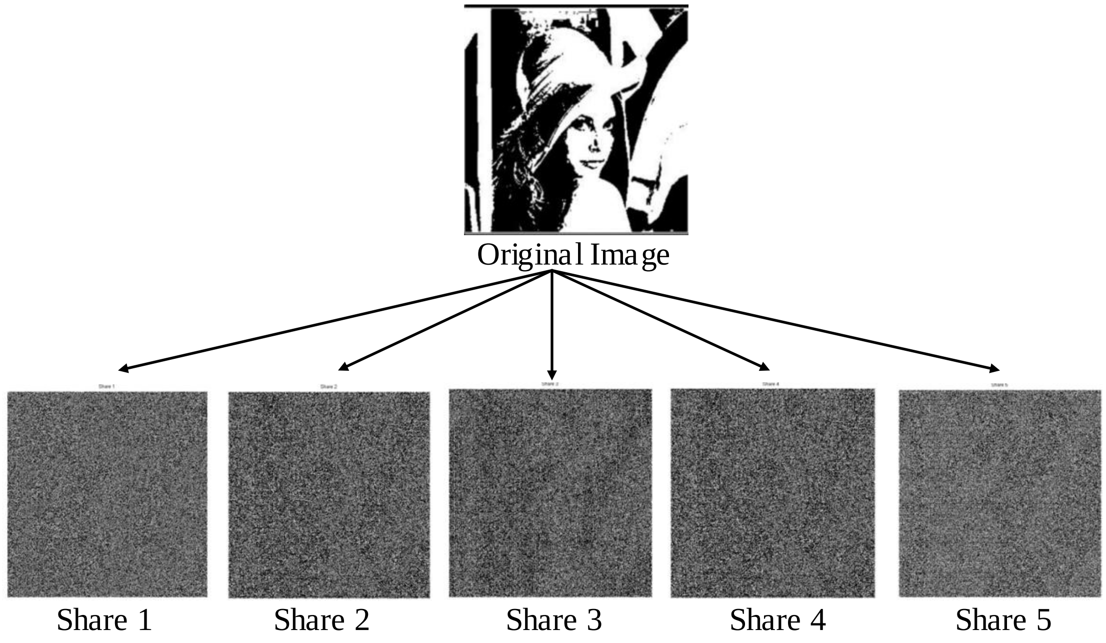
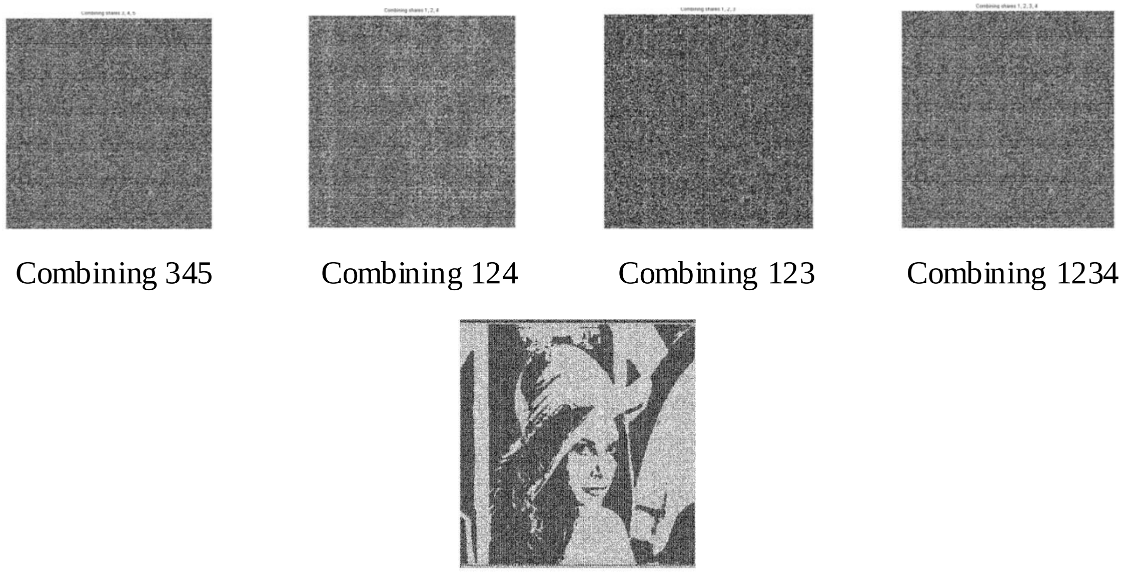


Figure-12: Illustration of 5-out-of-5 scheme



Combining 345 Combining 124 Combining 123 Combining 1234

Final Decoded Image (12345)

Figure-13: Regenerated Images from 5-out-of-5 scheme

3.6 A General k-out-of-n Scheme

A general k-out-of-n scheme is designed from k-out-of-k scheme. Let C be k out of k visual secret sharing scheme with parameters m, r, α. The scheme C consists of two collections of $k \times m$ Boolean matrices and $C_0 = T_1^0, T_2^0, ... T_r^0$ and $C_1 = T_1^1, T_2^1, ... T_r^1$. H is a collection of l functions $\forall h \in H, h : \{1 .... n\} \rightarrow \{1 .... k\}$. Let B be the subset of $\{1 .... n\}$ of size k and β_q is probability that randomly chosen function $h \in H$ yields q different values on B, $1 \leq q \leq k$.

We construct from C and H a k out of n scheme $C^{|}$ with parameters $m^{|} = m . l$, $\alpha^{|} \geq \beta_k \alpha$, $r^{|} = r^l$

1) The ground set is $V = U \times H$

2) Each $1 \leq t \leq r^l$ is indexed by a vector $(t_1, t_2, \ldots . t_l)$ where each $1 \leq t \leq r$.

3) The matrix S_t^b for $t = (t_1, t_2, \ldots . t_l)$ where $b \in \{0, 1\}$ is defined as

$$S_t^b[i,(j,h)] = T_{tj}^b[h(i),j]$$

Contrast

Contrast should be $\geq \beta_k \alpha$

1) k rows is S_t^b, mapped to $q < k$ different values by h

2) Hamming weight of OR of q rows is *f(q)*

3) Difference is αm white and black pixels occurs when h is one and happens at β_k

4) WHITE: $H(v) \leq l(\beta_k(d - \alpha m) + \sum_{q=1}^{k-1} \beta_q . f(q))$

5) BLACK: $H(v) \geq l(\beta_k + \sum_{q=1}^{k-1} \beta_q . f(q))$

Security

Security properties of the k-out-of-k scheme imply the security of k-out-of-n scheme because we are using $(k,\ k)$ scheme to create $(k,\ n)$ scheme. The expected hamming weight of OR of q rows, $q < k$ is $l \sum_{q=1}^{k-1} \beta_q . f(q)$ irrespective of WHITE or BLACK pixel.

3.7 Secret Sharing Scheme for Grayscale Images

Grayscale images are different from black and white images. Grayscale is a range of shades of gray without apparent color. According to their physical characteristics, different media use different ways to represent the color level of images. The computer screen uses the electric current to control lightness of the pixels. The diversity of the lightness generates different color levels. The general printer, such as dot matrix printers, laser printers, and jet printers can only control a single pixel to be printed (black pixel) or not to be printed (white pixel), instead of displaying the gray level. As such, the way to represent the gray level of images is to use the density of printed dots. The method that uses the density of the net dots to simulate the gray level is called “Halftone” and transforms an image with gray level into a binary image before processing. Every pixel of the transformed halftone image has only two possible color levels (black or white). Because human eyes cannot identify too tiny printed dots and, when viewing a dot, tend to cover its nearby dots, we can simulate different gray levels through the density of printed dots, even though the transformed image actually has only two colors – black and white.

Several schemes for grayscale images [20] and for color images [20, 21] have been proposed. However, all of these earlier works result in a decrypted image of reduced quality. I here proposed a new gray- level visual cryptography scheme and the image quality in this proposed scheme is better than anything and provides high quality images including that of perfect (original) quality to be reconstructed.

3.7.1 Gray-Level Visual Cryptography

In my scheme I convert each grayscale block into a binary block. First of all each pixel value in a grayscale block is transformed into binary representation. For example take a grayscale block and transform into binary blocks.

$$\begin{bmatrix} 111 & 159 & 20 \\ 254 & 10 & 198 \\ 40 & 215 & 100 \end{bmatrix}$$

Its corresponding binary blocks are as follows:

[0 1 1 0 1 1 1 1] [1 0 0 1 1 1 1 1] [0 0 0 1 0 1 0 0];

[1 1 1 1 1 1 1 0] [0 0 0 0 1 0 1 0] [1 1 0 0 0 1 1 0];

[0 0 1 0 1 0 0 0] [1 1 0 1 0 1 1 1] [0 1 1 0 0 1 0 0].

Take each binary block and go for different possible combinations of that block, and try to design the block into different shares. For example take a grayscale block and divide the block into shares and apply the above scheme.

3.7.2 Two-out-of-Three Scheme using Grayscale Images

This proposed scheme is totally different from that of previous schemes. Here I design the shares such a way that when combining any two shares will reveal the original bit information, but not the whole share just half of each single share will give me high quality image when reconstructed. I will explain this scheme by taking a value from the grayscale block and divide that value into shares.

254: [1 1 1 1 1 1 1 0]

	1st half	2nd half
Share1:	0 1 0 1 0 1 0 0	1 1 0 1 1 0 1 0
Share2:	1 0 1 0 1 0 1 0	1 1 1 0 1 1 1 0
Share3:	0 0 1 0 0 1 0 0	1 0 0 1 0 1 0 0

Table-2: Grayscale bits are transformed into Binary bits

Share1(1st half): 0 1 0 1 0 1 0 0

Share2(1st half): 1 0 1 0 1 0 1 0

1 1 1 1 1 1 1 0 = 254

Share3(1st half): 0 0 1 0 0 1 0 0

Share1(2nd half):1 1 0 1 1 0 1 0

1 1 1 1 1 1 1 0 = 254

Share2(2nd half): 1 1 1 0 1 1 1 0

Share3(2nd half): 1 0 0 1 0 1 0 0

1 1 1 1 1 1 1 0 = 254

Combining any two half shares will give me exact bit and by doing the same procedure for the whole grayscale block gives me perfect high quality image when reconstructed without any loss of contrast.

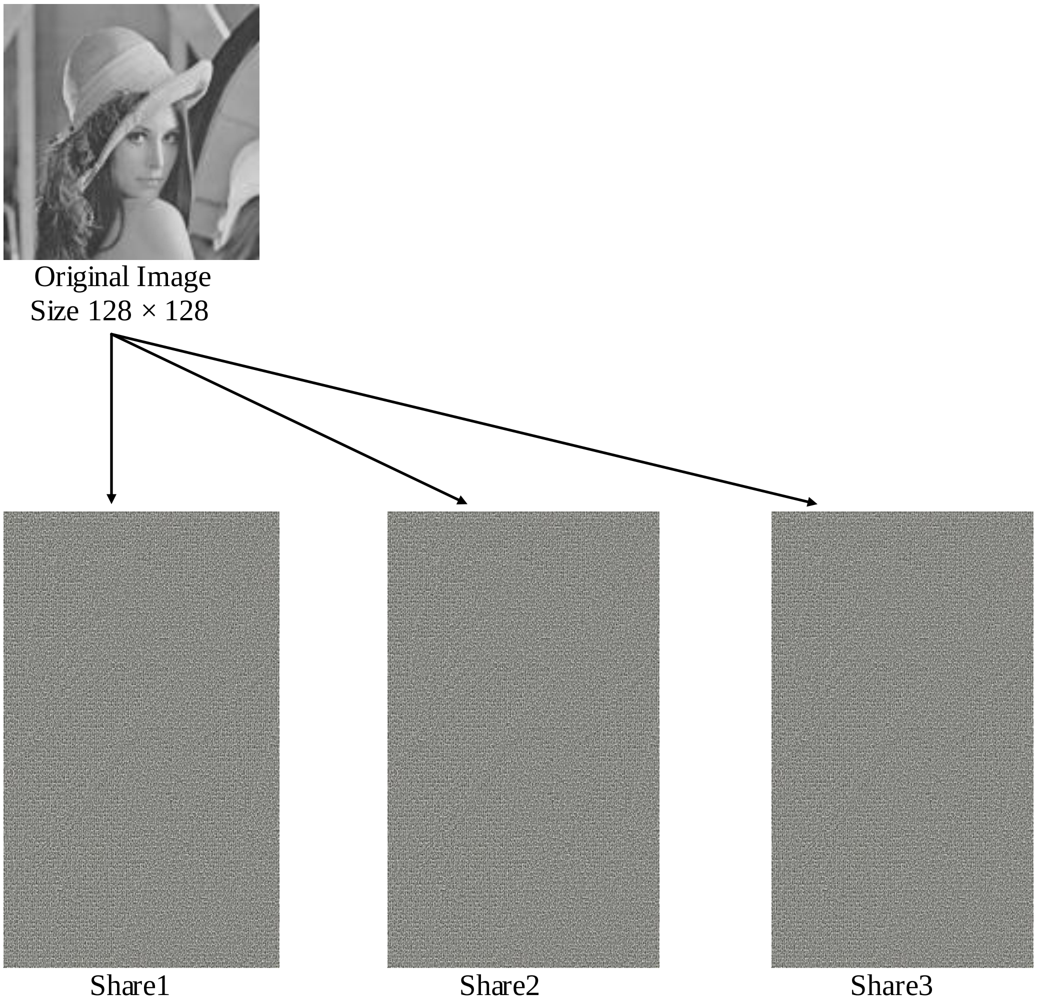
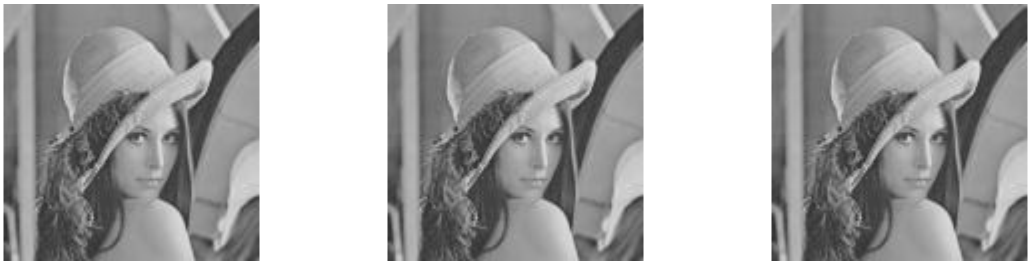


Figure-14: Generating three separate shared transparencies for gray-level visual cryptography

The beauty of this scheme is, when you combine the direct shares you can't see a perfect gray-scale image only when you combine the half shares, the original quality of the image will be revealed without any loss of generality.

3.7.3 Simulation Results for 2-out-of-3 Scheme using Grayscale



Share1(1st half)

&

Share2(1st half)

Share3(1st half)

&

Share1(2nd half)

Share2(2nd half)

&

Share3(2nd half)

Figure-15: Stacking of gray-level visual cryptography

CHAPTER IV

4. CONCLUSION

Visual cryptography provides a secure way to transfer images. The advantage of visual cryptography is that it exploits human eyes to decrypt secret images with no computation required.

Visual Cryptography allows easy decoding of the secret image by a simple stacking of the printed share transparencies. However, there are some practical issues that need c areful consideration. The transparencies should be precisely aligned in order to obtain a clear reconstruction of the secret image. There is also some unavoidable noise introduced during the printing process. Furthermore, the stacking method can only simulate the OR operation which always leads to a loss in contrast. The loss of contrast can be rectified by further processing.

As visual cryptography schemes operate at the pixel levels, each pixel on one share must be matched correctly with the corresponding pixel on the other share. Superimposing the shares with even a slight change in the alignment results in a drastic degradation in the quality of the reconstructed image.

The thesis described two new probabilistic schemes for black and white and gray-scale images. For black and white images recursive information hiding technique was used in which smaller secrets are hidden in the shares of larger secrets without an expansion in the size. This hiding technique can be applied to many applications in real and cyber world. For gray-scale images I proposed a new scheme that gives perfect quality images but it is not directly a scheme that is a straightforward superposition of shares.

REFERENCES

[1] Kafri, O and Keren, E. 1987. Encryption of pictures and shapes by random grids. *Optics Letters* 12: 377-379.

[2] Naor, M. and Shamir, A. 1995. Visual Cryptography. Advances in Cryptography-Eurocrypt, 950: 1-12.

[3] Shamir, A. 1979. How to Share a Secret. *Communications of the ACM*. 22: 612-613.

[4] Horng, G., Chen, T. and Tsai, D. 2006. Cheating in Visual Cryptography. *Design, Codes and Cryptography* 38: 219-236.

[5] Gnanaguruparan, M. and Kak, S. 2002. Recursive Hiding of Secrets in Visual Cryptography. *Cryptologia* 26:68-76.

[6] Parakh, A. and Kak, S. 2008. A Recursive Threshold Visual Cryptography Scheme. *Cryptology ePrint Archive, Report* 2008/535.

[7] Ching-Nung, Y. and Tse-Shih, C. 2004. Aspect Ratio Invariant Visual Secret Sharing Schemes with Minimum Pixel Expansion. *Pattern Recognition Letters* 26: 193-206.

[8] Stinson, D. 1995. Cryptography Theory and Practice. *CRC Press*.

[9] Katoh, T. and Imai, H. 1996. Some Visual Secret Sharing Schemes and Their Share Size. *Joint Conference of 1996 International Computer Symposium*, Kaohsiung, Taiwan, R.O.C., pages 19-21.

[10] Verheul, E.R. and Van Tilborg, H.C.A. 1997. Construction and Properties of *k* out of *n* Visual Secret Sharing Schemes. *Designs, Codes and Cryptography*, 2: 179-196.

[11] Droste, S. 1996. New Results on Visual Cryptography. *Advances in Cryptology-CRYPTO `96, Lecture Notes in Computer Science*, 1109: 401-415.

[12] Prisco, R. and Santis, A. 2006. Cheating Immune Threshold Visual Secret Sharing. *LNCS* 4116: 216-228.

[13] Blakely, G. R. 1979. Safeguarding Cryptographic Keys. *Proceedings of the National Computer Conference, American Federation of Information Processing Societies Proceedings*. 48: 313-317.

[14] Kak, S. 1982. On Asymmetric Secret Sharing. *LSU ECE Technical Report*. May.

[15] Alon, N. and Spencer, J. 1992. The Probabilistic Method, Wiley-Interscience, 2nd edition.

[16] MacWilliams, F. J. and Sloane, N. J. A. 1977. The Theory of Error Correcting Codes, North Holland, Amsterdam.

[17] Parakh, A. and Kak, S. 2010. A Tree Based Recursive Information Hiding Scheme. To appear in proceedings of IEEE ICC 2010 – *Communication and Information System Security Symposium* ('ICC'10 CISS'), May 23-27, Cape Town, South Africa.

[18] Parakh, A. and Kak, S. 2009. Recursive Secret Sharing for Distributed Storage and Information Hiding, Advanced Networks and Telecommunication Systems (ANTS), *IEEE 3rd International Symposium* on, pages 1-3, 14-16.

[19] Parakh, A. and Kak, S. 2009. Space Efficient Secret Sharing: A recursive approach. *Cryptology ePrint Archive, Report* 2009/365.

[20] Innes Muecke. 1999. Greyscale and Colour Visual Cryptography, Thesis of degree of Master of Computer Science, Dalhouse University – Daltech.

[21] Blude, C., De Santis, A. and Naor, M. 2000. Visual Cryptography for Grey Level Images, *Information Processing Letters*, Vol. 27, pp. 255-259.

[22] Alfred, J, Paul, C and Scott, A. 1965. Handbook of Applied Cryptography, *Library of Congress Cataloging-in-Publication Data.*

VITA

Sandeep Katta

Candidate for the Degree of

Master of Science

Thesis: SECRET SHARING IN VISUAL CRYPTOGRAPHY

Major Field: Computer Science

Biographical:

Education:

Completed the requirements for the Master of Science in Computer Science at Oklahoma State University, Stillwater, Oklahoma in December, July, 2011.

Completed the requirements for the Bachelor of Engineering in Electronics and Communication Engineering at Anna University, Chennai, India in 2008.

Experience:

Graduate Assistant / Web Developer, System Administrator & Technical Support, Distance Education, OSU, Aug 2010 – Present

• Design & maintenance of distance learning website using Joomla, a Content Management System - http://de.ceat.okstate.edu
• Provide technical support to students, faculty and classroom.
• Schedule & recording class videos, troubleshoot all kinds of hardware and software problems and work on D2L (Desire to Learn) environment.
• Responsible for purchase, installation and configuration of all new hardware and software

Graduate Assistant / Web Developer, Engineering Technology Management, OSU, July 2009 – Present

• Designed a website for Engineering Technology Management - http://etm.ceat.okstate.edu/
• Maintain & organize a database using MS Access for student records

Professional Memberships: Computer Society of India

Name: Sandeep Katta Date of Degree: July, 2011

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: SECRET SHARING IN VISUAL CRYPTOGRAPHY

Pages in Study: 43 Candidate for the Degree of Master of Science

Major Field: Computer Science

Scope and Method of Study:

This thesis examines techniques for recursive hiding scheme for 3 out of 5 secret sharing and a probabilistic 2 out of 3 secret sharing scheme for gray scale images. In recursive hiding of secrets several messages can be hidden in one of the shares of the original secret image. The images that are to be hidden are taken according to their sizes from smaller to the largest. The first small secret image is divided into five different shares using visual cryptography. These shares are placed in the next level to create the shares of larger secret information. The shares at each consecutive level are distributed so that no one has access to all the shares of the smaller images, unless at least three participants come together to reveal the secret information, resulting in 3 out of 5 scheme. In the proposed protocol for gray scale images, the quality of the image is perfect when it is reconstructed for the construction of the final image based on the binary OR operation.

Findings and Conclusions:

Through simulation and analysis, it is demonstrated that recursive hiding and gray scale secret sharing serve as steganographic channels that can be used to embed invisible watermarks, convey secret keys or encode authentication information.

ADVISER’S APPROVAL: Dr. Subhash Kak