A SECURE AND EFFICIENT MECHANISM FOR MULTIPLE SECRET SHARING IN VISUAL CRYPTOGRAPHY

A DISSERTATION

Submitted in partial fulfillment of the requirements for the award of the degree of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

JAYA

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE -247 667 (INDIA)
JUNE, 2011

CANDIDATE'S DECLARATION

I hereby declare that the work, which is being presented in the dissertation entitled "**A SECURE AND EFFICIENT MECHANISM FOR MULTIPLE SECRET SHARING IN VISUAL CRYPTOGRAPHY**" towards the partial fulfillment of the requirement for the award of the degree of **Master of Technology** in **Computer Science and Engineering** submitted in the Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee, Roorkee, Uttarakhand (India) is an authentic record of my own work carried out during the period from July 2010 to June 2011, under the guidance of **Dr. Anjali Sardana, Assistant Professor,** Department of Electronics and Computer Engineering, IIT Roorkee.

The matter presented in this dissertation has not been submitted by me for the award of any other degree of this or any other Institute.

Date: 9/6/11

Place: Roorkee

Jaya

(JAYA)

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Date: 9/6/11

Place: Roorkee

Anjali

(Dr. Anjali Sardana)
Assistant Professor
Department of Electronics and Computer Engineering
IIT Roorkee.

ACKNOWLEDGEMENTS

First and foremost, I would like to extend my heartfelt gratitude to my guide and mentor **Dr. Anjali Sardana**, Assistant Professor, Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee, for her invaluable advices, guidance, encouragement and for sharing her broad knowledge. Her wisdom, knowledge and commitment to the highest standards inspired and motivated me. She has been very generous in providing the necessary resources to carry out my research. She is an inspiring teacher, a great advisor, and most importantly a very nice person.

I am greatly indebted to all my friends, who have graciously applied themselves to the task of helping me with ample moral supports and valuable suggestions.

On a personal note, I owe everything to the Almighty and my parents. The support which I enjoyed from my father, mother and sisters provided me the mental support I needed.

Jaya

JAYA

Abstract

Cryptography is the art of protecting information by conversion of the data into a disordered code that can be sent across a public or private network and deciphered with the correct key which is kept with the designated receiver. An array of cryptographic techniques has been proposed for providing data security. However, most of the traditional cryptographic methods require complex algorithms for encryption and decryption. Visual cryptography is a technique for sharing images between two or more participants. It achieves the goals of security such as confidentiality and authentication without any complex computations. Here an image that has to be kept confidential (secret image) is divided into certain number of images called shares. The concept of share eliminates the use of key and encrypted data used in conventional cryptographic methods. The secret image can be reconstructed just from the stacking of the shares. Traditional visual cryptography methods produce random shares which are susceptible to attackers. Some methods have been proposed to generate innocent-looking shares so that attacker does not get doubtful by looking at the random pattern of the share used in conventional visual cryptography methods. The shares look like valid images and the adversary cannot judge whether they are part of secret image. However many of these techniques use additional data structure and take much time in encoding the secret.

In this dissertation entitled "A SECURE AND EFFICIENT MECHANISM FOR MULTIPLE SECRET SHARING IN VISUAL CRYPTOGRAPHY", a secure and efficient visual cryptography technique is proposed. The proposed addresses the major challenges in producing meaningful shares for secret sharing process. This technique overcomes the problems of privacy invasion, high computation time and storage requirement. The proposed technique uses the concept of meaningful shares to enhance the security of the scheme. It uses an efficient algorithm to improve performance and needs no additional data structure to reduce storage overhead. The technique is extended to share multiple secrets together. This reduces the number of shares produced and the storage requirement is further optimized.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction and Statement of the Problem

1.1 Introduction

Internet has become the primary source of transmitting confidential data such as military information, financial documents, etc. With such prolific use of Internet, information security gains high priority. Cryptography is one of many security providing tools that are used to make the communication over network reliable. With cryptographic methods, the data becomes disordered after being encrypted and can then be recovered by a correct key. Without the correct key, the encrypted source content cannot be detected even if unauthorized person steals the data. The disadvantage of conventional cryptographic methods is that they need a lot of time and computation power for performing encryption and decryption. So, some new scheme should be looked forward to, which can provide confidentiality with simpler techniques.

The concept of secret sharing was developed by Adi Shamir in 1979 [1]. He discussed the problem of dividing a secret data D into n pieces such that exactly k out of n pieces are required to reconstruct the data. This idea was extended for images in 1994 when Naor and Shamir [2] proposed a new cryptographic area called visual cryptography. It divides a secret image into a collection of shares. The decrypted message is obtained from stacking of the shares. The most notable characteristic of this scheme is to have a computation-free decryption. The idea of the Visual Cryptography Scheme (VCS) proposed is to split an image into n random shares which separately reveal no information about the original secret image other than the size of the secret image. These shares are noise-like secure images which can be transmitted or distributed over an untrusted communication channel. The image is composed of black and white pixels. When all *n* shares are overlaid, the original image appears.

Over the time, some extensions have been done to the basic VCS. It can also be used for applications which do not want to trust every participating entity in the process, using General Access Structure scheme [3]. Another interesting extension of the original model is to have a method to generate innocent-looking shares so that attacker cannot get doubtful by looking at the random pattern of the share. Visual Cryptography is expanded to encode multiple secret images together so that overhead of keeping too many shares can be reduced. One other advancement in this field has been done to encode multi-pixels at each encoding run in order to reduce the share size and make the performance better. This technique is known as multi-pixel encoding [4].

1.2 Motivation

The task of keeping information secure has gained the top priority with Internet growing day-by-day. Visual cryptography is a simple and effective technique for providing information security. Some applications of visual cryptography have been suggested such as means of transmitting financial documents [5], electronic balloting system [6], person authentication system [7], authentication in WiMAX [8] etc. But visual cryptography is not very popular as it has some restrictions. Here the concept of visual cryptography is taken because of its simplicity. Visual cryptography is capable to do much more than the proposed allocations in [5, 6, 7, 8] with its techniques constantly evolving over the time. The motivation behind this dissertation work is to propose a secure and efficient visual cryptography scheme so that it can be applied in more general applications which need information security.

1.3 Problem Description

Problem statement: To design and implement a secure and efficient algorithm for multiple secret sharing to generate meaningful shares for color images.

The problem can be divided further into following sub-problems:

1. Design an algorithm to produce meaningful shares for color images: Though several methods exist to produce meaningful shares, some of them do not apply for color images and some suffer from high time taken in encoding and decoding process. The aim is to design an algorithm for color images with reduced encoding-decoding time.

2. Extend the technique for multiple secret sharing to improve performance: Extending the technique with multiple secret sharing results in an improved scheme which creates very few number of share images for all the secret images to be shared. So less storage and less maintenance is needed for the shares.
3. Validate the proposed technique and compare it with existing schemes.

1.4 Organization of the Report

This dissertation report comprises of five chapters including this chapter that introduces the topic and states the problem. The rest of the report is organized as follows.

Chapter 2 gives the background of basic visual cryptography, description of two basic models as (k,n)-threshold method and (2,2)-scheme. It contains some extensions of basic visual cryptography model and brief literature review of related work including research gaps.

Chapter 3 contains the details of the proposed technique.

Chapter 4 gives the implementation details of the proposed technique, details of experiments performed and description of dataset used. It contains results obtained from implementation and provides experimental and theoretical analysis of the scheme.

Chapter 5 concludes the dissertation work and gives suggestions for future work.

Chapter 2

Background and Literature Review

This chapter describes the visual cryptography method and two basic schemes for it-(k,n)-threshold and (2,2) scheme. Some extensions to visual cryptography have evolved over the time such as gray-level encoding, color visual cryptography, meaningful share creation, multiple secret sharing, etc. The chapter contains relevant information related to these extensions and provides an exhausted comparative study of methods based on these extensions. Limitations of these methods are found and research gaps are identified.

2.1 Basic Visual Cryptography

Basic visual cryptography scheme is described as a visual secret sharing problem in which the secret message can be viewed as nothing more than a collection of black and white pixels. Visual cryptography schemes are characterized by two main parameters: the *pixel expansion,* which is the number of subpixels each pixel of the original image is encoded into, and the *contrast* which measures the "difference" between a black and a white pixel in the reconstructed image.

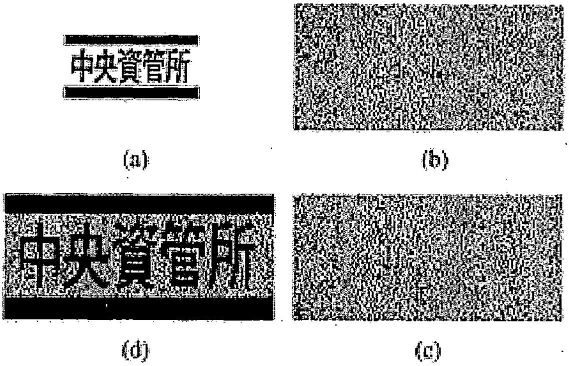


Figure 2.1: Reconstruction of secret image through shares (a) Secret message; (b) First share; (c) Second share; (d) Recomposed message Y.- C. Hou [9]

Each share is comprised of collections of *m* black and white subpixels where each collection represents a particular original pixel. The shares are constructed by randomly splitting the pixels into subpixels on the share. Two subpixels laid on top of each other (a white-black and white black laid on each other) produces essentially a white pixel or blank pixel to the human eye, while two subpixels laid next to each other (a black-white and a white-black laid onto each other) produce a full black pixel to the human eye.

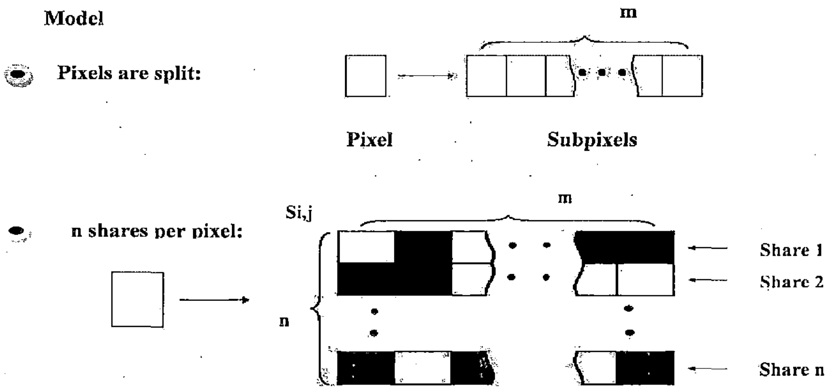


Figure 2.2: Pixel expansion

Naor and Shamir [2] devised the following scheme, illustrated in the figure below. The algorithm specifies how to encode a single pixel, and it would be applied for every pixel in the image to be shared.

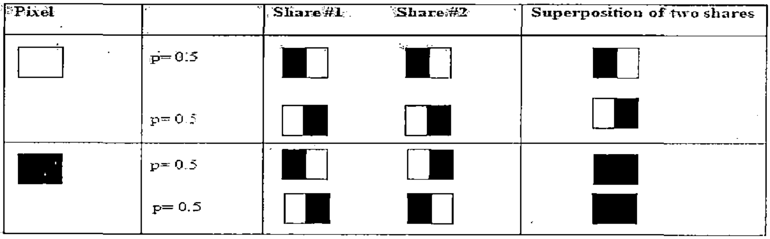


Figure 2.3: Black and white pixel construction

Considering first share, one of the two subpixels in P is black and the other is white. Each of the two possibilities "black-white" and "white-black" is equally likely to occur, independent of whether the corresponding pixel in the secret image is black or white. Thus the first share gives no clue as to whether the pixel is black or white. The same argument applies to the second share. Since all the pixels in the secret image were encrypted using independent random coin flips, there is no information to be gained by looking at any group of pixels on a share, either. This demonstrates the security of the scheme.

Two share blocks of a white secret pixel are the same while those of a black secret pixel are complementary. If the original pixel P is black, then two black subpixels are produced when the two shares are superimposed; if P is white, then one black subpixel and one white subpixel is produced when the two shares are superimposed. Thus, it can be said that the reconstructed pixel (consisting of two subpixels) has a gray level of 1 if P is black, and a gray level of 1/2 if P is white. There will be a 50% loss of contrast in the reconstructed image, but it should still be visible.

2.1.1 (k,n)-threshold Method

Suppose there is a situation as follows. There are n entities working on a highly secret project and a locked document can be opened only if k or more entities are interested to do so. Naor and Shamir [2] proposed (k,n)-threshold scheme to solve such kind of problems. Here a secret message is encoded to create n share images and for decrypting the image, atleast k shares must be superimposed. Some preliminary notations are listed below to understand the scheme better.

$\boldsymbol{n}$ → Number of shares

$\boldsymbol{k}$ → Threshold value

$\boldsymbol{m}$ → Pixel Expansion

$\boldsymbol{\alpha}$ → Relative Contrast

$\boldsymbol{C0}$ → Collection of $n \times m$ Boolean matrices for shares of White pixel

$\boldsymbol{C1}$ → Collection of $n \times m$ Boolean matrices for shares of Black pixel

V → ORed k rows

$\boldsymbol{H(V)}$ → Hamming weight of V

$\boldsymbol{d}$ → number in $[1,m]$

Two main properties of a visual cryptography scheme are pixel expansion, the number of subpixels used to encode a single pixel; and contrast, the relative difference between a black and white pixel in the image generated. k out of n visual secret sharing scheme consists of two collections of n x m Boolean matrices $C0$ and $C1$. To share a white pixel, one of the matrices in $C0$ is randomly chosen and to share a black pixel, one of the matrices in $C1$ is randomly chosen. The chosen matrix defines the color of m subpixels in each one of the n transparencies. The solution is considered valid if following three conditions are met.

Contrast conditions

- For S in $C0$ (WHITE): $H(V) \leq \mathrm{d} - \alpha\mathrm{m}$
- For S in $C1$ (BLACK): $H(V) \geq \mathrm{d}$

Security condition

- The two collections of $q \times m$ $(1 \leq q \leq k)$ matrices, formed by restricting $n \times m$ matrices in $C0$ and $C1$ to any q rows, are indistinguishable i.e no information can be achieved for less than k shares.

2.1.2 2 out of 2 scheme

An example of the encoding of white and black pixels in a 2 out of 2 scheme can be seen in Figure 2.4. Here two shares out of the two generated would be needed to recover the original image. Since only two shares are generated, $\boldsymbol{n} = 2$. Figure 2.4 represents a single white or black pixel in the original image and shows the subpixel assignments that would be given to shares 1 and 2 respectively. The number of subpixels per share used to represent the original pixel is four ($\boldsymbol{m} = 4$). Share images are 2 x 2 times larger than the original secret image. Finally, it represents the overall visual effect when shares 1 and 2 are correctly aligned on top of one another.

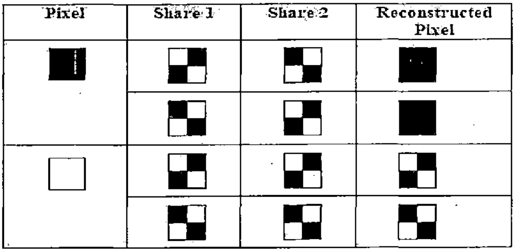


Figure 2.4: Sharing and stacking of black and white pixels

The 2 out of 2 visual cryptography scheme can be thought of as a private key cryptosystem. The secret printed message is encoded into two random looking shares. One of the two shares will be a printed page of ciphertext which can be sent by mail or fax, whereas the other share serves as the secret key. The original image is revealed by stacking together the two transparencies.

2.2 Visual Cryptography Extensions

Some extensions of visual cryptography have evolved over the time. Some of them are gray level encoding, color visual cryptography, meaningful share creation, multiple secret sharing, etc. These extensions make visual cryptography more general, secure and efficient.

2.2.1 Gray Level encoding

The images which are used in day-to-day life are not just black-and-white images but they contain various shades of a color. Naor and Shamir [2] proposed a method to encode continuous tone image with pixels having 256 gray levels. An original pixel with grey level c is divided into c black and 255-c white subpixels. Then, the basic VCS for black and white images can be applied to encode each subpixel. Each pixel in each one of the two transparencies is represented by a rotated half circle. When the two half circles (with rotation angle a and b) are aligned carefully, the superposition of the two half circles can range in color from medium gray (representing white) to completely black (representing black) depending on the relative angle a-b between the two rotated half circles.

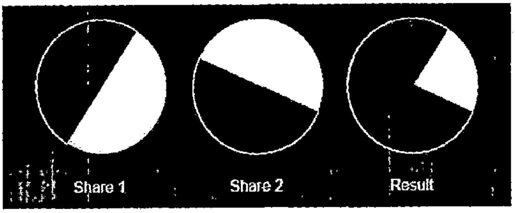


Figure 2.5: Shares as rotated half circles and their stacking

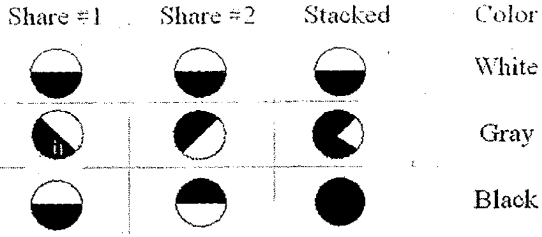


Figure 2.6: Shares as rotated half circles for different gray levels

Verhuel and Tilborg [10] suggested another method and used gray-level of pixels to form shares instead of using only black and white values. The scheme has a big disadvantage as the size of the decoded image increases hugely. If (k,n)-threshold method is applied with g gray-levels then the image size increases by a factor of g^(k-1).

Lin and Tsai [7] proposed a method for grey images which uses the technique of digital image halftoning to convert a grey level image to an approximately binary image by applying space-filling curve ordered dithering (SFCOD) algorithm. Then, the basic visual cryptography methods can be applied to create the shares. The way to represent the gray level of images is to use the density of printed dots; for example, the printed dots in the bright part of an image are sparse, and those in the dark part are dense. The method that uses the density of the net dots to simulate the gray level is called "Halftone". Because human eyes cannot identify too tiny printed dots, different gray levels can be simulated through the density of printed dots.

2.2.2 Color Visual Cryptography

2.2.2.1 Few Proposed Solutions

An image to be encoded generally contains several colors. Until the year 1997 although the shares could be stacked to recover the secret image without any computation, visual cryptography schemes were applied to only black and white images. Verhuel and Van Tilborg [10] were the first researchers to develop visual cryptography schemes for color images. They used the concept of arcs to construct a colored VC scheme. A major disadvantage of this scheme is that the number of colors and the number of subpixels determine the resolution of the revealed secret image. If the number of colors is large, coloring the subpixels will become a very difficult task. Rijmen and Preneel [11] proposed that each pixel be expanded to a 2x2 block to form 2 shares. Each 2x2 block is filled with red, green, blue and transparent, respectively. The color of a small sub-pixel is hard to notice and so four-pixel color is treated as an average color. But it is not appropriate to fill the blocks with red, green, blue, and white colors [12].

2.2.2.2 Hou's Method

Hou [9] proposed three VCS for color images. In all the three schemes, secret color image is decomposed into three primitive color images – cyan, magenta and yellow first, then halftoning of those three images is done and finally the encryption is performed. First scheme, the four-share color VCS generates one black mask randomly as the fourth share which contains as many 2 x 2 blocks as the number of pixels in the secret image. The second method expands each pixel of a halftone image into a 2 x 2 block on two sharing images and fills it with cyan, magenta, yellow and transparent, respectively. Using these four colors, two shares can generate various colors with different permutations. To reduce the trouble of having four shares as in method 1, and to have a better image quality than in method 2, third scheme was developed. This method creates only two share images and does not lose too much contrast for color visual cryptography. The scheme uses the technique of gray-level visual cryptography [2] and then applies basic (2,2) VCS on each of the C,M and Y images to produce six intermediate shares. It then combines C1, M1, Y1 to get share 1 and C2, M2, Y2 to get share 2.

The steps of the third method can be listed as follows:

Step 1: The scheme first decomposes the original image into three primitive-color images under the subtractive model, namely, *C* (Cyan), *M* (Magenta) and *Y* (Yellow). Figure 2.8 shows the three primitive-color components of the Lena image, where each image has 256 levels of the corresponding primitive color.

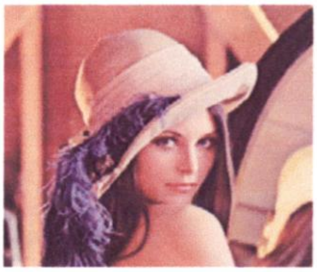


Figure 2.7: Original secret image (Lena)— 24-bit color
Bert W. Leung, Felix Y. Ng, Duncan S. Wong [13]

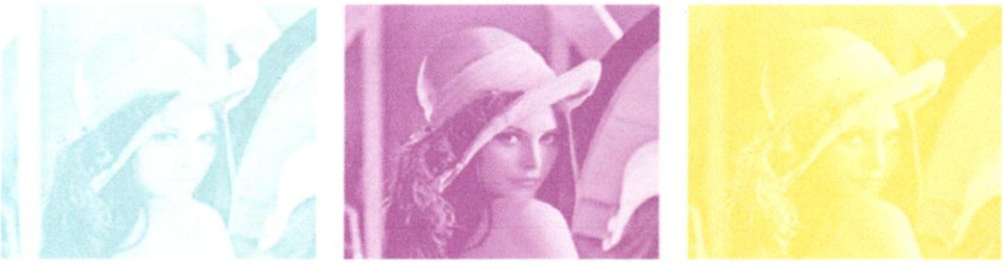


Figure 2.8: Primitive-color (*C, M* and *Y*) components— 256- level
Bert W. Leung, Felix Y. Ng, Duncan S. Wong [13]

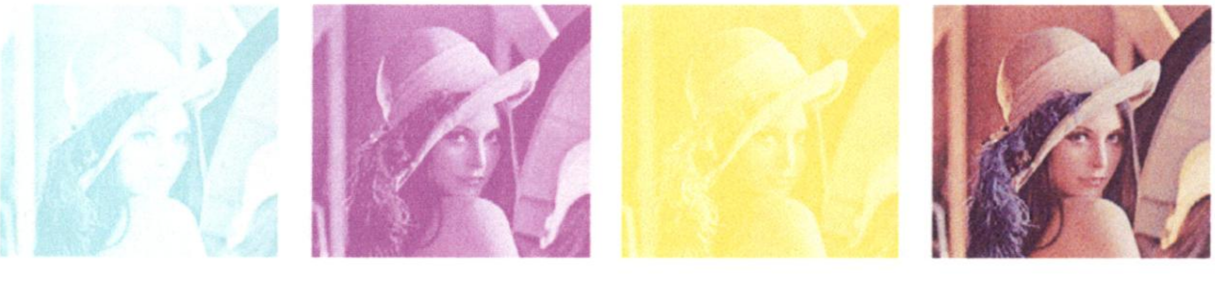


Figure 2.9: Dithered *C, M* and *Y* components and their superimposition
Bert W. Leung, Felix Y. Ng, Duncan S. Wong [13]

Step 2: After decomposition, each primitive-color image is dithered (e.g. by applying the Floyd–Steinberg algorithm) so that each image will have two color levels, namely the presence of the corresponding primitive color or the absence of it. Figure 2.9 shows the three dithered primitive-color images and an illusion of their superimposed image. Each pixel in a dithered primitive-color image has a 1-bit color depth. The superimposed image has therefore 3-bit color depth, that is, eight colors altogether.

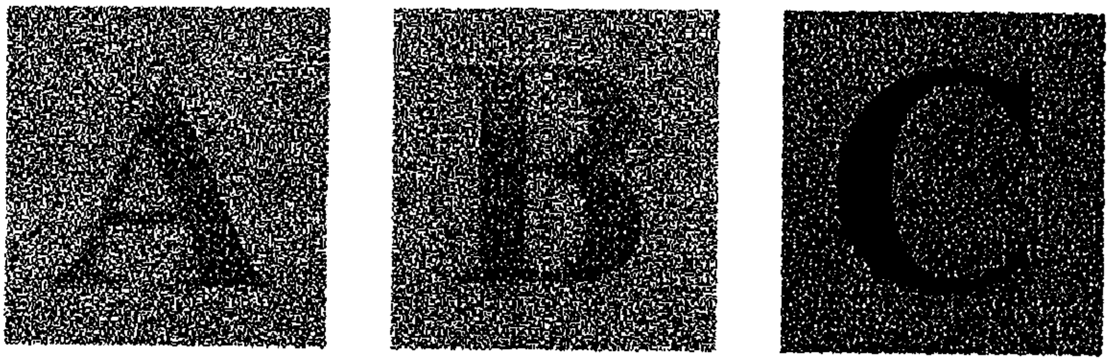
Step 3: Create 2 shares for each dithered component using basic VCS- C1, C2, M1, M2, Y1, Y3.

Step 4: Combine C1, M1, Y1 to create share 1 and C2, M2, Y2 to create share 2. Now each share will have 3-bit of color information for each pixel.

2.2.3 VC for Producing Meaningful Shares

2.2.3.1 Few Proposed Solutions

Although encryption can protect the privacy of the secret data, the encrypted data themselves appear random and meaningless and look a little different from normal data, which may attract intruders' attention. Intruders may interfere, intercept, or disturb the communication during data transmission and thus make the legal users unable to successfully receive the data in secrecy. To remove this problem, meaningful shares, which show some valid image, should be produced to fill in the security gap. Naor and Shamir [2] proposed a method to produce innocent looking shares to conceal the secret message. Under this solution, all the share images show some valid image which is different from the original secret. When they are superimposed, the decrypted image is the same as the original secret. The method can be understood as follows: Suppose the original secret message is 'C'. Two shares are created where share s1 shows the message 'A' and share s2 shows the message 'B. But when these shares are decoded, they generate the original message 'C'. Figure 2.10 shows an example where s1 and s2 are meaningful shares and their result is s1+s2 which is very different from the contents of s1 and s2.



(a) (b) (c)

Figure 2.10: Meaningful shares (a) share s1 (b) share s2 (c) s1+s2

In the scheme proposed by Naor and Shamir [2], pixel expansion is taken as 4. To encode a white pixel, share pixels are selected from the arrangement shown in Figure 2.11 and for a black pixel from Figure 2.12. In shares, white pixel is represented with 2 black subpixels and black pixel with 3 black subpixels. Two types of superimposed results are defined, white with 3 black subpixels and black with 4 black subpixels. It can be easily understood that each share can contain an arbitrary image which reveals no information whatsoever about the superimposed image.

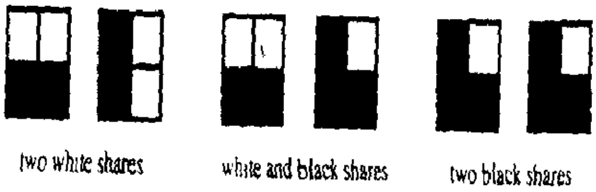


Figure 2.11: White pixel encoding

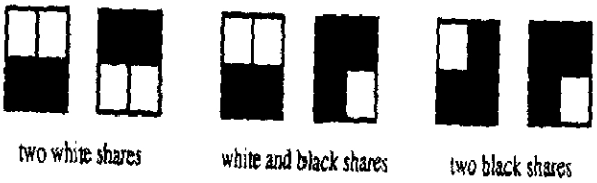


Figure 2.12: Black pixel encoding

Chang et al [14] suggested a scheme for color image hiding using a Color Index Table. For a secret color image, two significant color images are selected as cover images which are the same size as the secret color image. Then according to the pre-defined Color

Index Table, the secret image will be hidden into two camouflage images. One disadvantage of this method is that extra space is required to accumulate the Color Index Table. In this method, number of subpixels is also proportional to the number of colors in the secret image. When the number of colors in the secret image is more, the size of the shares becomes larger. To overcome this limitation, Chang and Yu [15] developed a secret color image sharing scheme based on modified visual cryptography. This scheme provides a more efficient way to hide a gray image in different shares. Here the size of the shares is fixed; it does not vary when the number of colors appearing in the secret image differs. This does not require any pre-defined Color Index Table.

Wu et al. [16] formulated a method in which size of each embedding image is about 1/k of that of the secret image (k- threshold), avoiding the need for much storage space and transmission time. Qualities of both the recovered secret image and the embedding images that contain the hidden shadows are acceptable. Chang et al. [17] proposed two methods to generate two shares for hiding a secret two-tone image in a gray-level cover image. Tsai et al. [18] developed a method to support true-color secret image with size constraint on shares. In this scheme through the combination of neural networks and variant visual secret sharing, the quality of the reconstructed secret image and camouflage images are visually the same as the corresponding original images.

2.2.3.2 Chang's Method

Chang et al. [17] proposed two methods to generate two shares for hiding a secret two-tone image. First toral automorphism is used to mix all the pixels in the cover image before embedding the secret share into the cover image in order to improve the security of the image hiding scheme. Toral automorphisms [19] is a kind of dynamic system where the state S changes with time t. Toral automorphism is a permutation function that moves each pixel from a grid to another grid. For example, assume a pixel is stored at (1, 4). After r sequences of toral automorphism operations, the pixel will be stored at a new coordinate (x_r, y_r). After doing the remaining ($R-r$) times of permutation operations, the original image can be achieved, where R denotes the recurrence time. According to the conclusions in [19], in most cases, R is equal to N-1 or N+1 when N is a prime number. Shares are embedded into two gray-level cover images.

In Method-2, one secret bit from a secret share is embedded into the 4-th bit of a cover pixel. Before embedding the secret bits into each cover pixel, all the cover pixels in a cover image are first mixed by using toral automorphism. The embedding structure of Method-2 is shown in Figure 2.13. After embedding, the inverse toral automorphism is applied to obtain the cover image shares. While decryption, each participant can collect the 4-th bit of each cover share pixel to obtain the hidden share. Method-2 is taken as the base concept and has been expanded further for the purpose of dissertation.

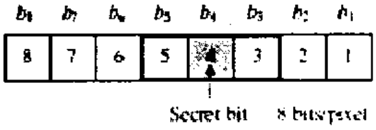


Figure 2.13: Chang Method-2
Chang et. al. [17]

2.2.4 Sharing Multiple Secrets

VC was proposed to deal with only one secret image at a time. Later it has been extended to share multiple secrets together. To share multiple secret images, many shares need to be generated and it takes a lot of time during transmission. This is not efficient and hence, some progress has been made to hide more secret images into two share images. Droste [20] took the problem of sharing more than one secret among a set of shares. Wu and Chang [21] developed a VCS to share two secret images together using two circle shares. First secret can be obtained by stacking the two shares and second secret by rotating share 1 by a rotation angle and then stacking it with share 2. The scheme was extended by Shyu et al. [22] so that multiple secrets can be shared. Share 2 is rotated with n different angles and stacked with share 1 to get n secret images. Feng et al. [23] proposed another scheme to hide *n* secrets and to reveal the secrets by stacking the share images at *n* aliquot angles.

Table 2.1 shows comparison of various visual cryptography methods to find the limitations of the existing methods.

Table 2.1: Comparison of various visual cryptography method

Author	Year	Encryption/ decryption	Pixel expansion (m)	Decoding time	(n,n) scheme support	Size of decrypted image	Quality of output image	No of secret image	Share type	True color support	Security enhanceement	Additional data structure	Meaningful shares
M. Naor and A. Shamir [2]	1995	An original pixel divided into 255 sub-pixels.	255	NA	Yes	255 times original image	Medium	1	Rectangular	No (grey level)	No	No	No
Rijmen and Preneel [11]	1996	Each pixel expanded to 2 x 2 block	4	NA	No	4 times original image	Medium	1	Rectangular	No (only 3-bit color)	No	No	No
E. Verhuel and V. Tilborg [10]	1997	Each subpixel takes one of grey-levels of 0,1,.., g-1 (g is no of grey levels)	Variable	NA	Yes	At least g^(k-1) times original image	Medium	1	Rectangular	No (grey level)	No	No	No
Chang et al. [14]	2000	With a predefined color index table, secret is hidden into 2 cover images.	529	more	No	m times original image	good	1	Rectangular	No	No	Yes	Yes
Chang-Yu [15]	2002	Based on modified visual cryptography. Decoding by XOR.	9	more	Yes	9 times original image	Medium	1	Rectangular	No	No	No	Yes
C. Lin and T. Tsai [8]	2003	After dithering of secret image, basic VC can be applied.	variable	NA	Yes	m times original image	Medium	1	Rectangular	No (grey level)	No	No	No
Hou [9] third scheme	2003	After dithering, 3 primitive color shares are generated.	4	NA	No	4 times original image	Medium	1	Rectangular	Yes	No	No	No

Wu et al. [16]	2004	Quantized-embeded image is divided into n shadows.	4	more	Yes	About 1/n times original image	good	1	Rect-angu-lar	No	Perm uta-tion	Yes	Yes
Wu and Chang [21]	2005	share *B* generated a/c to distribution of 2×2 extended block of share *A* at the relative position. Decryption by rotation at 90 degree.	4	more	No	m times original image	Medi-um	2	Circle	No	No	No	No
Shyu et al. [22]	2007	Secrets can be obtained one by one by stacking first share and rotated second shares with *n* different angles.	2*n	NA	No	NA	Medi-um	n>= 2	Circle	No	No	No	No
Feng et al. [23]	2008	A stacking relationship graph of secret pixels and share blocks is generated to indicate encryption functions.	9	NA	No	3n times original image	Medi-um	n>= 2	Rect-angu-lar	No	No	No	No
Tsai et al. [18]	2009	Size of secret is shrinked before encoding.	9	more	Yes	9 times original image	good	1	Rect-angu-lar	Yes	Perm uta-tion	Yes	Yes

2.3 Limitations of Existing Methods

The method in [14] needs a color index table as additional data structure which increases the overhead. This scheme supports only (2, 2) scheme. The methods in [14, 15, 16, 17] do not support true-color secret image. Method in [18] supports true-color images, however it needs additional data structure, the algorithm is complex and the encoding-decoding time is more. Further the quality of reconstructed image in methods [15, 16] is not very good.

The methods proposed for sharing multiple secrets by Wu and Chang [21] and Shyu et. al. [22] produce circle shares for the secrets which is not very general. The method in [21] can be used for sharing only 2 secrets. The pixel expansion in [22] is 2 times the number of secrets to be shared. The method suggested by Feng [23] takes pixel expansion as 9. This increases the share size 9 times.

2.4 Research Gaps

Based on Table 2.1, the research gaps identified in the previously proposed methods are as follows:

1. No single method achieves both the goal of being able to share multiple secrets along with producing meaningful shares.
2. The encoding and decoding time in methods for producing innocent shares is large.
3. The quality of the reconstructed image in methods for producing innocent shares is not very good.

Thus the challenges in the dissertation work are:

- To design an efficient scheme by sharing multiple secrets with innocent-looking shares
- To improve the computation (encoding-decoding) time
- To improve the quality of reconstructed image

Chapter 3
Proposed Technique

This chapter covers the proposed algorithm for sharing single as well as multiple secret images. The encryption and data extraction algorithm for secret sharing is described. The proposed technique aims to enhance the security by producing innocent-looking shares and improve the efficiency by encoding more than one secret together. The technique supports images with true colors so that it can be applied in general scenarios.

3.1 Single Secret Sharing

In the proposed algorithm for producing meaningful shares for single secret, 3 steps are performed for the entire process. The first step applies Hou's algorithm [9] to the secret image. It creates 2 random-looking shares for the secret image. However, the ultimate goal of the technique is to produce meaningful shares and that task is performed by the second module. Second step does the encryption process for the random-looking shares created by Hou's algorithm and produces meaningful shares. The third step performs the data extraction process which decrypts the meaningful shares to reconstruct the secret image. Figure 3.1 shows the steps of share creation and secret reconstruction.

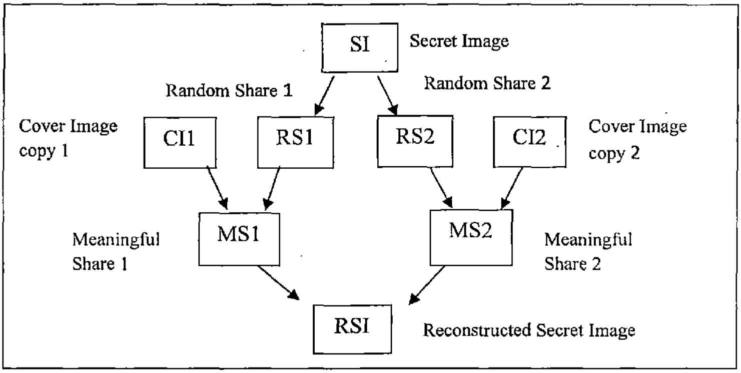


Figure 3.1: Share creation and secret reconstruction for single secret

To share a W x H sized secret image, a cover image of size 2W x 2H is taken so that the cover image size and share size is the same. Hou's algorithm [9] produces random-looking shares which are 2 times in height and 2 times in width than the original secret image as the pixel expansion of the method is 4. A 24-bit color image is chosen as the cover image which has 256 gray levels for each primitive color- red, green and blue. So each primitive color has 8-bit of color information in cover image. Figure 3.2 shows the block diagram for the entire process of sharing a single secret image.

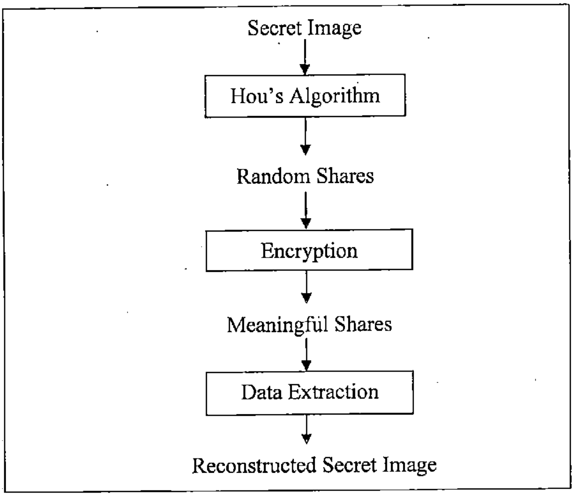


Figure 3.2: Block diagram for single secret sharing

3.1.1 Encryption

The encryption process for a single secret takes 2 random shares for the secret image and the cover image as input and produces 2 meaningful shares. Each pixel of the random share image produced by Hou's method [9] has 3-bit of color information- one for each primitive color, 0 representing absence and 1 representing presence of that primitive color. Each pixel of the cover image has 8-bit (256 level) of color information for each primitive color. A bit-position is selected for the secret image to embed in the cover image. The bit pattern is shown in Figure 3.3.

8	7	6	5	4	3	2	1

Figure 3.3: Bit pattern

The same bit position is used for both the shares. While embedding share 1, for one particular primitive color this bit-value is XORed with same primitive color bit of the random share image. This is done for all three primitive colors and the process is repeated for all the pixels of share 1. Now for share 2, another copy of same cover image is taken and the same procedure is reiterated. Figure 3.4 shows the block diagram for encryption of a single secret.

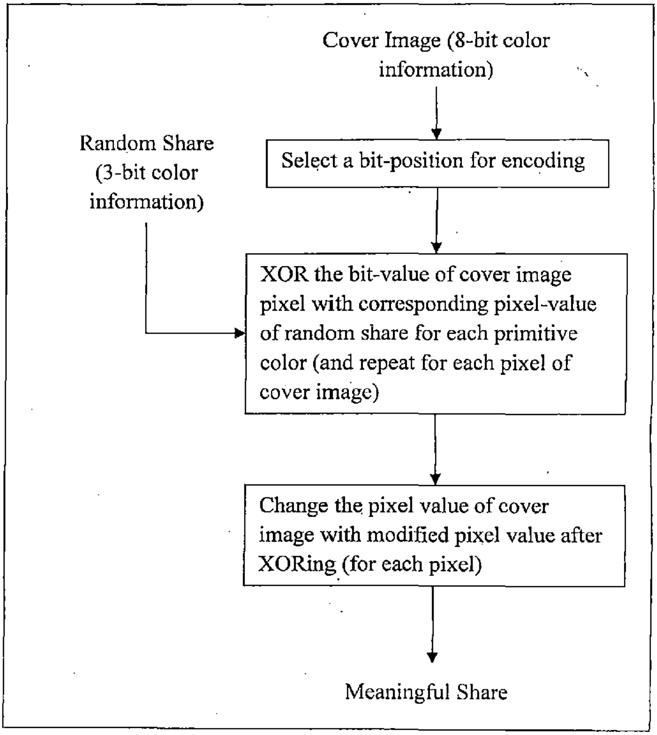

Cover Image (8-bit color information)
Random Share (3-bit color information)
Select a bit-position for encoding
XOR the bit-value of cover image pixel with corresponding pixel-value of random share for each primitive color (and repeat for each pixel of cover image)
Change the pixel value of cover image with modified pixel value after XORing (for each pixel)
Meaningful Share

Figure 3.4: Block diagram for encryption

In the proposed technique XOR is used instead of toral automorphism as in [17] for enhancing security of the shares. The reason is that XOR operation is easy to perform and takes very less computation time than toral automorphism. The secret bit cannot be revealed from the cover image as it is the XOR-result of cover-pixel bit and share-bit. The bit to be replaced should be one of the lower order bits so that the pixel value of the cover image does not change much. In order to have the least distortion, least significant bit-position of the cover image should be considered for embedding of the secret image.

3.1.2 Data Extraction

The data extraction process takes 2 meaningful shares input and reconstructs the secret image embedded. For decoding purpose, the bit-value of the selected bit-position of modified-share 1 and modified-share 2 are XORed and the bit value of the original halftoned image is achieved. Figure 3.5 shows one of the possible combinations of share 1 and share 2 using basic (2, 2) visual cryptography method proposed by Naor and Shamir [2] for white pixel, and the reconstructed pixel using XOR operation. Figure 3.6 shows the scheme for black pixel. It can be observed that the reconstructed pixel has 4 white sub-pixels for a white pixel as shown in Figure 3.5, and 4 black sub-pixels for a black pixel as shown in Figure 3.6. Thus the output image quality is better using this technique as XOR operation allows for perfect reconstruction of pixels. It means that there is no loss in contrast.

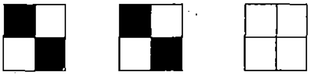


Figure 3.5: White pixel reconstruction

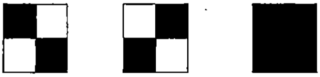


Figure 3.6: Black pixel reconstruction

Figure 3.7 shows the block diagram for data extraction process.

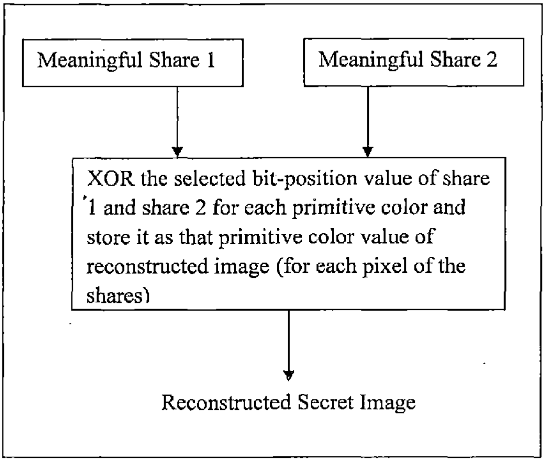

Meaningful Share 1
Meaningful Share 2
XOR the selected bit-position value of share 1 and share 2 for each primitive color and store it as that primitive color value of reconstructed image (for each pixel of the shares)
Reconstructed Secret Image

Figure 3.7: Block diagram for data extraction

The decryption process can be understood as follows. Suppose the secret image S is to be hidden. First 2 shares for the secret image are created as S1 and S2 using Hou's method [9] which are just random pattern of pixels. A cover image C is taken and two copies of the same cover image are created as C1 and C2. Then the random shares are XORed with the cover images, S1 with C1 and S2 with C2 during encryption to produce innocent-looking shares. C1 and C2 get modified after the encryption and become meaningful shares for the secret image S. For the decoding process, again XOR is performed with the bit-value of C1 and C2 of predefined bit-position used for encryption to generate the meaningful shares. The final result of decryption can be expressed as,

$$(S1 \oplus C1) \oplus (S2 \oplus C2) = (S1 \oplus S2) \oplus (C1 \oplus C2) = S1 \oplus S2$$

C1 and C2 are identical images as they are just 2 copies of same cover image. Hence result of $C1 \oplus C2$ becomes 0 and the whole expression effectively results in $S1 \oplus S2$ which constructs the final image. Table 3.1 shows the pixel reconstruction using the proposed technique. It can be observed that the final result of the whole algorithm turns out to be the XORed result of share 1 and share 2.

Table 3.1: Pixel reconstruction using proposed technique

Share 1	Share 2	Cover Image	Modified Share 1	Modified Share 2	Final XOR Result
0	0	0	0	0	0
0	1	0	0	1	1
1	0	0	1	0	1
1	1	0	1	1	0
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	1	0	0	0

3.2 Multiple Secret Sharing

The proposed method can be used to share single secret as well as multiple secrets in the same cover image. To share multiple secrets together, 2 random shares are created for each secret using Hou's third method [9]. So total 2*n random shares are created, where n is the number of secret images. Now one random share of each secret is embedded in one copy of cover image and the second random share is embedded in another copy of the cover image. This way, only 2 shares are produced for all the secrets that are shared and instead of 2*n shares only 2 shares need to be managed. While secret extraction, both the meaningful shares are needed for decryption and all the secrets embedded can be reconstructed successfully using only these 2 shares. Each secret is embedded at different bit-positions. As 8 bit-positions are available for embedding of secrets, total 8 secret images can be shared together in 2 meaningful shares. Thus the proposed technique reduces the space overhead for storing all the shares for secret images and requires less maintenance. Figure 3.8 shows the steps of share creation and secret reconstruction for multiple secret sharing.

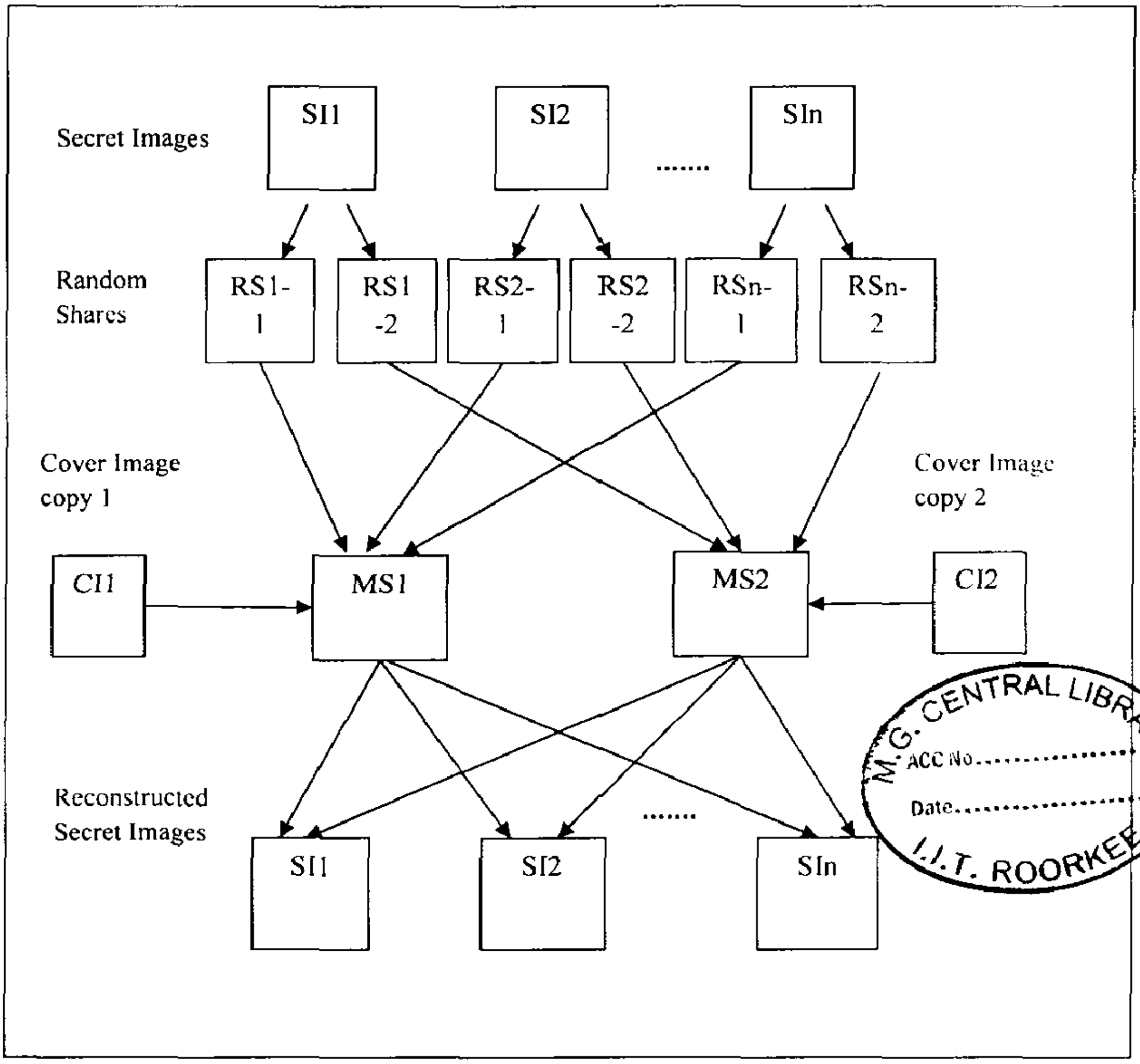


Figure 3.8: Share creation and secret reconstruction for multiple secrets

The encryption of multiple secrets follows the same pattern as for single secret sharing shown in Figure 3.3. A different bit-position is selected for each secret image and all the steps for embedding a single secret image in the cover image are repeated for each secret to be shared here. The proposed technique can accommodate maximum 8 secret images in the same cover image. While extraction of the secrets also the same step of XOR operation in meaningful shares is performed as in case of single secret sharing shown in Figure 3.7 (for each bit-position where the secret is hidden). Figure 3.9 shows the block diagram for the entire process of random share creation, encryption and data extraction for multiple secret sharing.

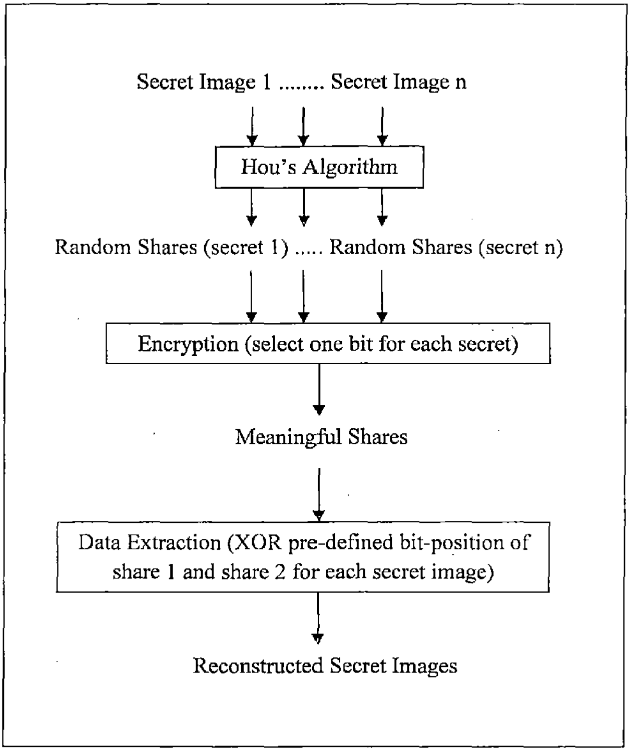

Secret Image 1 .... Secret Image n
Hou's Algorithm
Random Shares (secret 1) ..... Random Shares (secret n)
Encryption (select one bit for each secret)
Meaningful Shares
Data Extraction (XOR pre-defined bit-position of share 1 and share 2 for each secret image)
Reconstructed Secret Images

Figure 3.9: Block diagram for multiple secret sharing

3.2.1 Encryption

The encryption process for multiple secrets takes 2 random shares for each secret image to be shared and the cover image as input and produces 2 meaningful shares. While embedding the shares in cover image, one bit-position is selected for each secret image to embed. The same bit-position is used for both shares belonging to the same secret. First random share of each secret is embedded into the first copy of cover image and second random share is embedded into the second copy of cover image. Thus the extension will result in 2 innocent-looking shares which will contain multiple secrets.

The technique can be used to share up to 8 secret images together but in that case the shares produced will become random pattern of pixels and will no longer remain meaningful. To keep the shares meaningful, optimum number of secret images should be embedded in the cover image. The optimum number can be found out after analyzing the meaningful share distortion from cover image and the security needs of the application. For example, if 4 secret images are shared together in same cover image then it can change the pixel value maximum by a value of 15. If 5 images are to be shared then it can change the original pixel value of cover image by a value of 31. The value of optimum number of secrets that can be shared together in same cover image depends on the application where the method is to be used. Table 3.2 shows maximum possible distortion in cover image pixel-value with respect to number of secret images. The secret images are embedded at least significant bit-position available in cover image pixel. Pixel distortion increases with more number of secret images to be shared because pixel values change more in that case.

Table 3.2: Maximum Pixel distortion with number of secret images

No of secret images	Maximum possible change in pixel value (secrets embedded at lower bit-positions)
1	1
2	3
3	7
4	15
5	31
6	63
7	127
8	255

3.2.2 Data Extraction

The data extraction process takes 2 meaningful shares as input and reconstructs all the secret images embedded. The secret extraction process in case of multiple secrets follows the same pattern as in case of single secret sharing shown in Figure 3.7. For each secret shared, one particular bit-position was used. That selected bit value (for each primitive color) of meaningful share 1 and meaningful share 2 are XORed and the bit value for that primitive color of the original halftoned secret image is achieved. This way, each secret image is reconstructed from the meaningful shares.

3.3 Overall Algorithm

The algorithm for the proposed technique for sharing secrets (single as well as multiple) has 3 modules to perform the entire work of encryption and decryption. The first module produces random shares using Hou's algorithm [9]. The second module transforms these random shares into meaningful shares by hiding them in cover images. The third module is the data extraction process which reconstructs the secret images from the meaningful shares.

The algorithm can be summarized as follows:

Step 1: Create 2 shares for each secret image of size W x H using Hou's third method for color visual cryptography.

Step 2: Choose a cover image of size 2W x 2H. Make 2 copies of the same cover image.

Step 3: Select an unused bit-position for each secret image (lower order bits should be preferred).

Step 4: Take the bit-value of each pixel in the first copy of cover image and XOR it with corresponding pixel-value of share 1 (for each primitive color).

Step 5: Repeat step 4 for all the pixels of share 1.

Step 6: Repeat the steps 1-5 for share 2 with the second copy of the cover image.

Step 7: Repeat the steps 1-6 for each of the secret images to be shared.

Figure 3.10 shows the flowchart for the overall proposed technique.

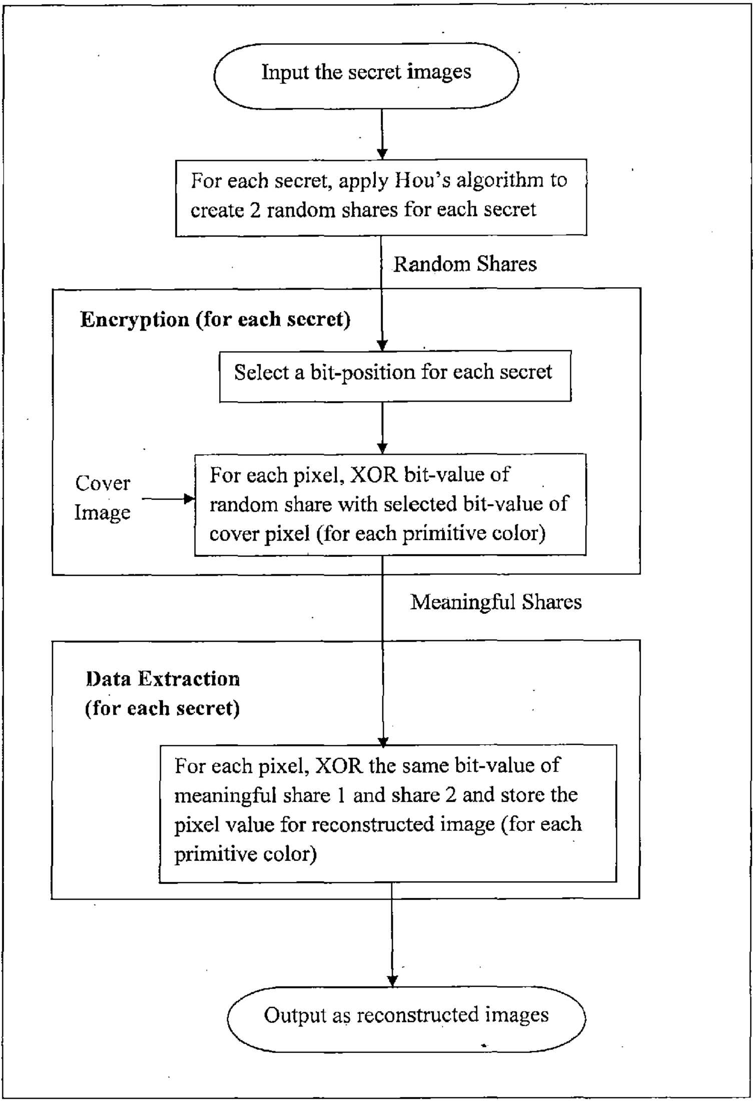

Input the secret images
For each secret, apply Hou's algorithm to create 2 random shares for each secret
Random Shares
Encryption (for each secret)
Select a bit-position for each secret
Cover Image
For each pixel, XOR bit-value of random share with selected bit-value of cover pixel (for each primitive color)
Meaningful Shares
Data Extraction (for each secret)
For each pixel, XOR the same bit-value of meaningful share 1 and share 2 and store the pixel value for reconstructed image (for each primitive color)
Output as reconstructed images

Figure 3.10: Flowchart for proposed algorithm

Chapter 4

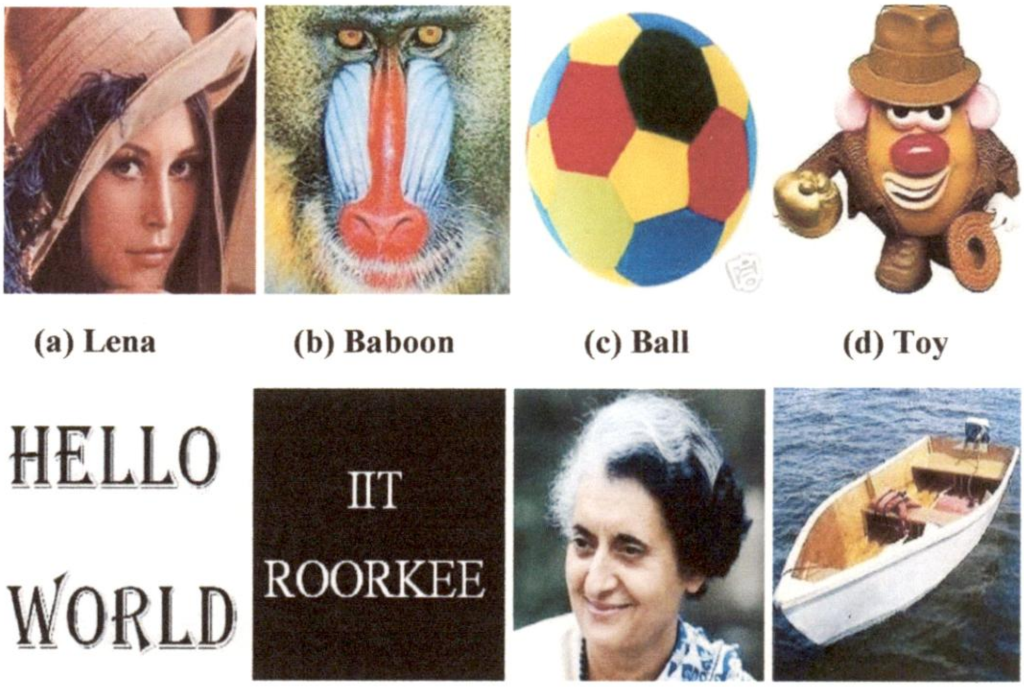
Results and Discussions

This chapter contains the implementation details, experimental analysis and analytical validation for the proposed technique. The implementation of the proposed technique is done and tested with dataset of various secret images of Figure 4.1 having different background and foreground color components from each other. The cover image used for implementation and analysis is shown in Figure 4.2 unless mentioned something else. The results are obtained and the experimental analysis and analytical validation has been performed. The first set of experimental analysis is done to test the correctness of the algorithm in being able to reconstruct the secret images, single as well as multiple secret images. The second set of analysis is performed to compare the results of the proposed technique with the results of previously proposed Chang [15] method for single secret. The third set of experimental analysis is done to know the distortion produced in the meaningful share after embedding of single secret image as well as multiple secret images. The fourth set of analysis provides distortion in the meaningful share after embedding of a single secret image in different cover images. In the analytical validation, security enhancement of the proposed technique is analyzed and the performance analysis is done in comparison to previously proposed methods. Finally the time complexity of the algorithm is analyzed comparing it with Chang [15] method. In the end, advantages of the proposed technique are discussed.

4.1 Experimental Analysis

4.1.1 Data Set

The images shown in Figure 4.1 are taken as some of the chosen secret images to test and validate the results of the proposed technique. These images are of wide range with respect to their color components. For the implementation purpose, the size of all the secret images is set to 200 x 200. Figure 4.2 is chosen as the cover image which is of size 400 x 400. This image has 256 gray-levels corresponding to each primitive color that is red, green and blue. It means that the cover image has 24-bit color information, 8 bits of information for each primitive color. As the pixel expansion in the technique is taken as 4, the share size becomes 4 times than the original secret image size and hence the cover image size is also kept as 2 times in width and 2 times in height than the secret image size.



(a) Lena (b) Baboon (c) Ball (d) Toy

(e) Hello (f) Text (g) Indira Gandhi (h) Boat

Figure 4.1: Secret Image (a) Lena (b) Baboon (c) Ball (d) Toy (e) Hello (f) Text message (g) Indira Gandhi (h) Boat

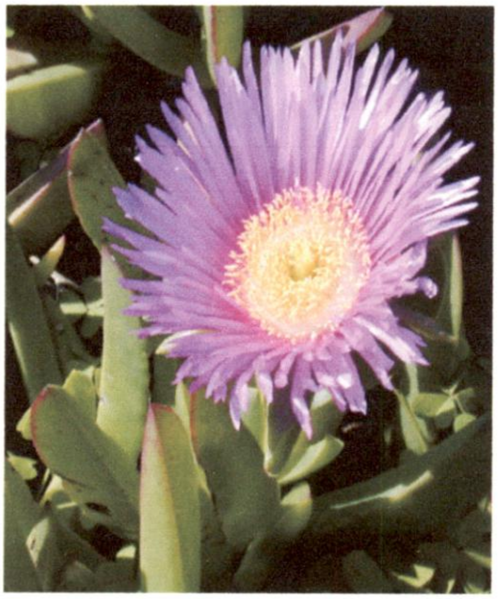


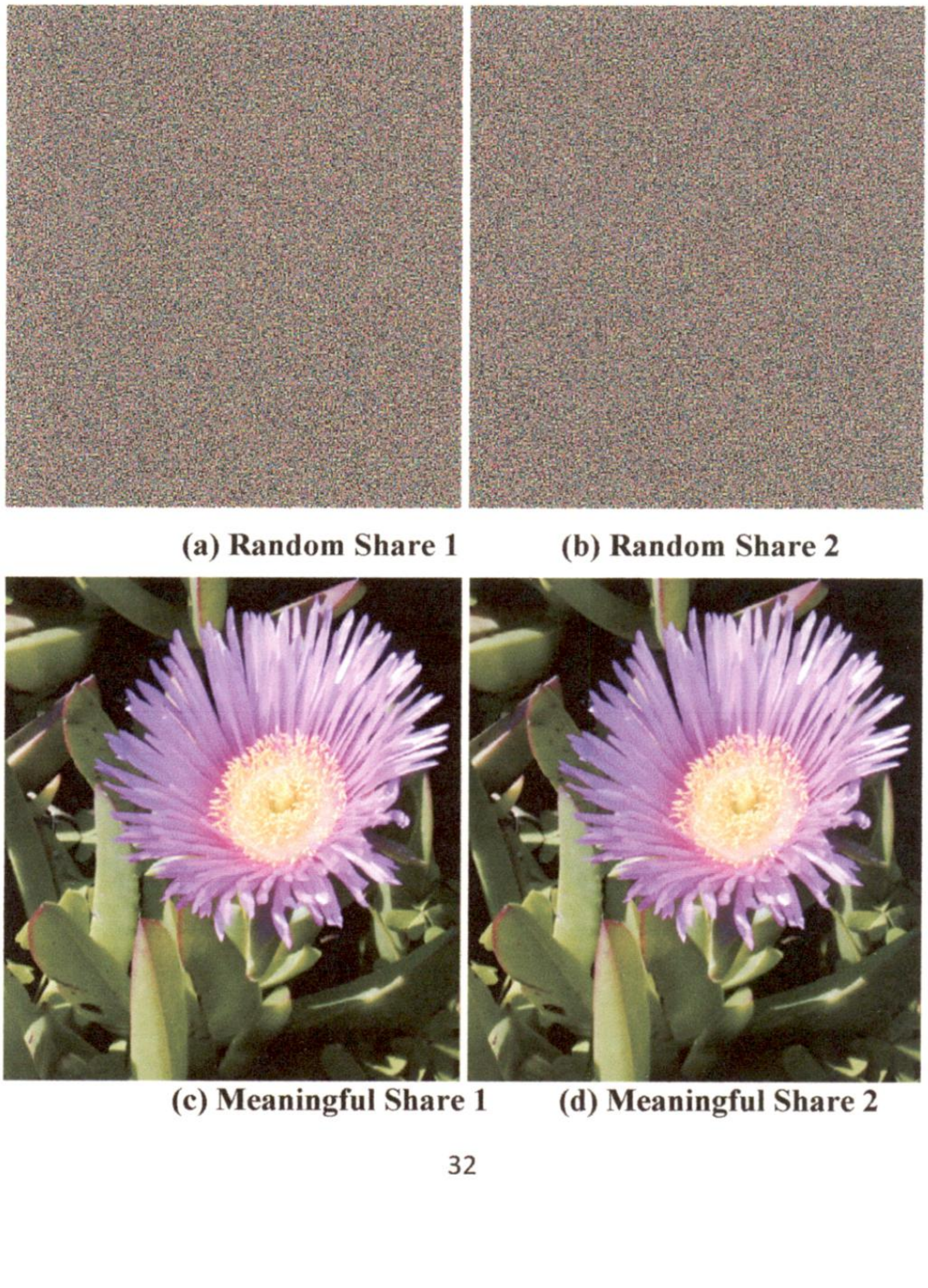
Figure 4.2: Cover Image

4.1.2 Experimental Validation

The technique is tested for single secret and for multiple secrets shared together.

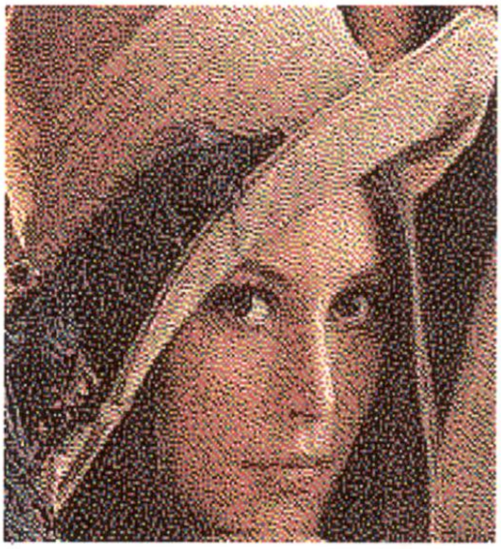
4.1.2.1 Single Secret

The secret image to embed is taken as Lena image in Figure 4.1(a). The cover image is of Figure 4.2. Figure 4.3(a), (b) shows the random shares created by Hou's method and Figure 4.3(c), (d) show the innocent-looking shares produced after embedding of the secret image in cover image. It can be easily understood that meaningful shares truly enhance the security in secret sharing as an attacker can doubt the random shares but not the meaningful shares. The bit-position used for embedding is 1. The innocent shares look exactly like the cover image and human-eye cannot see any difference.



(a) Random Share 1 **(b) Random Share 2**

(c) Meaningful Share 1 **(d) Meaningful Share 2**

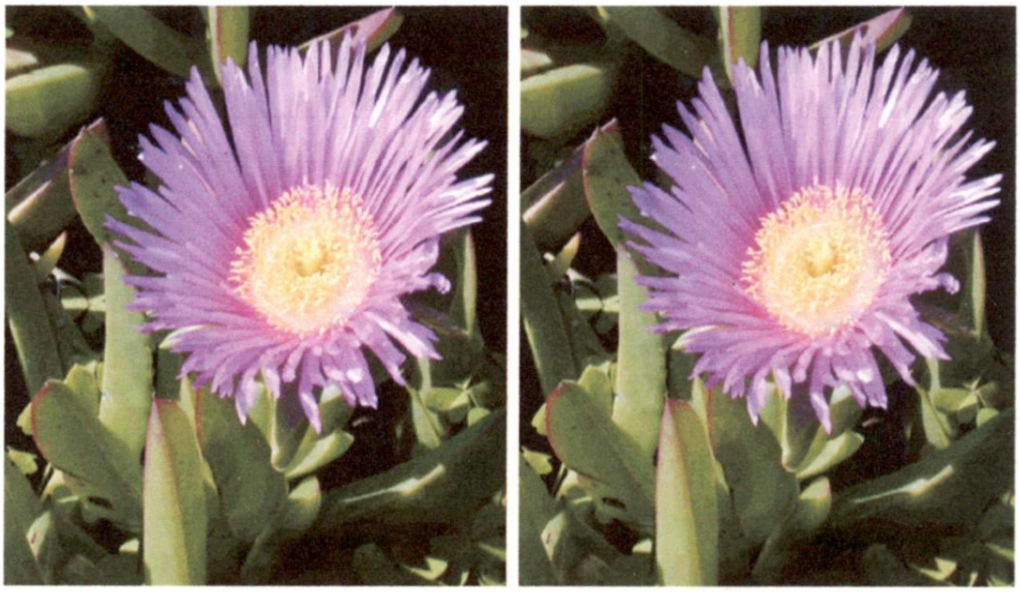


(e) Recovered Image

Figure 4.3: (a) Random Share 1 (b) Random Share 2 (c) Innocent Share 1 (d) Innocent Share 2 (e) Recovered Image

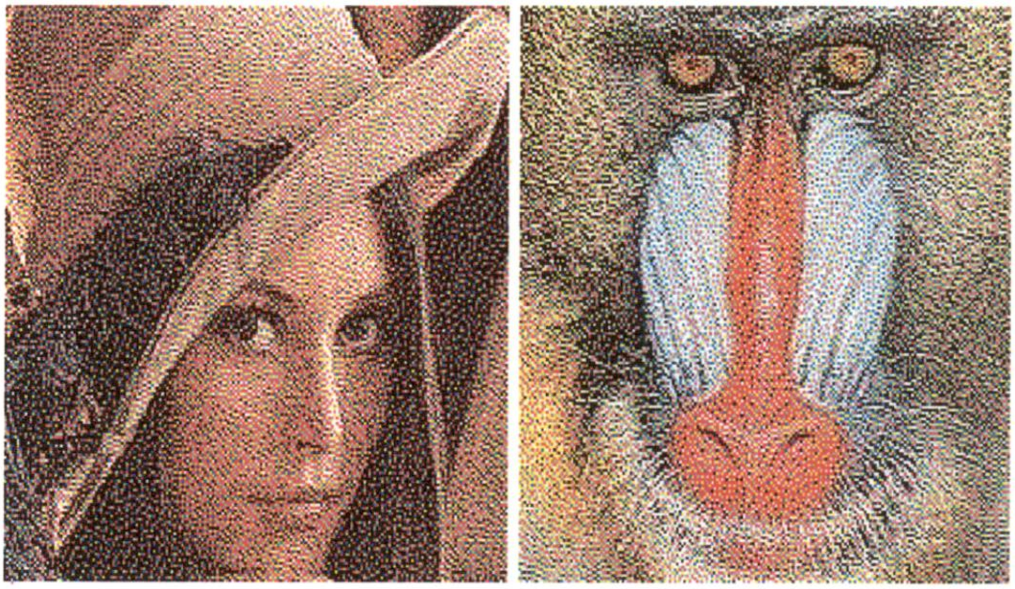
4.1.2.1 Multiple Secrets

The secret images to embed in the cover image are taken as Lena image, Baboon image, Ball image and Toy image listed in Figure 4.1(a), (b), (c), (d). So the total number of secret images here is 4. The cover image is of Figure 4.2. Figure 4.4 shows the meaningful shares produced after embedding of the secret images in cover image. Figure 4.5 shows the reconstructed secret images after decryption. The bit-position used for embedding is 1, 2, 3 and 4. It can be seen that in this case also, the meaningful shares in Figure 4.4 look very much like the cover image in Figure 4.2 and the distortion produced in the meaningful shares after embedding of the secrets is very less. Further, the reconstructed image quality does not vary with the number of secret images to be shared. The proposed technique reconstructs images with same quality independent from the number of secrets embedded. This fact can be verified from comparing the recovered Lena image from Figure 4.3(c) and Figure 4.5(a).The image quality also remains the same irrespective of the bit-position used for the secret image while embedding of the secret. The total number of random shares produced here are 8, 2 for each secret image. These random shares need not be stored as reconstruction of secret images is done just from the 2 meaningful shares produced after encryption, shown in Figure 4.4.

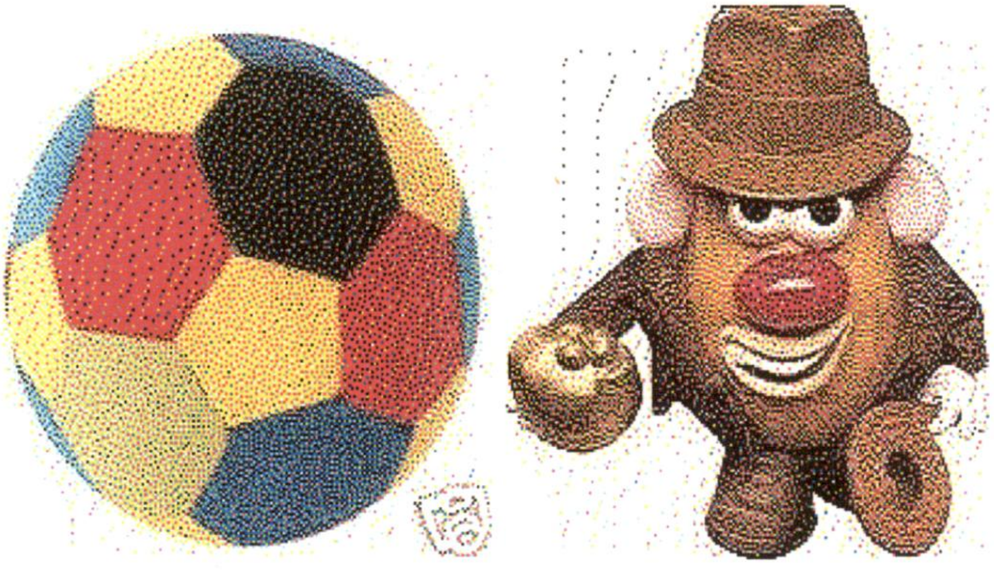


Innocent Share 1 **(b) Innocent Share 2**

Figure 4.4: (a) Innocent Share 1 (b) Innocent Share 2



(a) Lena **(b) Baboon**

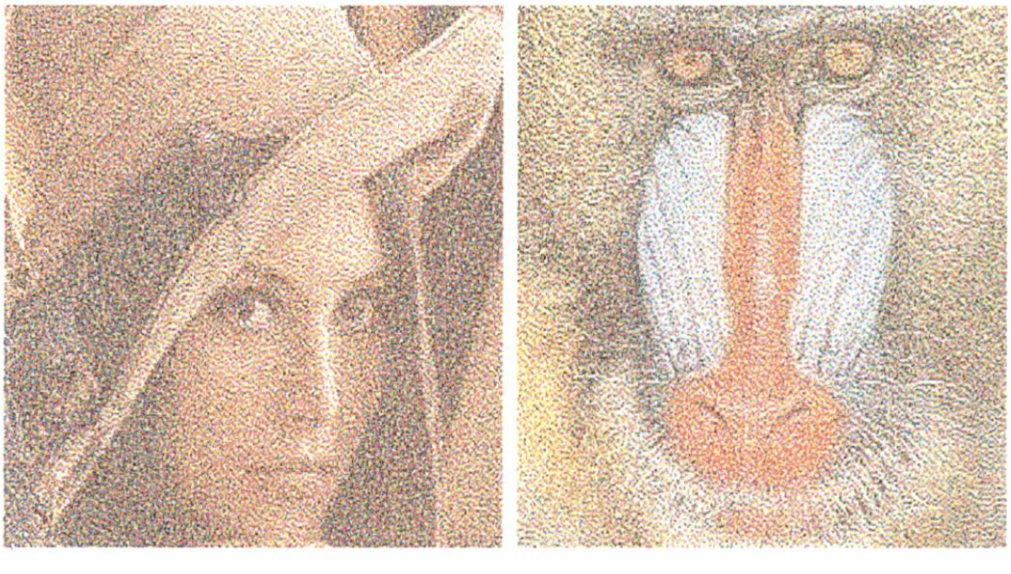


(c) Ball (d) Toy

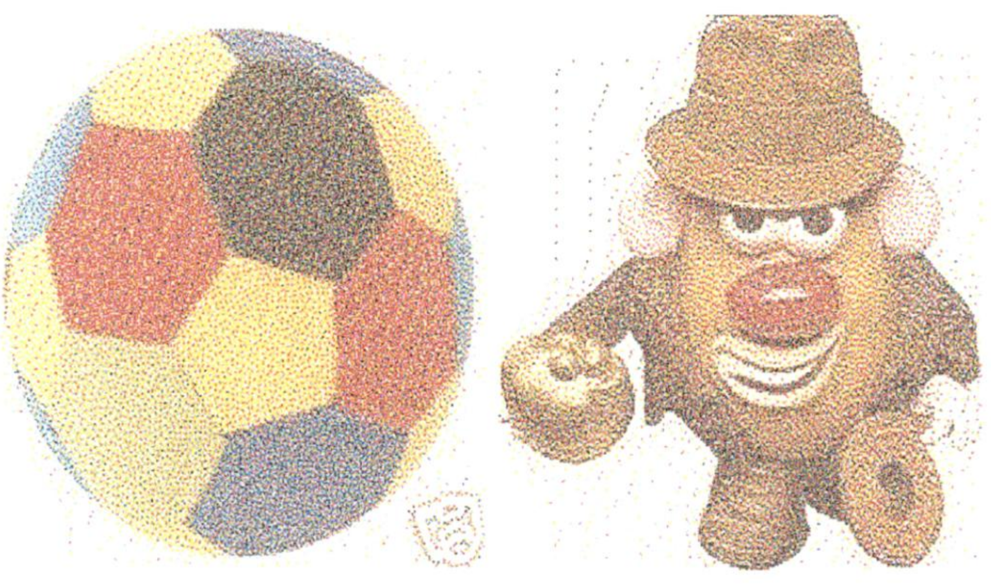
Figure 4.5: Recovered Image (a) Lena (b) Baboon (c) Ball (d) Toy

4.1.3 Reconstructed Secret Quality Analysis

The proposed method aims to reconstruct the secret image with improved quality. The image quality can be compared visually. The secret images are taken as Figure 4.1(a), (b), (c), (d). The results in Figure 4.6 for Chang algorithm can be compared with the results of proposed algorithm in Figure 4.5. It can be observed from the result shown in Figure 4.6 that the proposed method reconstructs the secret image with a better image quality and contrast than the reconstructed image from Chang's [17] method.



(a) Lena (b) Baboon



(c) Ball **(d) Toy**

Figure 4.6: Results for Chang algorithm (a)Lena (b) Baboon (c) Ball (d) Toy

4.1.4 Threshold Analysis

Here the distortion in the meaningful shares is analyzed with single secret embedded at different bit-positions and with multiple secrets shared together. The cover image gets distorted after the secret images are embedded in it. The amount of distortion depends upon the bit-position where the secret is embedded and the number of secret images that are embedded together. When the secret is hidden at lower bit-position, distortion produced in the meaningful shares is less as pixel value changes by a smaller value. Distortion is high when higher bit-position is used for embedding the secret because the change in bit-value causes higher changes in the pixel-values. Table 4.1 shows the average distortion of the meaningful shares produced with respect to the original cover image when a single secret image is hidden at different bit-positions. It can be observed that distortion increases by almost two times when the next higher bit-position is used to embed the secret image. Here distortion is calculated for the change in pixel-values of a single primitive color component. Distortion for other primitive color components will follow the same pattern as it will increase almost 2 times on using the next higher bit-position.

Table 4.1: Average distortion in cover image with secret at different bit-positions

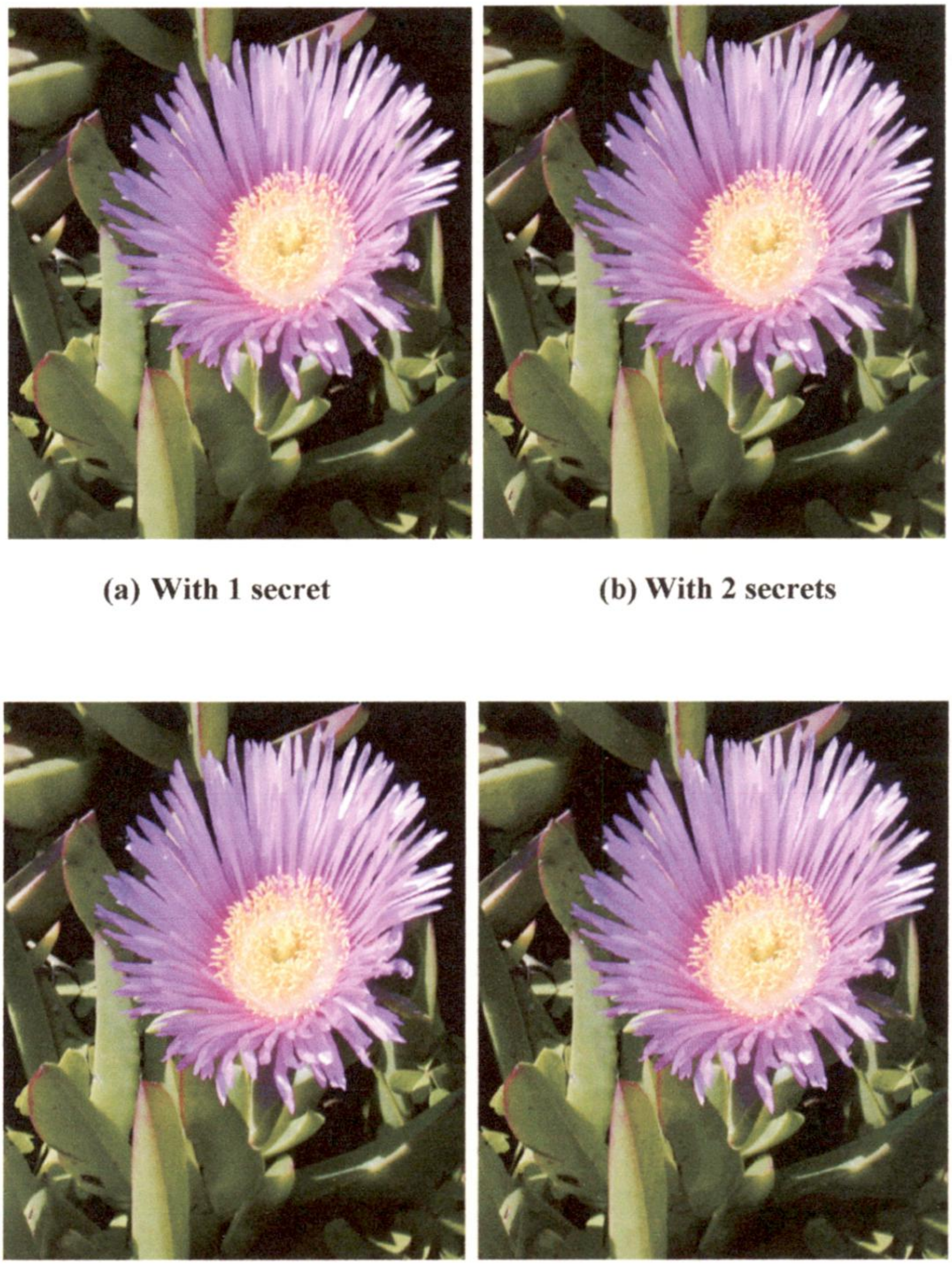
Bit-position	Distortion (in percentage)
1	0.474623005
2	0.949246010
3	1.898492021
4	3.796984043
5	7.593968087
6	15.187936174
7	30.375872349
8	60.751744699

As 8 bit-positions are available for the secrets to hide, total 8 secrets can be shared together (one at each bit-position) in a single cover image. Table 4.2 shows the distortion in the cover image when multiple secrets are shared together. As the number of secret images to be embedded together increases, the distortion also increases. However the proposed technique provides an efficient way to share upto 4 or 5 secrets as in that case distortion is not very high. The cover image gets fully distorted when all bit-positions are used to embed secrets. In that case, the shares produced are no longer meaningful.

Table 4.2: Average distortion in meaningful share with multiple secrets shared

No of secrets	Distortion (in percentage)
1	0.474623005
2	1.190615540
3	2.503422773
4	5.070195852
5	10.351622673
6	20.831571542
7	42.288874190
8	90.132498724

Figure 4.7 shows the meaningful shares produced when different number of secret images is shared in the cover image. As the number of secret images increase, the distortion increases. This fact is also supported from Table 4.2.

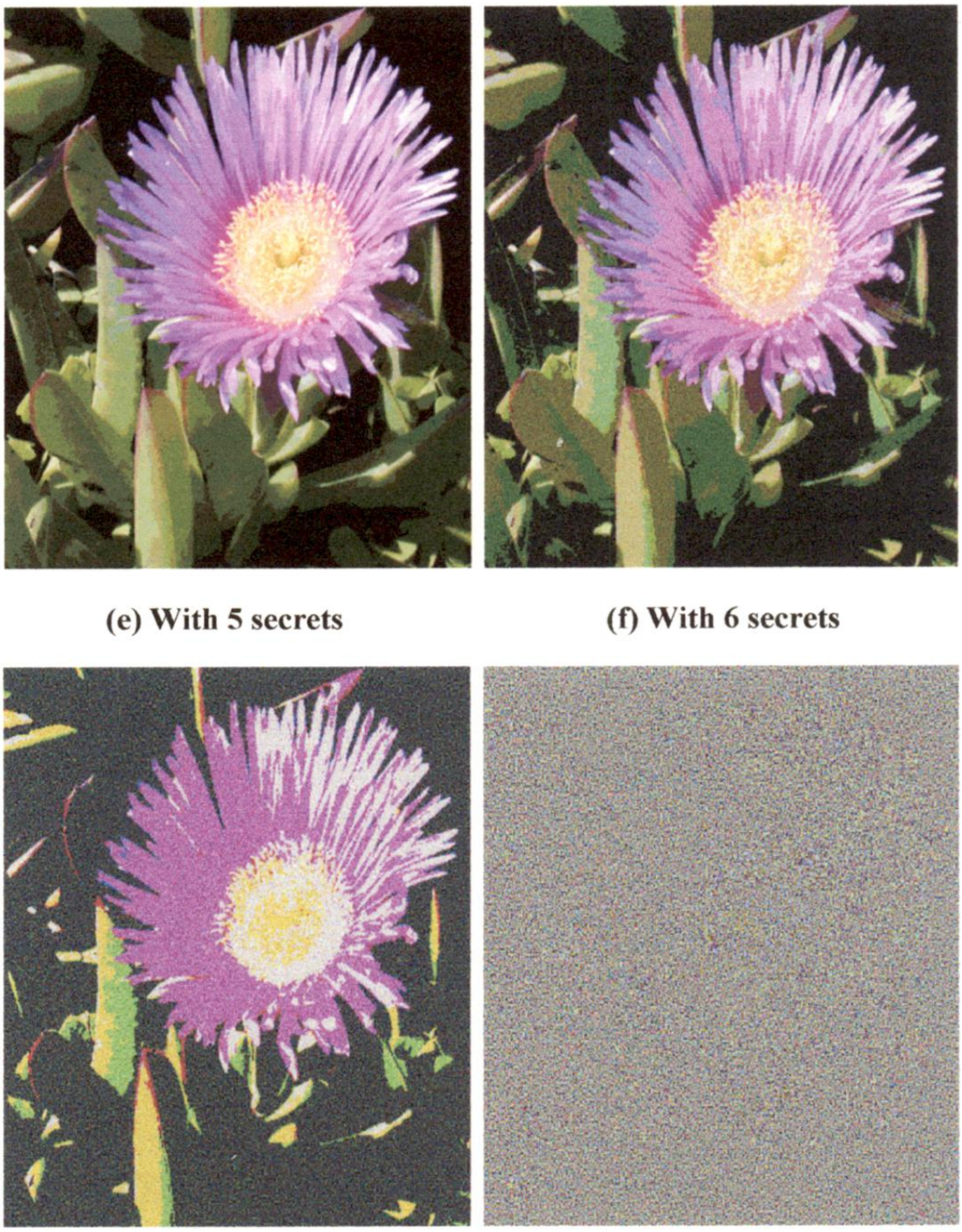


(a) With 1 secret

(b) With 2 secrets

(c) With 3 secrets

(d) With 4 secrets



(e) With 5 secrets

(f) With 6 secrets

(g) With 7 secrets

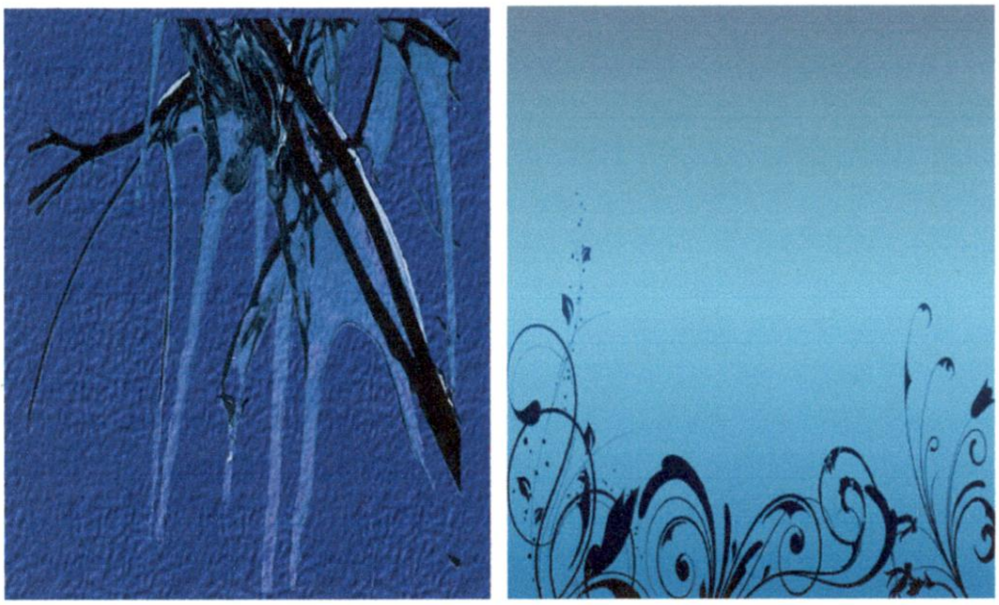
(h) With 8 secrets

Figure 4.7: Innocent share with (a)1 secret (b)2 secrets (c)3 secrets (d)4 secrets (e)5 secrets (f)6 secrets (g)7 secrets (h)8 secrets

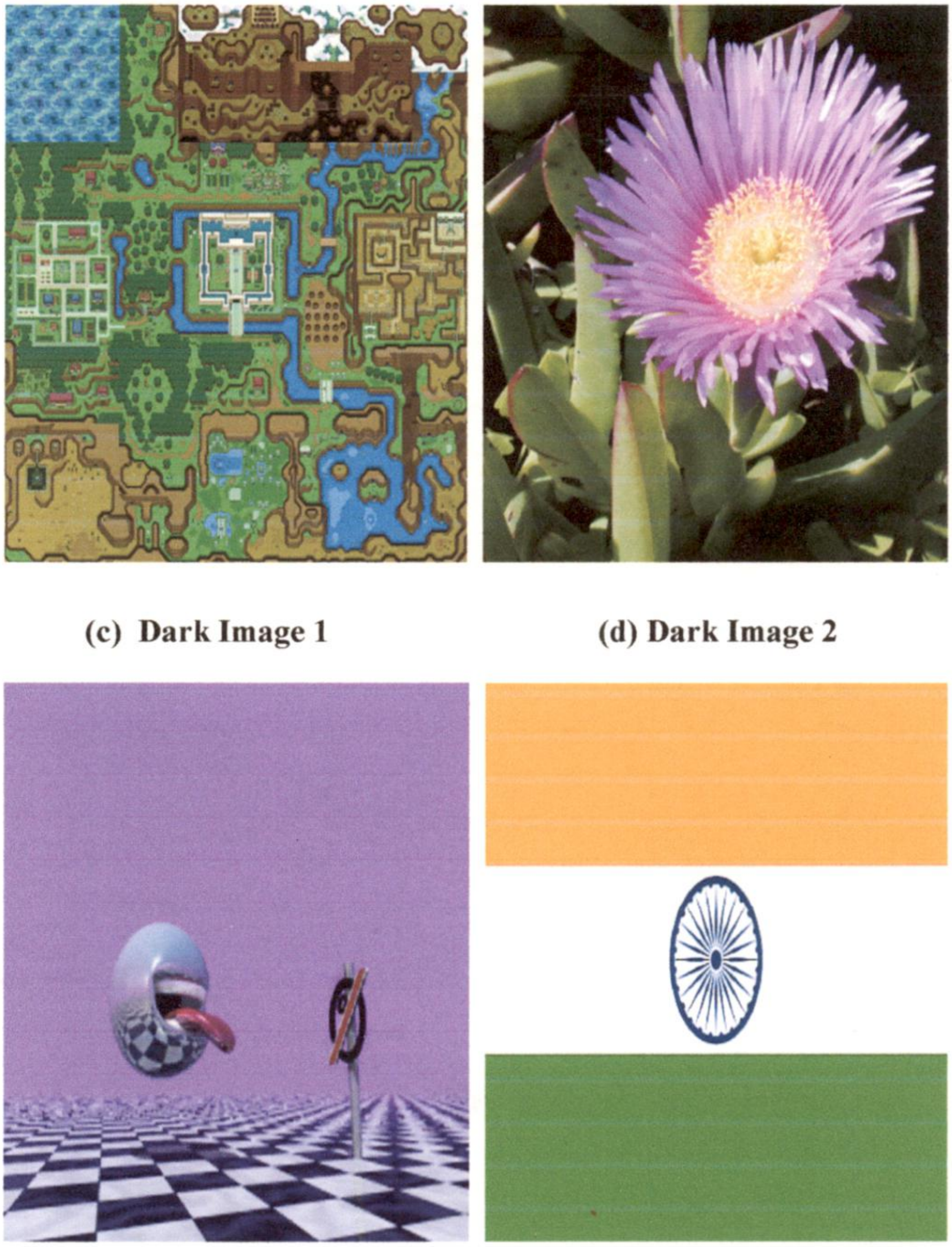
The number of secret images that can be hidden in the cover image depends on the security needs of the application. If the application is highly prone to attackers, distortion more than 5% cannot be tolerated and so maximum 4 secret images should be shared. In this case, innocent share does not differ much from the original cover image. This can be verified from Figure 4.7(e) and 4.7(f). The innocent share changes significantly from the original cover image when number of secret images is 6 and above. The created share is completely distorted when 8 secrets are embedded in the cover image and it does not remain meaningful anymore. Table 4.2 can be used to decide the number of secrets to be shared together after analyzing the security risks of the application.

4.1.5 Meaningful Share Distortion Analysis

The distortion also varies with the type of cover image. It depends on the color components of the cover image. Figure 4.8 contains some images which are tested for the distortion produced after secret embedding. The distortion is found out for single secret embedded in the cover image. For multiple secrets, same trend will follow. Further the secret is embedded at bit-position 8 where the distortion is maximum for single secret sharing. This is done so that distortion variation can be analyzed more clearly. Distortion is calculated for the change in pixel-values of a single primitive color component.



(a) Single Colored 1 **(b) Single Colored 2**



(c) Dark Image 1 **(d) Dark Image 2**

(e) Light Image 1 **(f) Light Image 2**

Figure 4.8: Cover images

The images in Figure 4.8 have varying color elements from each other. To have a better analysis, the images are arranged in three categories- single colored, dark image and light image. These images were tested for distortion produced in them after secret image embedding. Table 4.3 shows distortion of the cover images with secret embedded at different bit-position.

Table 4.3: Distortion in different cover images

Bit-position	Image 4.8(a)	Image 4.8(b)	Image 4.8(c)	Image 4.8(d)	Image 4.8(e)	Image 4.8(f)
1	1.087448	1.080198	0.578331	0.474623	0.357557	0.292992
2	2.174896	2.160397	1.156662	0.949246	0.715114	0.585984
3	4.349793	4.320794	2.313324	1.898492	1.430228	1.171968
4	8.699586	8.641588	4.626648	3.796984	2.860455	2.343937
5	17.399171	17.283176	9.253296	7.593968	5.720910	4.687875
6	34.798343	34.566352	18.506593	15.187936	11.441820	9.375750
7	69.596687	69.132705	37.013186	30.375872	22.883640	18.751500
8	139.193374	138.265411	74.026372	60.751744	45.767281	37.503000

Images with only one or two primitive color components have high distortion with respect to that primitive color which is not present in the cover image. This fact can be verified from the distortion in Figures 4.8(a) and 4.8(b) which contains blue color predominantly. It can also be observed from Table 4.3 that distortion is more in cover images with dark color components. Light colored images have comparatively less distortion. This can be seen from Figures 4.8(e) and 4.8(f). Images in 4.8(e) and 4.8(f) are very light colored images and distortion in that case is less whereas distortion is high for images in 4.8(c) and (d) which are dark-colored. The reason here is that light pixel has high value of gray level for primitive colors and dark pixel has low value of gray level. The white pixel has gray value of (255, 255, 255) for RGB and a black pixel has gray value of (0, 0, 0). A slight change in dark pixel leads to high value of distortion and the image gets more deformed. As the gray level is high in light pixel, deformation is less in that case.

Hence the cover image should not be such that it contains only one or two primitive colors. The cover image should contain all three primitive colors. Also a light-colored cover image should be preferred over dark-colored image so that distortion is minimal and meaningful shares do not differ much from the cover image.

4.2 Analytical Validation

4.2.1 Security Analysis

The proposed scheme does the security enhancement of the shares with the XOR operation. XOR is computationally less intensive than the permutation method used by Chang [17], Wu et. al. [16] and Tsai et. al. [18]. The same XOR operation also performs the encryption and so no extra work is to be done for security enhancement of the shares. The method XORs the original bit-value of the cover image pixel with the pixel-value of the random share created. The produced result becomes the bit-value in the corresponding pixel of the modified cover image. This modified cover image itself is the meaningful share produced. For example, if the bit-value of the selected bit-position in the cover image pixel is 0 and the pixel-value of the color share is 0 for a primitive color, this gives a bit-value of 0 after the modification done by XOR. But this result can also be produced if the values are 1 and 1. Thus the XORed result 0 gives no clue as to whether the pixel-values are 0 or 1. The same argument applies to the second condition. If the bit-value in the cover image is 0 and the pixel-value of the color share is 1, this gives a XOR value of 1. The same result is produced if the values are 1 and 0 respectively. So each possible bit-value in the modified cover image has two possibilities and both of these are equally likely. Thus no information can be gained by looking at any group of pixels on an innocent share. This proves the security of the scheme. Table 4.4 shows two possible combinations for each bit-value of cover pixel value.

Table 4.4: Security analysis of the scheme

Cover pixel bit-value	Pixel-value of color share	Modified cover pixel bit-value
0	0	0
1	1	0
0	1	1
1	0	1

4.2.2 Performance Analysis

The proposed technique uses XOR operation for encryption and decryption so it does not need any data structure and thus reduces any additional storage overhead. Further, the security enhancement of the meaningful shares is done by the XOR operation and not by the permutation as in [16, 17, 18] and so it takes less time for encoding. The proposed technique supports true color images as opposed to [15, 16, 17] which support either black-and-white images or images with limited colors. Finally, decoding again needs only XOR operation of the pixels and so it also takes less time than the methods which need additional data structure lookup to decode the image. The image quality is also good as XOR allows for perfect reconstruction of the pixels. Table 4.5 shows comparison of the proposed technique with some of the existing techniques for producing meaningful shares.

Table 4.5: Comparison of various VCS for producing meaningful shares

Authors	Year	True-color support	(n,n)-scheme supported	Security enhancement	Additional data structure needed
Chang-Yu [15]	2002	No	No	No	Yes
Wu et al. [16]	2004	NA	Yes	Permutation	Yes
Chang et al. [17]	2005	No	No	Permutation	No
Tsai et al. [18]	2009	Yes	Yes	Permutation	Yes
Proposed method	-	Yes	No	XOR	No

Table 4.6 shows comparison of the proposed technique with some of the existing techniques for multiple secret sharing. Method in [21] can share only 2 secrets together. The proposed technique can share upto 8 secret images in a single cover image. Methods in [21, 22] produce circle shares which is not general. Pixel expansion in [22] is 2*n and so the share size gets larger with more number of secret images. Pixel expansion in the proposed method is 4 which is less than [22, 23] and so the share size is small compared to them.

Table 4.6: Comparison of various VCS for sharing multiple secrets

Author	Year	No of secret images	Pixel expansion	Share type
Wu and Chang [21]	2005	2	4	Circle
Shyu et al. [22]	2007	n>=2	2n	Circle
Feng et al. [23]	2008	n>=2	9	Rectangular
Proposed method	-	Upto 8	4	Rectangular

4.2.3 Time Complexity Analysis

The proposed algorithm differs from the algorithm developed by Chang [17] in several ways. Chang algorithm first mixes the pixels using toral automorphism and then replaces their pixel values with the generated share's pixel values. Toral automorphism is a permutation function that moves each pixel from a grid to another grid. The proposed method merely XORs the pixel value of the cover image with the pixel value of the created share. The time complexities for encoding process of both the algorithms can be described as below:

Chang algorithm: 1. Perform the toral automorphism of the pixels.

2. Replace each with the created share's pixel value.

3. Perform reverse toral automorphism.

The first step performs toral automorphisms and the last step performs reverse toral automorphsms. This implies that the pixels are moved a total of R times (r times in first step and R-r times in last step), where R is the recurrence time of toral automorphism. If n is the total number of pixels, the time taken for the first and last step can be considered as O(R*n) as it is done for all the n pixels. R is constant, and so it turns out to be O(n). The time complexity for the second step is also O(n). The second step is done for each of the n pixels. So the total time can be written as c1*n+c2*n (where c1 and c2 are constants).

Proposed algorithm: XOR the pixel value of the cover image with the pixel value of the created share.

The time complexity for the XOR step is O(n) as it is repeated for all the n pixels. So the total time can be written as c3*n (where c3 is a constant).

The time complexity of both the algorithms turns out to be the same. But it can be understood that the Chang's algorithm takes much more time as it also has to perform toral automorphism and inverse toral automorphism. The value of c1+c2 is much higher than the value of c3.

4.3 Advantages over Existing Techniques

The proposed technique improves the performance over previous methods as follows:

- *No additional data structure required*: As shown in Table 4.4, the proposed technique does not need any additional data structure for the purpose of encoding and decoding. Encoding and decoding needs only XOR operation to be done and spends no time in look up for additional data structure as in [15, 16, 18]. These methods also need this data structure to be sent along with the created shares for the decoding of secret image. There is no such requirement in this case. This improves the computation and transmission time. Further, no storage is required to keep the data structure.
- *Efficient encoding*: The proposed technique provides efficient encoding than the methods in [16, 17, 18] as those methods require permutation of pixels of the image first and then encoding is performed. Here the single operation XOR does both the work- security enhancement as well as the encoding. XOR is a much cheaper operation than the permutation followed by encoding and thus the encoding time is reduced.
- *Secure*: The method in [15] does not provide security enhancement for the meaningful shares. Some of the existing methods [16, 17, 18] do the security enhancement of the created innocent shares by permuting the pixels of the image.

Here the security enhancement is done by the XOR operation. The bit-value in the pixels of the innocent share has two choices for the corresponding arrangement of the random shares created from the original secret image and each possibility is equally likely. Hence the method is secure. Table 4.3 demonstrates the security of the technique.

algorithm generates image with a better quality than the Chang [17] scheme. Chang [17] scheme uses OR operation for decoding and the proposed method uses XOR. With XOR, pixel reconstruction is more accurate. Hence the image quality improves *significantly.*

- *Single algorithm sharing multiple secrets with meaningful shares*: The technique developed provides advantage of both the schemes- meaningful share creation and multiple secret sharing. Meaningful shares help in improving the security against the attackers. With meaningful shares, the attackers do not guess the presence of the secret image embedded into the cover image and so they do not try for decoding. Sharing multiple secrets together results in creation of lesser number of share images and so the overhead of creating, transmitting and maintaining too many shares is reduced. The space requirement for keeping those many shares is also reduced.
- *Share size*: The size of the shares is fixed and is independent of the number of colors appearing in the secret image. Further the pixel expansion in this case is 4 as opposed to the methods in [22] where pixel expansion is 2 times the number of secret images and in [23] where pixel expansion is 9. Here the share size increases only four times than the original secret image.

Thus the proposed technique improves in many factors such as computation time, security, space requirement and image quality than the previously proposed schemes and can therefore be used for real life applications such as transmitting financial documents and military information.

Here the security enhancement is done by the XOR operation. The bit-value in the pixels of the innocent share has two choices for the corresponding arrangement of the random shares created from the original secret image and each possibility is equally likely. Hence the method is secure. Table 4.3 demonstrates the security of the technique.

- *Good image quality*: The proposed algorithm generates image with a better quality than the Chang [17] scheme. Chang [17] scheme uses OR operation for decoding and the proposed method uses XOR. With XOR, pixel reconstruction is more accurate. Hence the image quality improves significantly.
- *Single algorithm sharing multiple secrets with meaningful shares*: The technique developed provides advantage of both the schemes- meaningful share creation and multiple secret sharing. Meaningful shares help in improving the security against the attackers. With meaningful shares, the attackers do not guess the presence of the secret image embedded into the cover image and so they do not try for decoding. Sharing multiple secrets together results in creation of lesser number of share images and so the overhead of creating, transmitting and maintaining too many shares is reduced. The space requirement for keeping those many shares is also reduced.
- *Share size*: The size of the shares is fixed and is independent of the number of colors appearing in the secret image. Further the pixel expansion in this case is 4 as opposed to the methods in [22] where pixel expansion is 2 times the number of secret images and in [23] where pixel expansion is 9. Here the share size increases only four times than the original secret image.

Thus the proposed technique improves in many factors such as computation time, security, space requirement and image quality than the previously proposed schemes and can therefore be used for real life applications such as transmitting financial documents and military information.

Chapter 5
Conclusions and Future Work

5.1 Conclusions

The proposed technique combines the feature of meaningful shares with multiple secret sharing to improve the computation time, space requirement and image quality. Meaningful share creation is very important in visual cryptography to improve the security measures by misleading the attacker. The technique has been validated using different types of secret and cover images to see the distortion produced in the created meaningful share. Distortion increases with the number of secret images. However this is an efficient way and can be used to share upto 4 or 5 secret images. This scheme is very much suitable for real-life applications such as transmitting financial and military information, which requires fast computation, less storage overhead and is prone to attackers.

The algorithm eliminates the requirement of any additional data structure for the encoding and decoding operations. The computation time improves as no look-up to that data structure is required. Further, no additional storage is required to keep the data structure. The security enhancement of the scheme is done using XOR operation instead of permutation used in previously proposed solutions for providing meaningful shares. This also reduces the encoding time. The proposed technique produces image with enhanced quality than many of the previously proposed method. The share size is fixed and does not vary with the number of colors present in the secret. The proposed technique draws the advantages of both the visual cryptography concepts- meaningful shares and multiple secret sharing.

5.2 Future Work

In the future the improvements can be done in the following areas:

- In (k,n)-threshold technique, n shares are produced where at least k out of n shares are needed to regenerate the secret image. (k,n)-threshold technique cannot be applied for this proposed method as only 2 shares are created for the secret images and both the shares are needed to reconstruct the secrets. Future work may adapt (k,n)-threshold technique using an alternative method for creating random shares.

- The number of secret images to be shared together cannot go beyond 8. Also in that case the shares will not be innocent. The algorithm can be improved to share more number of secret images.
- The method can be extended to regenerate the secret images with further improved image quality.

REFERENCES

[1] Adi Shamir, "How to Share a Secret," ACM, Laboratory for Computer Science, 1979.

[2] M. Naor and A. Shamir, "Visual cryptography, Advances in cryptology EUROCRYPT'94," in Lecture Notes in Computer Science. vol. 950, Springer, Berlin, 1995, pp. 1-12.

[3] G. Ateniese, C. Blundo, A. D. Santis, and D. R. Stinson, "Visual cryptography for general access structures," Information and Computation, vol. 129, pp. 86-106, 1996.

[4] Y. C. Hou, S.-F. Tu, "A Visual Cryptographic Technique for Chromatic Images Using Multi-pixel Encoding Method," Journal of Research and Practice in Information Technology, vol. 37, pp. 179-191, 2005.

[5] L. W. Hawkes, A. Yasinsac and C. Cline, "An Application of Visual Cryptography to Financial Documents," Technical report TR001001, Florida State University, 2000.

[6] D. Chaum, "Secret-ballot receipts: True voter verifiable elections," IEEE Security and Privacy, vol. 2, pp. 38-47, 2004.

[7] C. Lin and W. Tsai, "Visual cryptography for gray-level images by dithering techniques," Pattern Recognition Letters, vol. 24, pp. 349-358, 2003.

[8] A. Altaf, R. Sirhindi, A. Ahmed, "A Novel Approach against DoS Attacks in WiMAX Authentication using Visual Cryptography", The Second International Conference on Emerging Security Information, Systems and Technologies, SECURWARE, Cap Esterel, France 2008.

[9] Y. C. Hou, "Visual cryptography for color images," Pattern Recognition, vol. 36, pp. 1619-1629, 2003.

[10] E. R. Verhuel and V. Tilborg, "Construction and Properties of k out of n visual secret sharing schemes," Designs, Codes and Cryptography, vol. 11, pp. 179-196, 1997.

[11] V. Rijmen and B. Preneel, "Efficient colour visual encryption for shared colors of Benetton," in Eurocrypto'96, Rump Session, Berlin, 1996.

[12] C. A. Poynton, "Frequently asked questions about color," http://www.inforamp.net/~poynton.

[13] F. Y. N. Bert W. Leung, Duncan S. Wong, "On the security of a visual cryptography scheme for color images," *Pattern Recognition,* vol. 42, pp. 929-940, 2009.

[14] C. Chang, C. Tsai, and T. Chen, "A New Scheme For Sharing Secret Color Images In Computer Network," Proceedings of International Conference on Parallel and Distributed Systems, pp. 21–27, 2000.

[15] Chin-Chen Chang, Tai-Xing Yu, "Sharing A Secret Gray Image In Multiple Images," Proceedings of the First International Symposium on Cyber Worlds (CW.02), 2002.

[16] Y.S. Wu, C.C. Thien, J.C. Lin, Sharing and hiding secret images with size constraint, Pattern Recognition, vol. 37, pp. 1377–1385, 2004.

[17] Chin-Chen Chang, Jun-Chou Chuang, Pei-Yu Lin, "Sharing A Secret Two-Tone Image In Two Gray-Level Images", Proceedings of the 11th International Conference on Parallel and Distributed Systems (ICPADS'05), 2005.

[18] D.-S. Tsai, G. Horng, T.-H. Chen, Y.-T. Huang, "A Novel Secret Image Sharing Scheme For True-Color Images With Size Constraint," Information Sciences, vol. 179, pp. 3247–3254, 2009.

[19] G. Voyatzis and I. Pitas, "Applications of toral automorphisms in image watermarking", *Proceedings of International Conference on Image Processing*, Lausanne, Switzerland, Vol. 2, 1996, pp. 237–240.

[20] S. Droste, "New results in visual cryptography," Advances in cryptology – CRYPTO '96, Lecture Notes in Computer Science, No. 1109, pp. 401–415, 1996.

[21] H.-C. Wu and C.-C. Chang, "Sharing visual multi-secrets using circle shares," Computer Standards & Interfaces, vol. 28, pp. 123-135, 2005.

[22] S. J. Shyu, S.-Y. Huang, Y.-K. Lee, Ran-Zan Wang and K. Chen, "Sharing multiple secrets in visual cryptography," Pattern Recognition, vol. 40, pp. 3633-3651, 2007.

[23] J.-B. Feng, H.-C.Wu, C.-S Tsai, Y.-F.Chang, and Y.-P. Chu, "Visual secret sharing for multiple secrets," Pattern Recognition, vol. 41, pp. 3572-3581, 2008.

LIST OF PUBLICATIONS

[1] **Jaya** and Anjali Sardana. "Multiple Secrets Sharing With Meaningful Shares," International Conference on Advances in Computing and Communications (ACC 2011), Publisher: LNCS Springer. (Registered)

[2] **Jaya** and Anjali Sardana. "Visual Cryptography – A survey," IEEE symposium on Computer and Informatics (ISCI 2011), Publisher: IEEE Xplore. (Accepted)