B.Sc. in Computer Science and Engineering Thesis

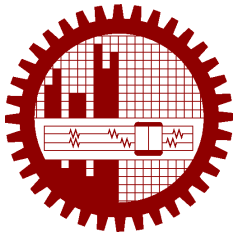
Secured High Capacity Image Steganography Using XOR Operation

Submitted by

Md. Tanjim Rejoanul Islam
201305013

Supervised by

Dr. Md. Shohrab Hossain



Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

Dhaka, Bangladesh

October 2018

CANDIDATES' DECLARATION

This is to certify that the work presented in this thesis, titled, "Secured High Capacity Image Steganography Using XOR Operation", is the outcome of the investigation and research carried out by us under the supervision of Dr. Md. Shohrab Hossain.

It is also declared that neither this thesis nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Md. Tanjim Rejoanul Islam
201305013

CERTIFICATION

This thesis titled, **"Secured High Capacity Image Steganography Using XOR Operation"**, submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in October 2018.

Group Members:

Md. Tanjim Rejoanul Islam

Supervisor:

Dr. Md. Shohrab Hossain
Associate Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

ACKNOWLEDGEMENT

At first, I would like to thank my thesis advisor Dr. Md. Shohrab Hossain of the department of CSE at Bangladesh University of Engineering and Technology. He consistently allowed this thesis to be my own work, but steered me in the right direction whenever he thought I needed it.

I would also like to thank my previous advisor Dr. Mohammad Mahfuzul Islam, former professor of department of CSE, BUET who advised me a lot in the first stage of this thesis and without him, it was not possible for me to start this thesis .

I would also like to acknowledge all of my friends that help me by giving invaluable advice in various parts of this thesis.

Finally, I must express my very profound gratitude to my parents for providing me with unending support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Dhaka
October 2018

Md. Tanjim Rejoanul Islam

Contents

CANDIDATES' DECLARATION i

CERTIFICATION ii

ACKNOWLEDGEMENT iii

List of Figures vi

List of Tables vii

List of Algorithms viii

ABSTRACT ix

1 Introduction 1

1.1 Steganography . . . 1
1.2 Motivation and Problem Statement . . . 1
1.3 Summary of Existing Works and Their Limitations . . . 2
1.4 Contributions of the Thesis . . . 2
1.5 Organization of the Thesis . . . 2

2 Background and Literature Review 4

2.1 Terminologies . . . 4
2.1.1 Image . . . 4
2.1.2 Grey scale image . . . 5
2.1.3 RGB image . . . 5
2.1.4 Carrier or cover image . . . 5
2.1.5 Stego medium . . . 6
2.1.6 Embedded or payload . . . 6
2.1.7 Steganalysis . . . 6
2.1.8 Capacity . . . 6
2.1.9 Robustness . . . 6
2.1.10 Undetectability . . . 6
2.1.11 Invisibility . . . 7

2.1.12 Security . . . 7
2.2 Data Hiding Techniques . . . 7
2.2.1 Injection . . . 7
2.2.2 Substitution . . . 8
2.2.3 Generation . . . 8
2.3 Types of Substitution . . . 8
2.3.1 Spatial Domain Techniques . . . 8
2.3.2 Transform Domain Technique . . . 8
2.4 Related Works . . . 9
2.4.1 XOR Replacement Based Steganography . . . 9
2.4.2 Key Based Steganography . . . 14
2.4.3 LSB Based Steganography . . . 15
2.4.4 Wavelet Based Steganography . . . 15
2.4.5 XOR Cipher Based Steganography . . . 15
2.4.6 Phase Based Steganography . . . 15
2.5 Limitations of Existing Works . . . 16

3 Proposed Methodology **17**
3.1 Novelty of the Proposed Method . . . 17
3.2 Embedding Algorithm . . . 17
3.3 Pseudocode of the Embedding Algorithm . . . 18
3.4 Retrieving Algorithm . . . 21
3.5 Pseudocode of the Retrieving Algorithm . . . 21
3.6 Advantage of the proposed algorithm . . . 22

4 Result **24**
4.1 Performance Metrics . . . 24
4.1.1 MSE . . . 24
4.1.2 PSNR . . . 25
4.2 Baseline Parameter Setting . . . 25
4.3 Tables and Figures . . . 25
4.4 Summary of finding . . . 29

5 Conclusion **30**
5.1 Conclusion . . . 30
5.2 Future Work . . . 30

References **32**

List of Figures

2.1 A picture of an image with pixels 4
2.2 A gray scale image 5
2.3 An RGB image 5
2.4 Types of steganography 9
2.5 Data embedding using xor for the last 3 bits 10
2.6 Data retrieving using xor for the last 3 bits 11
2.7 Data hiding using xor for the last 3 bits 12
2.8 Data hiding using xor for the last 3 bits 13

3.1 Embedding Algorithm 20
3.2 Retrieving Algorithm 22

4.1 Standard test images 27
4.2 The stego images of Baboon for n=1, n=2, n=4, n=8 27
4.3 The stego images of F-16 for n=1, n=2, n=4, n=8 28
4.4 The stego images of Pepper for n=1, n=2, n=4, n=8 28
4.5 The stego images of Sailboat on lake for n=1, n=2, n=4, n=8 28
4.6 The stego images of Splash for n=1, n=2, n=4, n=8 28

List of Tables

4.1 Performance result for proposed method for n = 8 25
4.2 Performance result for proposed method for n = 4 26
4.3 Performance result for proposed method for n = 2 26
4.4 Performance result for proposed method for n = 1 26
4.5 Comparisons of PSNR Values for Hiding Capacity of 1 bpp 29

List of Algorithms

1 Embedding 19
2 Retrieving 21

ABSTRACT

Data or information hiding without seeking much attention is one of the most important requirements now-a-days. The existing steganography approaches are not fully satisfying as that require a lot of complexity or distort the cover image more without hiding a good amount of information. In this thesis, we have developed a simplistic way of hiding a good amount of data without distorting the cover image significantly. We have implemented our proposed approach using MATLAB. Our results show that PSNR values for our proposed method is high enough to say that our method is a balanced one in every aspect considering the payload and size of the cover image. Our approach can be useful to hide a large amount of data inside a small image without significant distortion.

Chapter 1

Introduction

Steganography has gained significant importance over years for its art of writing hidden messages in such a way, that no one, apart from the sender and intended recipient, suspects the existence of the message, it is a form of security through obscurity. The term steganography is derived from two Greek words steganos and graphein. Steganos means covered and graphein means to write. It is a process of hiding information in some sort of cover. Steganography techniques hide the existence of the message.

1.1 Steganography

This chapter introduce us to the soul content of the entire thesis. The meaning of Steganography, its purposes and various works and their limitations about this topic are discussed shortly. Moreover, our goal and motivation behind this work and contributions of our effort to the modern world of Steganography also take place. At last the whole overview of this thesis give a basic idea of it.

- Protection of private data is the main application of steganography. The urgency of protecting a private file is raising day by day.
- Confidential communication is another important aspect of today's world. Almost all the country need private communication with another country without any kind of interception.

1.2 Motivation and Problem Statement

Steganography has various types of importance. Some of them are :

- Large file can be transmitted through the network without drawing attention.
- Ensure security all over the network.
- Transmission load is reduced.

We have chosen steganography over other topics because of its significance for sending data and information over the network by hiding in plain sight. Moreover, steganography ensures privacy and prevent the leakage of critical information which may cause serious troubles.

There are so many processes and styles of steganography techniques are available now a days which are also very efficient but they have some major issues to solve such as data embedding capacity is not so high when the distortion is insignificant and vice versa. Moreover unsecured embedding technique is another headache in the world of steganography.

1.3 Summary of Existing Works and Their Limitations

Most of the XOR based steganography works are based on LSB (least significant bit) of the cover image. XOR based steganography methods generally work directly with the bits of the cover file and that is why they are mainly consist of spatial domain techniques.

Image distortion is one of the main limitations of XOR based steganography. Besides, low embedding capacity and not using all the bits of a pixel are also significant.

1.4 Contributions of the Thesis

- We have proposed a method that uses a full or part of a pixel to hide data. So, the efficiency is high and embedding capacity is increased.
- We have implemented our proposed method using MATLAB.
- We have compared our method with existing ones in terms of PSNR. The PSNR value of our method is significantly higher than some recent works.

1.5 Organization of the Thesis

The rest of this thesis is organized in the following way:

Chapter 2 contains some terminologies which are essential for any steganography techniques and also some related works of XOR based steganography.

Chapter 3 introduces the proposed methodology of this thesis for hiding data using XOR operation. This chapter consists of two parts. One for describing the data embedding technique and the other is for data retrieving.

Chapter 4 carries the implementation process and result analysis of the proposed method.

Chapter 5 which is the last chapter, concludes the paper

Chapter 2

Background and Literature Review

In this chapter is all about the essential terms of Steganography which will help us to understand the rest of the thesis. There are also descriptions of some common techniques of data hiding which are broadly used in various parts of this topic. Finally there are some short descriptions of thesis works which are related to our topic.

2.1 Terminologies

2.1.1 Image

An image is nothing but a picture that has been created or copied and stored in some sort of electronic form. An image can be described in terms of vector graphics or raster graphics . An image which is stored in raster form is also known as a bitmap. An image is a collection of numbers that constitute different light intensities in different areas of the image. This numeric representation forms a grid and the individual points are referred to as pixels (picture element).



Figure 2.1: A picture of an image with pixels

2.1.2 Grey scale image

Grey scale images use 8 bits for each pixel and are able to display 256 different colours or shades of grey. In this thesis, we work on only grey scale images.



Figure 2.2: A gray scale image

2.1.3 RGB image

RGB images are typically stored in 24-bit files and use the RGB colour model, also known as true colour. RGB images are also known as digital color image. All colour variations for the pixels of a 24-bit image are derived from three primary colours: red, green and blue, and each primary colour is represented by 8 bits. Thus in one given pixel, there can be 256 different quantities of red, green and blue.



Figure 2.3: An RGB image

2.1.4 Carrier or cover image

Cover image means an original image where the secret data or information will be hidden. This image may or may not be distorted. Distortion means the deviation of the cover image after hiding the data from the original image. This distortion largely depends on algorithm and process of the data hiding. This also hugely depends on embedding capacity. As a general rule of thumbs distortion is inversely proportional to embedding capacity.

2.1.5 Stego medium

This is the medium in which the secret data or information is hidden. The medium may be a image, text, audio or video. Steganography may be of various types on basis of these mediums. In this thesis, we deal with image. In other words, our stego medium is an image. So, it is in the kind of image steganography. The cover image may be a png or jpg format image.

2.1.6 Embedded or payload

The secret data or information are known as the embedded or payload. The payload itself may be an image, video, audio or text. The distortion of the cover image depends on the size of the payload. Generally distortion is proportional to the size of the payload.

2.1.7 Steganalysis

Steganalysis is the process of detecting hidden information inside the cover image. A robust steganography algorithm should be less vulnerable to any kind of steganalysis. Sometimes strong steganalysis is required to detect a hidden message that is supposed to be a crime or terror information.

2.1.8 Capacity

The notion of capacity in data hiding indicates the total number of bits hidden and successfully recovered by the system. A good system is supposed to have a high capacity with a small size of cover image.

2.1.9 Robustness

Robustness refers to the ability of the embedded data to remain intact if the stego-system undergoes transformation, such as linear and non-linear filtering; addition of random noise; and scaling, rotation, and loose compression.

2.1.10 Undetectability

The embedded algorithm is undetectable if the image with the embedded message is consistent with a model of the source from which images are drawn. For example, if a Steganography method uses the noise component of digital images to embed a secret message,it should do

so while not making statistical changes to the noise in the carrier. Undetectability is directly affected by the size of the secret message and the format of the content of the cover image.

2.1.11 Invisibility

This concept is based on the properties of the human visual system or the human audio system. The embedded information is imperceptible if an average human subject is unable to distinguish between carriers that do contain hidden information and those that do not. It is important that the embedding occurs without a significant degradation or loss of perceptual quality of the cover.

2.1.12 Security

It is said that the embedded algorithm is secure if the embedded information is not subject to removal after being discovered by the attacker and it depends on the total information about the embedded algorithm and secret key.

2.2 Data Hiding Techniques

There are mainly three different techniques that can be used to hide information in a cover object:

- Injection
- Substitution
- Generation

2.2.1 Injection

Injection technique hides data in part/s of a file that are ignored by the processing application. Therefore file bits that are relevant to an end-user are not modified and leaving the cover file entirely usable. For example, we can add additional bytes in an executable or binary file because those bytes don't affect the process and the end-user may not even realize that the file contains additional hidden information. However, using an insertion technique changes file size according to the amount of data hidden and therefore, if the file looks unusually large, it may arouse suspicion.

2.2.2 Substitution

The main theme of substitution technique is to replace some of the cover files specific bits to the bits of the hidden information. Thus, this process hamper the purity of the cover file and in some case may make it unable to be used perfectly but the most important and useful characteristic of this process is it does not change the size of the file and hence avoid any kind of suspicion that may arrive in injection process. The main two drawbacks are the limitation of the capacity and distortion of the cover file.

2.2.3 Generation

Generation techniques do not use an existing cover file rather this technique generates a cover file for the sole purpose of hiding the message. The main flaw of the insertion and substitution techniques is that people can compare the stego object with any pre-existing copy of the cover object (which is supposed to be the same object) and discover differences between the two. We will not have that problem when using a generation approach, because the result is an original file, and is therefore immune to comparison tests.

2.3 Types of Substitution

In this thesis, we approach with substitution technique. So some types of substitution should be discussed. There are mainly two types of substitution techniques:

- Spatial Domain Techniques
- Transform Domain Techniques

2.3.1 Spatial Domain Techniques

Special domain techniques are consist of using the bits of cover file to directly represent the bits of hidden information. In this thesis, we use this kind of technique to hide data.

2.3.2 Transform Domain Technique

During transform domain techniques the cover image has to go through a transformation into its frequency domain and then the coefficients are changed to hide the secret information.

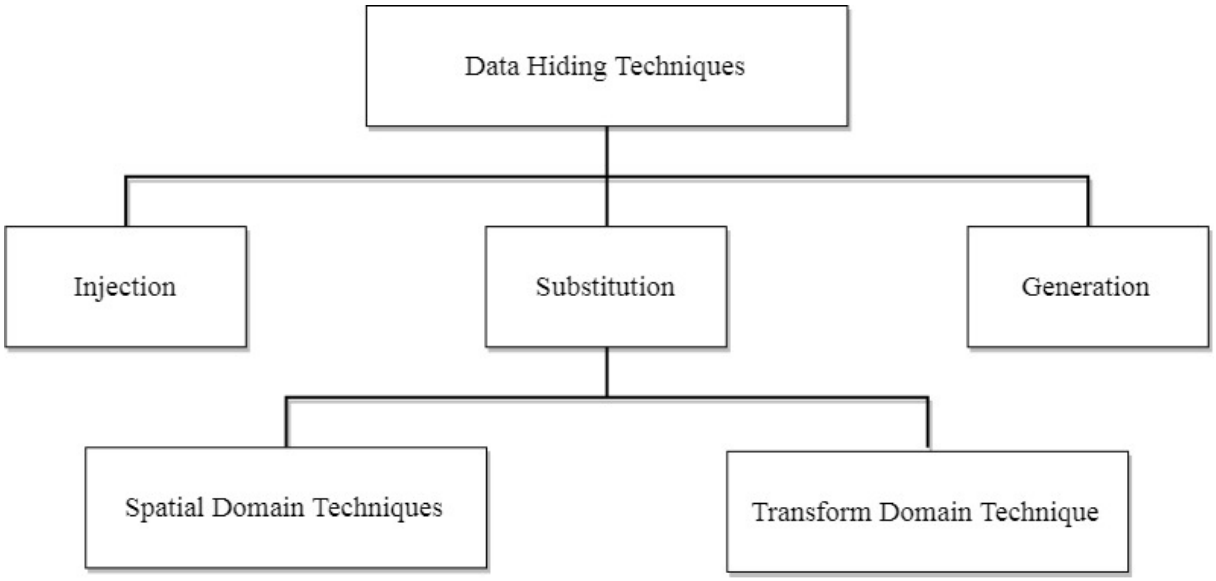


Figure 2.4: Types of steganography

2.4 Related Works

2.4.1 XOR Replacement Based Steganography

Joshi et al. [1] proposed a three bit XOR steganography system for concealing messages into gray scale images.In this method, last three bits of pixel value offer 100 percentage of message addition. This new technique uses the XOR operation between the message and original image. They also protect the message from any intruder by keeping it decoded in the image. The embedding and retrieving algorithms are shown in Fig. 2.5 and Fig. 2.6.

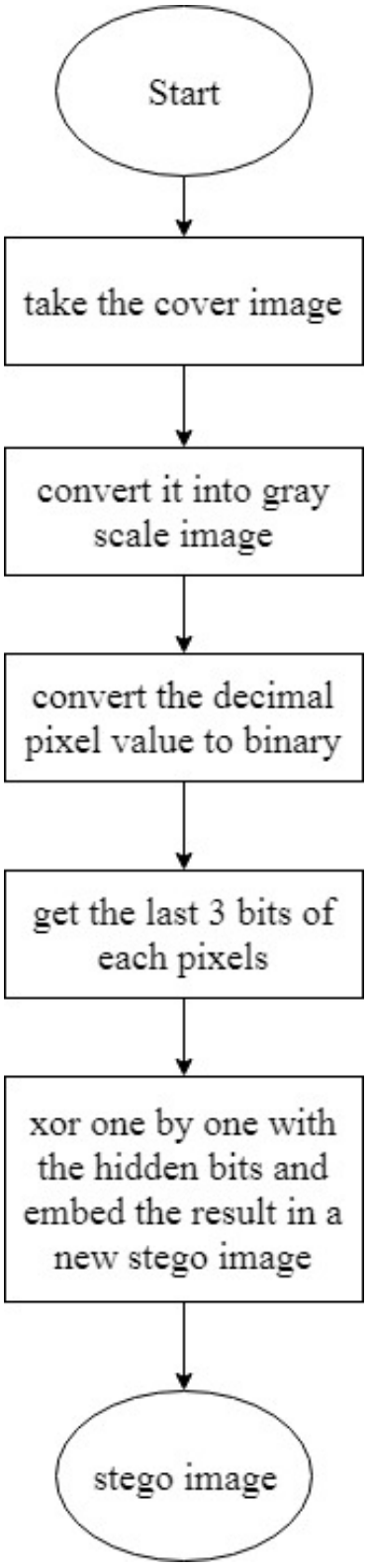


Figure 2.5: Data embedding using xor for the last 3 bits

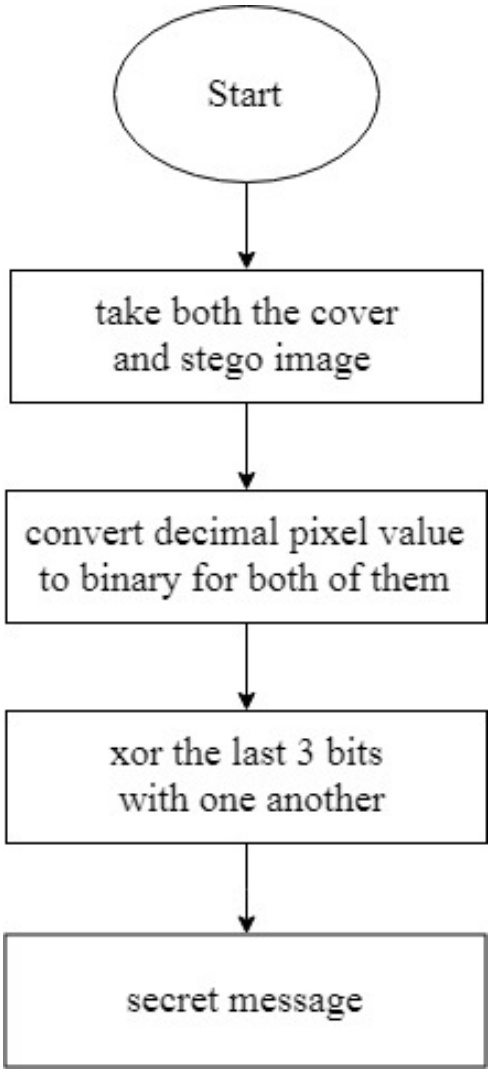


Figure 2.6: Data retrieving using xor for the last 3 bits

Joshi et al. [2] present a new technique for image steganography in spatial domain. Their proposed method extracts two LSBs and two MSBs of the selected pixel value. Then perform the XOR operation on first and last bit and second bit and seventh bit. On the basis of result of these two XOR operations every bit of secret data is embedded one by one on LSB of selected pixel value. The objective of this technique is to maximize the PSNR rate and invisibility of the secret data with minimal modification to the cover image.The embedding and retrieving algorithms are shown in Fig. 2.7o$_e mbedandFig.2.8$

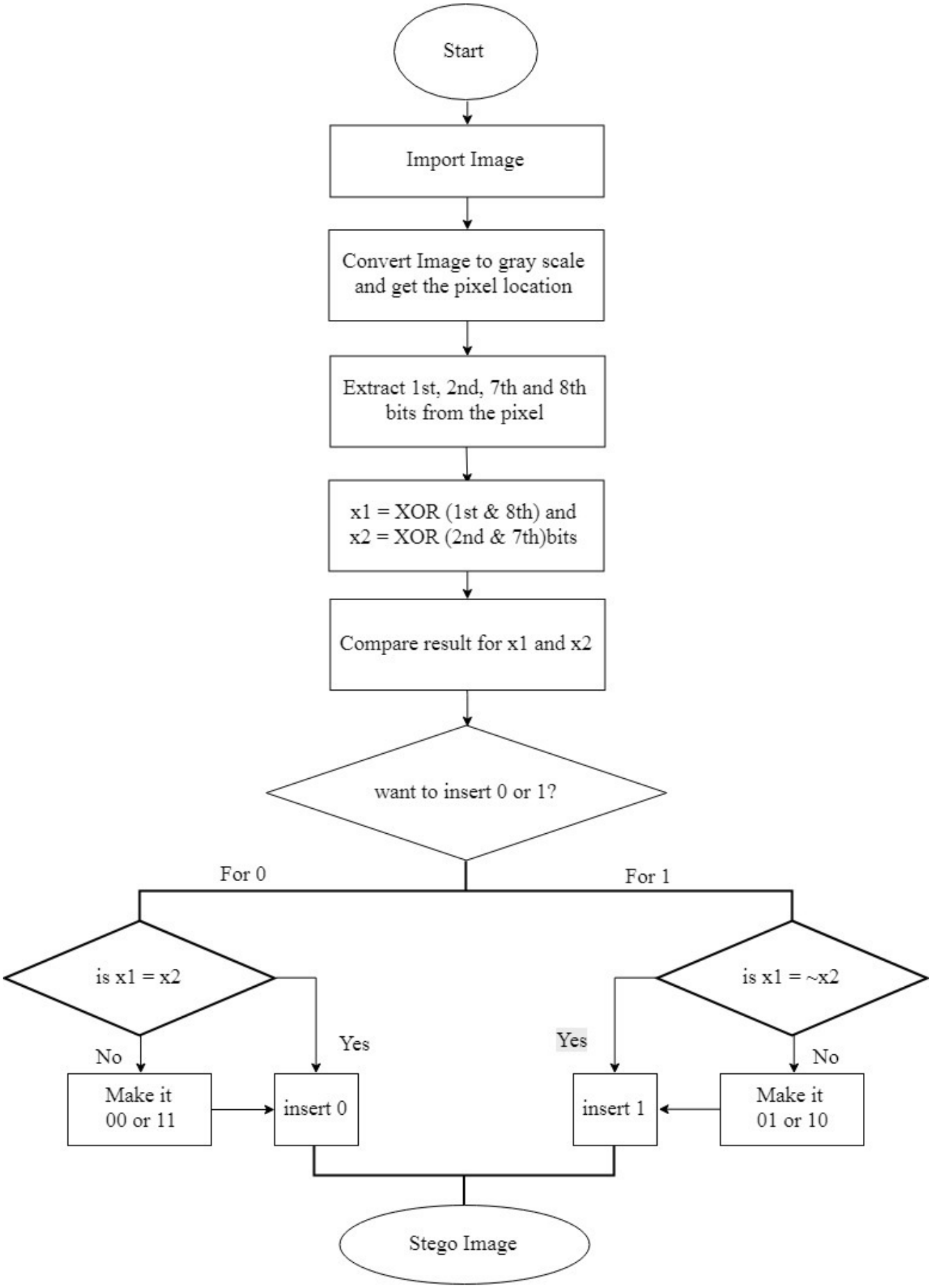


Figure 2.7: Data hiding using xor for the last 3 bits

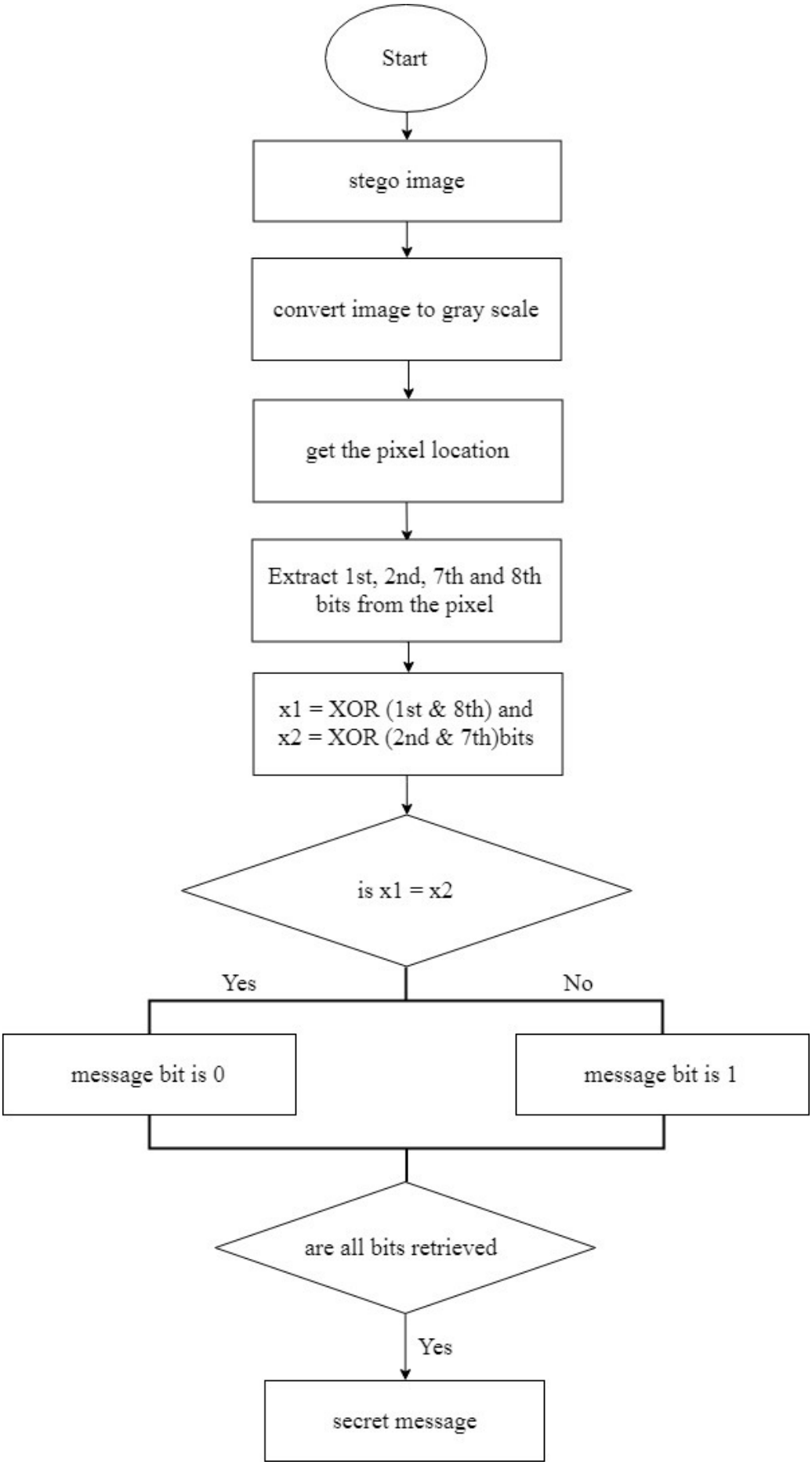


Figure 2.8: Data hiding using xor for the last 3 bits

Vandana et al. [3] conducted their research to solve pixel expansion and poor image quality of visual cryptography.In this research, effective technique of share generation based on XOR-based visual cryptography for General Access Structures is introduced. Perfect restoration of the secret, no pixel expansion and no code book requirement are the advantages that the algorithm is expected to have. The generated shares are then covered in an image using steganography which provides additional security.

Chi-Shiang Chan et al. [4] propose a hybrid data hiding method to embed three secret bits to four cover pixels. In the proposed method, a Hamming code word consists of three parity check bits whose values are identical to three secret bits and four data bits whose values are derived by performing XOR Function on four cover pixels. In order to keep the Hamming code word satisfying the condition of (7, 4) Hamming Code, at most one cover pixel would be modified. If the modified bit is a data bit of the code word, it is directly modified, and three secret bits can be embedded to the cover pixels successfully. However, if the modified bit is a parity check bit, namely the secret bit, instead of modifying the parity check bit, two cover pixels are needed to be modified while the proposed method only requires one bit modification by adopting the XOR Function. Our hybrid method only need to modify one cover pixel by adding/subtracting its values to/from one, and two resultant bits of XOR Function are able to be adjusted to the desired bits. In this way, the number of modified pixels can be further reduced.

Chin-Chen Chang et al. [5] use the XOR operation to solve the increment of compression codes. The XOR operator does not generate any negative number, thus the proposed method does not record the signal value.

2.4.2 Key Based Steganography

Tapodhir et al. [6] propose a new model of key-based text steganography. Various methods in this type of key-based steganography do not work when the cover file contained single letters and words which have the same starting and ending letter. The proposed approach works on this limitation and conceals a message, without degrading cover, by using XOR operation on the start and end letter of words of the cover.

Amitava Nag et al. [7] propose a new location transformation based encryption technique. They redistribute the pixel values to different location using affine transform technique with four 8-bit keys. The transformed image then divided into 2 pixels 2 pixels blocks and each block is encrypted using XOR operation by four 8-bit keys. The total key size used in their algorithm is 64 bit which proves to be strong enough. The experimental results proved that after the affine transform the correlation between pixel values was significantly decreased.

2.4.3 LSB Based Steganography

Arun et al. [8] introduce a design of image steganography using LSB XOR substitution method. They use a random 8 bit secret key that initially XOR with the RGB colors which leads to hold the embedding of data sharing and then after replace the LSB of pixel we get extracted data which is an actual message from encoding method.It has the storage capacity to hide a large number of characteristics than the existing features of data hiding and also data sharing.

Mohanjeet et al. [9] worked on the center weighted least significant bit method with hidden object encryption to embed the small images in the medium or large sized images. The proposed method employs the spatial domain embedding method using the RGBR for embedding each 8-bit pixel into the cover image hence consume 2 pixels for hiding each 2-bit pixel. The multi-layered bitwise XOR method has been used along with the same sized key to realize the lightweight Elliptic Curve cryptography.

2.4.4 Wavelet Based Steganography

Shaheen S. Patel et al. [10]realize that now-a-days it is very important that patients privacy is protected while data are being transmitted over the public network as well as when they are stored in hospital servers.So they introduce a new technique which is the hybrid of encryption and wavelet-based ECG steganography technique. Encryption allows privacy and ECG steganography allows to hide one sensitive data into other insensitive host thus guaranteeing the integration between ECG and the rest.

2.4.5 XOR Cipher Based Steganography

S Arul Thileeban [11] proposes using XOR Cipher to encrypt the binary data in images pixel by pixel rather than securing it with an application so that it cannot be exploited or cracked easily. The proposed model explains multiple ways to encrypt the Image using XOR Cipher and the analysis shows that by using the proposed model, the images are properly encrypted.

2.4.6 Phase Based Steganography

Chang-Mok Shin et al. [12] propose that The multilevel image can be divided to binary images that have same gray levels. They convert each binary image to binary phase image by phase encoding and encrypt these images with binary random phase images by binary phase XOR operation. They make encrypted gray image by combining each binary encrypted images. They implement the decryption process with joint transform correlator (JTC) and verify the proposed

method by computer simulation.

2.5 Limitations of Existing Works

- Most of the XOR based embedding approach deal with least significant bit of any pixel and hence left the rest of the bits unused for hiding. [1]
- Distortion is a common factor for many of the XOR based embedding procedure. Regardless of the size of the hidden information many procedures distort the cover image. [2]
- Embedding capacity is another main factor in stegamography, specially in spatial domain. Most of the procedures suffer from low embedding capacity for keeping the image less distorted. [3]

Chapter 3

Proposed Methodology

In this chapter, we introduce our full method of Steganography in details. The first part we describe the data embedding technique and the second part describe the data retrieving technique.

3.1 Novelty of the Proposed Method

Our proposed method has the following novelty compared to previous ones. We use variable number of bits of a pixel of an image to embed a particular character of a text file. This allows the system to choose the value of n (number of bit of a pixel) according to the type of the image or according to the result which allows it to embed data more efficiently than any other fixed pixel embedding algorithm. Moreover, our proposed method can choose a pixel among the whole image to embed data rather than pick it randomly or serially which allows us to keep the distortion as low as possible.

3.2 Embedding Algorithm

Here we present our proposed steganography meyhod using XOR operation.

- First we need three things to do this. A cover image, a text file (which has to be hidden) and a 2D array of size m*n where m is the number of pixel in a row and n is the number of pixel in a column of the cover image.
- Then the cover image is taken where we hide the text. The cover image may be a png or jpg format image. Then this cover image is transferred to gray scale image (if it is not already a gray scale image).

- Then we go through the text file character by character (along with space, punctuation mark etc.) and take the corresponding ASCII value of the character.
- Then we go through the whole cover image pixel by pixel and execute XOR operation between the pixel and the ASCII value of the character.
- After going through the whole image with that particular ASCII value we embed it where the XOR operation gives the lowest value and make the corresponding entry of the 2D array to the number of the character in the text file from first.
- For example, suppose we are trying to embed the first character of the text file into the image. After going through the whole image we find that the pixel of the fifth row and sixth column gives the lowest value after XOR operation with the character. Then we replace the pixel with that ASCII value and make the (5, 6) entry of that 2D array equals to 1 as it is the first character of the text file.

3.3 Pseudocode of the Embedding Algorithm

Let the length of the hidden message is n, the number of row and column of the cover image is r and c respectively. The first seven line of the algorithm runs in constant time. Then the next three nested loops run in O(n), O(r) and O(c) time respectively. In the third nested loop the calculation can be done in constant time. The rest of the first loop at the end of the algorithm can also be done in constant time. So, the total run time of the embedding algorithm is O(n*r*c). The pseudocode of the algorithm is shown in algorithm 1 and the flowchart of the algorithm is shown in Fig.3.1

Algorithm 1 Embedding

```
c ← coverImage
stegoImage ← c
row ← rowofc
column ← columnofc
length ← lengthofHiddentext
key[length][2]
contain[row][column] ← 0
for i ← 1 to length do
    min ← 99999
    for j ← 1 to row do
        for k ← 1 to column do
            if contain[row][column] ≠ 1 then
                if ithcharacter xor c[row][column] < min then
                    min ← ithcharacter xor c[row][column]
                    r ← j
                    c ← k
                end if
            end if
        end for
    end for
    stegoImage[r][c] ← ithcharacter
    key[i][1] ← r
    key[i][2] ← c
    contain[r][c] ← 1
end for
```

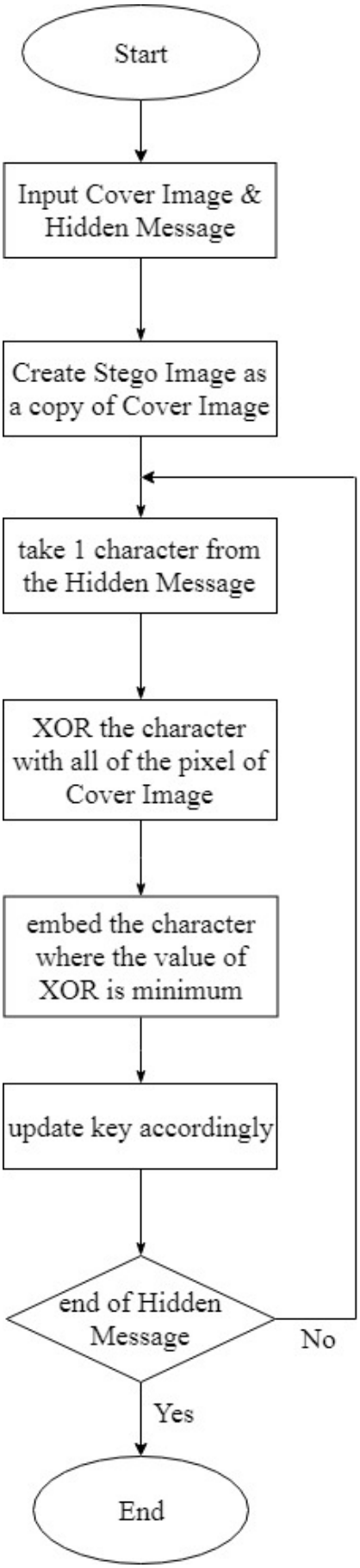


Figure 3.1: Embedding Algorithm

3.4 Retrieving Algorithm

The retrieving procedure is as follows:

- Firstly the receiver receives both the image and the 2D array.
- The array works like a key for that image. The corresponding values of the pixels are taken as ASCII and are arranged according to that 2D array.
- After going through the whole stego image the receiver gets the hidden message.

3.5 Pseudocode of the Retrieving Algorithm

Let the length of the row that means the number of pair of row and column is r. Then the first three line of the retrieving algorithm run in constant time. The next for loop of the algorithm runs in O(r) time. Inside the for loop all the calculation can be done in constant time. So, the overall run time of the retrieving algorithm is O(r). The pseudocode of the algorithm is shown in algorithm 2 and the flowchart of the algorithm is shown in Fig.3.2.

Algorithm 2 Retrieving

$s \leftarrow stegoImage$
$k \leftarrow key$
$row \leftarrow rowofkey$
for $i \leftarrow 1$ to row **do**
  $r \leftarrow key[i][1]$
  $c \leftarrow key[i][2]$
  append $s[r][c]$ to retrieved message
end for

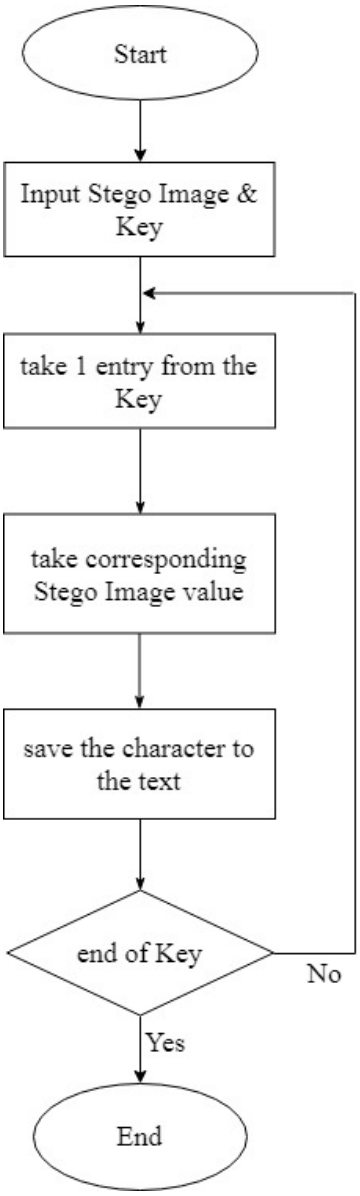


Figure 3.2: Retrieving Algorithm

3.6 Advantage of the proposed algorithm

Our proposed algorithm performs better for various reasons. They are explained below:

- Our proposed embedding method can vary the number of least significant bit for embedding data in a particular pixel. So, we can choose the value of n according to our advantage to perform better.
- We choose a particular pixel to embed a particular character of the hidden message by checking all the available pixel of the image. This allows us to choose the best pixel for that particular character to embed.

- We can use a lot more bits of the cover image to embed data than most other algorithm which increases the embedding capacity of our system.
- We try to embed data so that the value of the mismatch between the image pixel and hidden byte is minimum which causes minimum distortion for the stego image. This makes our proposed system immune to steganalysis.
- We have implemented the proposed method for greyscale image only but it can also be extended to RGB image which may perform better than greyscale approach as there are more pixel and combination to hide data.
- Our method works even better when the payload or the size of the hidden message is not so large compared to the size of the image. In this scenario we can even embed data maintaining zero distortion.
- It is impossible to retrieve data from the stego image without the help of the key which is generated during embedding. So, our system is also sound in security issues.

Chapter 4

Result

We have implemented this steganograpy technique using MATLAB. We have embedded only text in png image. We use five images of miscellaneous catagory from SIPI image database that are Baboon, F-16, Peppers, Sailboat on lake and Splash. All of them are of 128 x 128 size and in RGB format. First we convert them into greyscale and then embed 16384 bits of data in each of them according to the algorithm. Following four tables contain the values of MSE and PSNR for different values of n which denotes the number of bits in each pixel of the image that is used to embed the text.

4.1 Performance Metrics

We have derived two performance metrics that are most commonly used in image steganography.

4.1.1 MSE

MSE is the Mean Square Error representing the distortion between the original cover image C and stego image S sized r * c. The equation to calculate MSE is :

$$MSE = \sum\sum(C_{ij} - S_{ij})^2/r \times c \tag{4.1}$$

where C_{ij} and S_{ij} are the grayscale values of the pixels located at the i th row and the j th column of the cover image C and stego image S.

4.1.2 PSNR

We used the Peak Signal-to-Noise Ratio (PSNR) to measure the distortion between the original cover image C and the stego image S . The PSNR is defined by :

$$PSNR = 10 \times \log_{10}(255^2/MSE) \tag{4.2}$$

4.2 Baseline Parameter Setting

We have set the image size and hidden text size as our baseline parameter. All the grayscale images are of 128*128 size and the size of the text file is 2048 bytes.

4.3 Tables and Figures

n = 8(whole pixel)					
Cover Image	MSE	PSNR	%of pixel used	% of pixel changed	Time(s)
Baboon	6.8296	39.7869	12.5	3.2	30.24
F-16	60.0816	30.3434	12.5	7.58	43.05
Peppers	1.3978	46.6765	12.5	4.39	33.65
Sailboat on lake	14.0394	36.6573	12.5	7.61	42.65
Splash	1.0781	47.8041	12.5	2.2	29.2
Average	16.6853	40.2536	12.5	4.99	35.758

Table 4.1: Performance result for proposed method for n = 8

In table 4.1 we have presented the result of our proposed method using n=8 which means we have used the whole pixel to embed a single character.

n = 4					
Cover Image	MSE	PSNR	%of pixel used	% of pixel changed	Time
Baboon	0.0087	68.7216	25	0.87	44.08
F-16	0.0182	65.5328	25	1.82	48.73
Peppers	0.0124	67.2	25	1.24	47.38
Sailboat on lake	0.0104	67.9705	25	1.04	45.72
Splash	0.0165	65.9453	25	1.65	50.65
Average	0.01324	67.074	25	1.324	47.312

Table 4.2: Performance result for proposed method for n = 4

In table 4.2 we have presented the result of our proposed method using n=4 which means we have used the half of a pixel to embed half of a single character.

n = 2					
Cover Image	MSE	PSNR	%of pixel used	% of pixel changed	Time
Baboon	0	Infinity	50	0	24.31
F-16	0	Infinity	50	0	21.77
Peppers	0	Infinity	50	0	32.48
Sailboat on lake	0	Infinity	50	0	28.85
Splash	0	Infinity	50	0	26.82
Average	0	Infinity	50	0	26.846

Table 4.3: Performance result for proposed method for n = 2

In table 4.3 we have presented the result of our proposed method using n=2 which means we have used the quarter of a pixel to embed the quarter of a single character.

n = 1(single bit)					
Cover Image	MSE	PSNR	%of pixel used	% of pixel changed	Time
Baboon	0.0360	62.5665	100	3.6	131.89
F-16	0.0494	61.1902	100	4.94	135.34
Peppers	0.0494	61.1955	100	4.94	130.77
Sailboat on lake	0.0463	61.4726	100	4.63	135.28
Splash	0.0475	61.3596	100	4.75	156.9
Average	.0457	61.5569	100	4.57	139.034

Table 4.4: Performance result for proposed method for n = 1

In table 4.4 we have presented the result of our proposed method using n=1 which means we have used a single bit of a pixel to embed the single bit of a single character.



Figure 4.1: Standard test images

In Fig. 4.1, five standard test images are shown. The first one is Baboon, the second one is the F-16, the third one is Peppers, the fourth one is Sailboat on lake and the last one is Splash.

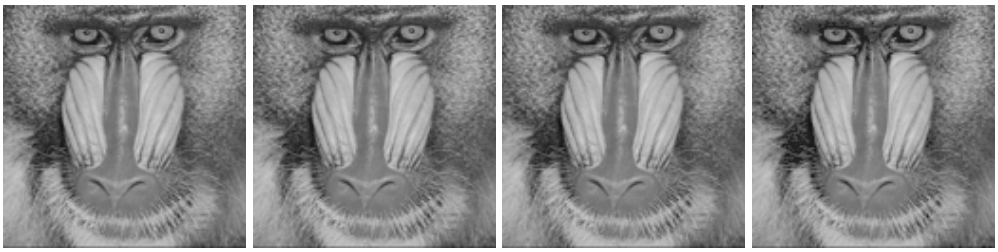


Figure 4.2: The stego images of Baboon for n=1, n=2, n=4, n=8

In Fig. 4.2, we are showing the stego images of Baboon for various values of n. Here the image is most distorted for n=8 and least distorted for n=2 because it is better to hide a character by dividing it into several parts rather than embedding it directly but for n=1 the parts are too much for the size of the picture and so distortion is not so low.



Figure 4.3: The stego images of F-16 for n=1, n=2, n=4, n=8

In Fig. 4.3, we are showing the stego images of F-16 for various values of n. Here the image is most distorted for n=8 and least distorted for n=2 because of the same reason.



Figure 4.4: The stego images of Pepper for n=1, n=2, n=4, n=8

In Fig. 4.4, we are showing the stego images of Peppers for various values of n. Here the image is most distorted for n=8 and least distorted for n=2 because of the same reason.



Figure 4.5: The stego images of Sailboat on lake for n=1, n=2, n=4, n=8

In Fig. 4.5, we are showing the stego images of Sailboat on lake for various values of n. Here the image is most distorted for n=8 and least distorted for n=2 because of the same reason.

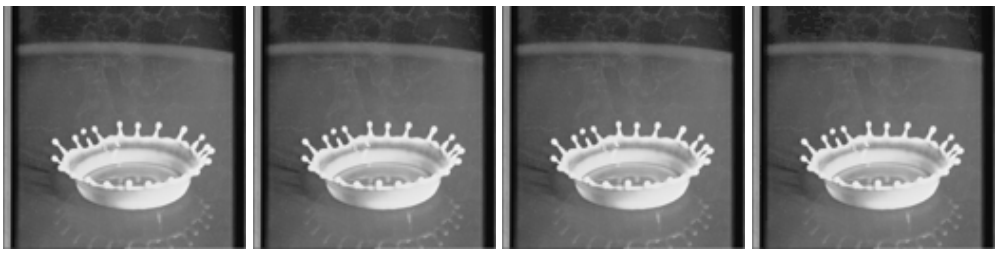


Figure 4.6: The stego images of Splash for n=1, n=2, n=4, n=8

In Fig. 4.6, we are showing the stego images of Splash for various values of n. Here the image is most distorted for n=8 and least distorted for n=2 because of the same reason.

Now, we want to make some comparisons with other recent works to show the superiority of our proposed method. The performance of our method in terms of PSNR at the hiding capacity of 1 bpp is compared with some methods with the results appearing in table 4.5

Methods	Cover Images			
	Baboon	F16	Pepper	Average
Simple LSB [13]	51.14	51.15	51.13	51.14
LSB-M-R [14]	52.39	52.39	52.39	52.39
OPAP [15]	51.14	51.15	51.13	51.14
EMD [16]	52.10	52.12	52.11	52.11
Section-wise [17]	52.92	52.92	52.92	52.92
Proposed n = 8	39.79	30.34	46.68	38.94
Proposed n = 4	68.72	65.53	67.2	67.15
Proposed n = 2	Infinity	Infinity	Infinity	Infinity
Proposed n = 1	62.57	61.19	61.2	61.65

Table 4.5: Comparisons of PSNR Values for Hiding Capacity of 1 bpp

4.4 Summary of finding

From the above tables we can find some interesting point. they are given below:

- Generally it is better to keep the value of n lower than the size of a character in bit because lower value of n can help us to hide a character in more than one pixel.
- If the value of n is such low that the number of used pixel is very near to the number of pixel of the image then the stego image can be distorted a good amount.
- The value of n should be choose according to the size of the cover image and hidden message for the best performance.

Chapter 5

Conclusion

In this chapter, we summarize the contributions of this thesis and the possible impact and discuss the directions of future work.

5.1 Conclusion

A steganography method should be such that it can embed data into an image without any distinguishable change that can be noted by a human eye. As this is the key requirement for this thesis, we try to maintain that throughout the thesis.

Though many steganography methods has been developed over the years, many of them cannot fulfill all of the demands perfectly. So, it becomes necessary to develop a method that can give optimum results.

In this thesis, we have proposed a high capacity steganography method based on XOR operation. This method also maintains a key to the hidden message which ensures security.

After analyzing the result and comparison with other well established methods, we can conclude that the proposed method has some significant advantages and is appropriate for various steganography based applications.

5.2 Future Work

- We have presented our method for only grayscale image. A future work may be on RGB image image for proposed scheme.
- Generally most of the image have similar bit pattern in a certain area. If one letter can be embedded in an area then there is a high probability that the next time for the same letter

it can be embedded in the same area. So, an intelligent method for finding a suitable byte for a particular character can reduce the run time significantly.

- Now our proposed scheme has a large size of key. Reducing the size of the key is another challenge for the future.
- We hide only text file in this thesis. Hiding image or any other kind of file can be another interesting task in future.

References

[1] K. Joshi and R. Yadav, "New approach toward data hiding using xor for image steganography," in *Ninth International Conference on Contemporary Computing (IC3)*, (Noida, India), pp. 1–6, IEEE, August 11-13, 2016.

[2] K. Joshi, P. Dhankhar, and R. Yadav, "A new image steganography method in spatial domain using xor," in *Annual IEEE India Conference (INDICON)*, (New Delhi, India), pp. 1–6, IEEE, December 17-20, 2015.

[3] V. Purushothaman and S. Sreedhar, "An improved secret sharing using xor-based visual cryptography," in *Online International Conference on Green Engineering and Technologies (IC-GET)*, (Coimbatore, India), pp. 1–4, IEEE, November 19, 2016.

[4] C.-S. Chan and C.-Y. Chang, "Hiding secret in parity check bits by applying xor function," in *Cognitive Informatics (ICCI), 9th IEEE International Conference on*, (Beijing, China), pp. 835–839, IEEE, July 7-9, 2010.

[5] C.-C. Chang, Y.-H. Huang, and W.-C. Chang, "Reversible data hiding for vq indices using xor operator and soc codes," in *International Conference on Machine Learning and Cybernetics (ICMLC)*, (Guangzhou, China), pp. 340–345, IEEE, July 12-15, 2015.

[6] T. Acharjee, A. Konwar, R. K. Ram, R. Sharma, and D. Goswami, "Xorsteg: A new model of text steganography," in *International Conference on Communication and Electronics Systems (ICCES)*, (Coimbatore, India), pp. 1–4, IEEE, October 21-22, 2016.

[7] A. Nag, J. P. Singh, S. Khan, S. Biswas, D. Sarkar, and P. P. Sarkar, "Image encryption using affine transform and xor operation," in *International Conference on Signal Processing, Communication, Computing and Networking Technologies*, (Thuckafay, India), pp. 309–312, IEEE, July 21-22, 2011.

[8] C. Arun and S. Murugan, "Design of image steganography using lsb xor substitution method," in *International Conference on Communication and Signal Processing (ICCSP)*, (Chennai, India), pp. 0674 – 0677, IEEE, April 6-8, 2017.

[9] M. Kaur and M. Juneja, "A new lsb embedding for 24-bit pixel using multi-layered bit-wise xor," in *International Conference on Inventive Computation Technologies (ICICT)*, (Coimbatore, India), pp. 1–5, IEEE, August 26-27, 2016.

[10] S. S. Patel and S. V. Sankpal, "Secure patients data transmission using xor ciphering encryption and ecg steganography," in *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, (Chennai, India), pp. 1311–1315, IEEE, March 3-5, 2016.

[11] S. A. Thileeban, "Encryption of images using xor cipher," in *IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, (Chennai, India), pp. 1–3, IEEE, December 15-17, 2016.

[12] C.-M. Shin, D.-H. Seo, K.-B. Cho, H.-W. Lee, and S.-J. Kim, "Multilevel image encryption by binary phase xor operations," in *CLEO/Pacific Rim. The 5th Pacific Rim Conference on Lasers and Electro-Optics (IEEE Cat. No.03TH8671)*, (Taipei, Taiwan), p. 426, IEEE, December 15-19, 2003.

[13] T. Sharp, "An implementation of key-based digital signal steganography,," in *4th International Workshop on Information Hiding, LNCS 2137*, pp. 13–26, 2001.

[14] J. Mielikainen, "Lsb matching revisited,," in *IEEE Signal Processing Letters,*, p. 285287, IEEE, April 10, 2006.

[15] C. K. Chan and L. M. Cheng, "Hiding data in images by simple lsb substitution,," in *Pattern Recognition, Vol. 37, No. 3*, p. 469474, 2004.

[16] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction,," in *IEEE Communications Letters,*, pp. 781 – 783, IEEE, December 04 , 2006.

[17] Y. S. J. Wang, K. C. H. Xu, H. J. Kim, and S. H. Joo, "An improved section-wise exploiting modification direction method,," in *Signal Processing, Vol. 90, No. 11,*, pp. 2954–2964, November 2010.

Generated using Undegraduate Thesis LaTeX Template, Version 1.4. Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh.

This thesis was generated on Saturday 20th October, 2018 at 6:12am.