KADİR HAS UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
COMPUTER ENGINEERING

DEVELOPING NOVEL TECHNIQUES FOR SPATIAL DOMAIN LSB IMAGE STEGANOGRAPHY

DANISH SHEHZAD

DOCTOR OF PHILOSOPHY THESIS

ISTANBUL, JANUARY, 2019

DANISH SEHZAD **PHD Thesis** **2019**

DEVELOPING NOVEL TECHNIQUES FOR SPATIAL DOMAIN LSB IMAGE STEGANOGRAPHY

DANISH SHEHZAD

PHD THESIS

Submitted to the Graduate School of Science and Engineering of Kadir Has University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Engineering

ISTANBUL, JANUARY 2019

DECLARATION OF RESEARCH ETHICS / METHODS OF DISSEMINATION

I, DANISH SHEHZAD, hereby declare that;

- this PhD Thesis is my own original work and that due references have been appropriately provided on all supporting literature and resources;
- this PhD Thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- I have followed "Kadir Has University Academic Ethics Principles" prepared in accordance with the "The Council of Higher Education's Ethical Conduct Principles"

In addition, I understand that any false claim in respect of this work will result in disciplinary action in accordance with University regulations.

Furthermore, both printed and electronic copies of my work will be kept in Kadir Has Information Center under the following condition as indicated below:

☑ The full content of my thesis/project will be accessible only within the campus of Kadir Has University.

DANISH SHEHZAD

DATE: 08/01/2019

KADIR HAS UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

ACCEPTANCE AND APPROVAL

This work entitled **DEVELOPING NOVEL TECHNIQUES FOR SPATIAL DOMAIN LSB IMAGE STEGANOGRAPHY** prepared by **DANISH SHEHZAD** has been judged to be successful at the defense exam held on 08/01/2019 and accepted by our jury as **DOCTORATE'S THESIS**.

APPROVED BY:

Assoc. Prof. Dr. Tamer Dag (Kadir Has University) ____________
(Advisor)

Prof. Dr. Feza Kerestecioğlu (Kadir Has University) ____________

Assoc. Prof. Dr. Songül Albayrak (Yildiz Technical University) ____________

Assoc. Prof. Dr. Tansal Güçlüoğlu (Yildiz Technical University) ____________

Asst. Prof. Dr. Taner Arsan (Kadir Has University) ____________

I certify that the above signatures belong to the faculty members named above.

____
(Assoc. Prof. Dr. Ebru Demet AKDOĞAN)
Dean of Graduate School of Science and Engineering
DATE OF APPROVAL: (08/01/2019)

TABLE OF CONTENTS

DEVELOPING NOVEL TECHNIQUES FOR SPATIAL DOMAIN LSB IMAGE STEGANOGRAPHY

ABSTRACT

Steganography is one of the most noteworthy information hiding mechanism, which is used as an alternative to cryptography in order to provide adequate data security. Image steganography is one of the key types of steganography where a message to be transmitted is hidden inside a cover image. The most commonly used techniques for image steganography rely on LSB Steganography. In this thesis, new techniques are developed for LSB image steganography to achieve maximum security, optimal data capacity along with provisioning of efficient steganography mechanism.

In the first part of this work, a novel technique based on pairs matching is developed for LSB image steganography. In this technique MSBs along with LSBs are used in a delicate method for data hiding for the first time. The message bits from the secret information are compared with all defined pixel pairs and replace the least two significant bits with respective matched pair number. This technique shows good quality of stego image along with adequate peak signal to noise ratio and provides high payload of secret message. In the second part, threshold-based LSB image steganography technique is developed. This technique also works in spatial domain and categorizes the pixels based on threshold defined categories. Maximum four bits and minimum one bit is embedded in pixel based on its category. The prominence in THBS is on security and payload as it uses bits proficiently for data embedding. ETHBS allows efficient execution of the algorithm along with provisioning of optimal security. In the last part 1LSB Image steganography technique based on blocks matrix determinant is developed. It is a technique in which data is embedded by making minimal changes in image pixels. This technique is 1LSB substitution technique that works on matrix determinant of 2 by 2 blocks of image pixels. This technique ensures high PSNR and ensures good quality of stego image.

Keywords: Steganography, least significant bits, cover image, stego image, peak signal to noise ratio, mean square error, matrix determinant.

SPATIAL DOMAIN LSB GÖRÜNTÜ STEGANOGRAFİSİ İÇİN YENİ TEKNİKLER GELİŞTİRME

ÖZET

Steganografi, yeterli veri güvenliği sağlamak için kriptografiye alternatif olarak kullanılan en önemli bilgi gizleme mekanizmasindan biridir. Görüntü steganografisi, gönderilecek bir mesajin kapak resmi içerisinde gizlendiği steganografideki temel türlerden biridir. Görüntü steganografisi için en yaygin kullanilan teknikler LSB steganografisine dayanır. Bu tezde, azami(maksimum) güvenliği ve verimli steganografi mekanizması ile optimal veri kapasitesi sağlamak amacıyla LBS görüntü steganografisi için yeni teknikler geliştirilmiştir.

Bu çalışmanın ilk bölümünde, LSB görüntü steganografisi için çift eşleştirmeye dayalı yeni bir teknik geliştirilmiştir. Bu teknikte MSB'ler LSB'ler ile birlikte ilk kez veri saklamak için hassas bir yöntemde kullanılmaktadır. Gizli bilgiden gelen mesaj bitleri(bits) tüm tanımlanmış piksel çiftleriyle karşılaştırılır ve en az iki önemli bit(bits) bire bir eşleşen sayı çifti ile değiştirilir. Bu teknik, stego görüntüsünün gürültü oranı için yeterli uç sinyali ile birlikte iyi kalitesini gösterir ve yüksek miktarda gizli mesaj taşıma kapasitesi sağlar.İkinci kısımda eşik tabanlı LSB görüntü steganografi tekniği geliştirilmiştir.Bu yöntem aynı zamanda uzamsal alanda da çalışır ve eşik limitine dayalı pikselleri kategorize eder.Kendi kategorisine dayalı olarak en fazla dört ve en az bir bit piksele gömülüdür. THBS'deki önem güvenlik ve taşıma kapasitesi üzerinedir çünkü veri yerleştirme(gömülmesi) için bitları(bits) becerikli bir şekilde kullanır. ETHBS, en uygun güvenliğin sağlanması ile birlikte alagoritmanın verimli yürütülmesine olanak sağlar. Son bölümde , blokların matrix determinantına dayanan 1LSB görüntü steganografisi tekniği geliştirilmiştir. Görüntü piksellerinde minimum değişiklik yaparak verilerin gömülü olduğu bir tekniktir. Bu teknik, 2x2 blokluk görsel piksellerinin matrix determinantı üzerinde çalışan 1LSB yerine koyma tekniğidir. Bu teknik yüksek PSNR ve stego görselinde iyi kaliteyi garantiler.

ACKNOWLEDGEMENTS

All praise to Almighty Allah that enabled me to reach this milestone. I would like to express my sincere gratitude to my advisor Assoc. Prof. Dr. Tamer Dağ for his continuous support, encouragement and guidance during my research work. Besides my advisor, I am obliged to Assoc. Prof Dr. Songül Albayrak and Assist Prof Dr. Taner Arsan for their help and valuable feedback during my research. I would like to show my gratitude to Dr. Zeki Bozkus who guided me through my Ph.D. I would like to thank all the department members of Computer Engineering, especially our head of department Prof Dr. Feza Kerestecioğlu for his timely encouragement and support during my Ph.D.

My deepest gratitude to my parents, wife, siblings and friends for their prayers and support throughout the work and who were always there when I need them.

Dedicated to my beloved parents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

LSB	Least Significant Bit
MSB	Most Significant Bit
PNG	Portable Network Graphics
JPEG	Joint Photographic Experts Group
GIF	Graphics Interchange Format
BMP	Bitmap Image File
MSE	Mean Square Error
RMSE	Root Mean Square Error
UIQI	Universal Image Quality Index
SNR	Signal To Noise Ratio
MAE	Mean Absolute Errors
PSNR	Peak Signal to Noise Ratio
AD	Average Difference
MD	Maximum Difference
NAE	Normalized Absolute Error
PVD	Pixel Value Differencing
DFT	Discrete Fourier Transform
HVS	Human Visual System
SSIS	Spread Spectrum Image Steganography
DCT	Discrete Cosine Transform
DWT	Discrete Wavelet Transform
IWT	Integer Wavelet Transform

INTRODUCTION

1.1 Overview

Steganography is the art and science of hiding secret information inside cover of another medium. The main objective of digital steganography is to send secret data over the internet without being noticed by any intruder during transmission [1]. Along with extensive usage of internet, privacy for data communication has become more vulnerable. Demand for secure information transfer over internet has become need of the hour. Lack of trust about security has led internet clients and companies to look for alternatives in the form of advanced steganography techniques besides customary cryptography methods. Images, text, audio and video files are most common files that are transferred across the network. Redundancy in digital files gives opportunity to conceal the data inside a cover medium. Images are the most commonly used cover types for steganographic purposes. Least significant bits (LSB) steganography is most frequently used mechanism for image steganography. LSB image steganography utilizes the pliability of LSBs replacement without effecting visual quality, providing reasonable data hiding capacity and optimal level of imperceptibility [2].

In LSB image steganography, there are two different sub methods known as non-adaptive and adaptive. The non-adaptive method conceals data without caring for the image content, while adaptive method conceals data by taking content of image into account and hides data in those parts which are less vulnerable to be noticed. Firstly mentioned method has more data hiding capacity whereas second is less susceptible to be detected. Different perspectives are taken into account about the significance of a steganography technique. Imperceptibility, security and data hiding capacity are considered as the most important features of a steganographic technique [3, 4]. Steganalysis is the use of statistical methods to detect the presence of secret information inside cover medium. The failure of a steganographic algorithm is its weakness to be detected rather that secret message is unleashed, because main purpose of steganography is to avoid the possibility to be notified about the availability of secret data.

This thesis provides novel techniques for LSB image steganography from three perspectives; first technique is developed for increasing capacity of data hiding, second complex and efficient steganographic algorithm development; and finally provisioning of a secure steganography mechanism.

1.2 Motivations

The use of image steganography has increased drastically in last decade. Due to redundancy in image pixels, images are most frequently and promising steganographic covers. Key challenge for LSB steganography is its undetectability. Along with new techniques being proposed for steganography, novel steganalysis methods are also foreseen which empowers the possibility of detection of secret data. Steganalysis techniques are able to detect location of secret data inside cover and even extract or estimate data which is concealed. Using the available data embedding capacity of LSBs along with being unnotified to steganalysis techniques makes steganography a challenging area of research.

The use of a pixel’s multiple bits for data embedding can increase capacity but also makes imperceptibility a tough task along with managing complex changes in pixel’s bits. Also, use of MSB along with LSB is a challenging task as a small change in the value of MSBs can affect image quality drastically. There is always a need to develop new steganography techniques or improve existing techniques so that better data security can be achieved.

1.3 Research Aim

This research work aims to provide LSB image steganography techniques that guarantee high data hiding capacity and upright level of security. For this purpose firstly, the consideration was taken into account for the use of MSBs along with LSBs to ensure novel and secure image steganography. Consequently, a new steganography technique is proposed where most significant pixel bits are used in an intelligent manner by using pair matching and data is embedded in LSBs. Similarity bits pair matching ensures high level of data hiding in a manner where data is implanted in such a way that it is non-detectable by an unauthorized user.

Human visual system (HVS) has a weakness that it cannot detect changes in low level colors. The aim of this research work is to use weakness of HVS to hide more data in low level colors. To develop a technique in which image pixels are categorized according to their values based on predefined threshold limits. Data bits to be embedded vary pixel to pixel depending upon the category in which a specific pixel falls. A single pixel can store one to four bits according to its color intensity level. This technique ensure high security and good payload but pixel by pixel encoding process makes it computationally complex. Efficient execution of a steganographic algorithm is equally as important as that of its imperceptibility and undetectability. This research work also aims to remove pipelining hazard from THBS, allow achieving same level of algorithm complexity along with reducing steganography execution time. The third aim of this research is to develop a new 1LSB steganography technique which ensures maximal data security; complexity of algorithm should certify the maximum undetectability and security of hidden data.

1.4 Contributions

The contribution of this thesis can be divided into three extents depending upon proposed techniques application. Firstly, a novel image steganography technique based on similarity of bits pairs is proposed. This technique is a novel technique and is different from all the previous techniques because it focuses on hiding data inside image pixels based on similarity of bits pairs and uses MSBs for this purpose. In this technique, data bits of the message to be secured are arranged in pairs and image pixel bits are also arranged pairwise. Data bits to be secured are compared with pixels bit pairs and based on respective similarity embedding is done. If there is no similarity between the data bits pair and pixel pairs, secret data pair is embedded into the 0th pair and pair number is saved in 2LSB. The technique is implemented in Matlab and when compared with other existing techniques showed that the proposed technique is secure and it hides higher payload of secret information with good quality of stego images and significant peak signal to noise ratio.

Secondly, threshold based image steganography (THBS) is developed, a technique that has four levels of data hiding in pixel and data bits concealment vary from pixel to pixel. This allows usage of HVS weakness where more data is concealed in low color levels. So, the higher the value of a pixel less data can be concealed and lower the value of pixel more data is concealed. This technique ensures complexity, suitable payload and data security. The prominence in existing threshold-based technique is on security and payload, but pixel by pixel encoding or decoding process makes the steganography process computationally complex and slow. Pixel by pixel encoding and decoding process makes the steganography process computationally complex. THBS technique despite its usefulness was unable to fully utilize the parallel architecture and was reluctant to execute efficiently. ETHBS diminishes the serial dependency of THBS and allows the execution of process on parallel machines in proficient manner. The results exhibit that ETHBS ensures adequate security level and payload along with exploiting efficient methodology for its execution.

Lastly, in this dissertation, a technique based on blocks matrix determinant is developed for LSB image steganography. In this technique image is divided into 2x2 blocks; and from each pixel four least significant bits are selected, these target bits are converted into decimal values to find the determinant of these values for every block. If the data bit which we are hiding is 0 and the determinant of the block is even or if the data bit is 1 and the determinant of the block is odd then no change occur. But if the data bit is 0 and determinant of block is odd then change the 1st bit of block pixels to make determinant of block even. Similarly, if data bit is 1 and determinant of block is even then change 1 bit of pixel to get determinant of the block odd. Proposed methods when compared with other techniques have shown that this technique is more secure and reliable.

1.5 Thesis Organization

This thesis is arranged in chapters as described below:

- Chapter 1 gives overview of research work, research aim, motivation and overall contributions in the area of image steganography.

- Chapter 2 introduces information security mechanism. It briefly elaborates techniques used for data hiding. Mainly techniques of cryptography, watermarking and steganography are explained. Furthermore steganography and its types are illustrated in detail.
- Chapter 3 gives details about existing LSB image steganography techniques, their benefits and limitations. It also explains that how image steganography techniques are analyzed. Number of image performance metrics that are used for evaluation of quality of image is also in this chapter. Different classical LSB steganography methods are analyzed and compared to illustrate the inevitability for the development of new LSB steganography techniques, which also results in commencement of research questions for this dissertation.
- Chapter 4 explains the firstly developed technique for LSB Image Steganography using similarity of bits pairs (SOBP). It explains the encoding and decoding mechanism that how pixel and data bit pairs matching is used to hide secret data. Furthermore this technique is explained with encoding examples. In the end stego image quality is compared and analyzed the results are compared with other image steganography techniques.
- Chapter 5 elaborates a novel threshold based image steganography (THBS). It explains the encoding and decoding process. It shows how data embedding and extraction is done to and from each pixel. It elaborates how one to four data bits are hidden inside image pixels. Comparison of results is done between THBS and other LSB image steganography techniques.

  It further identifies the limitation in the execution process of THBS and elaborates the relation between parallelism and image steganography mechanism. It explains how parallel processing can help in fast execution of steganography followed by how efficient threshold based image steganography (ETHBS) removes pipelining hazard and allows parallel execution of THBS. The implementation details and comparison was done in the end of this chapter to show the performance improvement of proposed technique.
- Chapter 6 explains a new developed technique mainly focusing on the improvement of security in LSB image steganography based on blocks matrix

determinant method (MDS). It also explains with the help of examples and compares the results based on key image performance metrics.

- Chapter 7 gives the comparative evaluation of the developed techniques based on their data hiding capacity, imperceptibility and security.
- Chapter 8 gives the concluding remarks. It defines the benefits of the developed techniques, determines the thesis contributions in the area of LSB image steganography and unleashes the future directions of this dissertation.

2. INFORMATION SECURITY

This chapter gives a brief overview of the three techniques that deal with information security: cryptography, steganography and water marking. Since the area of concern in this dissertation is steganography, it is discussed in detail.

The security of data and its communication across the network is a leading challenge in this modern time of internet and digital communication. Information security is a mechanism to protect information from any unauthorized access. Information hiding refers to hiding secret message in a digital medium. Information hiding techniques can be divided into three categories Cryptography, Watermarking and Steganography as shown in Figure 2.1.

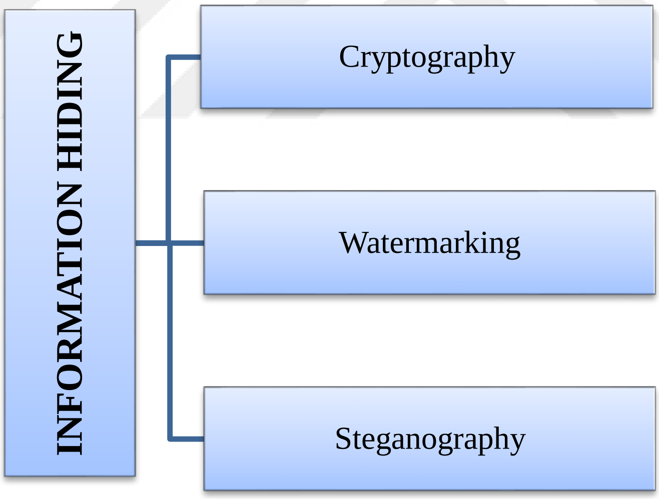


Figure 2.1 Information Hiding Techniques

2.1 Cryptography

Cryptography is a field of computer science and mathematics in which message is encrypted with the help of encryption key to make message immune and secure. Cryptography is one of the techniques used since ancient times to secure secret information from unauthorized access [1, 2]. Cryptographic techniques protect the contents of secret information by converting them into unreadable and non-understandable format during transmission. The process of cryptography is shown in Figure 2.2.

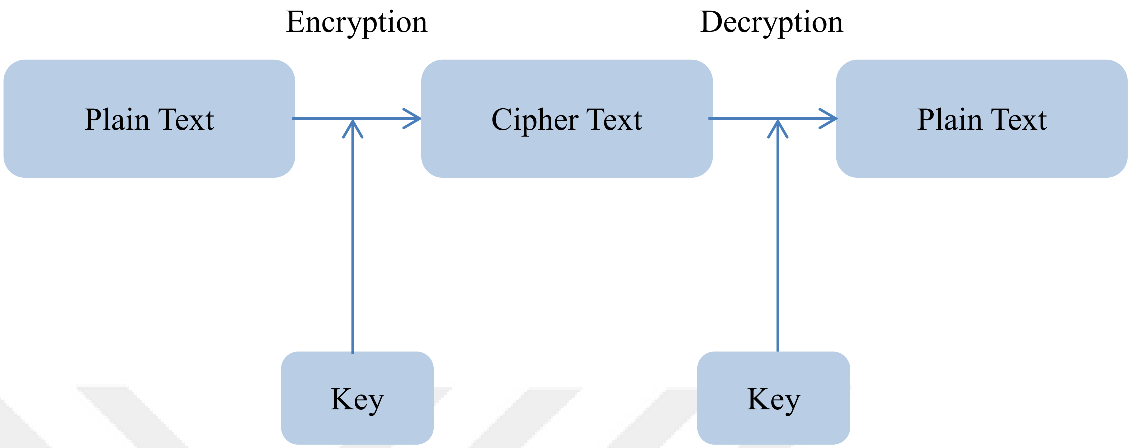


Figure 2.2 Cryptography Model

Plain text is encrypted with the help of a defined key. The message after the encryption is called cipher text. The users across the network can observe the communication of cipher text but unavailability of the key does not allow them to decrypt message. Decryption is a reverse process in which cipher text is converted back to original or plain text using decryption key. Encryption/ decryption keys can be same under symmetric/private key cryptography or different under asymmetric/public key cryptography. There exist different types of cryptography techniques; few of the main are substitution, transposition and RSA. Substitution ciphers perform cryptography mechanism by substituting text letters with other letters. A simple example can be of replacing an alphabet in plain text with its immediate next letter. Transposition ciphers transpose letters or characters of secret message to generate encoded data called as cipher text. Whereas, RSA was developed in 1977 by Ronal Rivest, Adi Shamir and Len Adelman. First letters from their names were chosen to name this public key cryptography mechanism. Data communication across the network is visible but due to encryption data cannot be extracted from the cipher without decryption key. In terms of payload cryptography supports high payload as compared to steganography and watermarking. Cryptography is widely used mechanism used for secure data communication purposes.

2.2 Water Marking

Watermarking is a technique of embedding intellectual property rights details to text, images, audio, video or other multimedia data. The purpose of watermarking is provisioning data authenticity and integrity [1]. In digital watermarking, hidden message may or may not have relation with carrier signal but it provides authenticity. Watermarking can be noticeable where the ownership of data is observable as logos or texts or invisible where watermarking is not perceivable and is hidden inside data. As the primary purpose of watermarking is to keep propriety, ownership information so payload of watermarking techniques is less as compared to cryptography and steganography. Watermarking techniques are mainly categorized into spatial and frequency domain techniques. Comparison between three information hiding techniques is shown in Table 2.1.

Table 2.1 Comparison between Data Hiding Techniques

	Steganography	**Cryptography**	**Watermarking**
Techniques	Spatial domain, transform domain	Transposition, RSA, Substitution	Spatial domain, frequency domain
Visibility	No as data is hidden within cover medium	Yes but data is encrypted	Depends on technique used
Capacity	Differs along with steganography method variation	Capacity of data hiding is high but high payload can reduce chances of data decryption	Usually low
Detection	Not easy to detect as it is hard depends on steganographic method used	Not easy to detect depends upon method used for encoding	Not easy to detect
Strength	Hiding message without altering message	Hide message by altering message through assigned key	Extend information and become attribute of cover medium
Imperceptibility	High	High	High
Applicability	Universal	Universal	Universal

2.3 Steganography

Steganography hides the secret message inside a cover of another medium. The word steganography is a collection of two words “steganos” and “grafia”. “Steganos” means cover and “grafia” means writing [3]. In other words steganography can be defined as the art and science of covered writing. In the case of image steganography, images are used to hide secret information without affecting the visual quality of the image. Steganography is the name of the technique that hides one type of information inside cover of other information [4]. Therefore, protecting the content of message is not enough, hiding the existence of secret messages is also important for information protection. The techniques used for hiding the existence of secret information inside the cover of another medium are called steganographic techniques.

In general steganography process model, secret message and cover are fed to encoder where steganography techniques are applied. After encoding stego object which is in the form of normal cover is sent through communication channel to the receiver. During the communication presence of secret message is hidden under cover. On the receiving end, stego system decoder uses steganography technique’s decoding to unleash secret message from stego object. The process is shown in Figure 2.3.

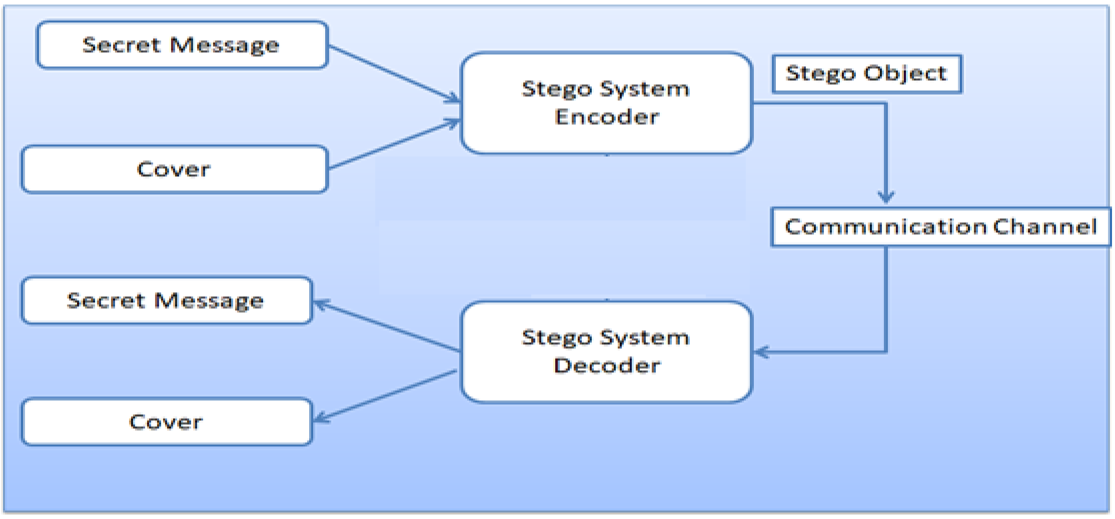


Figure 2.3 Steganography Process Model

The first usage of steganography dates back to the Greeks when wooden tablets were used for writing the secret information and then hide the writing by covering it with the wax. Pirates used the idea of tattooing secret information on the body parts and also used shaved heads such that when the hair would grow it would conceal the existence of secret messages. The Germans used a unique technique during the Second World War for sending sensitive information through the insecure communication channels. They used a unique way of using microdots to create high quality images and information was hidden using those microdots, at that time it was almost impossible for the intruders to decrypt that information. The use of steganography is increased today and is used for provisioning of security in various fields and applications. One of the major uses of steganography is in copyright enforcement so as to protect the use of copyright data. Another principal application of steganography is in the field of medical, where to control the amalgamation of patient's data; there key information is stored in their medical images from where the required information can be retrieved when required [6].

Simmons [7] proposed the modern formulation steganography in terms of the prisoner's problem. Two prisoners wish to hatch an escape plan by communicating secretly. The secret communication between them would be carried through a warden who can either be passive or active. Passive warden would examine the communication and would try to deduce if there is any secret information in normal messages between inmates. If the warden suspects any covert communication he would inform the concerned authority about the secret communication. Active warden would alter their communication in order to remove the secret information from their communication. Steganographic techniques are different from cryptographic techniques, the former hides the very existence of information where later protects the content of information by converting it to unreadable form [8].

Images are one of the most frequently transferred data types on internet; this is why images are used most commonly for steganography purpose. A large number of techniques have been proposed for image steganography [9]. All the available techniques have their strong and weak aspects specifically in terms of security and payload. For instance, some applications may require high security of information and can compensate on the amount of information to be hidden and vice versa. The efficiency and complexity of steganography mechanism also effects the selection of steganography technique.

2.3.1 Steganography Protocols

One general principle used for explanation of steganographic process is where Alice wants to send secret message *m* to Bob and chooses cover *c* for hiding secret message inside it through using stego key *k.* So, *c* is modified to stego object *s* with the help of *k*. This should be done in a careful manner so that any third party aware of only transfer of cover remains unaware of secret data transfer. There are three types of protocols in literature for steganography; pure key steganography, private key steganography and public key steganography [9]. These three types of steganography protocols are explained as following.

2.3.1.1 Pure Steganography

In pure steganography there is no prior exchange of secret key between sending and receiving parties. Generally describing the process, when C is cover image and M is message and encryption process is represented as $E: C \times M \rightarrow S$, resulting into S stego image. Whereas in decoding process $D: S \rightarrow M$, where secret information is extracted from stego image. It is obvious that size of cover is always greater than secret message. The sending and receiving parties only have information of encoding and decoding algorithm and these algorithms are not known outside the communication parties.

2.3.1.2 Secret Key Steganography

In pure steganography, the security of the messages depends on its system’s secrecy, where no information other than of encoding and decoding algorithm is required for starting communication process. In secret key steganography similar mechanism as that of symmetric cipher is followed. Here the security of steganographic system relies on stego-key sharing between sending and receiving parties. Sharing of this secret key is important where no one can extract data from stego cover without having prior knowledge of stego-key. So, if the stego key is known by receiver it can use reverse process to decrypt or uncover the secret message from received stego cover. Any receiving party without having knowledge of secret key cannot extract secret message from the received stego-object depict that sharing of stego-key is important in secret key steganography.

2.3.1.3 Public Key Steganography

Like pure steganography, public key steganography do not depend on sharing of secret key between communication parties. In this type of steganography two types of keys are used like public key cryptography. These keys include a public key and a private key, where public key is stored in public database. Public key is used for hiding secret message whereas private key is used for extraction of secret data from the cover object. This type of steganography utilizes the fact that decoding function can be applied to any cover object irrespective of whether it contains data or not. This makes it undistinguishable across the channel whether the cover objects are with or without secret data. Another perspective using public key steganography is from cipher text where secret data is encoded into cipher text and on cipher text application of public key encryption makes it more secure.

2.4 Types of Steganography

Steganography can be classified into various types based on the type of the cover medium which carries the secret information. These mediums usually include image, video, audio, text and protocol [3]. Information medium that have more redundancy are considered as more appropriate for the usage as steganography medium. Redundancy varies in terms of the medium and can be bits representing the pixels or samples of audio or video file. Redundant bits can be altered without affecting the file extensively to utilize them for steganography purposes. File formats such as image, audio, and video have a high degree of redundancy that is why they are used ordinarily for steganography process. Figure 2.4 shows common formats that are currently used for steganography.

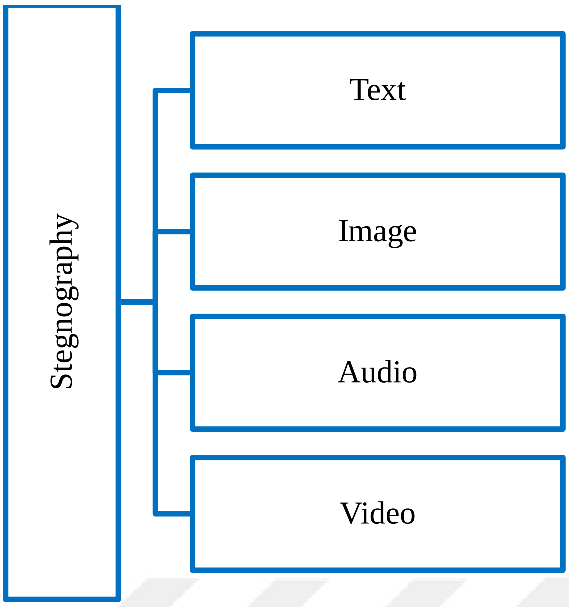


Figure 2.4 Types of Steganography

2.4.1 Text Steganography

Steganography in text just concentrates on modifying some of text attributes. This can either be done by changing attributes of text or text formatting. The list below discusses some ways to implement the text steganography [9].

Hiding data in normal text is done in many ways. One way is to add tabs and white spaces to the end of the lines of the documents. Another method was effectively used in parts of training where even if content has been printed and copied on paper for ten times, the secret message could still be recovered [10-11].

Another way of putting a secret message in a text is by using an openly accessible cover source, a book or a newspaper, or using a code which comprises for instance of a combination of a page number, a line number and a character number. By this way no data saved inside the cover source prompts to a hidden message.

One of the mainly used steganographic approach is setting backgrounds color and font color. This technique is engrossed for Microsoft word document. Pick predefined hues and set textual style and background hues of unseen characters such as space, tabs or the carriage return characters. Red, green and blue values are of 8 bits means we have enabled scope of 0 to 255. This approach needs no additional data to hide required bits [12].

2.4.2 Audio Steganography

Audio steganography is a technique used to hide secret message inside audio files. This is on account of the human auditory system (HAS) that has the capability for listening a wide range of audio signals. The main short coming of HAS befalls when it tries to distinguish the sound subjugated to encode secret messages without being recognized [13, 14]. Methods mostly used in steganography for audio rely on LSB coding and Spread Spectrum.

Least significant bit steganography uses LSBs to hide information in sound, as changes would not make obvious changes to the sounds [15]. The parity coding technique separates a signal down into distinct areas of samples and encodes each piece from the secret message in a sample area's parity bit. Phase coding takes care of the shortcoming of the noise prompting strategies for audio steganography. Phase coding utilizes the way that the phases of sound are not as capable of being heard to the human ear as commotion seems to be. This procedure encodes the message bits as phase shifts in the phase range of a computerized signal, accomplishing an unclear encoding in terms of signals to detect noise ratio [16]. The basic spread spectrum (SS) technique endeavors to spread hidden data over the audio signal frequency spectrum as much as possible. [17].

2.4.3 Video Steganography

In video steganography, a video file would be entrenched with supplementary information to hide secret messages. A stego video signal is a combination of hidden message information and cover video file. Intermediate signal is joined with the content information to get encoding. The accompanying information is incorporated as duplicate control information, which is taken from the electronic devices of the consumer and is used to disable copying [18].

The transitional signal also carries pseudo arbitrary key information in order to hide encoding and decode which requires equivalent key to extract hidden data from encoded content. In few executions regulation information is implanted in the content signal with auxiliary data. This encoding is strongly against scaling resampling and other types of degradation; the goal is that the supplementary information can be recognized from the substance which may have been corrupted [19]. Few of the most widely known approaches of video steganography are listed and discussed below.

A. Least Significant Bit Insertion

Least Significant Bit Insertion is the most basic and known approach for almost all types of steganography. In LSB video steganography digital video file is taken as distinct frames and changes the picture of every video outline. LSBs of images are used to store secret data. The technique upgrades the capacity of the hidden message yet may compromise security requirements for example data integrity [20, 21].

B. Real Time Video Steganography

The steganography of this type includes hiding data on the run time output video on the device. The technique considers every frame shown at any point regardless of what is revealed in message, after that image is distributed into blocks. If pixel shades of the blocks are similar at that point, it changes color characteristics of number of these pixels to some extent. It is easy to identify missing parts of information by labeling each frame with a sequence number. First the displayed frame should be recorded and then use the relevant program to extract the information from the video file [22].

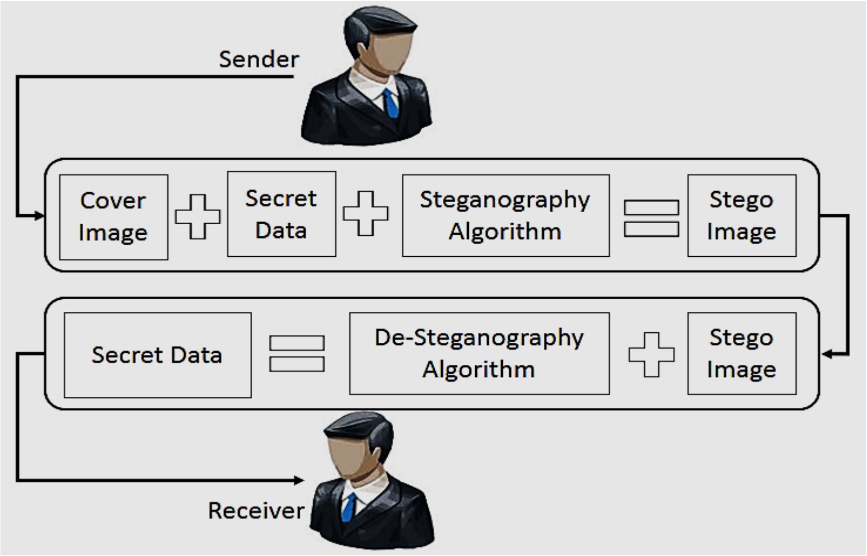


Figure 2.5 Image Steganography Process Block Diagram

2.4.4 Image Steganography

Images are the most favorable medium used as cover files for steganography. Data is embedded in image by exploiting the weakness of Human Visualization System (HVS). The general process of image steganography is shown in Figure 2.5. On the sender side, secret data is concealed in image using any steganography technique. The image having secret information is now called stego image. Stego image is sent through communication channel to the destination. On the receiver side, extraction of secret data is done depending on the decoding method defined by the steganography technique. There are different types of image formats and for each file format there exists some image steganographic techniques.

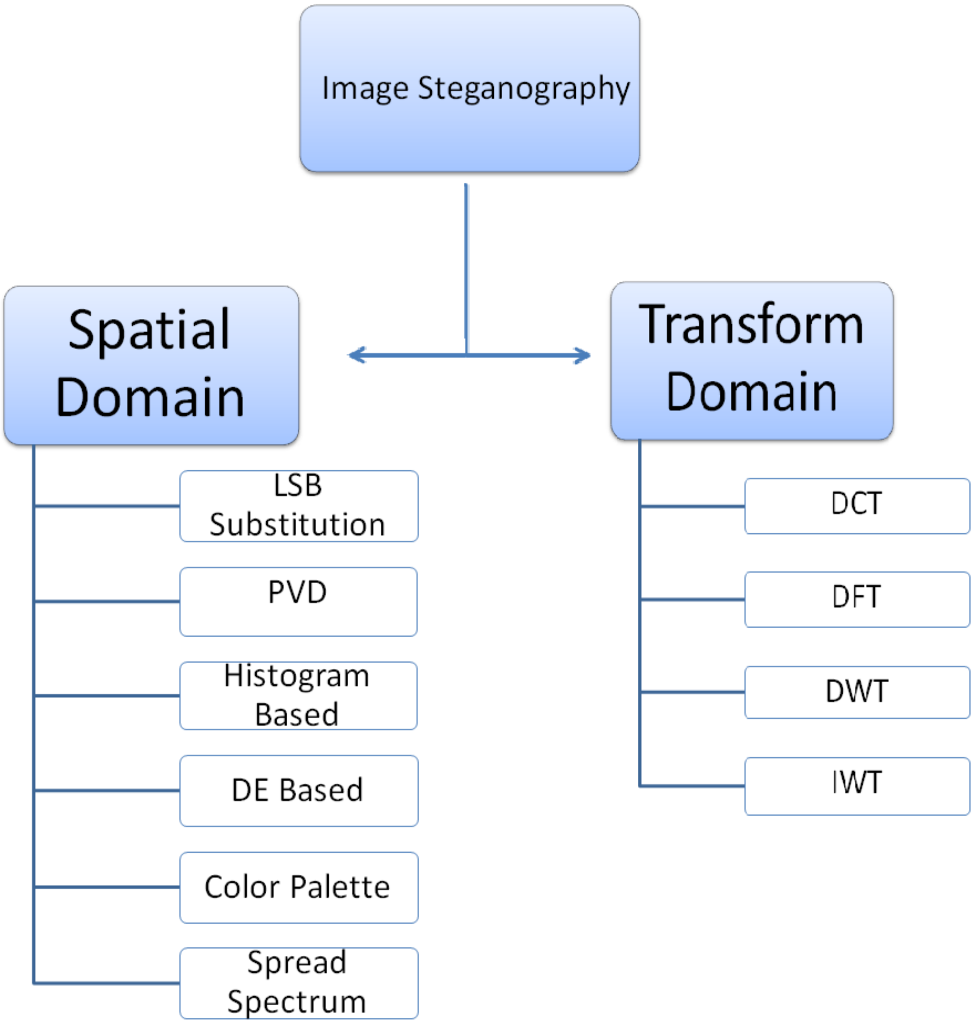


Figure 2.6 Image Steganography Domains

There are two main domains for image steganography i.e. transform domain and image domain. Transform domain image steganography is used for data hiding inside frequency domain. Images are first transformed and then data hiding process is performed. Whereas in image domain also known as spatial domain data is embedded in intensity of pixels directly. The key types of image steganography are shown in Figure 2.6.

2.4.4.1 Steganography in Spatial Domain

Image domain also known as spatial domain techniques embed information in the intensity or colors of image pixels. In image domain technique bitwise insertion of secret information is done in the image pixels. Lossless image formats are considered most appropriate for hiding information in image domain steganography; whereas embedding techniques vary along with deviation in image formats being used [23]. In this domain of image steganography modification is done in pixel values directly without doing any transformations into frequency domain. This domain is simple in terms of data embedding where data is embedded without modification of complete cover. All spatial domain steganographic techniques are applicable to lossless media types. Lossless compression is a class of data compression algorithms that allows the original data to be perfectly reconstructed from the compressed data. By contrast, lossy compression permits reconstruction only of an approximation of the original data. In the case of images widely used lossless image compressed formats are png, bmp, jpeg2000 and GIF etc. In this dissertation for experiments png images are used.

A. LSB Substitution Method

In spatial domain based steganography there are various types; but basically, all relies on modifying bits in image pixels for hiding data. Most noteworthy steganography technique used today is the least significant bit steganography (LSB) that modifies and embeds data in least significant bits of pixels. The key point for LSB steganography is that changes made in pixels for hiding data are not visible to human eye and modifications do not lead to noticeable image distortion. Various steganography techniques are being used for securing data in images [24]. LSB is the most broadly used technique to hide the data.

B. Pixel Value Differencing (PVD) Method

PVD has high data embedding capacity along with ensuring good quality of stego image. PVD is based on consideration of values difference between pixels. It distinguishes smooth and edge areas of image. In edges difference between pixel values is greater so more data bits are embedded without effecting quality of image. In smooth areas pixel differences is less so less data bits are concealed in those areas as they are more sensitive to change [25].

C. Histogram Based Method

In this method data is embedded using histogram where pixel locations are generated through histogram shifting. Mostly stegnalysis techniques based on histogram analysis are used to find modifications in images. This method preserves changes in histogram while doing modification in pixel bits, so that no or minimal changes could be noticed if stegnalysis of image is performed [26].

D. Difference Expansion (DE) Method

Liu et al. [27] proposed DE method with simple location map and bilinear interpolation. This results in finding appropriate locations and improvement of stego image quality. Difference expansion methods used in steganography has limitation of limited payload and limited locations for data concealment. Limitation of this type of techniques is incorrect pixel production during encoding of data.

E. Color Palette Based Method

Steganography method for palette based images was first proposed by Fridrich [28]. These images include Graphics interchange format (GIF). Based on color palette single bit is embedded. Pseudorandom number selection is considered as key in which random scan is used to choose pixels for data embedding and color matching helps to hide data. Different techniques have been proposed in this domain. This type of steganography has good stego image quality and produce less distortion.

2.4.4.2 Steganography in Transform Domain

Transform domain techniques make use of frequency components of an image. Image is first converted into frequency domain and then message is embedded [30].Transform domain image steganography techniques are based on manipulation of orthogonal transform of image. The transform has two components; magnitude and phase. Magnitude comprises of frequency content whereas phase is used for conversion to spatial domain. These techniques embed information bits in specific areas of images which are more robust and independent of image file formats. There exist few main image steganography techniques in spatial domain which include image steganography with Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) and Integer Wavelet Transform (IWT).

2.5 Evaluation of Steganographic Techniques

All the available steganography algorithms for steganography have their own beneficial features and limitations, it is very important to select the most appropriate algorithm in accordance with requirements. To select the appropriate steganographic algorithm for a specific application, it has to meet some key necessities [31]. The most important requirement for an algorithm is its imperceptibility. The higher the imperceptibility of a steganographic algorithm the better it is. There are many key requirements which a steganographic algorithm meets [32-34]. Few main characteristics of them are given below.

1.5.1 Payload Capacity

The first requirement of a steganographic algorithm is that it should embed data of desired capacity into image unlike watermarking which hides small amount of copyright data [35]. The payload represents the total amount of data embedded inside stego image in bits. It is calculated by using following Equation 2.1.

$$Payload = \sum_{j=1}^{M} \sum_{k=1}^{N} Bits\ (x'_{j,}k) \qquad (2.1)$$

Where j, k represents pixels location and $x'_{j,}k$ gives amount of data hidden in stego image.

1.5.2 Imperceptibility

Data hiding capacity and imperceptibility are inversely proportional to each other. The higher the fidelity of a stego image, the better is the imperceptibility and imperceptibility is contented when there is minimal difference between original and stego image [34]. Imperceptibility of a steganographic method is determined based on medium quality metrics which mainly include PSNR, MSE and UIQI. For the developed techniques, all the key parameters for image steganography are tested to certify the suitability of the proposed method. These parameters are discussed with detail in section 3.2.

1.5.3 Security

Security of a steganographic technique is evaluated through statistical and visual attacks [36]. Stegnalysis is mainly about discovering or even identifying the concealed data. Steganographic systems generally leave detectable traces on images. This is because often the steganographic process alters media properties and introduces degradations or abnormal characteristics that can be used by stegnalytic techniques as steganographic signatures for detection. Although these cannot be detected by the human eye due to careful application of steganographic mechanisms, the signatures left can be electronically discovered. These signatures can also be even used to identify the steganographic tools and techniques; thereby aiding the investigator in the retrieval of the hidden information. Spatial domain techniques embed data directly into image domain; these techniques are not robust as transform domain techniques which transform image into frequency domain against statistical attacks. Most widely used steganographic technique is to make use of the LSB data, because of the vulnerability that LSBs appear random to observer although it may contain hidden patterns.

It should be noticed that stegnalysis algorithms in essence are called successful if they can detect the presence of a message [36]. The message itself does not have to be decoded. Indeed, the latter can be very hard if the message is encrypted using strong steganography technique. However, recently there have been methods in the literature like RS Stegnalysis which in addition to detecting the presence of a message are also able to estimate the size of the embedded message with great accuracy. The most frequently used stegnalysis techniques for spatial domain include visual stegnalysis, histogram analysis, RS analysis, sample pair method and least square methods [37-39].

1.5.3.1 Histogram Analysis

It is considered a statistical attack since the histogram of an image shows a graph of the number of pixels at each different intensity value found in that image. This attack allows distinguishing the difference between the cover and stego images, if there is a message embedded in channels. For a 24-bit color image, 256 different intensities for each of the 3 channels (red, green, blue) are possible. Therefore, a histogram for each channel can be drawn separately, or an average histogram of all channels can be produced to analyze the difference between histogram of original image and histogram of stego image to determine the effect of data concealment by a steganographic technique [37].

1.5.3.2 RS Stegnalysis

RS stegnalysis is a reliable and accurate method proposed by Fridrich et al. [38, 39] for detecting and estimating LSB data embedding in color and gray scale images. The secret message length is derived by inspecting the loss less capacity, capacity for lossless data embedding in LSBs. It is statistical method which divides image into small groups. Each group can be classified as regular (R) or singular (S) depending upon smoothness. The pixel values of a group are compared with partially altered version of the same group. The percentage of regular or singular groups are calculated to determine positive mask or negative mask and then LSBs in the group are flipped i.e 0 to 1 or 1 to 0. Again the percentage of R and S is calculated. Randomizing the LSBs (for example by LSB replacement) decreases the lossless embedding capacity in the LSB plane. Thus Fridrich et al. uses the lossless capacity as a sensitive measure of embedded bits in the LSB plane. Even though the LSB plane appears random it is related to the other bit planes. This relationship is nonlinear and the lossless capacity is used to measure the relationship fairly well. In normal cover image, the numbers of regular groups is greater than that of singular groups. RS analysis utilizes the fact where after the concealment of data in image R and S group of pixels have tendency of becoming equal.

2.6 Significance of Steganography

Along with devastating increase of data communication new security technologies became need of hour as data become more susceptible to malicious contents and intruding parties. Traditionally, concerns of security have been dealt with cryptographic mechanisms over the internet, where messages are encrypted using stego keys and after the approval of authenticity of the receiver messages are decrypted by using authentication code.

Cryptography now has becomes a mature information security area after rigorous development over the last decade. However in cryptography, encrypted messages are more vulnerable for attackers as the messages are exposed to the attackers which easily recognize that encrypted messages having secret information are directed across the network. Steganography arises where the need for concealment of data is the main aim so that across the network it is not known that secret information is being transferred. This makes it less susceptible to attacks. Steganography is the younger sister branch of cryptography which is in developmental stages and is used as alternative of cryptography. Steganography mechanisms hide secret information in a way that data transfer across the network seems to be normal data but inclusively containing secret information. Steganographic algorithms can be utilized by armed forces for confidential data exchange purposes. Another application of these algorithms is in medical image processing, which enable hospitals to keep patient medical information and details along with his/her medical images/signals, such as X-Rays, ECG, CT-Scan and Ultra Sound reports.

Steganography is currently being used by security agencies for ensuring the security of their important data. Steganography is also used in countries and organizations as alternative where data encryption is illegal or the regimes where encryption of data can lure complications for the communication parties [40]. Many governments have created or currently are creating laws to prohibit or limit the use of cryptographic systems. The primary purpose of such limitations is shown to be the law enforcement and to avoid unethical activities, but application of these laws makes the privacy and security of internet users much vulnerable and relatively weak. Civil liberties and social rights activists consider these limitations as inhuman where each individual has right of their information security and data privacy. So there are both rough and smooth sides of these limitations for the civil society and the law enforcement agencies. This is where steganography comes between them and secure data communication through concealment of data inside cover mediums and allows data transfer to remain unnoticed across the network; thus limits the information about secret data transfer between communicating parties.

3. LEAST SIGNIFICANT BIT IMAGE STEGANOGRAPHY

This chapter explains least significant bits (LSB) image steganography and different key techniques that exist in the literature for LSB image steganography. It also includes key image performance metrics based on which quality of stego image is compared with the original cover image and excellence of a steganography technique is measured.

3.1 Existing LSB Image Steganography Techniques

In spatial domain based steganography there are various types; but basically, all relies on the modification of bits in image pixels for hiding secret data. Most noteworthy steganography technique used today is the least significant bit steganography that modifies and embeds data in least significant bits of pixels. The key point for the LSB steganography is that changes made in pixels for hiding data are not visible to human eye and modifications do not lead to noticeable image distortion. Various steganography techniques are being used for securing the data in images [41]. LSB is the most broadly used technique to hide the data [42]. Following are the common LSB steganographic techniques available in the literature.

3.1.1 Difference Expansion LSB Technique

This technique explores redundancy in data and stores a single bit of a message in two pixels of cover image [43]. This technique embeds fewer information bits in cover image and also has a low visual quality of stego images as compared to other techniques.

3.1.2 Edge-Based LSB Techniques

Edge-Based LSB technique is another steganographic technique. In this particular technique, secret messages are stored in those areas of images like corner and edges where pixels values are different from neighbors [44]. Low embedding capacity is the only drawback of the LSB technique. Edge based LSB techniques have high embedding capacity with low security which makes it less important to use. Data stored in LSB technique usually depends upon differences of two-pixel values [45].

3.1.3 LSB in GIF Images

Graphics Interchange Format (GIF) images are drawing based images, where colors of image are stored in a color lookup table or alpha channel. In this technique, data is concealed in the LSB of GIF images, but this could result in the variation in colors of the image. The major drawbacks of this technique are low payload and openness to visual and statistical attacks [46].

3.1.4 Hiding behind the Corners

This technique hides the secret information by taking into account the focus areas of the original cover image and hides message bits in the less focused areas such as corners [47]. The main disadvantage of this technique is that its embedding capacity is very low.

3.1.5 Neighborhood Pixel Information Hiding Technique

In this technique, information is embedded inside pixels based on the neighboring pixels values information. There are three main neighborhood methods used for hiding information through this technique which include Diagonal neighborhood method, four and eight neighborhood methods. This technique is useful in terms of high payload and PSNR but if the stego image is changed slightly during transmission the extraction of information on the receiver side becomes impossible [48].

3.1.6 Image Interpolation LSB Technique

Image interpolation technique is used to hide data in the interpolation area [49]. This technique hides more bits in complex regions of the image than in smooth regions. A major limitation of this technique is low data security. Data hidden behind the file using this technique can easily be obtained [50].

3.1.7 Pixel Value Differencing and Modulus LSB Technique

The pixel value differencing and modulus LSB technique is used to hide fewer amounts of data in carrier images. This technique uses the difference in pixels and a modulus function to secure data by changing the remainder or modulus value [25].

3.1.8 An Adaptive LSB Spatial Domain Steganography

An adaptive least significant bit steganography is proposed in [51], where extra care has been taken about noise sensitive areas for hiding data and to ensure better stego image quality. It differentiates normal object space and edge areas. Image brightness, edges and texture masking are considered for calculating k-bits to hide data using LSB. At non-sensitive image areas, the value of k increases, whereas in sensitive areas its value is kept minimal to keep standard image quality. For obtaining better stego-image quality pixel adjustment is done using LSB substitution method. This method results in fair quality and high hidden capacity for the image. However, the experimental data set used for the implementation of this technique is limited to images which are not complex and do not have voluminous noisy edges.

3.1.9 Texture Based Image Steganography

Texture based image steganography is proposed in [52]. In this technique texture area is divided into groups, one group is of simple texture area, while other is of complex texture area. In this technique simple texture area is used for hiding 3 bits of secret information in red, 3 in green and 2 bits for blue channel using LSB. For complex texture area 4 least significant bits are used to hide data. So 2 to 4 LSB depending upon the texture classification is used in this technique. This technique provides high data hiding capacity.

3.2 Image Quality Assessment Metrics

Image steganography techniques are compared with number of existing techniques available in the literature based on some key image performance metrics for imperceptibilty. These parameters are suggested by [53, 54]. The existing techniques compare few parameters as PSNR, MSE and RMSE but for the eminence of research in this thesis we use nine parameters for presenting the proficiency of the proposed work.

3.2.1 MSE (Mean Square Error)

The MSE of a stego image calculates the mean value of the squares of the errors in pixels, this error occurs during the steganography process. MSE illustrates the difference between the original cover image and stego image [55]. The value of MSE is calculated by the Equation 3.1.

$$MSE = \frac{1}{NM}\sum_{j=1}^{M}\sum_{k=1}^{N}(x_{j,k} - x'_{j,k})^2 \qquad (3.1)$$

Where N is number of rows and M is number of columns which represent dimensions of image, j and k are image coordinates. $x'_{j,k}$ is generated stego image and $x_{j,k}$ is the original cover image.

3.2.2 RMSE (Root Mean Square Error)

The RMSE of a stego image is calculated by taking the root of mean square of all errors between the stego and the original image [56]. The value of RMSE is calculated by the given Equation 3.2.

$$RMSE = \sqrt{\frac{1}{NM}\sum_{j=1}^{M}\sum_{k=1}^{N}(x_{j,k} - x'_{j,k})^2} \qquad (3.2)$$

Where N is number of rows and M is number of columns which represent dimensions of image, j and k are image coordinates. $x'_{j,k}$ is generated stego image and $x_{j,k}$ is the original cover image.

3.2.3 Universal Image Quality Index (UIQI)

UIQI is used to determine the quality of a stego image (Y) with respect to original image (X). It was proposed by Wang [57]. It is calculated by taking the product of its three main components (luminance, contrast, and structural component) raised by an exponent, when required. Its value will be 1.0 if both the cover and stego images are indistinguishable. Generally, the UIQI between two images X and Y is defined as follows in (3.3):

$$UIQI(x,y) = [l(x,y)^{\alpha}.c(x,y)^{\beta}.s(x,y)^{\ell}] \qquad (3.3)$$

Herein, α, β, and γ are parameters that represent the comparative consequence of its three components. By setting α= β = γ = 1, we get the UIQI as mentioned in (3.4).

$$UIQI(x,y) = \frac{(2\mu_x\mu_y + C1)(2\sigma_{xy} + C2)}{(\mu^2{}_x + \mu^2{}_y + C1)\,(\sigma^2{}_x + \sigma^2{}_y + C2)} \qquad (3.4)$$

Where x and y are corresponding windows inside the original image.

$\mu_x = \sum_{i=1}^{N} x_i$ is the mean of signal x.

$\sigma_x = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \mu_x)^2}{N-1}}$ is the standard deviation of signal x.

$\sigma_{xy} = \frac{\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)}{N-1}$ is the covariance of two signals x and y.

C_1,C_2 are two constants taken for stability purposes in the case where either term of the denominator is close to zero.

3.2.4 SNR (Signal To Noise Ratio)

SNR describes the level of noise that is added in the cover image after applying the steganography process by comparing it with the original image [58]. It can also be defined as ratio of mean value of the signal and standard devition of noise. It is calculated in Equation 3.5.

$$SNR = 10\ Log_{10} \sqrt{\frac{\sum_{j=1}^{M}\sum_{k=1}^{N}(x_{j,k})^2}{\sum_{j=1}^{M}\sum_{k=1}^{N}(x_{j,k} - x'_{j,k})^2}} \qquad (3.5)$$

Where j,k represents pixels location, N is row and M is column, $x'_{j,k}$ is generated stego image and x_{jk} is the original cover image.

3.2.5 MAE (Mean Absolute Errors)

MAE calculates the difference between original and stego values. It is one of the important parameter for information hiding testing [56]. It can be calculated by Equation 3.6 as shown below:

$$MAE = \frac{1}{MN}\sum_{j=1}^{M}\sum_{k=1}^{N}\left|x_{j,k} - x'_{j,k}\right| \qquad (3.6)$$

3.2.6 PSNR (Peak Signal to Noise Ratio)

The most common method used to evaluate the image quality by an objective method is the PSNR in which the ratio between the original and stego images is measured [59].

PSNR is very important parameter involved in the image processing and is used as one of the standards for the evaluation of different sorts of image quality evaluation methods. It is usually calculated in decibels (dB). The higher value of PSNR shows that the stego image is more contiguous to the original image. The value of PSNR is calculated by using the following formula.

$$PSNR = 10\ Log_{10} \frac{C^2 max}{MSE} \tag{3.7}$$

Where MSE denotes mean squared error and C_{max} holds maximum fluctuation in the image data file.

3.2.7 AD (Average Difference)

AD of two images is defined as the sum of absolute difference between original and stego image by dividing the number of pixels [60]. The value of AD is calculated by using the Equation 3.8.

$$AD = \sum_{j=1}^{M} \sum_{k=1}^{N} \left(\left| x_{j,k} - x'_{j,k} \right| \right) / MN \tag{3.8}$$

3.2.8 MD (Maximum Difference)

The MD of the two images is defined as the maximum absolute of difference between stego and original image [60]. The value of MD is calculated by using the Equation 3.9.

$$MD = Max \left(\left| x_{j,k} - x'_{j,k} \right| \right) \tag{3.9}$$

Where $x'_{j,k}$ is generated stego image and $x_{j,k}$ is the original cover image. The higher the value of maximum difference the poorer is the quality of stego image.

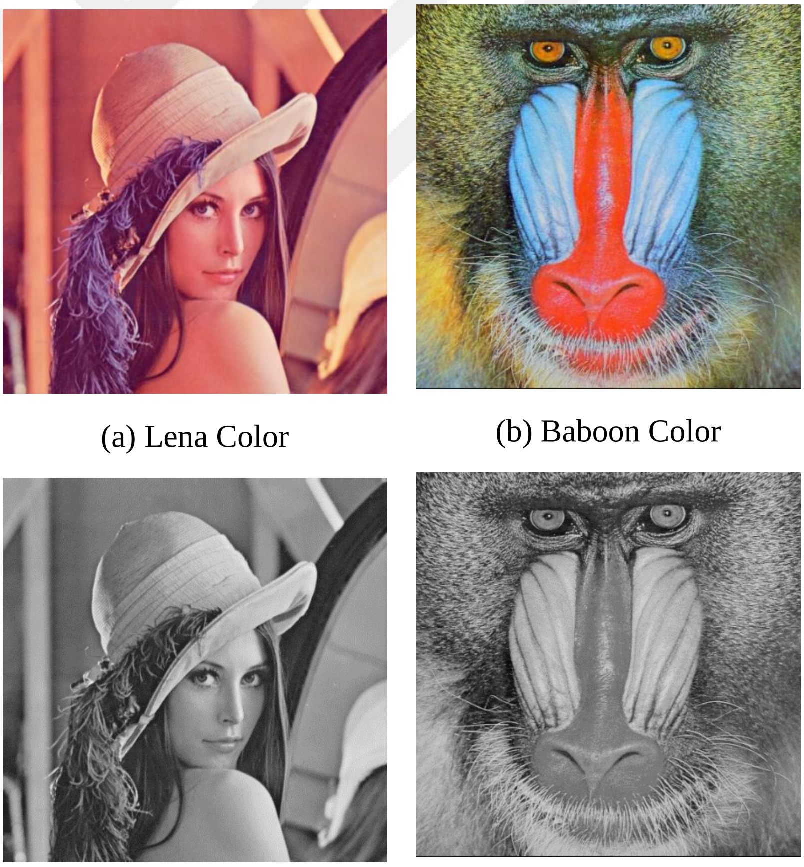
3.2.9 NAE (Normalized Absolute Error)

Normalized absolute error is defined as the ratio between sum of absolute errors between stego and original image [60]. The higher value of NAE means quality of stego image is poor. The value of NAE is calculated by using the given formula 3.10.

$$NAE = \frac{\sum_{j=1}^{M} \sum_{k=1}^{N} \left| x_{j,k} - x'_{j,k} \right|}{\sum_{j=1}^{M} \sum_{k=1}^{N} \left| x_{j,k} \right|} \tag{3.10}$$

3.3 LSB Image Steganography Problem Statement

There are number of steganographic techniques used for hiding information inside cover of images. The simplest of these techniques is classical LSB steganography techniques in which data bits replace least significant bits of the image pixels. Mostly 1-bit LSB substitution is considered but it varies as 2-bits, 3-bits, 4-bits and even 5-bits, but as the modification of number of bits per pixel are increased the quality of stego image is degraded and results in decreased PSNR and less imperceptibility. The main aim of steganography techniques is to remain undetected; but in classical LSB methods this is not achievable due to consistency of information hiding pattern. Classical LSB techniques also do not consider the pixel intensity while hiding secret information which helps to utilize weakness of human visual system and is used in different ways by modern LSB image steganography techniques.



(a) Lena Color

(b) Baboon Color

(c) Lena Gray

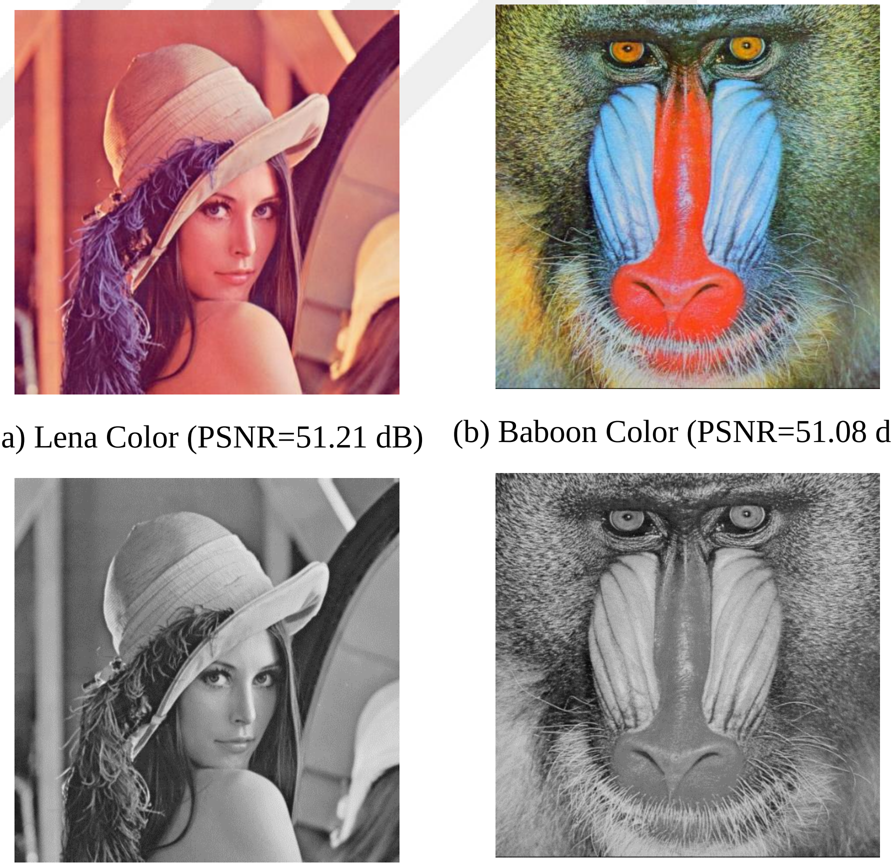
(d) Baboon Gray

Figure 3.1 Original Standard Images Used in Analysis

For the testing purposes, experiments were conducted using different classical LSB steganography methods. Specifically 4 different images were used and explained; the specification of these images are Lena gray image of sized 512 x 512, Baboon gray image of size 512 x 512, Lena colored image of sized 512 x 512 x 3 and Baboon colored image of size 512 x 512 x 3 as shown in Figure 3.1. These images are selected here because these images are widely used by steganographic community for experimental purposes. LSB image steganography techniques as 1-bit LSB, 2-bit LSB, 3-bit LSB and 4-bit LSB steganography techniques were applied on these images in Matlab and analysis were performed as following.

3.3.1 1-Bit LSB Steganography

1- Bit LSB technique can be defined as a technique in which a single bit is embedded across each pixel of image. Figure 3.2 shows the result of 1-LSB encoding. The Gray scale Lena and Baboon images hides 262,144 bits of secret data while colored images Baboon and Lena hides 786,432 bits of data embedded as payload.



(a) Lena Color (PSNR=51.21 dB) (b) Baboon Color (PSNR=51.08 dB)

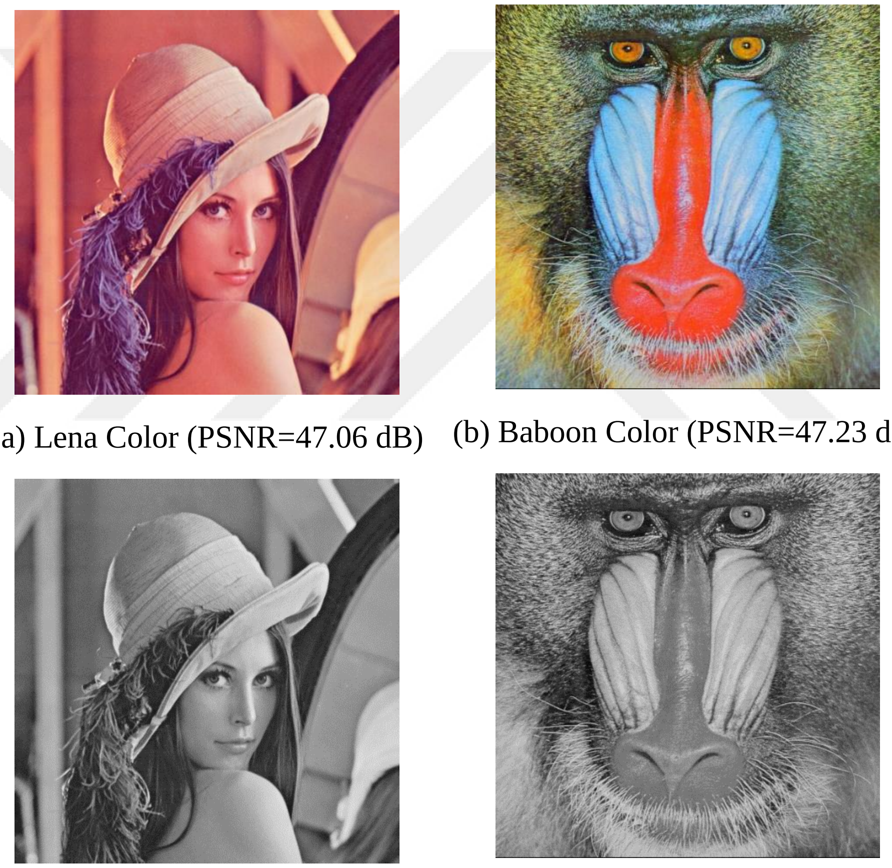
(c) Lena Gray (PSNR=51.15 dB) (d) Baboon Gray (PSNR=51.15 dB)

Figure 3.2 1-LSB Stego Images

The PSNR of Gray scale Lena and Baboon is 51.15 dB and 51.15 dB respectively, while the PSNR of Lena and Baboon color images is 51.21 dB and 51.08 dB respectively.

3.3.2 2-Bit LSB Steganography

Figure 3.3 shows the result of 2-LSB encoding. The Gray scale Lena and Baboon contain 524,288 bits of data and color Lena and Baboon contain the 1,572,864 bits of data as a payload. The PSNR of Gray scale Lena and Baboon is 44.77 dB and 44.77 dB respectively, while PSNR of color Lena and Baboon images is 47.06 dB and 47.23 dB respectively.



(a) Lena Color (PSNR=47.06 dB) (b) Baboon Color (PSNR=47.23 dB)

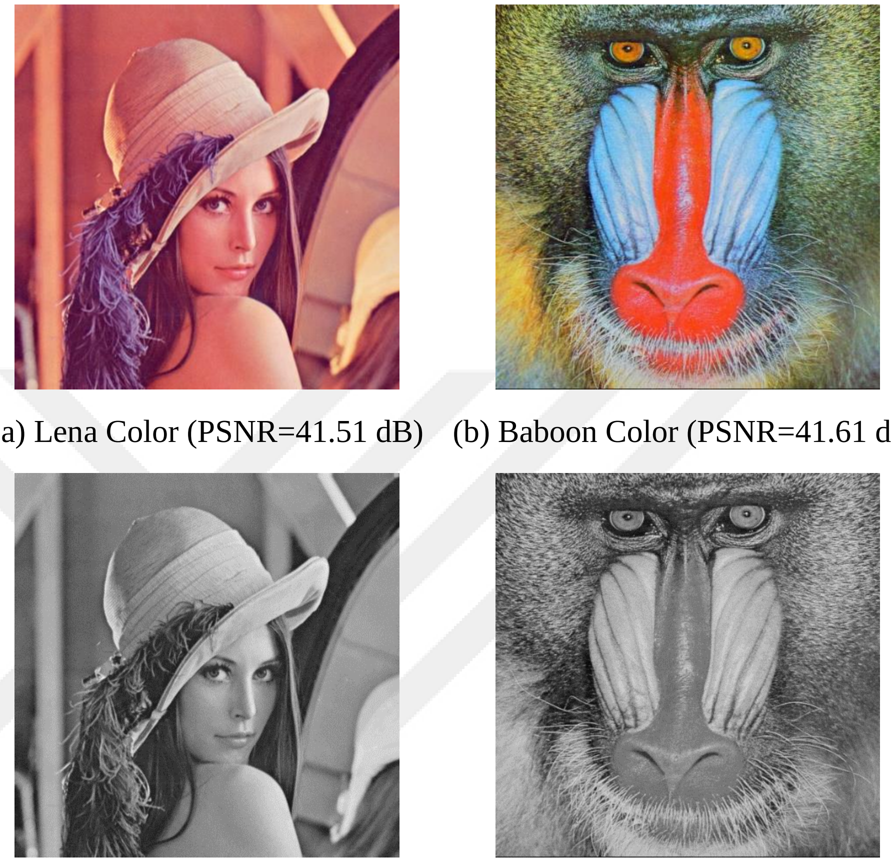
(c) Lena Gray (PSNR=44.77 dB) (d) Baboon Gray (PSNR=44.77 dB)

Figure 3.3 2-LSB Stego Images

3.3.3 3-Bit LSB Steganography

In this classical technique 3 bits are stored inside each pixel of cover image. Figure 3.4 shows the result of 3-LSB encoding. The Gray scale Lena and Baboon contain 786,432 bits of data and color Lena and Baboon contain the 2,359,296 bits of data as a payload.

The PSNR of Gray scale Lena and Baboon is 38.35 dB and 38.34 dB respectively, while The PSNR of color Lena and Baboon is 41.51 dB and 41.61 dB respectively.



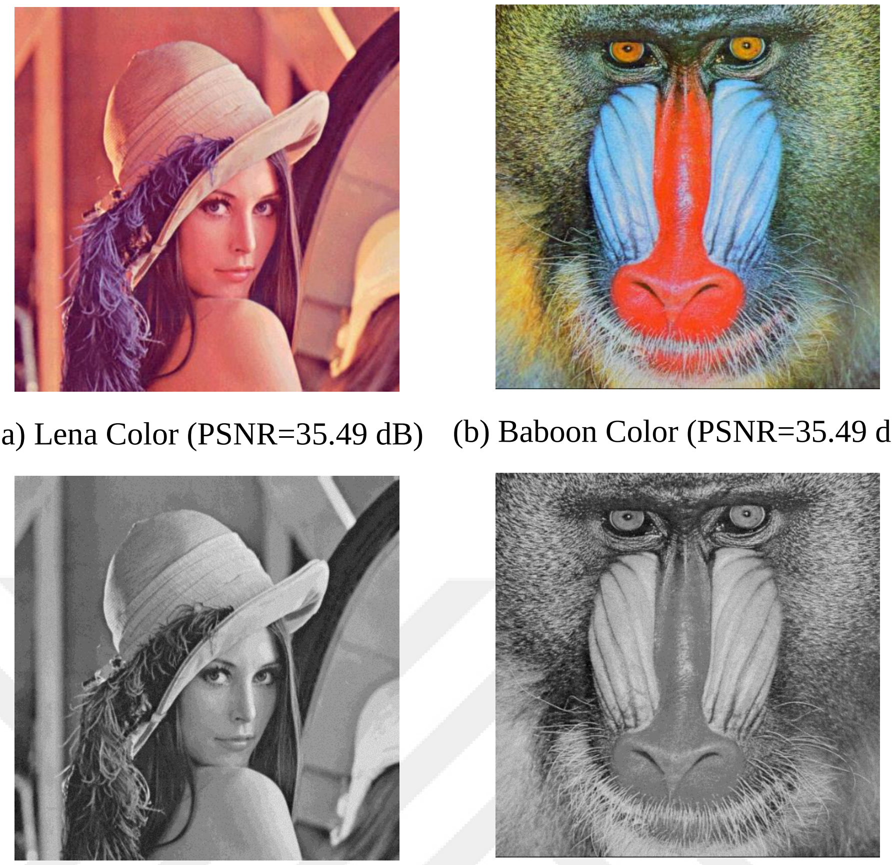
(a) Lena Color (PSNR=41.51 dB) (b) Baboon Color (PSNR=41.61 dB)

(c) Lena Gray (PSNR=38.35 dB) (d) Baboon Gray (PSNR=38.34 dB)

Figure 3.4 3-LSB Stego Images

3.3.4 4-Bit LSB Steganography

4-Bit LSB steganography allows hiding 4 bits per image pixel. Figure 3.5 shows the result of 4-LSB. The Gray scale Lena and Baboon contain 1,048,576 bits of data and color Lena and Baboon contain 3,145,728 bits of data as a payload. The PSNR of Gray scale Lena and Baboon is 32.01 dB and 32.13 dB respectively, while the PSNR of color Lena and Baboon is 35.49 dB and 35.49 dB respectively.



(a) Lena Color (PSNR=35.49 dB) (b) Baboon Color (PSNR=35.49 dB)

(c) Lena Gray (PSNR=32.01 dB) (d) Baboon Gray (PSNR=32.13 dB)

Figure 3.5 4-LSB Stego Images

In the above images we use 3-Bit and 4-Bit LSB Steganography, but it can be noticed that these images when hides more data; decreases the cover image quality and decrease in PSNR of the cover image resulting in easy detection and extraction of the secret information from the image.

Tables 3.1 and 3.2 provide comparison between PSNR and payload for the four test images for 1 to 4 bit embedding in LSB. It is shown that increasing the payload decreases PSNR for every test image.

3.3.5 Comparison of PSNR and Payload

For all the experiments performed above in this section; comparisons are performed to demonstrate that how the quality of image is degraded with increase of payload as shown in Table 3.1 for gray scale images.

Table 3.1 Gray Scale Lena and Baboon Stego Images Results

LSB bits	PSNR (dB)		Payload (Bits)	
	Lena	Baboon	Lena	Baboon
1	51.15	51.15	262,144	262,144
2	44.77	44.77	524,288	524,288
3	38.35	38.34	786,432	786,432
4	32.01	32.13	1,048,576	1,048,576

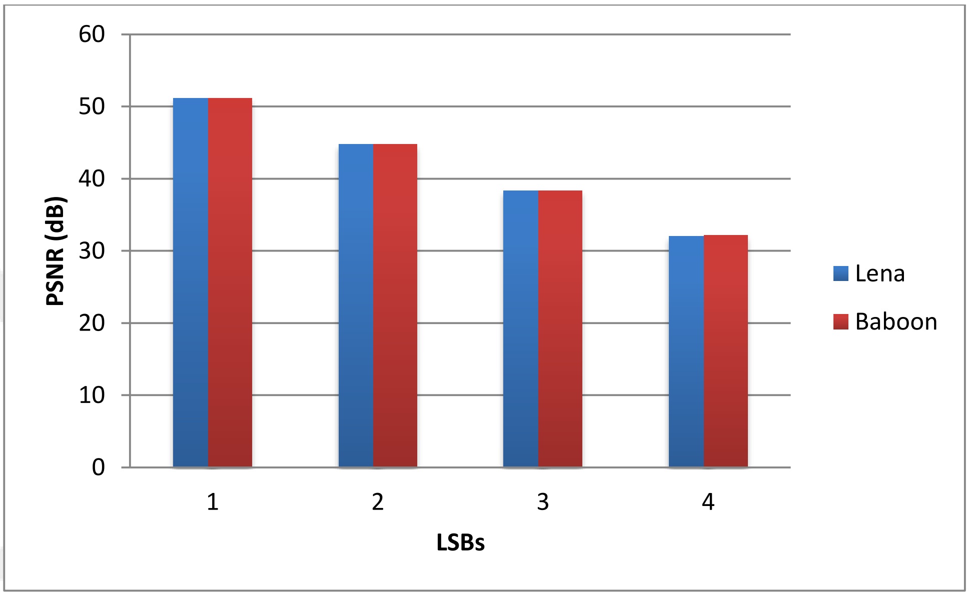


Figure 3.6 Gray Scale Lena and Baboon Stego Images PSNR

Figure 3.6 chart shows PSNR values of gray scale Lena and Gray scale Baboon. Values on x-axis show embedding bits and values on y-axis are PSNR values. It can be observed that while Increasing embedding data bits results in decrease of PSNR for both images and vice versa.

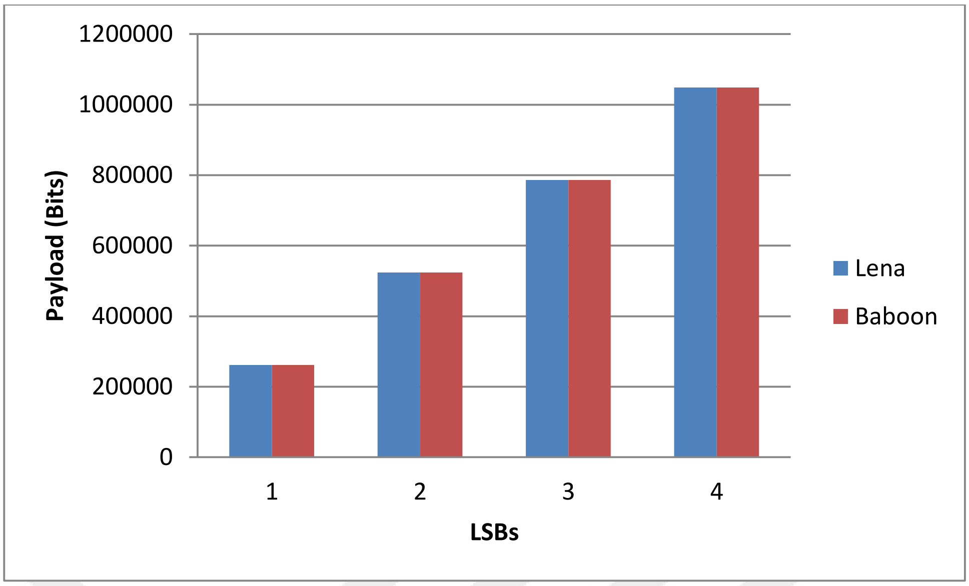
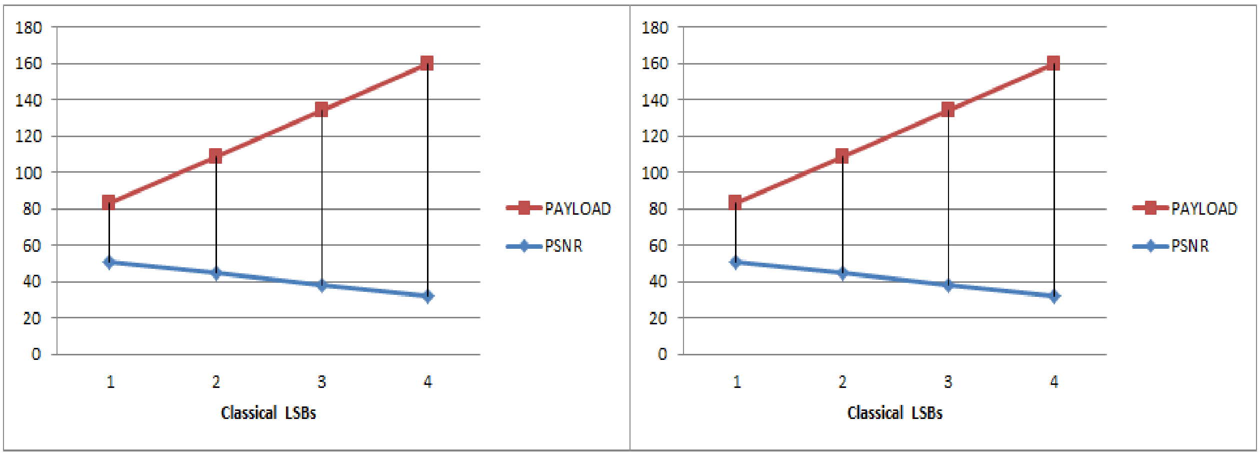


Figure 3.7 Gray Scale Lena and Baboon Stego Images Payload

Figure 3.7 shows the payload for gray scale Lena and gray scale Baboon. Increasing embedding bits in these image increases the payload of steganographic technique. Table 3.2 shows PSNR and payload values for 1 bit, 2 bit, 3 bit, and 4 bit embedding for color Lena and Baboon stego images. Figure 3.8 shows that there exists inverse relation between payload and PSNR as it can be seen that PSNR of stego gray image decreases along with increase in the payload. Hence, increase in payload decreases the stego image quality.



(a) (b)

Figure 3.8 Payload (KB) vs PSNR Comparison for Gray Images (a) Lena (b) Baboon

Table 3.2 Color Lena and Baboon Stego Images Results

LSB bits	PSNR (dB)		Payload (Bits)	
	Lena	Baboon	Lena	Baboon
1	51.21	51.08	786,432	786,432
2	47.06	47.23	1,572,864	1,572,864
3	41.51	41.61	2,359,296	2,359,296
4	35.49	35.49	3,145,728	3,145,728

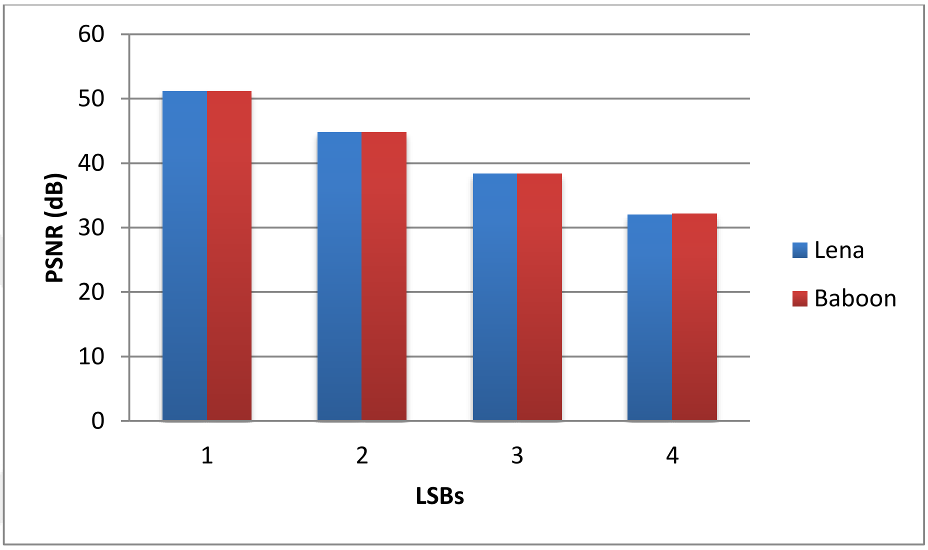


Figure 3.9 Color Lena and Baboon Stego Images PSNR

Figure 3.9 shows PSNR values for color Lena and color Baboon. It can be seen that by increasing the number of embedding bits in these images decreases the PSNR values of the resultant stego images.

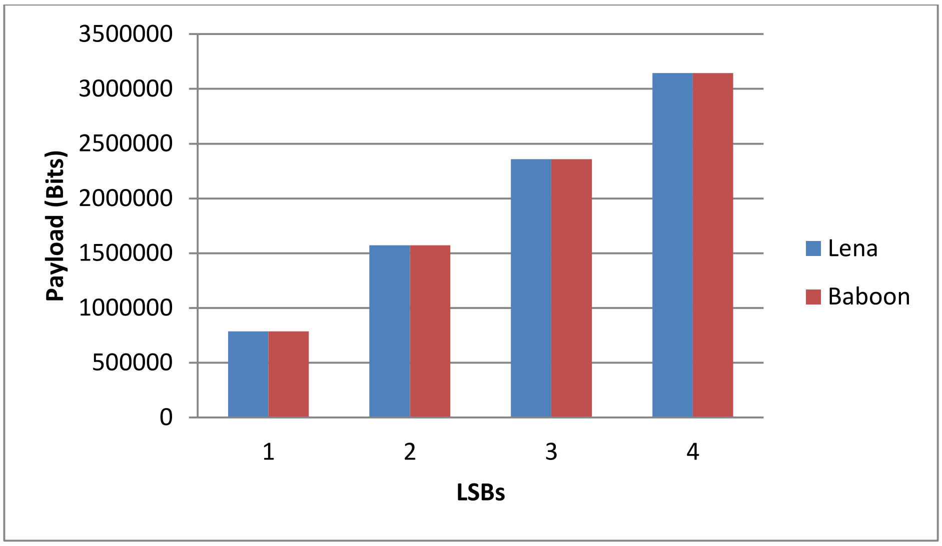
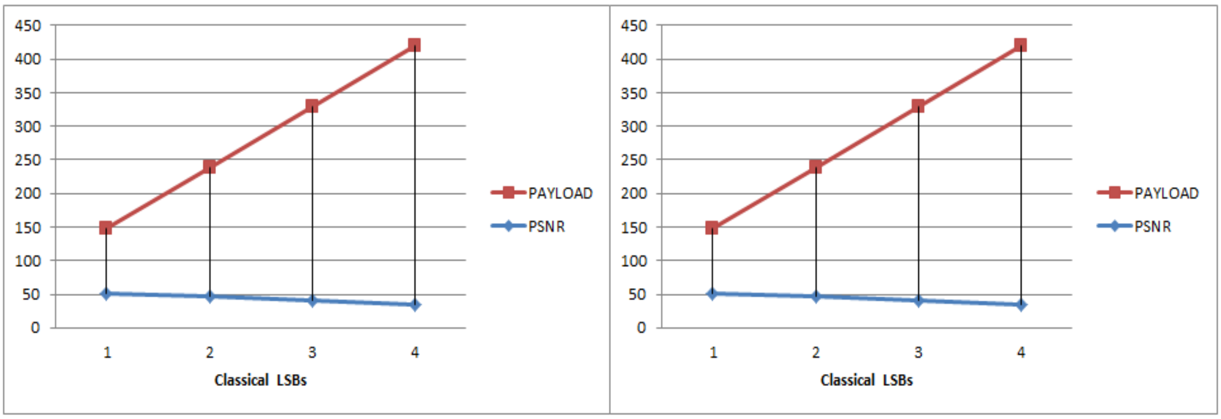


Figure 3.10 Color Lena and Baboon Stego Images Payload

The Figure 3.10 shows that payload of the stego images increases with increasing embedding of secret information bits. Figure 3.11 shows the relationship between payload and PSNR of stego color images. It shows that PSNR of stego image is decreased when there is increase in embedding capacity.



(a) (b)

Figure 3.11 Payload (KB) vs PSNR Comparison for Color Images (a) Lena (b) Baboon

3.4 Problem Identification in Classical LSB Image Steganography

The implementation of simple embedding in the least significant bits of cover images depicts that increasing payload of stego images decreases the peak signal to noise ratio and vice versa. Secondly, major limitation of classical LSB image steganography is that secret information bits are linearly added to LSBs of cover image pixels and hence can be easily detected and recovered from stego image. This can be done by the reversing the application process of clipping the LSB bits and then arranging them in order. Hence simple insertion of secret information bits in the LSB part of cover image pixels lacks the complexity required to thwart the attacker and secret information in such stego images is perceptible.

3.5 LSB Image Steganography Research Questions

From the above discussion following research questions arises:

- How to increase the hiding capacity of a steganographic technique while preserving high imperceptibility?
- Can MSBs be used with LSBs for data hiding?
- How to improve PSNR of stego Image or imperceptibility of secret information in cover image?
- Can a technique be developed which ensures minimal changes for data hiding, thus ensuring the original cover image quality to hide data?

4. IMAGE STEGANOGRAPHY USING SIMILARITY OF BITS PAIRS

This chapter explains the first contribution of this dissertation; a technique of image steganography using similarity of bits pairs (SOBP). This technique is novel in terms of bits usage for steganography purposes as it utilizes MSBs along with LSBs for crafty data embedding. Firstly encoding and decoding process is elaborated, followed by the experimental results and analysis for the developed technique.

This technique is a novel technique and is different from all the previous techniques because it focuses on matching of bits pairs. The existing techniques focused on the use of least significant bits or featured based data. In this technique, data bits of the message to be secured are arranged in pairs and image pixel bits are also arranged pairwise. For this purpose, the 3rd and 4th bits, 4th and 5th bits, 5th and 6th bits, 6th and 7th bits of pixels are arranged in pairs. The corresponding pairs are numbered as the 0th pair (3rd and 4th), 1st pair (4th and 5th), 2nd pair (5th and 6th) and the 3rd pair (6th and 7th). Data bits to be secured are compared with pixels bit pairs and based on respective similarity embedding is done. If there is no similarity between the data bits pair and pixel pairs, secret data pair is embedded into the 0th pair of pixel bits. The proposed technique is applied to fifty different test images and results for four images are considered here for the technique evaluation. The results show that the proposed technique is more secure and it hides higher payload of secret information with good quality of stego images and improved signal to noise ratio as well.

4.1 Mathematical Modeling for SOBP

The process of encoding and decoding through SOBP is mathematically described in this section. Suppose *M* denotes the secret message, message bits are divided into chunks through function *CH(M).* Pixels of cover image are divided into bits pair using function B_P and result is cover pairs denoted as $C_{P.}$ The message chunks M_C and cover pair C_P are passed to encoding function *E* which generates stego file *S*. The process is shown mathematically as below:

$$M_C = CH(M) \tag{4.1}$$

$$C_P = B_P(C) \tag{4.2}$$

$$S=E(C_P,M_C) \tag{4.3}$$

The recipient has to apply the reverse process for decoding to extract original message from stego image *S*. Stego file is passed through function BP which generates stego pairs number S_P from stego pixels. After that secret pair number (S_P) is passed through decoding function *D* that generates message chunks as M_C. *CH'* function is applied to combine bits chunks into message *M*. The process is mathematically shown below:

$$S_P=BP(S) \tag{4.4}$$

$$M_C=D(S) \tag{4.5}$$

$$M=CH'(M_C) \tag{4.6}$$

4.2 Data Encoding Process

The proposed technique reads the cover image and secret information bits. Then secret information is divided into chunks such that every chunk contains two bits in it. Then, each image pixel is divided into pairs and is assigned pair numbering. These pairs are the 3rd, 2nd, 1st and 0th pair, and assign bits 7th and 6th to 3rd, 6th and 5th to 2nd, 5th and 4th to 1st, 4th and 3rd to 0th pair number respectively.

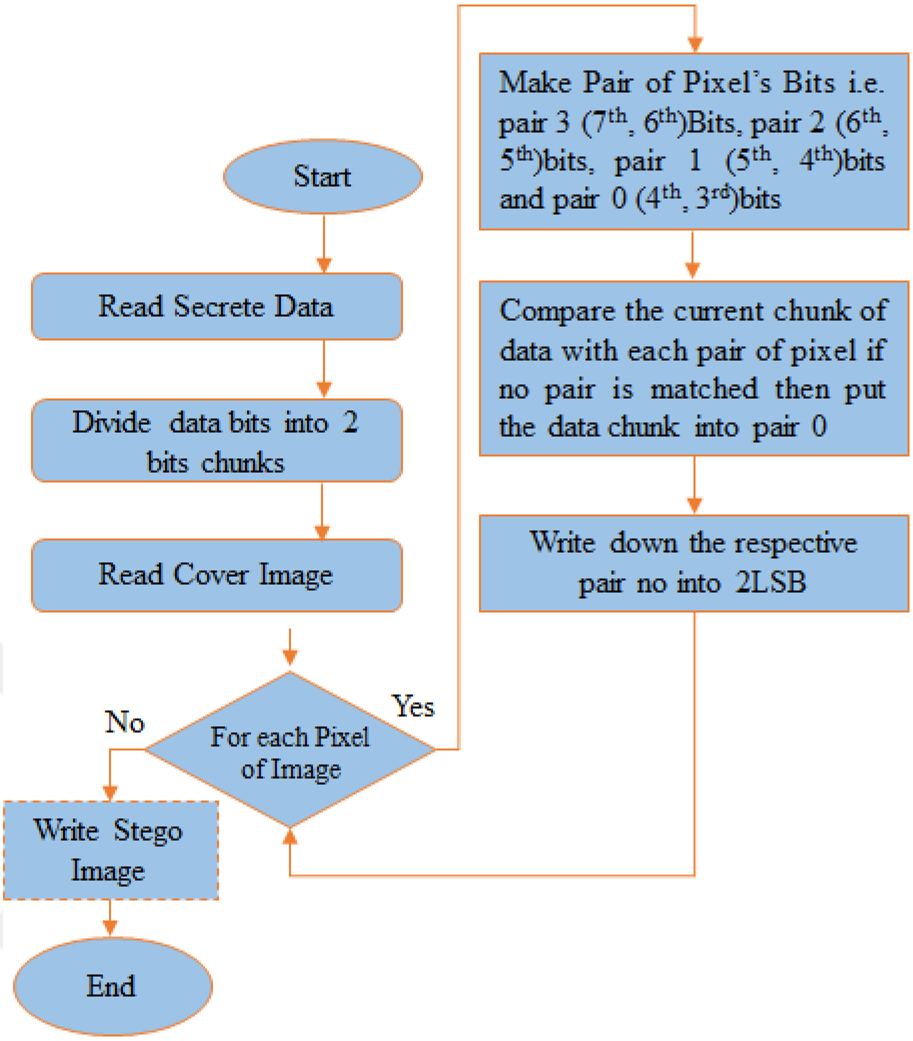

Start
Read Secrete Data
Divide data bits into 2 bits chunks
Read Cover Image
For each Pixel of Image
Yes
No
Make Pair of Pixel's Bits i.e. pair 3 (7th, 6th)Bits, pair 2 (6th, 5th)bits, pair 1 (5th, 4th)bits and pair 0 (4th, 3rd)bits
Compare the current chunk of data with each pair of pixel if no pair is matched then put the data chunk into pair 0
Write down the respective pair no into 2LSB
Write Stego Image
End

Figure 4.1 SOBP Embedding Process

Matching of data bits is done with the pixel bits, if data bits and 3rd pair bits are matched then replace two LSB of the respective pixel with the number of the 3rd pair as (1, 1). If these don't match then algorithm compares the data bits with 2nd pair, if data and the 2nd pair bits are matched then replace two LSB of the respective pixel with the number of the 2nd pair as (1, 0). Still, if there is no match then the technique proposes to match with a 1st pair with the data bits. If matched then replace two LSB of the respective pixel with the number of the 1st pair as (0, 1). If still no match is found then the 0th pair will be interchanged as original data secret bits, and two LSB of the respective pixel with the number of the 0th pair as (0, 0). The flow chart shown in Figure 4.1 describes the embedding process of secret information of proposed technique. The process of comparing data bits pair with pixel pair starts with comparison with 3rd pair and ends at 0th pair and the pair number that matches the data bit pair replaces 2LSBs of that pixel. If more than two pairs are matched than the first matched pair replaces 2LSBs.

Data Encoding Example

The process of embedding information is illustrated here with example.

The secret data bits to be embedded are 11110101.

First data bits pair to be embedded is 11 and first pixel value is 11010000.

- In this pixel 3rd pair is 10, 2nd pair is 01, the 1st pair is 10 and 0th pair is 00.
- Compare first data bits pair i.e. 11 of secret data with pixel pairs.
- As there is no match so put the data chunk in the 0th pair and put the pair number 00 in two LSB as shown in Figure 4.3.
- Now pixel becomes 1101**1100.**

This shows that 4 bits are modified where pixel pairs are not matched with data bit pairs, 2 for hiding data and 2 bits for storing pair number.

Similarly, for second chunk 11 and second pixel value is: 11100111.

- 3rd pair is 11, 2nd pair is 10, the 1st pair is 00 and 0th pair is 01.
- Compare the second chunk of secret data with pixel pairs.
- As 3rd pair matched with the second chunk, put 3 as 11 into two LSB of a pixel.
- Repeat all steps for every data chunk we get the stego pixels shown in Figure 4.3.

Now pixel value after embedding 2 bits becomes 111001**11**. In this case even none of the pixel bits are modified and 2 bits data is embedded.

As the secret data bits to be embedded are 11110101.

Similarly, for third chunk 01, respective third pixel value is: 00100001.

- 3rd pair is 01, 2nd pair is 10, the 1st pair is 00 and 0th pair is 00.
- Compare the second chunk of secret data with pixel pairs.
- As 3rd pair matched with the second chunk, put 3 as 11 into two LSB of a pixel.

Now pixel value after embedding 2 bits becomes 00100011. In this case only one of the pixel bits are modified and 2 bits data is embedded.

Similarly, for the fourth chunk 01 and second pixel value is: 11101001.

- 3rd pair is 11, 2nd pair is 10, the 1st pair is 01 and 0th pair is 10.
- Compare the second chunk of secret data with pixel pairs.

- As 3rd pair is 11 and data pair is 01 so not matched.
- 2nd pair is 10 and data pair is 01 so again pairs are not matched.
- 1st pair is 01 and data pair is also 01 so matched.
- Replace the pixel pair number with 2LSBs which should be for pair number 01

Now pixel value after embedding 2 bits becomes 11101001. In this case again even none of the pixel bits are modified and 2 bits data is embedded.

4.2 Data Decoding Process

On the receiver side, user receives the image and extracts the secret information bits from stego image according to the proposed technique's decoding algorithm in the following manner. First of all, the receiver reads stego image and separates the two LSB bits of each pixel, and defines the bits pairs according to the embedding process. After that decimal values of each two LSB are read and the data chunk is decoded from the respective bits pairs of the image pixel. If 2LSB contains (1,1) than data from 3rd pair of pixel is extracted, if it contains (1,0) than data from pixel's 2nd pair is taken as secret bits, if (0,1) is the value of 2LSB than data of the 1st pair is taken as secret bits. In the last case if 2LSB has (0, 0) then data bits are in 1st pair that is 3rd and 4th bit of corresponding image pixel. In this manner the whole pairwise data is extracted from the image pixel by pixel and written to the file on the receiver. The extraction process of secret data is shown in Figure 4.2.

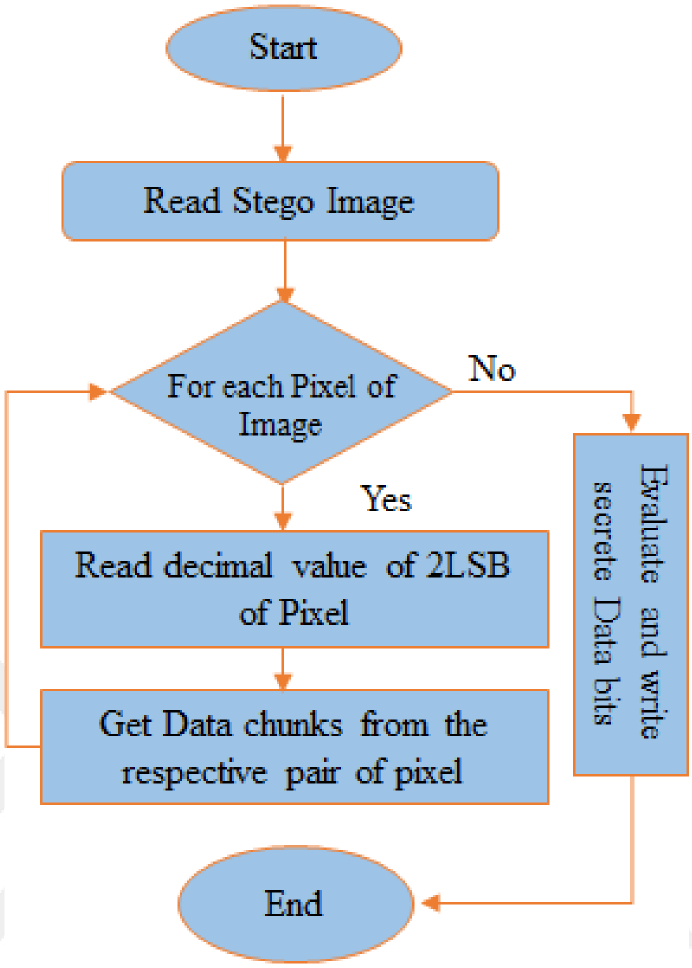
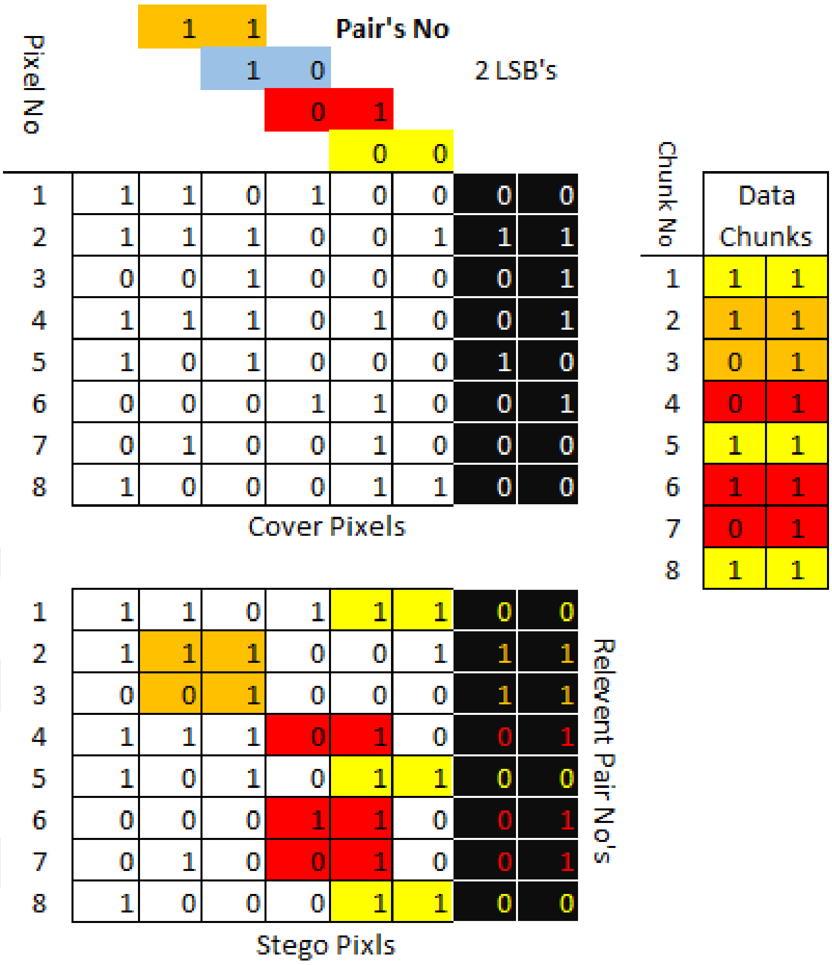

Start
Read Stego Image
For each Pixel of Image
No
Yes
Read decimal value of 2LSB of Pixel
Get Data chunks from the respective pair of pixel
Evaluate and write secrete Data bits
End

Figure 4.2 SOBP Extraction Process

For received image

- Separate the two LSB from each pixel and calculate the decimal values for pair number.
- For the first pixel, we get two LSB decimal values as 0.
- So get the data chunk from 0th pair.
- For second pixel we get the 11 from two LSB and the decimal value is 3.
- So get data chunk from 3rd pair.
- Similarly, repeat this process for every pixel after that combine and evaluate the secret data.



Pixel No								
1	1	1	0	1	0	0	0	0
2	1	1	1	0	0	1	1	1
3	0	0	1	0	0	0	0	1
4	1	1	1	0	1	0	0	1
5	1	0	1	0	0	0	1	0
6	0	0	0	1	1	0	0	1
7	0	1	0	0	1	0	0	0
8	1	0	0	0	1	1	0	0

Cover Pixels

Chunk No	Data Chunks	
1	1	1
2	1	1
3	0	1
4	0	1
5	1	1
6	1	1
7	0	1
8	1	1

1	1	1	0	1	1	1	0	0
2	1	1	1	0	0	1	1	1
3	0	0	1	0	0	0	1	1
4	1	1	1	0	1	0	0	1
5	1	0	1	0	1	1	0	0
6	0	0	0	1	1	0	0	1
7	0	1	0	0	1	0	0	1
8	1	0	0	0	1	1	0	0

Stego Pixls

Figure 4.3 SOBP Encoding and Decoding Process Example

Data Decoding Examples

Data decoding examples from the received stego pixels as shown in Figure 4.3 are explained here for first three pixels.

The first pixel received in stego image is 11011100.

- So, from this pixel two LSBs will be separated which are 00.
- 00 pair in decimal shows that data bits are stored in pair 0.
- Pair 0 which are actually 3rd and 4th LSBs contains bits the embedded bits 11.
- The extracted data chunk is written to the data file.

The second pixel received in stego image is 11100111.

- So, from this pixel two LSBs will be separated which are 11.
- 11 pair number in decimal shows that data bits are stored in pair 3.
- Pair 0 which are actually 6th and 7th LSBs contains bits the embedded bits 11.
- The extracted data chunk is written to the data file.

The third pixel received in stego image is 00100011.

- So, from this pixel two LSBs will be separated which are 11.
- 11 pair in decimal shows that data bits are stored in pair 3.
- Pair 0 which are actually 6th and 7th LSBs contains bits the embedded bits 11.
- The extracted data chunk is written to the data file.

The fourth pixel received in stego image is 11101001.

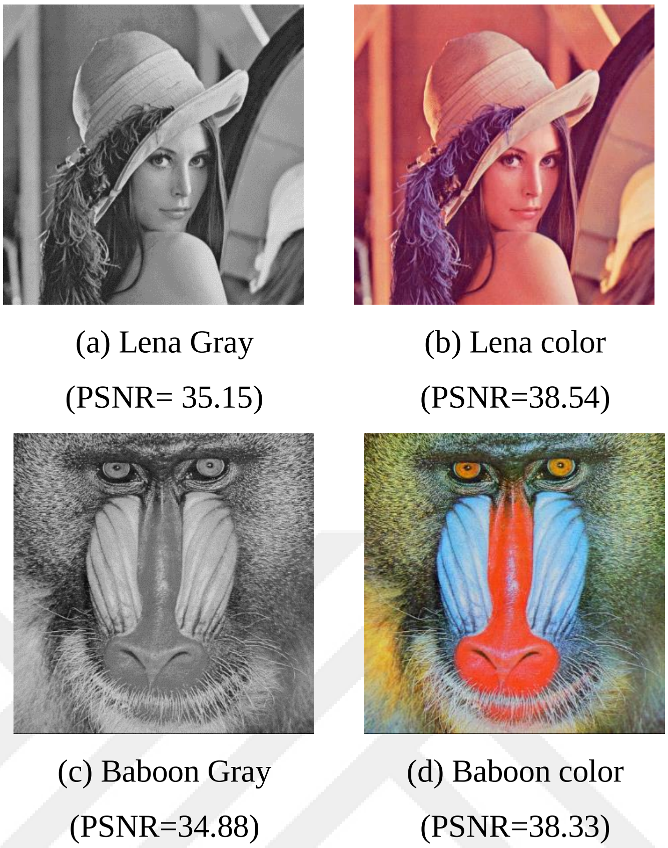
- So, from this pixel two LSBs will be separated which are 01.
- 11 pair in decimal shows that data bits are stored in pair 1.
- Pair 0 which are actually 4th and 5th LSBs contains bits the embedded bits 01.
- The extracted data chunk is written to the data file.

Concluding the technique explained above, two bits are embedded across each pixel whereas most of the existing steganography techniques are based on only single bit hiding inside pixel. This is one of the key beneficial features for the effectiveness of the proposed technique that it has more payload than the existing techniques and PSNR is more than 30 dB which is acceptable and over the HVS detection.

4.5 Experimental Results

The test results of four standard images are shown for the analysis and implementation of the proposed algorithm, as shown in Figure 4.4. The images selected are Gray level Lena of 512 x 512; Colored Lena of 512 x 512 x 3; Gray level Baboon 512 x 512; and Colored Baboon i.e. 512 x 512 x 3. Matlab 2016a on Intel core i5 with 4 cores and using 8 GB RAM is used to implement the proposed technique.

The stego images after embedding secret message for the dataset shown in Figure 4.4 and cover images are shown in Figure 3.1. It is clear from stego images that the variations in stego image after embedding secret message are undetectable by HVS. The human eye cannot find the difference between the cover images and the stego images. The results show that the proposed algorithms hide a higher capacity of secret information in the cover images with improved quality of the stego images and it improves the signal to noise ratio as well.



(a) Lena Gray

(PSNR= 35.15)

(b) Lena color

(PSNR=38.54)

(c) Baboon Gray

(PSNR=34.88)

(d) Baboon color

(PSNR=38.33)

Figure 4.4 SOBP Stego Images

4.5.1 Comparison with Existing Algorithms

The proposed technique is compared with a number of techniques available in the literature. The comparison is done on the basis of a number of image performance parameters which are explained in chapter 3. Tables 4.1, 4.2 and 4.3 show the comparison of the proposed technique with the existing techniques. The first column under "Techniques" lists the names of techniques while parameter wise comparison for each parameter is done row wise.

Table 4.1 Comparison Results for Lena Stego Gray Image

	Lena Stego Gray Image				Baboon Stego Gray Image			
Techniques	MSE	RMSE	PSNR (dB)	Payload (Bits)	MSE	RMSE	PSNR (dB)	Payload (Bits)
Ni et al. [61]	0.98	0.98	48.2	5,460	0.98	0.98	48.2	5,421
Hwang et al.[62]	0.97	0.98	48.22	5,336	0.97	0.98	48.22	5,208
Lin et al.[63]	1.42	1.19	46.6	59,900	1.12	1.05	47.61	19,130
Hu et al. [64]	0.87	0.93	48.69	60,241	0.95	0.97	48.34	21,411
Luo et al. [65]	0.85	0.92	48.82	71,674	0.34	0.58	48.36	22,696
Wu and Tsai. [66]	4.30	2.07	41.79	50,960	3.25	1.80	37.9	56,291
Vleeschouwer et al. [67]	8.18	2.86	39	24,108	8.18	2.86	39	2,905
Xuan et al. [68]	14.22	3.77	36.6	85,507	34.12	5.84	32.8	14,916
Celik et al. [69]	10.30	3.20	38	74,600	10.30	3.20	38	15,176
LSB4	40.88	6.39	32.05	1,048,576	39.76	6.30	32.16	1,048,576
SOBP	**19.82**	**4.45**	**35.19**	**524,288**	**21.10**	**4.59**	**34.92**	**524,288**

Tables 4.1 compare the results between proposed and the existing techniques for Baboon and Lena Gray stego images. Payload of proposed technique is more than the existing techniques as two bits are concealed in each pixel. The values of error are more while PSNR is less as compared to other techniques but the most prominent feature of this technique is its high payload while keeping acceptable low level of stego image quality. The main reason behind having less error values as compared to classic LSB 4 method is that in LSB4 4 bits are modified per pixel which affects stego image quality whereas in SOBP 4 bits per pixel are modified only when none of the pixel pair matches the data pair i.e. 2 bits for data and 2 bits in LSB for pair number.

Table 4.2 Comparison Results for Lena Stego Color Image SOBP

	Lena Color				Baboon Color			
Technique	MSE	RMSE	PSNR (dB)	Payload (Bits)	MSE	RMSE	PSNR (dB)	Payload (Bits)
Yalman et al. [50]	7.19	2.68	39.56	1,156,000	7.12	2.66	39.60	1,156,000
LSB4	18.35	4.28	35.52	3,145,728	18.33	4.28	35.53	3,145,728
SOBP	**9.09**	**3.01**	**38.57**	**1,572,864**	**9.54**	**3.08**	**38.36**	**1,572,864**

Table 4.2 shows comparison between proposed and existing LSB image steganography techniques for Baboon and Lena colored images. Results exhibit that payload capacity of the proposed technique is more than [50] and classic LSB4 method, while error values are comparatively low. For Baboon stego colored image maximum difference per pixel for LSB4 is 15 per pixel whereas for the proposed technique MD is also 15 where in SOBP each pixel has 2 to 4 bits modification. PSNR is also kept at acceptable level for the proposed technique.

One of the main characteristic of a steganography technique is to remain resilient against Image manipulation. SOBP is resistant to image manipulation like zoom in and then zoom out do not have effect on stego image quality. Whereas as like other steganography techniques if there is no loss of pixels during image rotation or when other manipulation techniques, the quality of SOBP generated stego image remains unaffected. Results show that the proposed technique has more payload than all the existing techniques while it maintains PSNR at the acceptable level which ensures god visual quality of mage along with minimal error. This technique is a novel technique and is different from all the previous techniques because it focuses on pattern matching of bits pairs. All the existing techniques focused on LSB or featured based data. In this technique, data bits of the message to be secured are arranged in pairs and image pixel bits are also arranged pairwise. Data bits to be secured are compared with pixels bit pairs and based on similarity embedding is done. If there is no similarity between the pairs, secret data pairs are embedded into the 0th pair of pixel bits. The proposed technique is applied to fifty different test images. The results show that the proposed technique is more secure and it hides higher payload of secret information with good quality of stego images that involves MSBs along with LSBs for data embedding for the first time.

5. THRESHOLD BASED LSB IMAGE STEGANOGRAPHY

This chapter defines a novel threshold-based steganography technique that utilizes the weakness of HVS for data hiding in images. The new LSB technique embeds secret information in the cover image based on its pixel values. The motivation of proposed approach is the weakness of HVS, i.e., HVS is unable to identify changes in low level colors. This idea is used for embedding information in the pixels of cover image. A threshold is set on the pixel values of the cover image and 4bits, 3bits, 2bits and 1bit of the secret information is embedded based on the threshold. The experimental results show that the proposed technique has high payload and PSNR as compared to the existing LSB steganographic techniques. The lower the pixel value the higher is its capacity to hide data. This technique achieves high payload and PSNR as compared to other existing techniques.

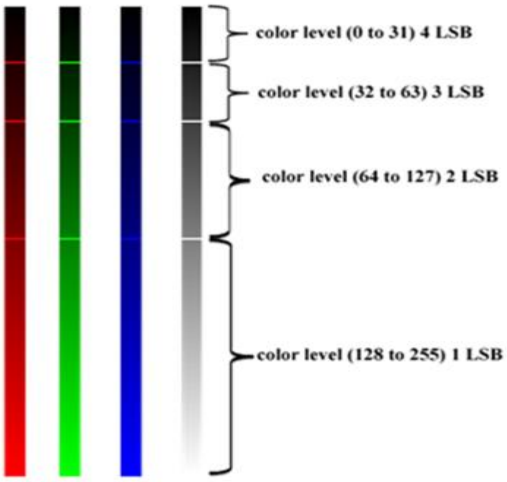

color level (0 to 31) 4 LSB
color level (32 to 63) 3 LSB
color level (64 to 127) 2 LSB
color level (128 to 255) 1 LSB

Figure 5.1 THBS Color Levels and Data Embedding Capacity

Figure 5.1 shows different threshold limit for colors and number of bits embedded per level. It shows that when pixel value is between 0-31 than 4 bits are concealed, 3 bits can be embedded when it falls in 32-63. As the value of pixel is increasing less bits are embedded so for pixel value between 64-127 only 2 bits are embedded and 1 bit is embedded when a pixel value is between 128-255. This modification enables maximum data capacity with minimal effects on image visual quality.

4.1 Mathematical Modeling for THBS

THBS encoding and decoding can be expressed mathematically and is explained in this section. Suppose M denotes the secret message and C is the cover image. Image pixels are categorized according to the defined threshold limits T_L through function F. The output is threshold pixels T_P. Secret message M *is* passed through function reshape R, that generates reshaped message M_R. In the final step threshold pixels T_P and reshaped message M_R are passed through encoding function E which encodes and generates stego file S.

$$T_P=F(T_L,C) \tag{5.1}$$

$$M_R=R(M) \tag{5.2}$$

$$S=E(T_P,M_R) \tag{5.3}$$

At the receiving side inverse process of encoding is applied to extract original message from stego image S. Stego image pixels are categorized according to the defined threshold limits T_L through function F. Threshold defined pixels T_P are passed through decoding function D to extract data bits. In the final step extracted bits are shaped and combined through D_S to get the original message M. The decoding process is mathematically shown below:

$$T_P=F(T_L,S) \tag{5.4}$$

$$M_R=D(T_P,S) \tag{5.5}$$

$$M=D_S(M_R) \tag{5.6}$$

5.1 THBS Encoding Process

The process of encoding starts with reading data and cover image file and runs loop for the total length of image pixels. This technique embeds data per pixel according to the defined criteria for color level shown in Figure 5.1. If pixel value is 0-31, 4 bits are taken from data bits and concealed into that pixel. The pointer of data is incremented by 4. Similarly for each pixel based on its value i.e. 0-31, 32-63, 64-127 or 128-255 data bits 4, 3, 2 or 1 are embedded pixel by pixel respectively. The process ends when it is done for all number of image pixels and stego image is written to the file. The process of encoding is shown in flowchart in Figure 5.2.

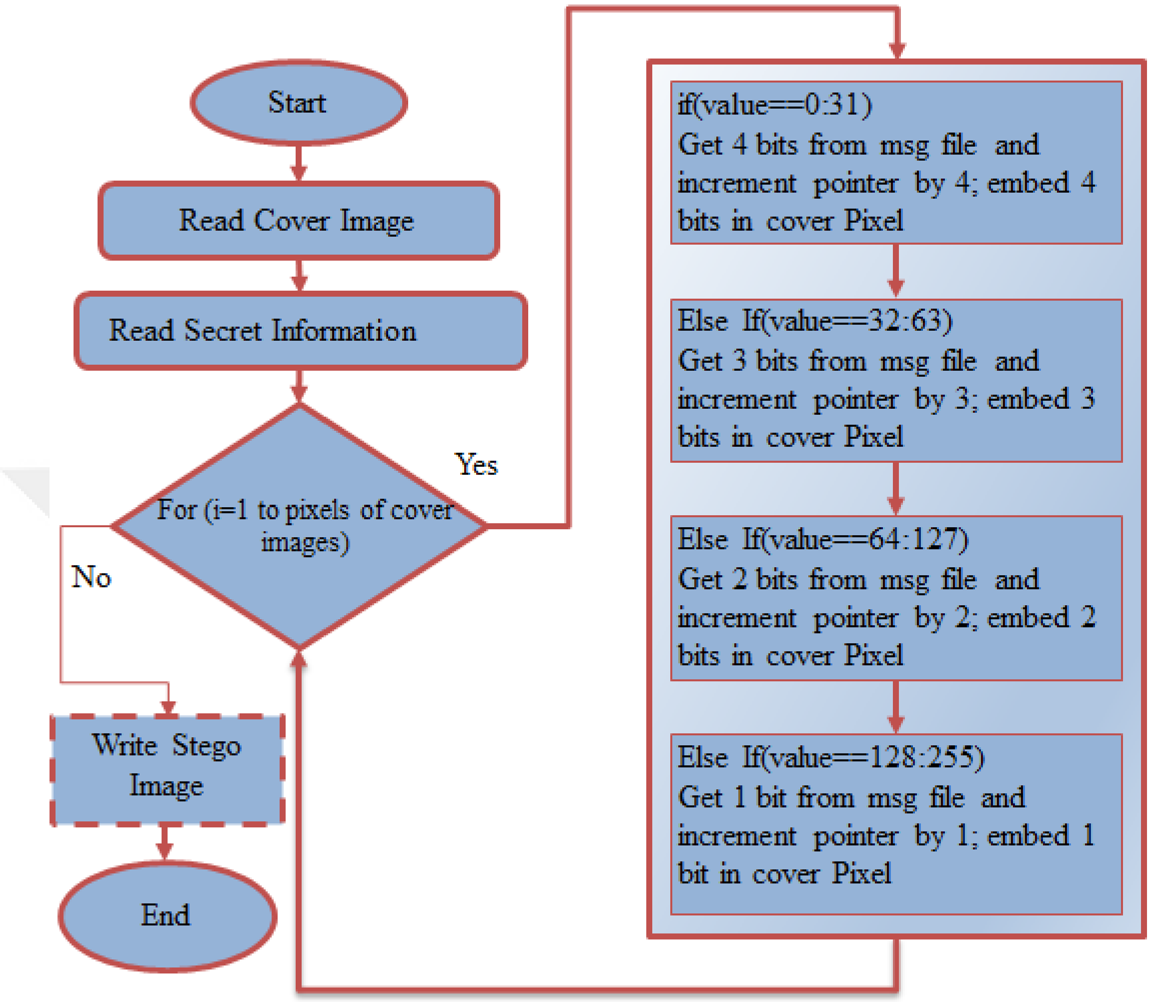

Start
Read Cover Image
Read Secret Information
For (i=1 to pixels of cover images)
Yes
No
Write Stego Image
End
if(value==0:31)
Get 4 bits from msg file and increment pointer by 4; embed 4 bits in cover Pixel
Else If(value==32:63)
Get 3 bits from msg file and increment pointer by 3; embed 3 bits in cover Pixel
Else If(value==64:127)
Get 2 bits from msg file and increment pointer by 2; embed 2 bits in cover Pixel
Else If(value==128:255)
Get 1 bit from msg file and increment pointer by 1; embed 1 bit in cover Pixel

Figure 5.2 THBS Encoding

Data Encoding Examples:

Let us consider four pixels, one from each defined category. The data bits to be embedded are 1010111011 whereas values of pixels selected are

A. 30 in binary 00011110.
B. 42 in binary 00101010.
C. 75 in binary 01001011.
D. 220 in binary 11011100.

A. For the first pixel having value of **30** the process of encoding is as follows:

- Pixel value lies between 0-31, so four bits can be embedded in this pixel according to defined encoding algorithm.
- Binary value of 30 is 00011110 and four least significant bits are selected 0001**1110** for data embedding.
- Pixel value after modification is 0001**0000**.
- Four bits to be embedded are **1010**111011
- After embedding new value of pixel becomes 0001**1010**, which in decimal is equal to 26. It can be noticed that 4 bits are hidden but only one bit is changed in pixel bits.
- Data pointer value is incremented by 4.

B. For the second pixel having value equal to **42** the process of encoding is as follows:

- As pixel value lies between **32-63**, so **three** bits can be embedded in this pixel according to defined encoding algorithm.
- Binary value of **42** is 00101010, and three least significant bits are selected 00101**010** for data embedding.
    - Pixel value after modification is 00010**000**.
    - Three bits to be embedded from data stream are 1010**111**011
    - After embedding new value of pixel becomes 00011**111**, which in decimal is equal to **47**. So, if we see inside pixel only two bits are modified.
    - Data pointer value is incremented by 3.

C. For the first pixel with a value of **75** the process of encoding is as follows:

- Pixel value lies between 64-127, therefore two bits can be embedded in this pixel according to defined encoding algorithm.
- Binary value of 75 is 01001011 and two least significant bits are selected 010010**11** for data embedding.
- Pixel value after modification is 010010**00**.
- Two bits to be embedded are 1010111**01**1
- After embedding new value of pixel becomes 010010**01**, which in decimal is equal to 73. It can be noticed that 2 bits are hidden but only one bit is changed in pixel bits.

- Data pointer value is incremented by 2.

D. For the fourth pixel with a value of **220** the process of encoding is as follows:

- Pixel value lies between 128-255, so only one data bit can be embedded in this pixel according to defined encoding algorithm.
- Binary value of 220 is 11011100 and four least significant bits are selected 1101110**0** for data embedding.
- Pixel value after modification is 1101110**0**.
- One data bit to be embedded is 101011101**1**.
- After embedding new value of pixel becomes 1101110**1**, which in decimal is equal to 221.
- Data pointer value is incremented by 1.

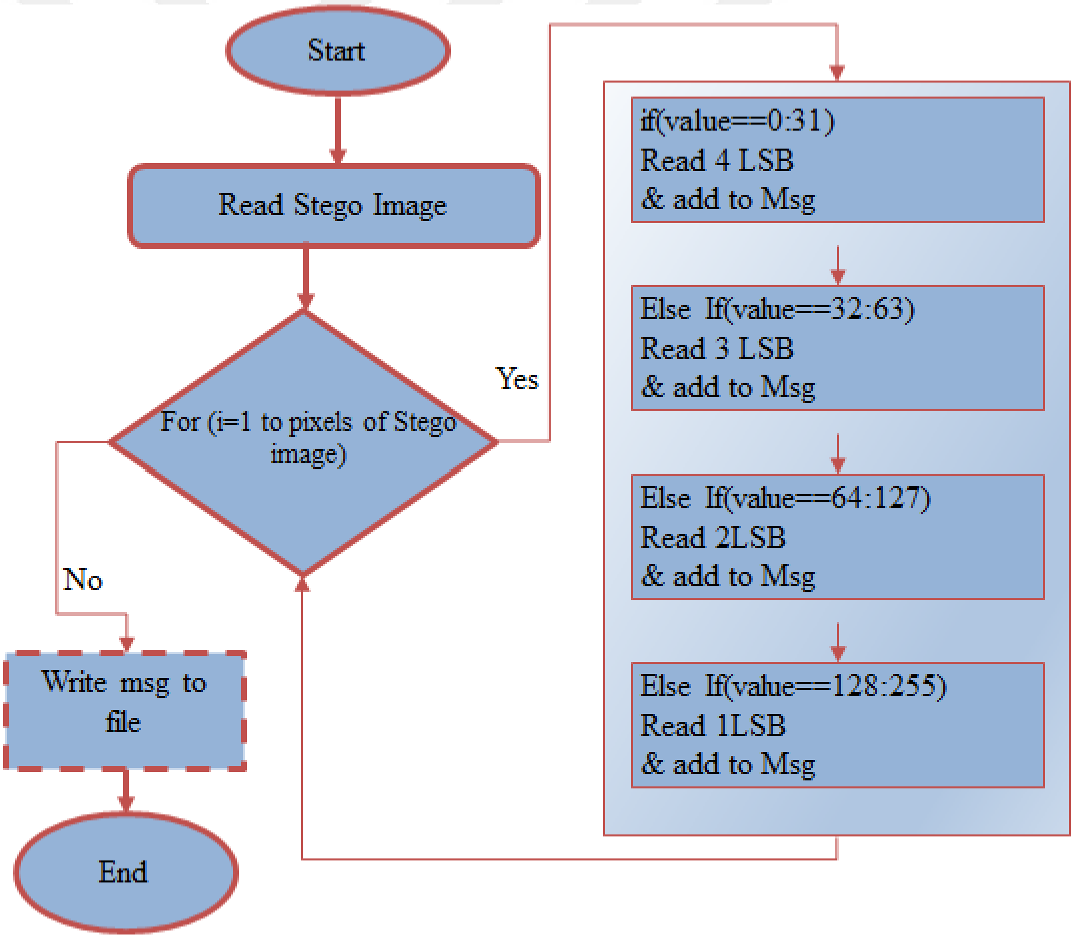


Figure 5.3 THBS Decoding Process

5.2 THBS Decoding Process

Decoding process of THBS starts with reading stego image. For the total length of pixels decoding process is performed. From each image pixel based on its color level data bits are extracted thus if pixel is between 0-31 then 4 bits, 32-63 then 3 bits, whereas for pixel value 64-127 extract 2 bits and if value of pixel is from 128-255 then 1 bit is extracted from that pixel. This process of data extraction is performed for all image pixels and data extracted is written to the file. The process is shown with help of flowchart in Figure 5.3.

Data Decoding Examples:

Four pixels encoded in above example are one from each defined category. The data bits are to be extracted from these four pixels based on their categories. The pixel values are

A. 26 in binary 00011010.

B. 47 in binary 00101111.

C. 75 in binary 01001001.

D. 221 in binary 11011101.

A. For the first pixel with a value of **26** the process of decoding is as follows:

- Pixel value lies between **0-31**, so four bits would be extracted from this pixel according to defined decoding algorithm.
- Binary value of 26 is 00011010
- Four least significant bits are extracted from this pixel 0001**1010** .
- The extracted bits **1010** are written to the data file.

B. For the second pixel with a value of **47** the process of decoding is as follows:

- Pixel value lies between **32-63**, so three bits would be extracted from this pixel according to defined decoding algorithm.
- Binary value of 47 is 00101111
- Three least significant bits are extracted from this pixel 00101**111.**
- The extracted bits **111** are written to the data file which now becomes 1010**111**.

C. For the third pixel with a value of **75** the process of decoding is as follows:

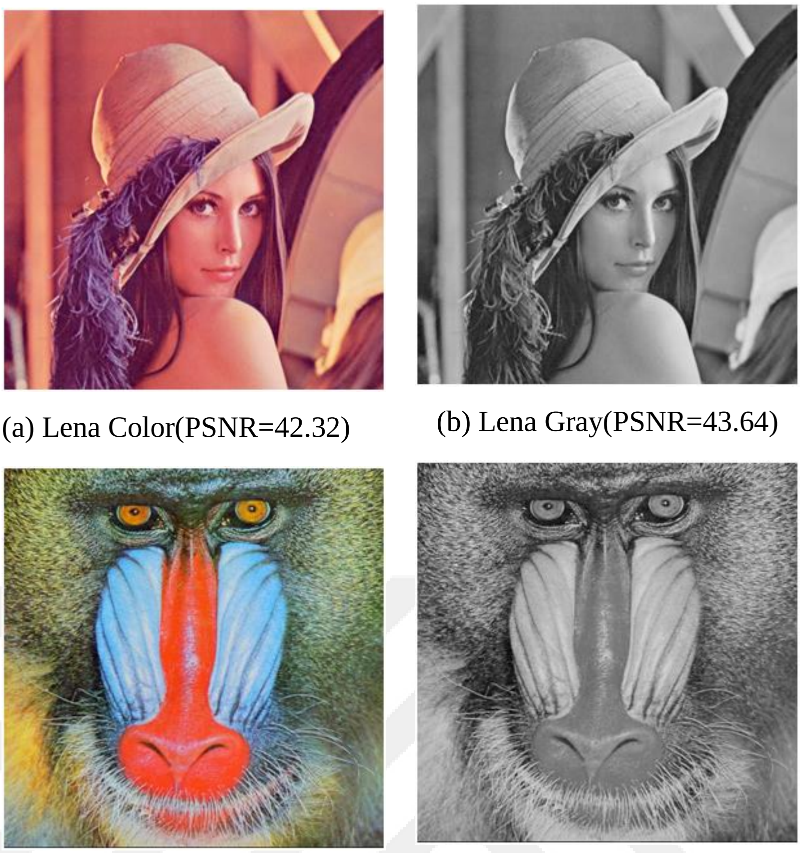
- Pixel value lies between **64-127**, therefore two bits would be extracted from this pixel according to defined decoding algorithm.
- Binary value of 75 is 01001001.
- Two least significant bits are extracted from this pixel 010010**01.**
- The extracted bits **01** are written to the data file, which now becomes 1010111**01**.

D. For the fourth pixel with a value of **221** the process of decoding is as follows:

- Pixel value lies between 128-255, therefore single bit would be extracted from this pixel according to defined decoding algorithm.
- Binary value of 221 is 11011101.
- One least significant bits are extracted from this pixel 1101110**1**.
- The extracted bits are written to the data file, which now becomes 101011101**1**.

5.3 Experimental Results

The proposed technique is implemented in Matlab 2016a on Intel core i5 with 4 cores and using 8 GB RAM. The experimental results of the proposed parallel technique show the noticeable improvement in terms of security as compared to existing techniques. Experiments were performed on fifty images and results of four standard images are elaborated that are widely used among the communities of steganography [73-76]. Figure 5.4 shows the quality of stego images that with HVS seems to be exactly the same as original cover images.



(a) Lena Color(PSNR=42.32)

(b) Lena Gray(PSNR=43.64)

(c) Baboon Color(PSNR=47.15)

(d) Baboon Gray(PSNR=44.38)

Figure 5.4 THBS Stego Images

THBS is compared with existing techniques based on fundamental image performance metrics and results are shown for four standard 512 x 512 size stego images in Tables 5.1-5.2. Column 1 in the aforementioned tables designates the techniques. Image performance metrics include Mean Square Error (MSE), Root Mean Square Error (RMSE), Universal Image Quality Index (UIQI), SNR and Mean Absolute Error (MAE) comparisons, which are shown in first five columns. Whereas followed by PSNR, Average Difference (AD), Maximum Difference (MD) and Normalized Absolute Error (NAE) depict the key performance metrics. The x value shows that authors of the proposed technique did not mention that parameter during their analysis. The payload of each stego image in bits is shown in the last column of the corresponding table.

Table 5.1 THBS Stego Lena and Baboon Color Image Comparison with Existing Techniques

	Lena Color Image				Baboon Color Image			
Technique	MSE	RMSE	PSNR (dB)	Payload (Bits)	MSE	RMSE	PSNR (dB)	Payload (Bits)
Yalman et al. [50]	7.19	2.68	39.56	1,156,000	7.12	2.66	39.6	1,156,000
Shehzad et al. [70]	9.09	3.01	38.57	1,572,864	9.54	3.08	38.36	1,572,864
LSB4	18.35	4.28	35.52	3,145,728	18.33	4.28	35.53	3,145,728
THBS	**3.24**	**1.77**	**42.32**	**1,355,199**	**1.22**	**1.13**	**47.15**	**1,329,318**

For Baboon colored image PSNR of proposed technique is 47.15 which is indication of high security level of the proposed mechanism, whereas payload is also greater as compared to [45]. For THBS payload is comparatively close to [68], whereas MSE, and RMSE are comparatively much low as compared to the THBS technique. Whereas for 4LSB technique where in each pixel 4 bits are modified for data embedding data capacity is more than double to that of this technique but MSE and PSNR shows high quality of stego image in THBS ensuring high level of security.

Table 5.2 THBS Stego Baboon Gray Image Comparison with Existing Techniques

	Lena Stego Gray Image				Baboon Stego Gray Image			
Techniques	MSE	RMSE	PSNR (dB)	Payload (Bits)	MSE	RMSE	PSNR (dB)	Payload (Bits)
Ni et al. [61]	0.98	0.98	48.2	5,460	0.98	0.98	48.2	5,421
Hwang et al.[62]	0.97	0.98	48.22	5,336	0.97	0.98	48.22	5,208
Lin et al.[63]	1.42	1.19	46.6	59,900	1.12	1.05	47.61	19,130
Hu et al. [64]	0.87	0.93	48.69	60,241	0.95	0.97	48.34	21,411
Luo et al. [65]	0.85	0.92	48.82	71,674	0.34	0.58	48.36	22,696
Wu and Tsai. [66]	4.30	2.07	41.79	50,960	3.25	1.80	37.9	56,291
Vleeschouwer et al. [67]	8.18	2.86	39	24,108	8.18	2.86	39	2,905
Xuan et al. [68]	14.22	3.77	36.6	85,507	34.12	5.84	32.8	14,916
Celik et al. [69]	10.30	3.20	38	74,600	10.30	3.20	38	15,176
LSB4	40.88	6.39	32.05	1,048,576	39.76	6.30	32.16	1,048,576
Shehzad et al. [70]	19.82	4.45	35.19	524,288	21.10	4.59	34.92	524,288
THBS	**2.84**	**1.64**	**43.64**	**433,224**	**2.33**	**1.54**	**44.38**	**410,636**

Table 5.2 shows comparison based on image performance metrics for Baboon gray and Lena gray images of size 512x512. For Baboon gray image MSE is 2.33 as compared to 3.25 in [66], whereas PSNR and Payload are also high as compared to other techniques. Payload for LSB4 is double as compared to THBS but low values of errors and PSNR depicts low stego image quality in LSB4 as compared to THBS. MSE for [70] is 19.82 and for THBS 2.84, but when payload is of both techniques is considered THBS has very good results when compared for both MSE and payload. The results shows that THBS technique while providing good level of data hiding capacity for the stego images ensures adequate level of PSNR and have less MSE. This illustrates that this technique is more secure with high payload.

5.4 Limitation of THBS : Efficient Execution

This chapter also targets limitation of Threshold based Steganography (THBS); which despite its usefulness for security purposes was unable to execute efficiently due to dependency in its execution process. This work explains the need of parallel processing and relation between parallelism and image processing and image steganography.

Steganographic algorithms executed through parallel processing, shows us one referred beneficial glimpse of parallelism. If adopted collectively both of them are major computer science research areas. This collaboration leads to develop two main advancements: protection of intellectual authentication of digital information and progress and use of efficient hardware devices, which has become inevitable in recent data processing and transmissions. In [71] implementation of various key sequential algorithms was done in parallel. The main emphasis of their work was segmentation, de-noising and improving the performance. The execution of processes is done on multi-core architectures for processing them in parallel by features calculation, noise reduction etc. For image processing [72] analyzed edge finding algorithm. They explored the weaknesses and strengths of parallel approach specifically for medical images and concluded that using parallel solutions along with parallel computing results in big performance improvement in image processing. In [73] analyzed the steganographic algorithms which can be parallelized. Their work analyzed the limitations and modifications were done through parallel processing for Wu-Lee steganographic algorithm for binary images. The results exhibited showed reduction in response time and improved the efficiency of algorithm.

Efficient THBS identifies the pipelining hazard, removes the dependency and allows execution of steganography mechanism on parallel processors in efficient manner. The encoding and decoding process and results before and after the removal of pipelining hazard are compared and discussed along with other image performance metrics comparison for ETHBS in this chapter. ETHBS along with ensuring optimal data security is able to obtain 25% performance improvement over the THBS encoding or decoding algorithm.

5.4.1 Parallel Computing and Image Steganography Evolution

Parallel computing enables simultaneous execution of multiple instructions, utilizes the principle of dividing large problems into smaller and executing small units concurrently [74-77]. There are different approaches for achieving adequate level of parallelism including bit level, instruction level, task level and data level parallelism [78]. The advancement in development of processors has blessed to utilize their high speed. Moore's Law cannot be adopted for the new situation shaped by hardware development, due to the fact of the incorporation of processing units in the same chip. Therefore, parallel programming systems are the requirement of the day [79-83].

In general, there are three key techniques for parallel processing that are adopted to obtain better performance and reduce the computational overhead.

5.4.1.1 Pipeline parallelism

In this type of parallelism, long procedures or tasks run in parallel, but still there are processes that are overlapped in terms of execution and need to run some parts sequential. This type of models relies on relational computation units. The outcome of one unit becomes input for another unit, the succeeding unit has to wait till the completion of previous unit [84, 85].

5.4.1.2 Independent or natural parallelism

In this type of parallelism, the execution units are independent and do not rely on each other [86]. The execution time in this sort of processes can be reduced by executing the process in parallel up to certain fine-grained level.

5.4.1.3 Inter-query and Intra-query parallelism

The tasks are totally independent and do not wait for others to complete. Inter query parallelism is achieved by executing distinct independent queries at same time [87-89]. The higher the number of processors the greater is the performance. To speed up the execution of a complex query it can be decomposed into smaller tasks and executed independently on parallel processing units.

5.5 Efficient-Threshold Based LSB Image Steganography

The proposed technique identifies the dependency in threshold based image steganography, distinguishes the dependent and independent steps and enables to achieve optimal performance by converting the serial process into independent parallel steps. This approach empowers to take the advantage of parallelism and when run on multicore systems, provides the obvious performance improvements, Figure 5.5. The steps of Efficient-THBS mechanism are explained in the following encoding and decoding sub-sections.

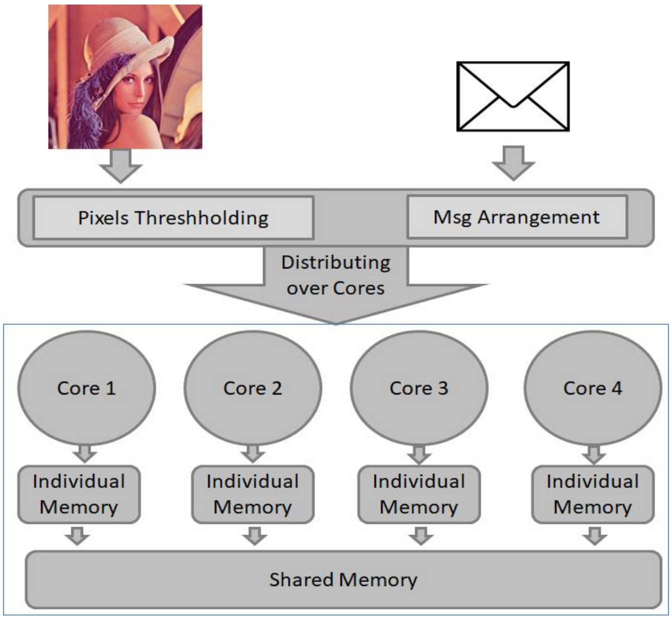


Figure 5.5 Distribution of Independent THBS Steps among Multiple Cores

5.5.1 Encoding Process

Encoding process in efficient THBS is as follows.

- Start by reading an image. After that, message to be hidden is read and converted into binary. Total length of image pixels is calculated.
- According to the values of each pixel they are divided into four categories 0-31, 32-63, 64-127, and 128-255.
- On the basis of color intensity it identifies their capacity levels and this step is completed for all the image pixels.
- Length of each pixel category is calculated.
- Once the capacity of each pixel and length of each category is known, bits from binary stream are embedded based on THBS mechanism inside image pixels independently using LSB1, LSB2, LSB3 and LSB4.

The identification of pixel’s category and defining its capacity allows the execution of parallel steganography process as illustrated in Figure 5.6. The removal of data dependency makes the steganography process efficient and independent allowing to obtain maximum benefit of parallel processing along with ensuring adequate security level.

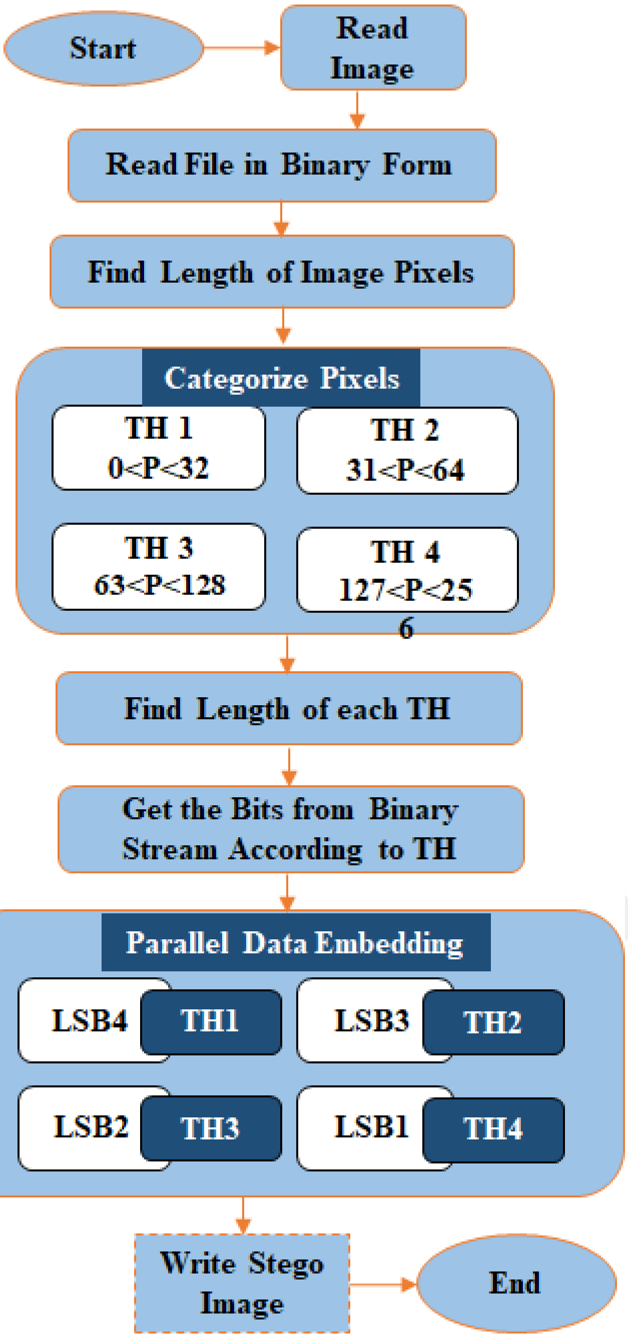


Figure 5.6 ETHBS LSB Image Steganography Encoding Process

5.5.2 Extraction Process

On the receiving end, the image with hidden data is received. The data extraction process starts by reading image signal and finding length of image pixels. For all the pixels their category is identified based on their values. The total length of each category is identified and according to the category of pixel, data bits are recovered from the LSBs of image pixels. The extraction process is parallel and efficient because dependencies have been removed in threshold value calculation step as illustrated in Figure 5.7. The data extracted is saved in another file and the process completes efficiently.

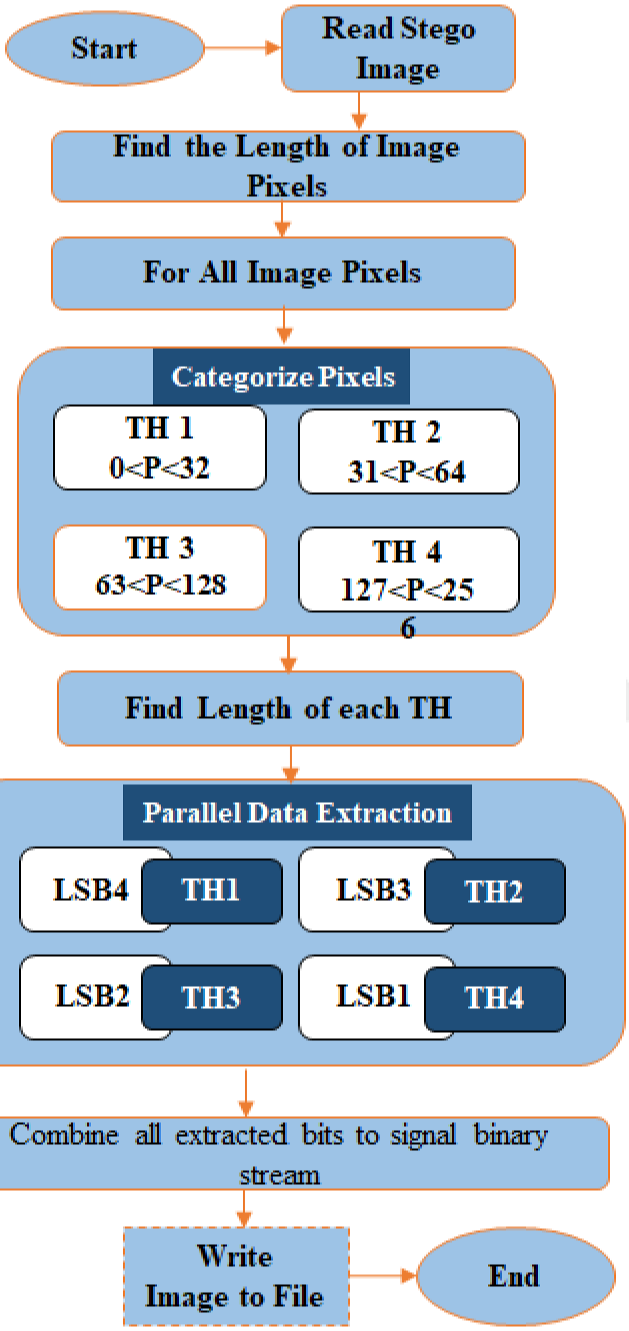


Figure 5.7 ETHBS LSB Image Steganography Extraction Process

5.5.3 Experimental Results

The proposed ETHBS technique is also implemented in Matlab 2016a on Intel core i5 with 4 cores and using 8 GB RAM. The images selected are Gray level Lena of 512 x 512; Colored Lena of 512 x 512 x 3; Gray level Baboon 512 x 512; and Colored Baboon i.e. 512 x 512 x 3. For examining the effect of proposed efficient mechanism on different size image size of test images are varied between 64 x 64, 128 x 128, 256 x 256 and 512 x 512. The experimental results of the proposed parallel technique show the noticeable efficiency improvement as compared to existing technique. Experiments were performed on different images and results of four standard images with variable sizes are elaborated that are widely used among the communities of steganography. In the case of efficient Threshold based LSB image steganography, the efficiency of proposed technique is measured as the ratio between proposed parallel execution runtime and THSB execution runtime as shown in Equation 5.7.

$$Efficiency = \frac{THBS\ Runtime}{ETHBS\ Runtime} \qquad (5.7)$$

Table 5.3 THBS vs. Proposed Technique Encoding Run-time Comparison

Image Name	Image Size	THBS Runtime (Sec)	Efficient Runtime	Improvement (Sec)	Efficiency %
Baboon Gray	64 x 64	79.25	63.28	15.97	25.23
	128 x 128	228.09	181.42	46.67	25.72
	256 x 256	934.62	745.85	188.77	25.30
	512 x 512	3.415.45	2,724.24	691.21	25.37
Baboon Color	64 x 64	162.72	129.62	33.10	25.53
	128 x 128	598.74	478.43	120.31	25.14
	256 x 256	2,104	1,666.03	437.97	26.28
	512 x 512	6,483.8	5,186.04	1,297.76	25.02
Lena Gray	64 x 64	48.84	38.96	9.88	25.35
	128 x 128	192.84	153.51	39.33	25.62
	256 x 256	760.69	609.68	151.01	24.76
	512 x 512	2,799.01	2,234.21	564.8	25.27
Lena Color	64 x 64	146.13	117.48	28.65	24.38
	128 x 128	577.12	457.96	119.16	26.01
	256 x 256	2,106.71	1,663.88	442.83	26.61
	512 x 512	6,364.97	5,085.94	1,279.03	25.14

Experiment results shown were performed on four different images by varying image sizes from 64 x 64, 128 x 128, 256 x 256 and 512 x 512. Table 5.3 shows the comparison results between the encoding process of the proposed Efficient-THBS and THBS mechanisms. Each experiment was performed five times for each reading and averages of the results. It shows that approximately 25% performance improvement is achieved over the existing Threshold based image steganography algorithm. Figure 5.8 shows the performance improvement of the proposed technique over the existing technique.

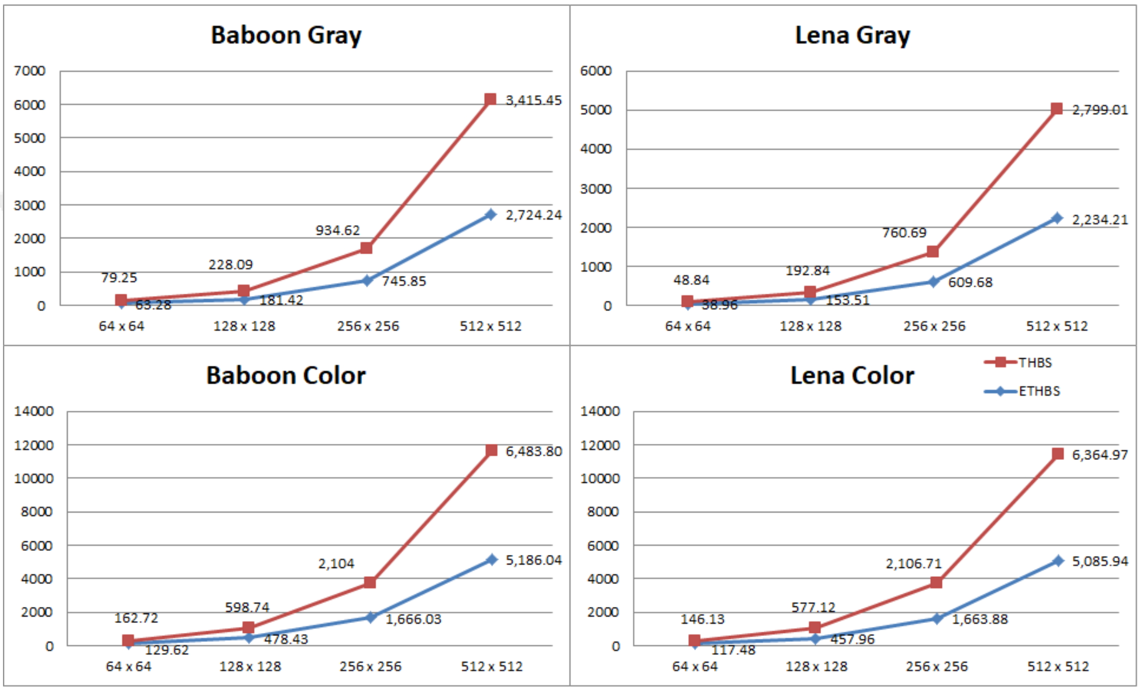


Figure 5.8 Run time Encoding Comparison between THBS and Efficient-THBS

THBS technique despite its usefulness was unable to fully utilize the parallel architecture and was reluctant to execute efficiently. This work also leads to the efficient threshold based image steganography which diminishes the serial dependency and allows the execution of process on parallel machines in proficient manner. Due to multiple levels of bits embedding in pixels this technique is robust against statistical attacks. The results exhibit that efficient THBS ensures adequate security level and payload along with exploiting efficient methodology for its execution.

6. IMAGE STEGANOGRAPHY BASED ON BLOCKS MATRIX DETERMINANT

In this chapter a new 1LSB image steganography technique is presented which is developed based on blocks matrix determinant (MDS) method. The main purpose of developing this technique is to provide maximum data security. Under MDS the image is divided into 2 x 2 blocks; and from each pixel four least significant bits are selected, these target bits are converted into decimal values. For these 4LSBs matrix determinant is calculated. If the data bit which we are hiding is 0 and the determinant of the block is even or if the data bit is 1 and the determinant of the block is odd then no change occurs. But if the data bit is 0 and determinant of the block is odd then 1 bit of a single pixel is hanged to make determinant of block even. Similarly, if data bit is 1 and determinant of block is even then change 1 bit of pixel to get determinant of the block odd. This technique results in minimal changes in image and provides better stego image quality. Proposed method is implemented in Matlab and experiments have shown that this technique is more secure and reliable when compared to other existing techniques.

If image is colored, then there are three channels, red, blue and green whereas gray scale image comprises of a single channel. After conversion of image into blocks the determinant of the block for 4LSBs of pixels are calculated. There are two cases either the determinant is even or odd. So for purpose of data embedding to and extraction from image there are two possibilities, block determinant is even there exist 0 in the block and in other case if determinant is odd there exist 1 in the block. The process of embedding and data extraction to and from image is explained along with examples as below.

The main purpose of the proposed MDS technique is to hide secret data while ensuring minimal effect on the cover image quality. The process of embedding data to the cover image and extracting data from the stego image is explained below.

6.1 Mathematical Modeling of MDS

Mathematical modeling for MDS is illustrated as following:

Let us consider *M* denotes the secret message and *C* is the cover image. Cover image is divided into blocks I_B according to the defined block size B_{size} i.e. 2 x 2. Determinant *Det(i)* of each of the block is calculated. According to the defined criteria of data bit M_{bit} vs. determinant, data bit is concealed in block to get stego block S_B, the blocks when combined result into the generation of a stego image *S*. Mathematically the process is shown below:

$$I_B = Block(B_{size}, C) \tag{6.1}$$

$$Det(i) = |I_{Bi}| \tag{6.2}$$

$$S_B = MD(Det(i), M_{bit}) \tag{6.3}$$

$$S = \sum S_B \tag{6.4}$$

On the receiving end stego image *S* is received and is divided into blocks S_B according to the defined size B_{size}. Determinant of each block of stego image is calculated. According to the determinant of corresponding block, data bit value is determined. In the last step data bits are combined to obtain original message *M* as show by following equations 6.5-6.8:

$$S_B = Block(B_{size}, S) \tag{6.5}$$

$$Det(i) = |S_{Bi}| \tag{6.6}$$

$$D_B = MD(Det_i) \tag{6.7}$$

$$M = \sum D_B \tag{6.8}$$

6.2 Encoding Process for MDS

The encoding process for the LSB image steganography based on blocks matrix determinant method follows the steps described below:

- If the cover image is a color image, then it is divided into RGB channels and read. If the cover image is a monochrome or a gray scale image, then only the single gray channel is read.
- The cover image is divided into blocks of size 2 x 2 pixels.
- For every block, the decimal value of the four LSBs of the four pixels forming the block is calculated.
- By using the decimal values calculated above, the determinant of each block is calculated.
- Since one data bit is inserted into one block, i^{th} data bit from data stream will be inserted virtually into the i^{th} block of the cover image.
- Based on the value of the determinant of the i^{th} block and i^{th} data bit, there exists four different cases of encoding.
    1. The i^{th} block determinant is even and the i^{th} bit of secret data is 0.
    2. The i^{th} block determinant is even and the i^{th} bit of secret data is 1.
    3. The i^{th} block determinant is odd and the i^{th} bit of secret data is 0.
    4. The i^{th} block determinant is odd and the i^{th} bit of secret data is 1.
- If the determinant and the data bit to be inserted are both odd or both even, then there is no need for modification of any pixel in the block. Thus, for case 1 or case 4 above, no changes are made in the pixel values of the block. When case 2 occurs, by changing pixel values of the block, the block determinant will be made odd. When case 3 occurs, by changing pixel values of the block, the block determinant will be made even.

The process of encoding is also illustrated as a flowchart in Figure 6.1.

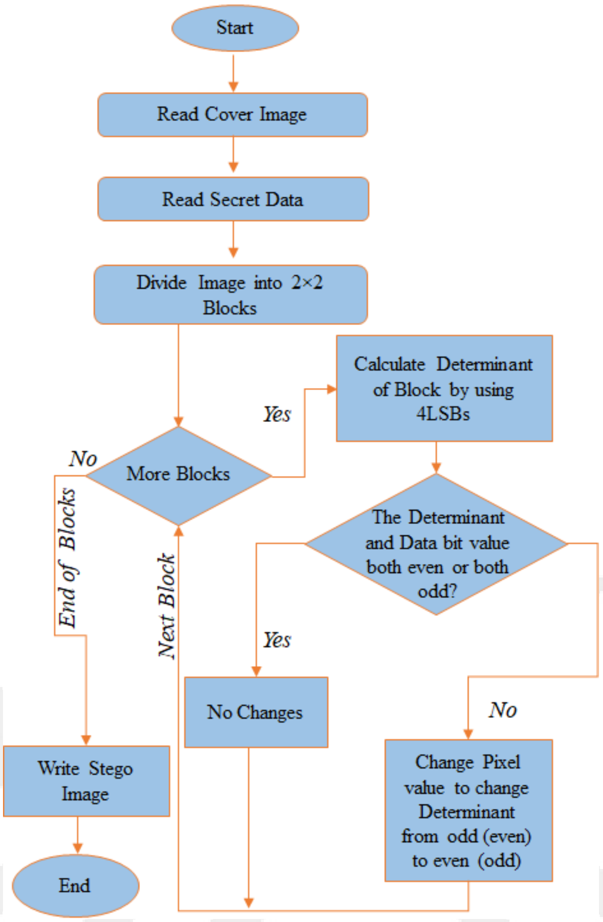

Start
Read Cover Image
Read Secret Data
Divide Image into 2×2 Blocks
More Blocks
Yes
No
Calculate Determinant of Block by using 4LSBs
The Determinant and Data bit value both even or both odd?
Yes
No
End of Blocks
Next Block
No Changes
Change Pixel value to change Determinant from odd (even) to even (odd)
Write Stego Image
End

Figure 6.1 The Flowchart for Encoding Process of MDS

The method of changing the pixel values of a block and to convert a determinant from odd to even or from even to odd is explained as follows:

Let $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ be a block of the cover image with the element values calculated from the 4LSBs of the corresponding pixels. The determinant of the block A can be calculated as shown below:

$$|A| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc \qquad (6.9)$$

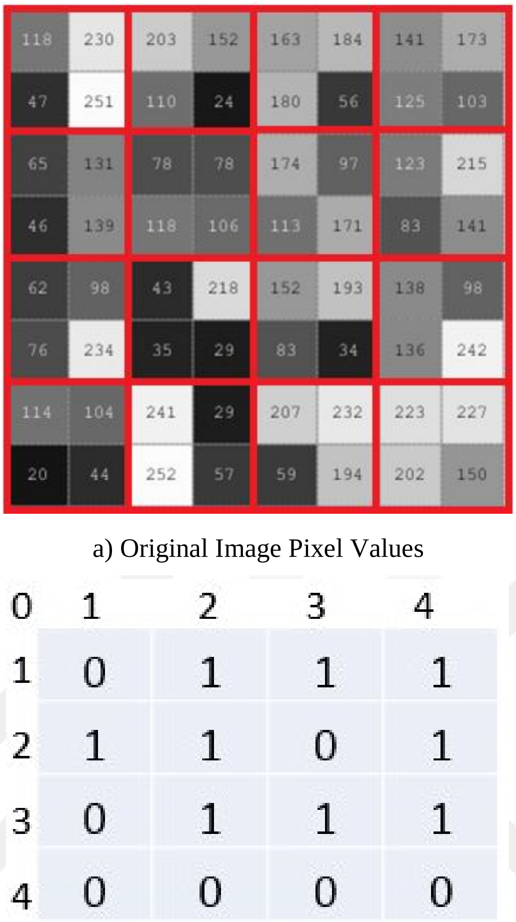
Suppose E represents an even number and O represents an odd number. Based on the elements of the block A, there are sixteen possible cases for which $|A|$ can generate even or odd as listed below [90]:

$$\begin{array}{ll}
1.\ If\ A=\begin{bmatrix}E & E\\ E & E\end{bmatrix}, then\ |A|=E & 9.\ If\ A=\begin{bmatrix}E & E\\ E & O\end{bmatrix}, then\ |A|=E\\
2.\ If\ A=\begin{bmatrix}O & E\\ E & E\end{bmatrix}, then\ |A|=E & 10.\ If\ A=\begin{bmatrix}O & E\\ E & O\end{bmatrix}, then\ |A|=O\\
3.\ If\ A=\begin{bmatrix}E & O\\ E & E\end{bmatrix}, then\ |A|=E & 11.\ If\ A=\begin{bmatrix}E & O\\ E & O\end{bmatrix}, then\ |A|=E\\
4.\ If\ A=\begin{bmatrix}O & O\\ E & E\end{bmatrix}, then\ |A|=E & 12.\ If\ A=\begin{bmatrix}O & O\\ E & O\end{bmatrix}, then\ |A|=O\\
5.\ If\ A=\begin{bmatrix}E & E\\ O & E\end{bmatrix}, then\ |A|=E & 13.\ If\ A=\begin{bmatrix}E & E\\ O & O\end{bmatrix}, then\ |A|=E\\
6.\ If\ A=\begin{bmatrix}O & E\\ O & E\end{bmatrix}, then\ |A|=E & 14.\ If\ A=\begin{bmatrix}O & E\\ O & O\end{bmatrix}, then\ |A|=O\\
7.\ If\ A=\begin{bmatrix}E & O\\ O & E\end{bmatrix}, then\ |A|=O & 15.\ If\ A=\begin{bmatrix}E & O\\ O & O\end{bmatrix}, then\ |A|=0\\
8.\ If\ A=\begin{bmatrix}O & O\\ O & E\end{bmatrix}, then\ |A|=O & 16.\ If\ A=\begin{bmatrix}O & O\\ O & O\end{bmatrix}, then\ |A|=E
\end{array} \qquad (6.10)$$

Therefore, if the determinant of the block needs to be converted from even to odd or from odd to even, then a combination from Equation 6.10 can be used. There are sixteen different combinations which result in either even or odd determinants. Out of sixteen, there are ten combinations which result into an odd determinant value and six combinations generate even determinant value.

To illustrate with the help of example, let us consider if the block is similar as case 6 of Equation 6.10, its determinant is even. But if the data bit to be inserted is 1, then the determinant of the block should be converted into odd. This can be done by modifying the block to a similar form as in case 8 or 14 of Equation 6.2. This only requires a single pixel's LSB modification inside the block. Thus, the LSB values of the pixels corresponding to matrix element c in case 8 or d in case 14 would be inverted to make the determinant odd. The selection of the case for the modification is kept random when there are multiple options ensuring minimal changes. The reliability of this technique is increased as the alteration in pixels' value of the resultant block is also kept at random. In addition, a change in pixel value is kept random as +1 or -1 which makes it more complex and challenging to break. This data concealment method results in a secure and a reliable steganographic mechanism which helps to improve the complexity of MDS technique and makes it more difficult for the intruders to notice the concealed data or to detect the original data from stego images.

The encoding method is elaborated with the help of following example. Let us consider an 8 x 8 monochrome cover image with the pixel values as shown in Figure 6.2.a. and the data to be embedded into the cover image is shown in Figure 6.2.b. The cover image is divided into blocks of size 2 x 2 pixels as shown in Figure 6.2.a. The process of embedding secret message bits for three different blocks, first block (block (1,1)), second block (block (1,2)) and third block (block (1,3)) respectively are as follows:



118	230	203	152	163	184	141	173
47	251	110	24	180	56	125	103
65	131	78	78	174	97	123	215
46	139	118	106	113	171	83	141
62	98	43	218	152	193	138	98
76	234	35	29	83	34	136	242
114	104	241	29	207	232	223	227
20	44	252	57	59	194	202	150

a) Original Image Pixel Values

0	1	2	3	4
1	0	1	1	1
2	1	1	0	1
3	0	1	1	1
4	0	0	0	0

b) Secret Message Bits

Figure 6.2 Data Encoding Example

A. Encoding for Block (1,1)

- The pixel values for the first block are 118, 230, 47 and 251. When represented in binary these values are 0111$\underline{0110}$, 1110$\underline{0110}$, 0010$\underline{1111}$ and 1111$\underline{1011}$ respectively. Thus, the block matrix for the first block is obtained by mining the 4LSBs from each pixel and represented as shown in Equation 6.11.

$$A = \begin{bmatrix} 6 & 6 \\ 15 & 11 \end{bmatrix} \tag{6.11}$$

The determinant for this block is even, as shown in Equation 6.12 below:

$$|A| = \begin{vmatrix} 6 & 6 \\ 15 & 11 \end{vmatrix} = 6 \times 11 - 6 \times 15 = -24 \tag{6.12}$$

As the determinant of the block is even and first secret data bit to be inserted is a 0, no changes will occur in this block. As a consequence, the data bit into first block is concealed into the cover image without changing a single pixel value of the block. Thus, for this block cover image and stego image pixel values will be the same.

B. Encoding for Block (1,2)

- The pixel values for the second block are 203, 152, 110 and 24. When represented in binary these values are 11001011, 10011000, 01101110 and 00011000 respectively. Thus, the block matrix for the first block is obtained by mining the 4LSBs from each pixel and represented as shown in Equation 6.13.

$$A = \begin{bmatrix} 11 & 8 \\ 14 & 8 \end{bmatrix} \tag{6.13}$$

The determinant for this block is even, as shown in Equation 6.14 below:

$$|A| = \begin{vmatrix} 11 & 8 \\ 14 & 8 \end{vmatrix} = 11 \times 8 - 14 \times 8 = -24 \tag{6.14}$$

As the determinant of the block is even and second secret data bit to be inserted is a 1, so change will occur in this block.

The form of the block matches with case 2 of Equation 6.10. To make the determinant odd, this block would be converted into case 10 of Eq. 6.10 by changing only one bit in the block. If the LSB of the last pixel is inverted, the block matrix of the third block will take the form as shown in Equation 6.15 below:

$$A = \begin{bmatrix} 11 & 8 \\ 14 & 9 \end{bmatrix} \tag{6.15}$$

Now, the determinant for this block is odd, as shown in Equation 6.16, thus giving the information that the inserted data bit into this block is a 1.

$$|A| = \begin{vmatrix} 11 & 8 \\ 14 & 9 \end{vmatrix} = 11 \times 9 - 8 \times 14 = -13 \tag{6.16}$$

In the stego image, the pixel values for the second block will be 203, 152, 110 and 25.

C. Encoding for Block (1,3)

- The pixel values for the third block are 163, 184, 180 and 56. When represented in binary these values are 10100011, 10111000, 10110100, 00111000 respectively. Thus, the block matrix for the third block is obtained by mining the 4LSBs from each pixel and represented as shown in Equation 6.17.

$$A = \begin{bmatrix} 3 & 8 \\ 4 & 8 \end{bmatrix} \tag{6.17}$$

The determinant for this block is even, as shown in Equation 6.18 below:

$$|A| = \begin{vmatrix} 3 & 8 \\ 4 & 8 \end{vmatrix} = 3 \times 8 - 8 \times 4 = -8 \tag{6.18}$$

For the third block, secret data bit to be inserted is 1 but determinant is even. The form of the block matches with case 2 of Equation 6.10. To make the determinant odd, this block would be converted into case 10 of Eq. 6.10 by changing only one bit in the block. If the LSB of the last pixel is inverted, the block matrix of the third block will take the form as shown in Equation 6.19 below:

$$A = \begin{bmatrix} 3 & 8 \\ 4 & 9 \end{bmatrix} \tag{6.19}$$

Now, the determinant for this block is odd, as shown in Equation 6.20, thus giving the information that the inserted data bit into this block is a 1.

$$|A| = \begin{vmatrix} 3 & 8 \\ 4 & 9 \end{vmatrix} = 3 \times 9 - 8 \times 4 = -5 \tag{6.20}$$

In the stego image, the pixel values for the third block will be 163, 184, 180 and 57.

To conclude the above example, when data bit and block determinant values are in accordance with each other no change is required but still one data bit is embedded in the block and when they are not in accordance with each other; change is made in block in a logical and calculated manner which ensure minimal change in pixels during data encoding. Thus, at larger scale it can be anticipated that in MDS to conceal a data bit in a single block the chance of modifying pixel's value is 50%, resulting in minimal effect and better quality of stego image.

6.2 Decoding Process for MDS

Once the stego image is received, the decoding process for the LSB image steganography based on blocks matrix determinant method follows the steps described below:

- If the stego image is a color image, then it is divided into RGB channels and read. If the stego image is a monochrome or a gray scale image, then only the single gray channel is read.
- The stego image is divided into blocks of size 2 x 2 pixels.
- For every block in the stego image, the decimal value of the four LSBs of the four pixels forming the block is calculated.
- By using the decimal values calculated above, the determinant of each block is calculated, and data bits are extracted.
    1. If the i^{th} block determinant is even, then the i^{th} bit of the secret data is 0.
    2. If the i^{th} block determinant is odd, then the i^{th} bit of secret data is 1.
- All extracted data bits are concatenated to form the secret data.

The process of decoding is also illustrated in Figure 6.3.

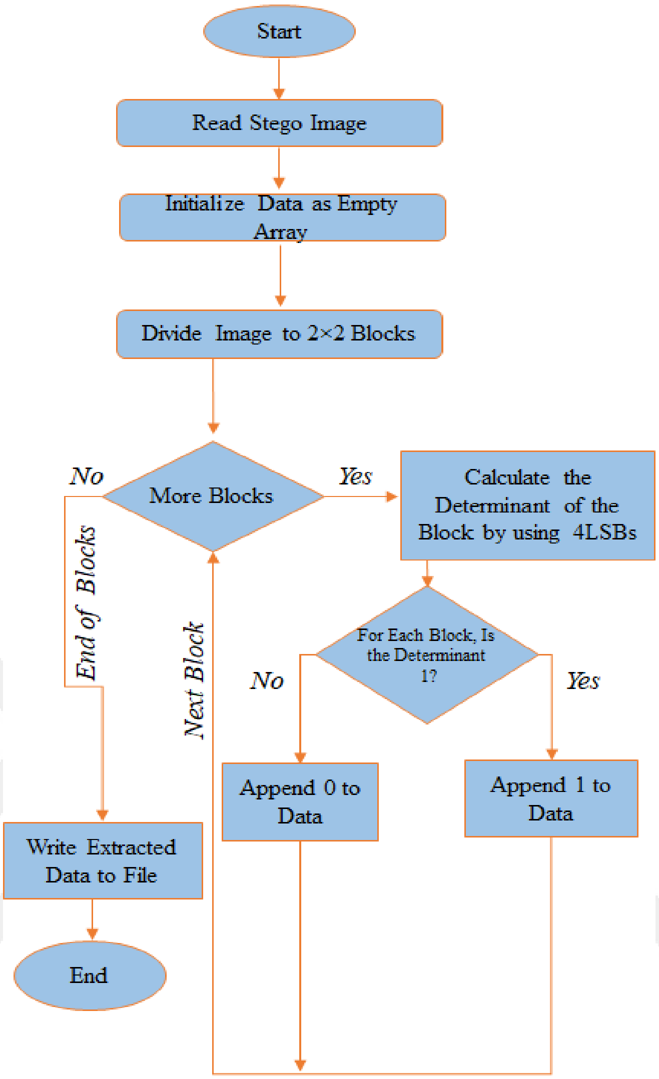

Start
Read Stego Image
Initialize Data as Empty Array
Divide Image to 2×2 Blocks
More Blocks
No
Yes
Calculate the Determinant of the Block by using 4LSBs
End of Blocks
Next Block
For Each Block, Is the Determinant 1?
No
Yes
Append 0 to Data
Append 1 to Data
Write Extracted Data to File
End

Figure 6.3 The Flowchart for Decoding Process of MDS

Data decoding process is also explained for three blocks is as follows:

A. Data Decoding for Block (1,1)

The received Block (1, 1) = $\begin{bmatrix} 118 & 230 \\ 47 & 251 \end{bmatrix}$

- 4LSB value is mined from each pixel

Block (1, 1) = $\begin{bmatrix} 118 & 230 \\ 47 & 251 \end{bmatrix} \Rightarrow \begin{bmatrix} 6 & 6 \\ 15 & 11 \end{bmatrix}$

- Determinant for block (1,1) is calculated

$\begin{vmatrix} 6 & 6 \\ 15 & 11 \end{vmatrix}$

⇨ (6 x 11 – 6 x 15)

⇨ (66-90)

⇨ -24

⇨ Even

Calculated determinant of block (1, 1) is even and according to data encoding scheme data bit hidden in even block is 0 so from block(1,1) data bit extracted is 0. Data bit is extracted and written in message file.

B. Data Decoding for Block (1,2)

Example of another block in which change in pixel value was made during encoding is as follows for block (1, 3).

Block (1,3) = $\begin{bmatrix} 203 & 152 \\ 110 & 24 \end{bmatrix}$

- 4LSB value is mined from each pixel

Block (1,3) = $\begin{bmatrix} 203 & 152 \\ 110 & 24 \end{bmatrix} \Rightarrow \begin{bmatrix} 11 & 8 \\ 14 & 9 \end{bmatrix}$

- Determinant is calculated

$\begin{vmatrix} 11 & 8 \\ 14 & 9 \end{vmatrix}$

⇨ (11 x 9 – 8 x 14)

⇨ (99 - 112)

⇨ -13

⇨ Odd

- Calculated determinant of block (1, 2) is odd and according to secret data encoding scheme, data bit to be extracted from block(1,2) is 1.

C. Data Decoding for Block (1,3)

Example of another block in which change in pixel value was made during encoding is as follows for block (1, 3).

Block (1,3) = $\begin{bmatrix} 163 & 184 \\ 180 & 57 \end{bmatrix}$

- 4LSB value is mined from each pixel

  Block (1,3) = $\begin{bmatrix} 163 & 184 \\ 180 & 57 \end{bmatrix} \Rightarrow \begin{bmatrix} 3 & 8 \\ 4 & 9 \end{bmatrix}$

- Determinant is calculated

  $\begin{vmatrix} 3 & 8 \\ 4 & 9 \end{vmatrix}$

  ⇨ (3 x 9 – 8 x 4)

  ⇨ (27 - 32)

  ⇨ -5

  ⇨ Odd

- Calculated determinant of block (1, 3) is odd and according to secret data encoding scheme, data bit to be extracted from block(1,3) is 1.

When encoding data bit in a single pixel the chance of changing pixel value in block is 50%. When data bit and block determinant are in accordance with each other no change is required but still 1 bit data is embedded in block. On receiving end it is defined that block having determinant even extracts 0 whereas block with determinant odd extracts 1.

6.4 Experimental Results

Experiments were performed on fifty standard images of size 512 x 512 were selected; results for four standard images are selected for analysis and elaboration of implementation of proposed technique. From colored images Lena color and Baboon color images are used, whereas from gray scale images Lena gray and Baboon gray images are selected. Matlab 2016a on Intel core i5 with 4 cores and using 8 GB RAM is used for implementation of the proposed technique. It is obvious from stego images quality that proposed technique is efficient and secure.

Table 6.1 Results for Baboon Stego Gray Image

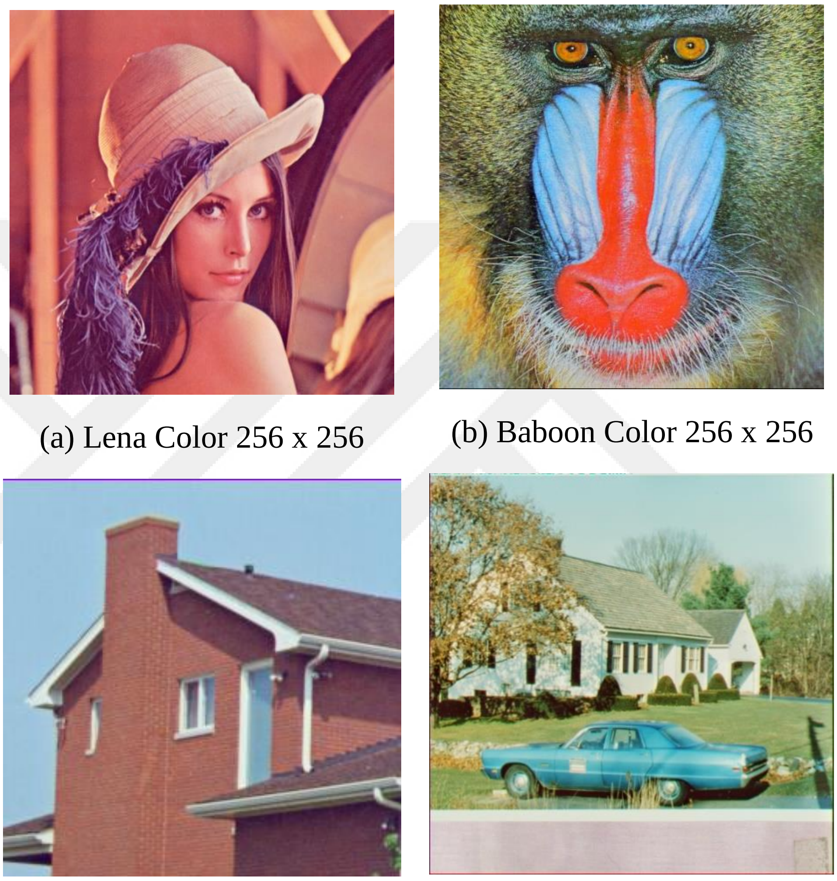
	Lena Stego Gray Image				Baboon Stego Gray Image			
Techniques	MSE	RMSE	PSNR (dB)	Payload (Bits)	MSE	RMSE	PSNR (dB)	Payload (Bits)
Ni et al. [61]	0.98	0.98	48.2	5,460	0.98	0.98	48.2	5,421
Hwang et al.[62]	0.97	0.98	48.22	5,336	0.97	0.98	48.22	5,208
Lin et al.[63]	1.42	1.19	46.6	59,900	1.12	1.05	47.61	19,130
Hu et al. [64]	0.87	0.93	48.69	60,241	0.95	0.97	48.34	21,411
Luo et al. [65]	0.85	0.92	48.82	71,674	0.34	0.58	48.36	22,696
Wu and Tsai. [66]	4.30	2.07	41.79	50,960	3.25	1.80	37.9	56,291
Vleeschouwer et al. [67]	8.18	2.86	39	24,108	8.18	2.86	39	2,905
Xuan et al. [68]	14.22	3.77	36.6	85,507	34.12	5.84	32.8	14,916
Celik et al. [69]	10.30	3.20	38	74,600	10.30	3.20	38	15,176
LSB4	40.88	6.39	32.05	1,048,576	39.76	6.30	32.16	1,048,576
Shehzad et al. [70]	19.82	4.45	35.19	524,288	21.10	4.59	34.92	524,288
THBS	2.84	1.64	43.64	433,224	2.33	1.54	44.38	410,636
MDS	**0.12**	**0.34**	**57.21**	**65,536**	**0.12**	**0.34**	**57.19**	**65,536**

Table 6.1 shows comparison of proposed technique with other existing techniques. Maximum difference for this technique per pixel is 1. The results show that for gray scale images proposed technique has higher PSNR while errors as MSE and RMSE in image are minimal for MDS technique.

Table 6.2 Results for Lena and Baboon Stego Color Images

	Lena Stego Color Image				Baboon Stego Color Image			
Techniques	MSE	RMSE	PSNR (dB)	Payload (Bits)	MSE	RMSE	PSNR (dB)	Payload (Bits)
Yalman et al. [50]	7.19	2.68	39.56	1,156,000	7.12	2.66	39.6	1,156,000
Shehzad et al. [70]	9.09	3.01	38.57	1,572,864	9.54	3.08	38.36	1,572,864
THBS	3.24	1.77	42.32	1,355,199	1.22	1.13	47.15	1,329,318
MDS	**0.09**	**0.3**	**58.19**	**196,608**	**0.09**	**0.3**	**57.49**	**196,608**

Tables 6.2 show comparison for Baboon and Lena colored images for proposed technique with other existing techniques including THBS and SOBP. The results show that proposed technique has higher PSNR and other key metrics values while MSE, RMSE and error in image is minimal. This technique is a powerful technique which has high quality stego image and is secure than all other existing LSB image steganography techniques.



(a) Lena Color 256 x 256

(b) Baboon Color 256 x 256

(c) Building Colore 256 x 256

(d) House Colore 256 x 256

Figure 6.4 Cover Images used for 1LSB Techniques Evaluation

To illustrate the effectiveness of MDS, experiments were also conducted by embedding fixed size data in different 1LSB image steganography techniques. A secret data of 6 KB is embedded in four different color images Baboon, Lena, Building and House of size 256 x 256 pixels and the results for stego images are compared with only 1LSB steganography techniques as shown in figure 6.4. For the evaluation of MDS encoding mechanism, when PSNRs are compared; MDS has the highest PSNR as shown in Table 6.3.

Table 6.3 Comparison of MDS with 1LSB Methods Based on PSNR (dB) by Hiding 6KB Data in (256 x 256 pixels sized) Color Images

Image	Classic 1LSB Method	[91]	[92]	[93]	[94]	[95]	**MDS**
Baboon	51.46	49.13	46.53	40.06	49.37	54.17	**58.26**
Lena	45.61	45.61	49.20	40.26	45.61	52.38	**57.55**
Building	49.84	46.72	46.82	40.30	48.34	52.37	**57.49**
House	51.48	51.48	47.68	40.27	51.47	52.37	**56.84**

Results for PSNR are shown graphically for Building, House, Baboon and Lena color images in Figure 6.5.

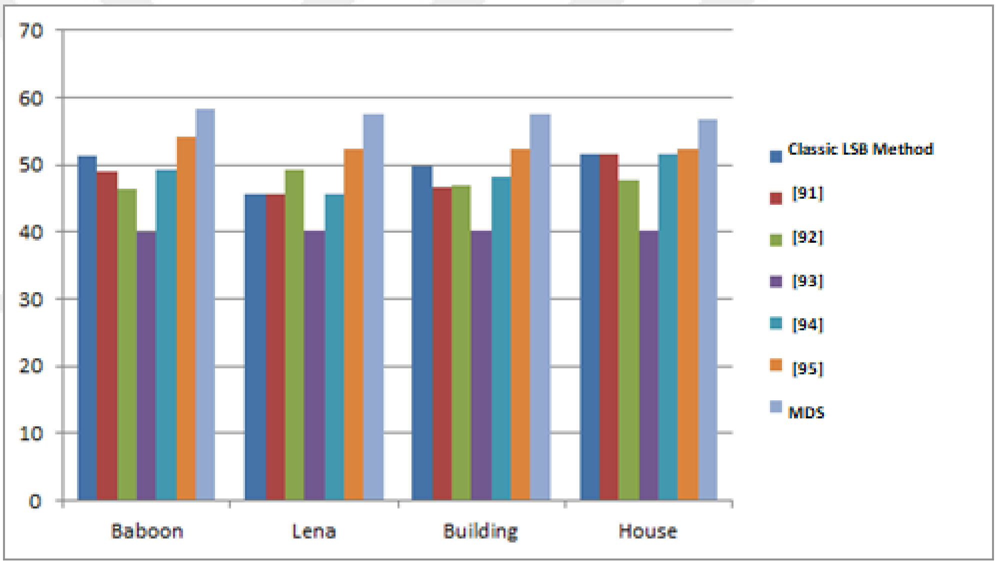


Figure 6.5 1LSB Techniques Comparison Based on PSNR (dB) by Hiding 6KB Data in (256 x 256 pixels sized) Color Images

The MSE values are also calculated after hiding 6 KB of data in these images. Minimal values for MSE show that MDS produces minimal changes/error in stego images as compared to other 1LSB images steganography methods as shown in Table 6.4.

Table 6.4 Comparison of MDS with 1LSB Methods Based on MSE by Hiding 6KB Data in (256 x 256 pixels sized) Color Images

Image	Classic 1LSB Method	[91]	[92]	[93]	[94]	[95]	**MDS**
Baboon	0.46	0.79	1.45	6.41	0.75	0.25	0.097
Lena	1.78	1.78	0.78	6.12	1.79	0.38	0.114
Building	0.67	1.38	1.35	6.07	0.95	0.38	0.116
House	0.42	0.46	1.11	6.11	0.46	0.38	0.135

The results for PSNR and MSE certify high security and stego image quality prominence of MDS as compared to existing 1LSB image steganography techniques.

6.4.1 Qualitative Analysis

In this sub-section, qualitative analysis is done for MDS. The visual quality of proposed technique is shown by comparing original images with stego images. Histogram comparison shows that for original image and stego image pixel values are almost same and distortion is not noticeable despite having data inside stego image, depicting the prominence of developed technique. The comparison of original and stego images and their histograms are shown in Figures 6.6-6.9.

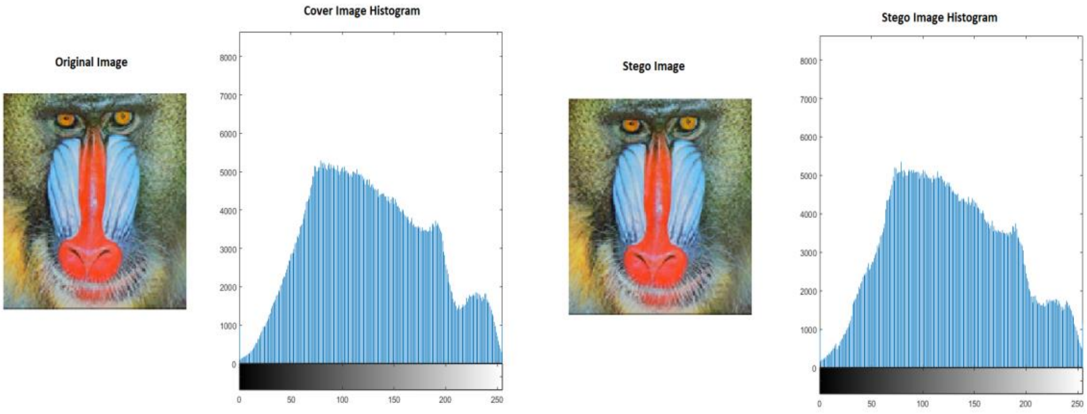


Figure 6.6 Qualitative Comparison for Baboon Color Original Image vs. Stego Image with Dimensions (512x512) and Histograms

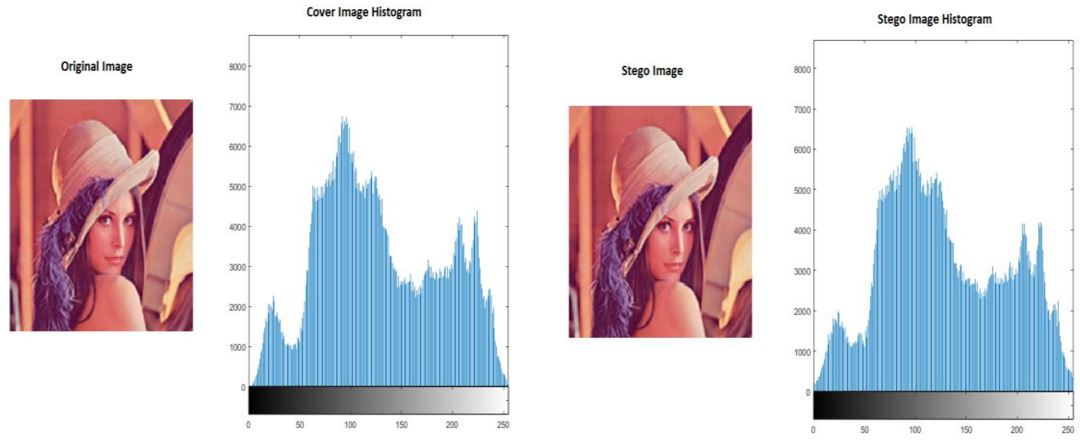


Figure 6.7 Qualitative Comparison for Lena Color Original Image vs. Stego Image with Dimensions (512x512) and their Histograms

Figures 6.6 and 6.7 shows the comparison between original and stego 512x512 pixels sized Baboon and Lena colored images. 24 KB data is stored inside original images using MDS encoding method. For the Baboon stego image generated by application of MDS encoding method PSNR of 58.19 dB and MSE of 0.09 was calculated. Whereas, for Lena stego image PSNR of 57.49 dB and MSE of 0.09 was noticed. The visual comparison between images and pixel by pixel comparison for original images and stego images using histogram shows that data embedment in these colored images did not have any noticeable affect the pixel values of original cover images confirming the quality of MDS method.

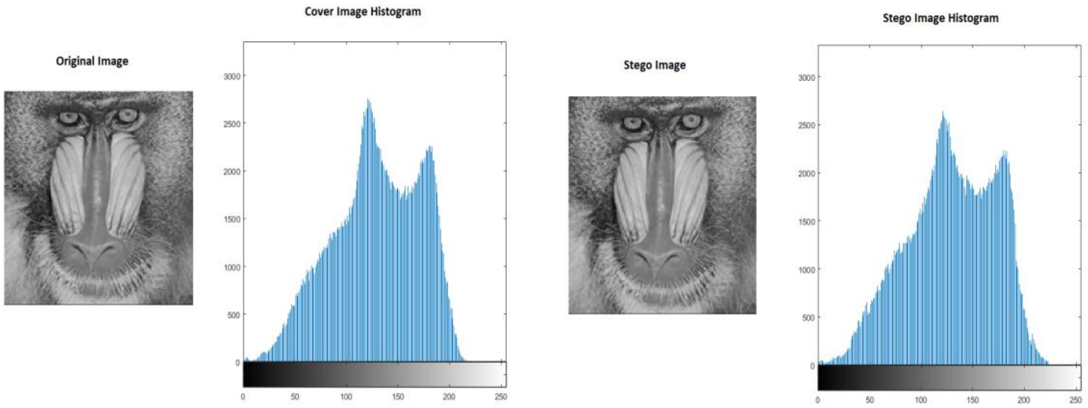


Figure 6.8 Qualitative Comparison for Baboon Gray Cover Image vs. Stego Image with Dimensions (512x512) and Histograms

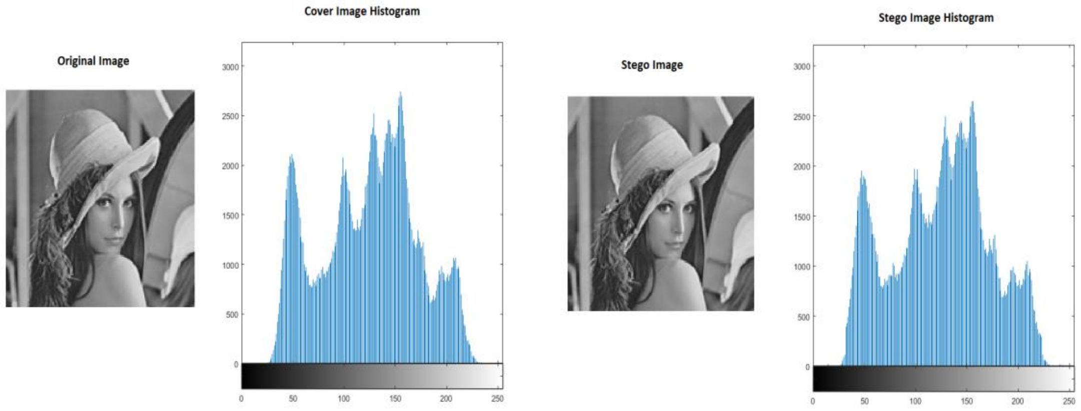


Figure 6.9 Qualitative Comparison for Lena Gray Cover Image vs. Stego Image with Dimensions (512x512) and Histograms

Histogram analysis were also performed on 512x512 pixels sized Baboon and Lena gray images as shown in Figures 6.8 and 6.9. Data of 8KB was embedded in these images. PSNR of 57.19 dB and 0.12 MSE were noted for Baboon stego image, while PSNR for Lena stego image was 56.98 dB and MSE calculated was 0.12. Histogram comparison in Figures 6.7 and 6.8 shows almost no change in the pixel values before and after data embedding in gray images.

The quality of above stego images and their histograms in comparison with original cover images show that developed MDS technique hides data inside image without affecting the image or if there is effect on image quality it is not noticeable. This certifies good stego image quality and high security of developed technique.

MDS is more secure and reliable than other prevailing techniques. This technique is robust against statistical attacks where usually every pixel is observed for a specific pattern, while in this technique for every four pixels only one bit of data is concealed. Also it does not change the value of LSB almost 50% of chances and still embed one data bit. Against image manipulation MDS is also resilient where if during manipulation if image pixels are not lost then MDS generated stego image has no effect on its quality. This makes this technique unique and versatile. Experiments have shown that this technique has high SNR, PSNR and low MSE and is more secure and reliable as compared to other techniques.

7. COMPARATIVE PERFORMANCE EVALUATION

In this chapter comparative analysis of all the three developed techniques is done along with LSB4 classical method. The developed techniques are compared based on their payload, imperceptibility and security. For the comparison of imperceptibility of techniques comparisons are done usually based on MSE, RMSE and PSNR but for better illustration here nine different image performance metrics are evaluated. For the security analysis techniques are evaluated using two types of statistical attacks histogram stegnalysis and RS stegnalysis.

7.1 Payload and Imperceptibility

Payload and imperceptibility are inversely proportional to each other. As more and more data is concealed inside image the steganographic technique become more perceptible. For imperceptibility comparison nine different image performance metrics are considered to be evaluated for image quality and errors; which include MSE, RMSE, SNR, MAE, PSNR, NAE, MD, AD and UIQI which are discussed in detail in section 3.1. Usually only PSNR and MSE are calculated for evaluation of imperceptibility, but for better comparative analysis in this dissertation we compared all the possible parameters for imperceptibility evaluation of these techniques. MD and AD gives the average and maximum difference between pixel values. MSE, RMSE, MAE and NAE are error values generated due to data concealment; whereas SNR and PSNR gives signal to noise ratio that helps to understand the effect of data concealment on image quality. Tables 7.1 and 7.2 gives the comparison for these parameter values for generated color stego images generated by these techniques along with classical LSB4 results.

Table 7.1 Results for Baboon Stego Color Image

Technique	MSE	RMSE	UIQI	SNR (dB)	MAE	AD	MD	NAE	PSNR (dB)	Payload (bits)
LSB4	18.40	4.29	0.9	8.04	5.15	0.47	15	0.04	32.14	3,145,728
SOBP	9.54	3.08	0.86	14.26	2.14	0.4	15	0.01	38.36	1,572,864
THBS	1.27	1.12	0.89	22.99	0.8	0.13	14	0.006	47.10	1,329,218
MDS	0.09	0.3	0.99	34.09	0.09	0.002	1.0	0.0007	58.70	196,608

Tables 7.1 and 7.2 shows that Payload for LSB4 is highest as compared to our developed techniques but in terms of image quality and imperceptibility it has least values. Whereas in the developed techniques MDS has higher stego image quality and high imperceptibility as compared to other two techniques because in SOBP two bits are embedded per pixel whereas in THBS data bits per pixel varies between one to four depending upon category of each pixel. MDS being 1LSB technique and storing single bit of data per block has the lowest data hiding capacity among these three developed techniques.

Table 7.2 Results for Lena Stego Color Image

Technique	MSE	RMSE	UIQI	SNR (dB)	MAE	AD	MD	NAE	PSNR (dB)	Payload (bits)
LSB4	18.42	4.29	0.75	11.4	3.47	0.5	15	0.03	35.48	3,145,728
SOBP	9.09	3.04	0.84	14.47	2.09	0.3	15	0.01	38.57	1,572,864
THBS	3.24	1.77	0.96	19	1.1	0.2	11	0.008	42.32	1,355,199
MDS	0.09	0.3	0.99	34.25	0.12	0.003	1.0	0.0008	58.35	196,608

Tables 7.1 and 7.2 also shows that maximum difference in SOBP like LSB4 generated images is 15 whereas in THBS generated image maximum difference between original and stego Lena image pixel is 11 whereas in Baboon stego image is 14. This difference is due to the flexibility in variation of data bits embedment depending upon THBS defined pixel categories. Also the variation among the parameter values for the stego image generated by same technique show that effect of data concealment vary from image to image depending upon the internal frame of an image. MDS has maximum difference (which is least as compared to other three techniques) of one bit in a single pixel. Stego images generated by MDS have least MSE, MAE and NAE errors and

highest UIQI, SNR, PSNR values depicting the high imperceptibility prominence of MDS among these three developed techniques.

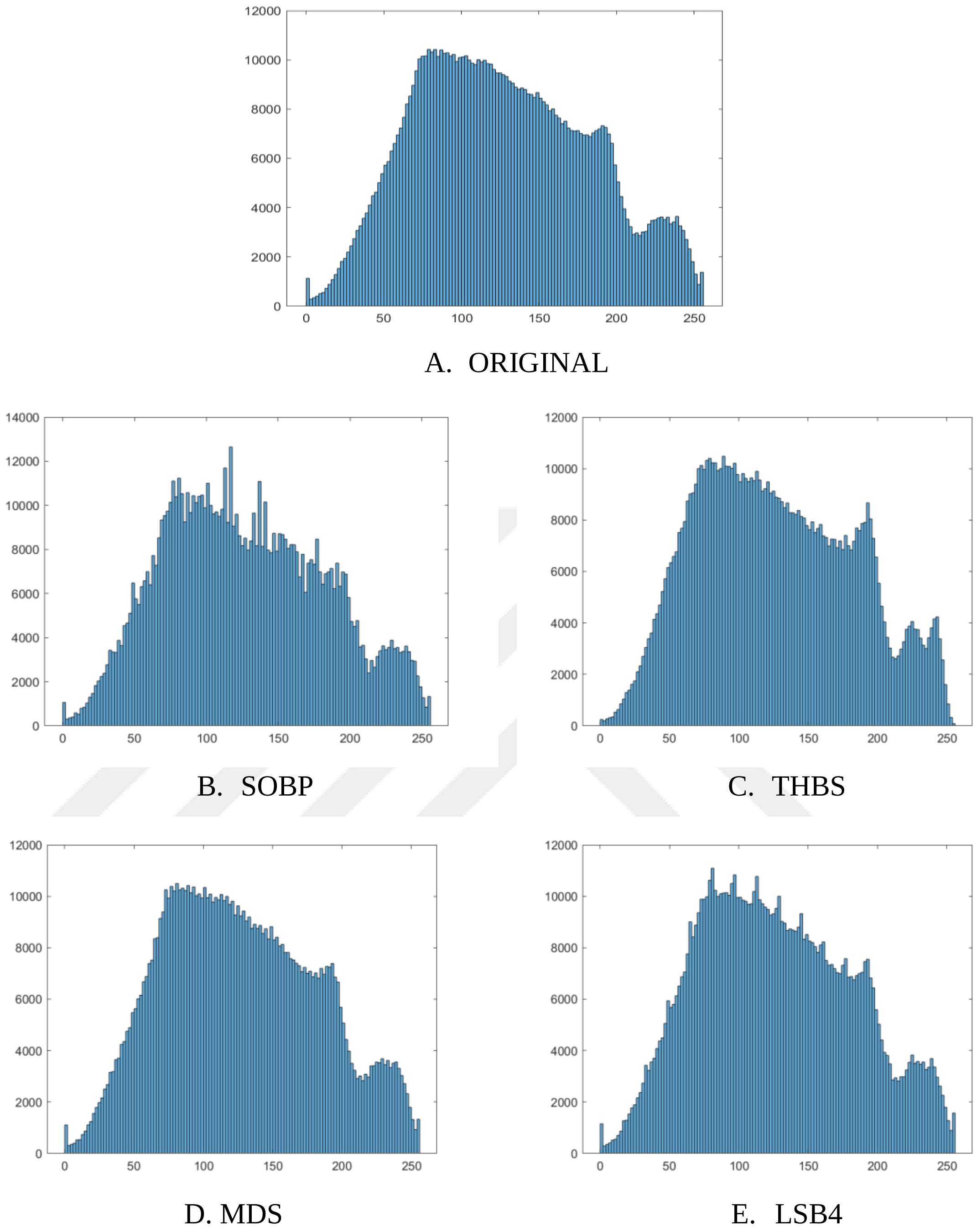
7.2 Security Analysis

A steganographic technique is said to be secure if along with being imperceptible and remaining unnoticeable it can also resist various statistical attacks that can unleash the data concealment. For the security analysis of developed techniques histogram analysis and RS stegnalysis are performed.

7.2.1 Histogram Analysis

Histogram analysis is one of the significant types of stegnalysis where pixel by pixel comparison depicts the quality of a steganographic technique. In this subsection, a qualitative security analysis is presented for all three developed techniques along with LSB4 where fix amount of data is concealed using these techniques in Baboon and Lena color images. The qualitative comparison in this section shows that for the original image and the stego image pixel values are almost the same for MDS generated images and the distortion is not noticeable depicting the high quality of MDS security.

For a 24-bit color image, 256 different intensities for each of the 3 channels (red, green, blue) are possible. Therefore, a histogram for each channel can be drawn separately, or an average histogram of all channels can be produced to analyze the difference between histogram of original image and histogram of stego image to determine the effect of data concealment on pixel intensities by a steganographic technique. Here in this chapter to have better evidence the latter is used where average histogram of all channels is shown. Histogram comparison is done for stego images generated by the application of all three developed techniques and LSB4 after embedding 196,608 bits i.e. 24KB into the original cover images as shown in figures 7.1 and 7.2.



A. ORIGINAL

B. SOBP

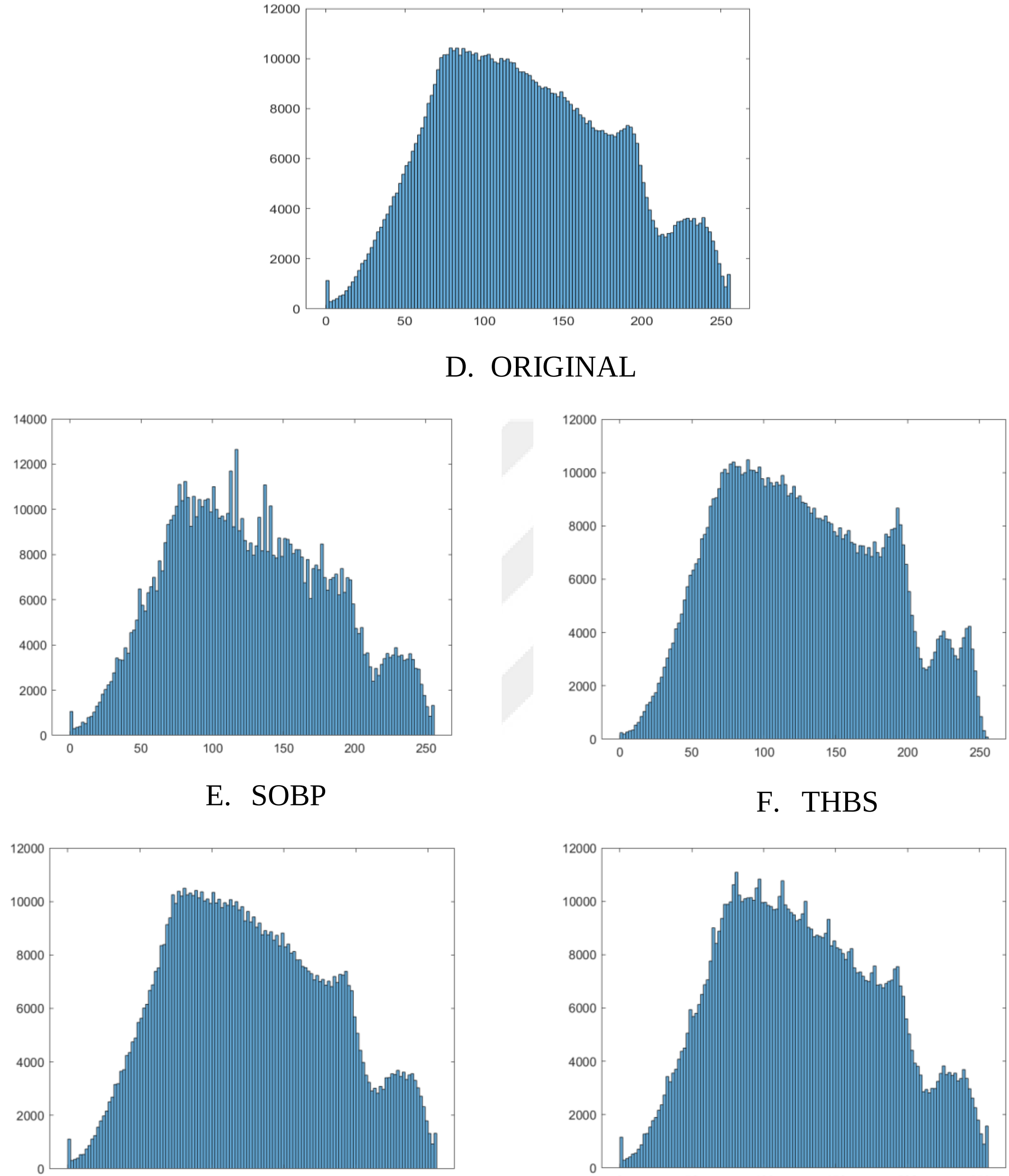
C. THBS

D. MDS

E. LSB4

Figure 7.1 Baboon Color Original and Stego Image Histograms

Figure 7.1 for Baboon 512 x 512 color image and figure 7.2 for Lena color image shows that in case of MDS generated stego image, resemblance of histograms with cover image is almost the same while the histograms of stego images generated by THBS is slightly modified whereas SOBP and LSB4 has clear and noticeable alterations in pixel values as shown in histograms. This point clearly shows the excellence of the MDS method; where same amount of data is embedded through different techniques but MDS ensures minimal change remaining more secure as compared to other steganographic methods.



D. ORIGINAL

E. SOBP

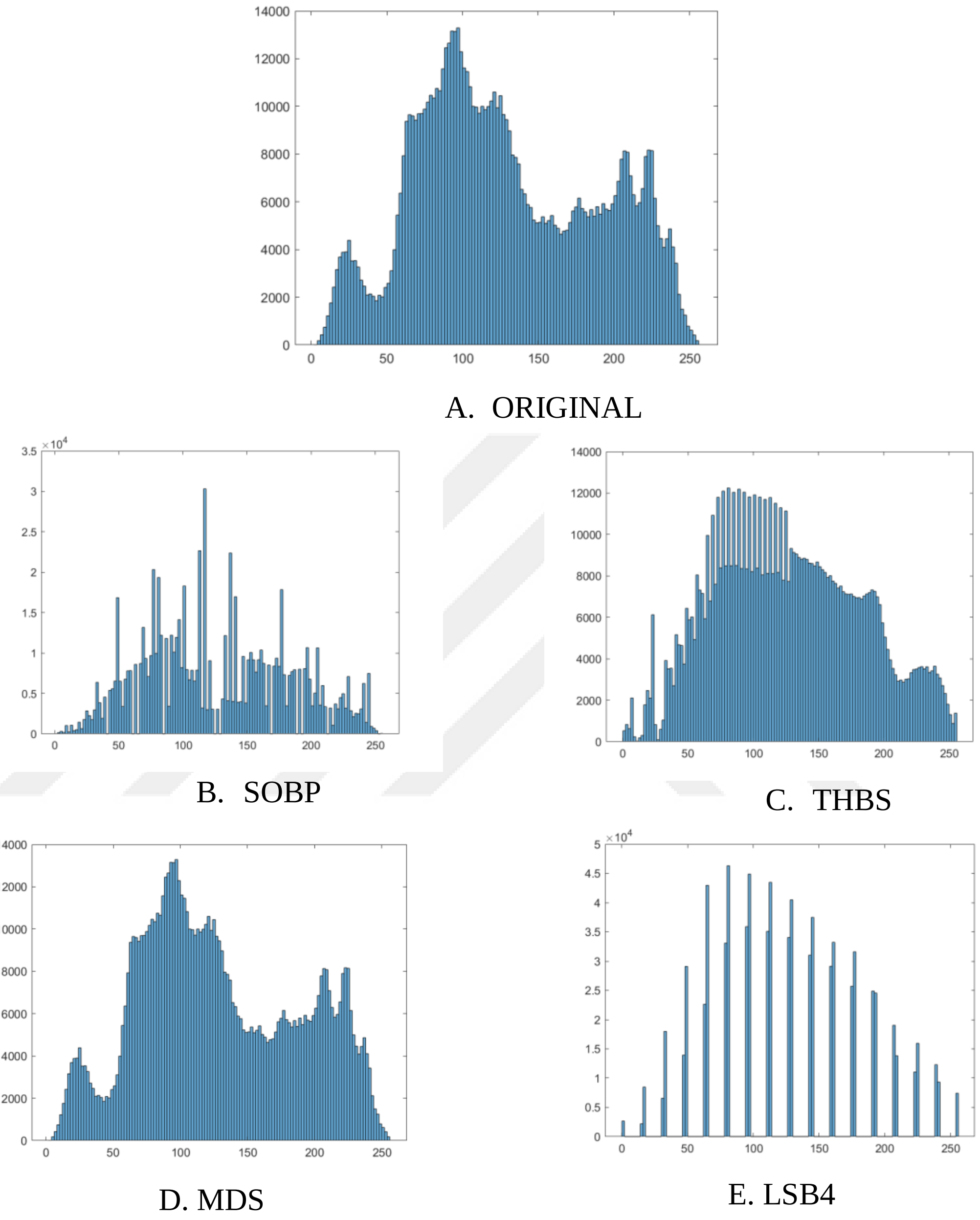
F. THBS

D. MDS

F. LSB4

Figure 7.2 Baboon Color Original and Stego Image Histograms

Histogram stegnalysis was also performed by embedding the maximum payload through developed techniques. Histograms for Baboon color original and stego images were generated for the THBS stego image having 162 KB of data, 192 KB for SOBP, for MDS 24 KB and for LSB4 384 KB data was concealed. Histogram comparison is done in figure 7.3 which clearly depicts that having the original image for all other techniques data concealment is visible and can be easily noticed. Only MDS data concealment has minimal effect on image and it does not affect image quality during data concealment.



A. ORIGINAL

B. SOBP

C. THBS

D. MDS

E. LSB4

Figure 7.3 Baboon Color Image Histograms with Maximum Payload

7.2.2 RS Stegnalysis of Developed Techniques

In this section stego images are passed through RS stegnalysis to determine the payload of these images. Stego images generated by all three techniques are passed through RS stegnalysis. According to Fridrich et al., the initial bias (estimated proportion of hidden data when no data is hidden), noise level of the cover image, and the placement of message bits in the image are main factors that influence the accuracy of the estimated message length. Original cover images may indicate a small non zero message length due to random variations. This initial non zero bias could be either positive or negative.

Table 7.3 Data Detection using RS Stegnalysis from Baboon and Lena Stego Color 512 x 512 Sized Image with Payload = 24 KB

	Lena			Baboon		
Technique	Detected Data (Bits)	Detected Data (KB)	Percentage Detection	Detected Data (Bits)	Detected Data (KB)	Percentage Detection
Original Image	6,776	0.82	—	36,786	4.49	—
Classical LSB4	19,046	18.60	77.5	19,630	19.17	79.87
SOBP	8,806	8.60	35.83	11,294	11.03	45.95
THBS	17,602	16.19	67.45	13,209	12.90	53.75
MDS	25,024	3.05	12.70	39,239	4.79	19.95

Table 7.3 depicts the data detection percentage through RS stegnalysis when same amount of data i.e. 24 KB is embedded in Baboon and Lena 512 x 512 color images. It was observed that while keeping the same amount of hidden data, the detection from MDS generated stego images is the least that verdict high security of MDS as compared to other techniques. There exist variation in bias i.e. data detected from original image in the case of Lena image 0.82 KB is detected which is minimal as compared to 4.49 KB detected from original Baboon image. This shows the variation of bias among different images.

Table 7.4 Data Detection using RS Stegnalysis from Baboon Stego Color 512x512 Sized Image

Technique	Payload (bits)	Payload (KB)	Detected Data (Bits)	Detected Data (KB)	Percentage Detection
Original Image	0	0	36,786	4.49	—
Classical LSB4	3,145,728	384	2,934,374	358.2	93.28
SOBP	1,572,864	192	355,784	43.43	22.61
THBS	1,329,318	162	641,432	78.29	48.32
MDS	196,608	24	39239	4.79	19.95

Table 7.4 shows that for classical LSB4 method RS analysis detects 93.98% data, whereas data detection for THBS is comparatively less, but best technique in terms of stegnalysis are SOBP and MDS were 22.61% and 19.95% data is detected respectively from Baboon stego image through RS stegnalysis.

Table 7.5 Data Detection using RS Stegnalysis from Lena Stego Color 512x512 Sized Image

Technique	Payload (Bits)	Payload (KB)	Detected Data (Bits)	Detected Data (KB)	Percentage Data Detection
Original Image	0	0	6,776	0.82	__
Classical LSB4	3,145,728	384	2,908,160	355	92.44
SOBP	1,572,864	192	330,408	40.33	21.01
THBS	1,355,199	165	914,936	111.68	67.27
MDS	196,608	24	25,024	3.05	12.70

Table 7.5 shows the results for stego Lena color image where RS stegnalysis was able to detect only 12.70% data from MDS generated image. Whereas for SOBP and THBS generated stego image RS analysis detects 21.01% and 67.27% data respectively. It can be noticed that bias differs between Lena and Baboon images and data detection also varies in terms of the same technique applied to different images.

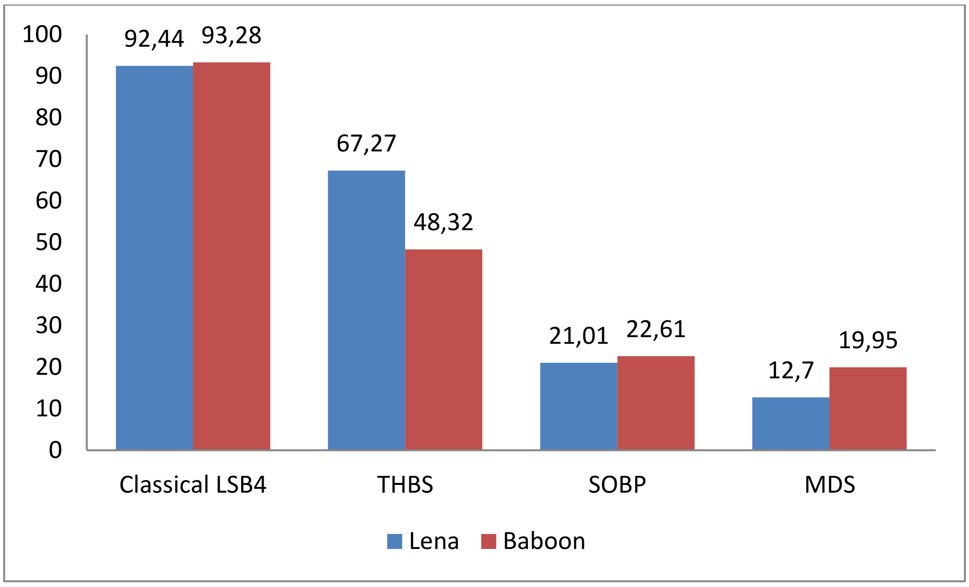


Figure 7.4 Data Detection Percentage for Baboon and Lena Color Images

The comparative results for percentage data detection from Baboon and Lena color stego images by RS stegnalysis is shown in figure 7.4. These results depicts that security of a same steganographic technique can vary from image to image as obvious in the case of THBS where 48.32% data is detected for Baboon as compared 67.27% from Lena stego image. This variation in the results is due to internal color intensities variation of images. In this case for above RS analysis it can be concluded that that MDS is more secure than other two techniques where RS stegnalysis can only detect minimal data <20% from MDS generated stego images as compared to LSB4, SOBP and THBS generated stego images.

8. CONCLUSIONS

Steganography is a mechanism to hide secret information inside cover of another medium. It is used since ancient times and now in the era of digital communication digital steganography is used. The common steganographic mediums used are text, image, audio and video for hiding secret information. Image steganography is one of the key types of steganography where a message to be sent is hidden inside the cover image. The most commonly used techniques for image steganography rely on LSB Steganography. In LSB image steganography, modification in the least significant bits of pixels enables the information transfer inside images without affecting the quality of the image. In this dissertation three new image steganography techniques are presented. The first technique proposed is a novel image steganography based on similarity of bits pairs. This is first technique that uses MSB along with LSBs for data hiding. Image pixel bits from 7th to 3rd bit are arranged in pairs. Data bits in pairs are compared with pixel pairs, if any of the pair matches with data pair, the pair number is saved in 2LSBs. If none of the pair is match with data pair, data bit pairs is saved in 2LSBs. Comparative analysis is performed along with existing techniques and results shows that this technique ensures more data capacity along with provisioning of acceptable level of security.

In second portion Threshold based image steganography; a technique that utilizes weakness of HVS is developed. This technique defines different color boundaries and categorizes image pixels. According to corresponding pixel's category data bits are embedded into pixels. It is complex technique for handling data bits and pixel values. The dependency in the execution process of this technique was the limitation of this technique that was source of pipelining hazard in its execution. Efficient Threshold based image steganography ensures it efficient execution by allowing the steganography process run in parallel in efficient manner along with maintaining security and complexity of algorithm.

Image steganography technique based on block matrix determinant is proposed in third section. The focus of this technique is to provide maximal data security. It is 1 LSB technique and has limited data capacity, but high quality of stego image and high PSNR ensures its secure mechanism.

All the three developed techniques have their own beneficial features. SOBP utilizes MSB along with LSB for the first time for embedding secret data. It achieves high data hiding capacity. THBS takes the advantage of weakness of HVS where variable length data bits are embedded in pixels. This allows to have complex steganographic algorithm, improved payload and PSNR. Lastly, MDS 1LSB technique is developed that ensures minimal change for data hiding in image and ensures high level of data security as compared to all other LSB image steganographic techniques.

An extension to this work can be performed by using steganography along with cryptography. A message to be hidden is encrypted using cryptography techniques and concealment of data inside cover through steganographic encoding can provide double folded security of secret data. In addition this work can lead to the development of a steganographic tool having options for different LSB and other image steganographic techniques, where a user according to their security and payload requirements can utilize specific steganographic technique.

REFERENCES

1. S. Katzenbeisser & F. Petitcolas. Information hiding techniques for steganography and digital watermarking. *Artech house.* 2000.
2. W. Bender, W. Butera, D. Gruhl, R. Hwang, F. J. Paiz, S. Pogreb, “Techniques for data hiding*”, IBM Systems Journal*, Volume 39, pp. 547 – 568, Issue 3-4, July 2000.
3. K. Robert. Steganography and steganalysis, *http://www.krenn.nl/univ/cry/steg/article.pdf*, January 2004.
4. A. Cheddad, J. Condell, K. Curran & Mc. Kevitt. Digital image steganography: Survey and analysis of current methods. *Signal processing*, 90(3), 727-752. 2010.
5. N. F. Johnson & S. Jajodia. Exploring steganography: Seeing the unseen. *Computer*, 31(2).1998.
6. T. Moerland. Steganography and steganalysis, Leiden Institute of Advanced Computing Science, *www.liacs.nl/home/tmoerl/privtech.pdf*
7. J. Simmons, Gustavus. The Prisoners' Problem and the Subliminal Channel, *in Proceedings of CRYPTO* '83, pp 51-67. Plenum Press 1984.
8. F.A.P.Petitcolas, R.J.Anderson, G.Kuhn: ”Information Hiding- a Survey”, *Proceedings of IEEE, vol.87, no.7,* pp.1062-1078, July, 1999.
9. M. Agarwal. Text steganographic approaches: a comparison. *arXiv preprint arXiv:1302.2718.* 2013
10. L. Y. Por & B. Delina. Information hiding: A new approach in text steganography. Proceedings. *WSEAS International Conference on Mathematics and Computers in Science and Enginee*ring (No. 7). World Scientific and Engineering Academy and Society. 2008.
11. M.H. Shirali-Shahreza. A new synonym text steganography. *IEEE, IIHMSP'08 International Conference on In Intelligent Information Hiding and Multimedia Signal Processing,* (pp. 1524-1526). 2008.
12. M. Shirali-Shahreza. Text steganography by changing words spelling. *IEEE, 10th International Conference on Advanced Communication Technology*. (Vol. 3, pp. 1912-1913). 2008.

13. P. Jayaram, H.R. Ranganatha & H.S. Anupama. Information hiding using audio steganography–a survey. *The International Journal of Multimedia & Its Applications (IJMA) Vol*, 3, 86-96. 2011.
14. M. Nosrati, R. Karimi & M. Hariri. Audio steganography: A survey on recent approaches. W*orld Applied Programming*, 2(3), 202-205. 2012.
15. P.K.Singh, R.K.Aggrawal. Enhancement of LSB based Steganography for Hiding Image in Audio, *International Journal on Computer Science and Engineering*, Vol. 02, No. 05, 2010.
16. N. Cvejic & T. Seppanen. Increasing the capacity of LSB-based audio steganography. *IEEE Workshop on Multimedia Signal Processing* (pp. 336-338). 2002.
17. K. Gopalan. Audio steganography using bit modification. *ICME'03. Proceedings of International Conference on Multimedia and Expo*, 2003.
18. M. Sadek, A.S. Khalifa & M.G. Mostafa. Video steganography: a comprehensive review. *Multimedia tools and applications,* 74(17), 7063-7094.2015.
19. Rhoads, B. Geoffrey. Video steganography. *U.S. Patent 6,026,193,* issued February 15, 2000.
20. M. Ramalingam. Stego machine–video steganography using modified LSB algorithm. *World Academy of Science, Engineering and Technology,* 74, 502-505. 2011.
21. A. Swathi, & S. A. K. Jilani. Video steganography by LSB substitution using different polynomial equations. *International Journal of Computational Engineering Research*, 2(5). 2012.
22. K.S. Jenifer, G. Yogaraj & K. Rajalakshmi. LSB approach for video Steganography to embed images. *International Journal of Computer Science and Information Technologies*, 5(1), 319-322. 2014.
23. T. Morkel, J.H. Eloff & M.S. Olivier. An overview of image steganography. *In ISSA* (pp. 1-11). 2005.
24. M. Khodaei & K. Faez New adaptive steganographic method using least-significant-bit substitution and pixel-value differencing. *IET Image processing*, 6(6), 677-686. 2012.

25. C.-M. Wang, N.-I. Wu, C.-S. Tsai, and M.-S. Hwang, "A high-quality steganographic method with pixel value differencing and modulus function," *Journal of Systems and Software*, vol. 81, pp. 150-158, 2008.
26. C. Vleeschouwer, J. Delaigle & B. Macq, Circular Interpretation on Histogram for Reversible Watermarking, *in Proceedings of the 4th IEEE International Workshop on Multimedia Signal Processing*, Cannes, pp. 345-350, 2001.
27. Liu, Yu-Chi, Hsien-Chu Wu, & Yu. Shyr-Shen. Adaptive DE-based reversible steganographic technique using bilinear interpolation and simplified location map. *Multimedia Tools and Applications* 52.2-3 (2011): 263-276.
28. J. Fridrich & Du. Rui. Secure steganographic methods for palette images. *International Workshop on Information Hiding*. Springer, Berlin, Heidelberg, 1999.
29. L.M. Marvel, C.G. Boncelet & C.T. Retter. Spread Spectrum image steganography. *IEEE Transactions on image processing*, 8(8), 1075-1083. 1999.
30. S. Sharma & U. Kumar. Review of Transform Domain Techniques for Image Steganography. *International Journal of Science and Research*, 2(2), 1. 2015.
31. Thomas, Priya. "Literature survey on modern image steganographic techniques." *International Journal of Engineering Research and Technology*. Vol. 2. No. 5 (May-2013). ESRSA Publications, 2013.
32. K. Bailey & K. Curran. An evaluation of image based steganography methods. *Multimedia Tools and Applications,* 30(1), 55-88., 2006.
33. A. Rehman, T. Saba, T. Mahmood, Z. Mehmood, M. Shah & A. Anjum. Data hiding technique in steganography for information security using number theory. *Journal of Information Science*. 2018.
34. V.L. Reddy, A. Subramanyam & P.C. Reddy. Implementation of LSB steganography and its evaluation for various file formats. *Int. J. Advanced Networking and Applications,* 2(05), 868-872.2011.
35. P. Moulin & M. K. Mihcak. *A framework for evaluating the data-hiding capacity of image sources*. IEEE Transactions on Image Processing, 11(9), 1029-1042. 2002.

36. R. Chandramouli, M. Kharrazi & N. Memon. Image steganography and steganalysis: Concepts and practice. *In International Workshop on Digital Watermarking*, Springer, Berlin. 35-49, 2003.
37. A.H. Mazinan & N. Sadati. Comparative Histogram Analysis of LSB-based Image Steganography. *WSEAS Transactions on systems and control.* 13, 103-112. 2018.
38. J. Fridrich, M. Goljan, and R. Du. Detecting lsb steganography in color and grayscale images, *IEEE Multimedia Special Issue on Security*, 22–28, OctoberNovember 2001.
39. J. Fridrich, R. Du, and L. Meng, “Steganalysis of lsb encoding in color images, *ICME*, New York, USA. 2000.
40. G.Chugh. A Review Article: Image steganography techniques. *Acta Technica Corvininesis-Bulletin of Engineering*, 6(3). 2013.
41. B. Li, J. He, J. Huang, & Y.Q. Shi. A survey on image steganography and steganalysis. *Journal of Information Hiding and Multimedia Signal Processing*, 2(2), 142-172. 2011.
42. R. Chandramouli & N. Memon . Analysis of LSB based image steganography techniques. IEEE, Proceedings of International Conference on Image Processing, Vol. 3, pp. 1019-1022. 2001.
43. J. Tian, "Reversible data embedding using a difference expansion, *IEEE Trans. Circuits Syst. Video Techn*, vol. 13, pp. 890-896, 2003.
44. K. M. Singh, L. S. Singh, A. B. Singh, and K. S. Devi, Hiding secret message in edges of the image, *International Conference on Information and Communication Technology*, 2007.
45. C.-H. Yang, C.-Y. Weng, S.-J. Wang, and H.-M. Sun, Adaptive data hiding in edge areas of images with spatial LSB domain systems, *IEEE Transactions on Information Forensics and Security*, vol. 3, pp. 488-497, 2008.
46. N. Tiwari & D. M. Shandilya. Evaluation of various LSB based methods of image steganography on GIF file format. *International Journal of Computer Applications* (0975–8887) 2010.

47. K. Hempstalk, Hiding behind corners: Using edges in images for better steganography, *Proceedings of the Computing Women's Congress, Hamilton, New Zealand,* pp. 11-19, 2006.
48. M. Hossain, S. Al-Haque, and F. Sharmin, Variable rate steganography in grayscale digital images using neighborhood pixel information," *The International Arab Journal of Information Technology,* vol. 7, pp. 34-38, 2010.
49. L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, Reversible image watermarking using interpolation technique, *IEEE Transactions on Information Forensics and Security,* vol. 5, pp. 187-193, 2010.
50. Y. Yalman, F. Akar & I. Erturk, An Image Interpolation based Reversible Data Hiding Method Using R-Weighted Coding, *Proceedings of the 13th International Conference on Computational Science and Engineering,* Hong Kong, pp. 346-350, 2010.
51. Y. K. Jain & R. Ahirwal, A Novel Image Steganography Method with Adaptive Number of Least Significant Bits Modification Based on Private Stego-Keys, *International Journal of Computer Science and Security (IJCSS),* vol. 4, 2010.
52. M. Hamid and M. L. M. Kiah, Novel Approach for High Secure and High Rate Data Hidden in the Image Using Image Texture Analysis, *International Journal of Engineering and Technology (IJET):* 0975-4042, 2009.
53. M. Chan: 'MATLAB Central: image error measurements', 2010.
54. S. Narayanan. 'MATLAB Central: image quality measures', 2011.
55. K. Zhang, S. Wang & Zhang. "New metric for quality assessment of digital images based on weighted mean square error". *International Society for Optics and Photonics,* City, 2002.
56. T. Chai & R. Draxler, Root mean square error (RMSE) or mean absolute error (MAE). *Geoscientific Model Development Discussions,* 7, 1525-1534. 2008.
57. Z. Wang, A. C. Bovik, "A universal image quality index", *IEEE Signal Processing Letters,* vol. 9, pp. 81-84, Mar. 2002.
58. K. Egiazarian, J. Astola, N. Ponomarenko, V. Lukin, F. Battisti, M. Carli. New full reference quality metrics based on HVS, *Proceedings of the Second International Workshop on Video Processing and Quality Metrics,* Scottsdale, USA, 2006.

59. K. R. Mayuresh Gulame, R. S. Joshi Kamthe, A Full Reference Based Objective Image Quality Assessment *IJAEEE*, Volume 2, Issue 6, 2013.

60. A. Sasivarnan, Jagan, Jaspreet Kaur, Divya Jyot & D.S. Rao, Image Quality Assessment In Spatial Domain, *IJCST*, Vol. 2, Issue 3, September 2011.

61. Z. Ni, Y. Shi, N. Ansari & S. Wei, Reversible Data Hiding, *IEEE Transactions on Circuits Systems and Video Technology*, vol. 16, no. 3, pp. 354-362, 2006.

62. M. Hussain & M. Hussain, A Survey of Image Steganography Techniques, *International Journal of Advanced Science and Technology*, vol. 54, pp. 113-124, 2013.

63. C. Lin & N. Hsueh, A Lossless Data Hiding Scheme based on Three-Pixel Block Differences, *Pattern Recognition*, vol. 41, no. 4, pp. 1415-1425, 2008.

64. Y. Hu, H. Lee & J. Li, De-based Reversible Data Hiding With Improved Overflow Location Map, *IEEE Transactions on Circuits Systems and Video Technology,* vol. 19, no. 2, pp. 250-260, 2009.

65. L. Luo, Z. Chen, M. Chen, X. Zeng & Z. Xiong, Reversible Image Water Marking Using Interpolation Technique, *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 187-193, 2010.

66. D. Wu & W. Tsai, A Steganographic Method for Images by Pixel-Value Differencing, *Pattern Recognition Letters*, vol. 24, no. 9, pp. 1613-1626, 2003.

67. C. Vleeschouwer, J. Delaigle & B. Macq, Circular Interpretation on Histogram for Reversible Watermarking, *in Proceedings of the 4th IEEE International Workshop on Multimedia Signal Processing,* Cannes, pp. 345-350, 2001.

68. G. Xuan, J. Zhu, J. Chen, Y. Shi, Z. Ni & W. Su, Distortionless Data Hiding based on Integer Wavelet Transform, *IEEE Letters,* vol. 38, no. 25, pp. 646-1648, 2002.

69. M. Celik, G. Sharma, A. Tekalp & E. Saber, Reversible Data Hiding, *in Proceeding of International Conference Image Processing*, pp. 157-160, 2002.

70. D. Shehzad & T. Dag. A Novel Image Steganography Technique Based on Similarity of Bits Pairs. *IEEE 8th Control and System Graduate Research Colloquium (ICSGRC)*, 2017.

71. S. Saxena, N. Sharma, S. Sharma, Image processing tasks using parallel computing in multi core architecture and its applications in medical imaging, *International Journal of Advanced Research in Computer and Communication Engineering,* Volume 2, Issue 4, ISSN: 2278-1021, 2013.
72. Petryniak, Rafal. Analysis of efficiency of parallel computing in image processing task. *Mechanika Czasopismo Techniczne.* 105.3-M : 185-193. 2008.
73. G. Silvana, M. Edlira, “Wu-Lee Steganographic Algorithm on Binary Images Processed in Parallel,” *International Journal of Video & Image Processing and Network Security IJVIPNS-IJENS*, Vol: 12 No: 03. 2012.
74. B. Wilkinson & M. Allen. Parallel programming: techniques and applications using networked workstations and parallel computers (Vol. 2). *New York: Prentice hall.* 2005.
75. W. Gropp, E. Lusk & A. Skjellum, Using MPI: portable parallel programming with the message-passing interface (Vol. 1). *MIT press.* 1999.
76. M.J. Quinn. Parallel Programming. *TMH CSE*, 526. 2011.
77. M. Hariri, R. Karimi & M. Nosrati. An introduction to steganography methods, *World Applied Programming*, 1, (3), pp. 191-195. 2011.
78. S. Singh, P. Kaur, K. Kaur; Parallel computing in digital image processing, *International Journal of Advanced Research in Computer and Communication Engineering* Vol. 4, Issue 1, January 2015.
79. P. Pacheco, An introduction to parallel programming, *Elsevier*, 2011.
80. G. Kamalakannan, Rajamanickam, High Performance Color Image Processing in Multicore CPU using MFC Multithreading, *International Journal of Advanced Computer Science and Applications*. Vol 4. No 12, 2013.
81. S. R. Sternberg, Parallel architectures for image processing. *In Real-Time Parallel Computing* (pp. 347-359). Springer, Boston, MA. 1981.
82. B. Wilkinson & M. Allen. Parallel programming: techniques and applications using networked workstations and parallel computers (Vol. 2). *New York: Prentice hall.* 2005.
83. A. Cochocki, & R. Unbehauen. Neural networks for optimization and signal processing. *John Wiley & Sons,* Inc. 1993.

84. M. Gordon, W. Thies & S. Amarasinghe, Exploiting coarse-grained task, data, and pipeline parallelism in stream programs. *ACM SIGOPS Operating Systems Review*, 40(5), 151-162. 2006.

85. W. Thies, V. Chandrasekhar & S. Amarasinghe. A practical approach to exploiting coarse-grained pipeline parallelism in C programs. *Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture* IEEE Computer Society. (pp. 356-369). 2007.

86. K. Muthukumar & M.V. Hermenegildo. Complete and efficient methods for supporting side effects in *independent*/restricted and-parallelism. 1989.

87. J. Liebeherr, E.R. Omiecinski & I.F. Akyildiz. The effect of index partitioning schemes on the performance of distributed query processing. *IEEE Transactions on Knowledge and Data Engineering*, 5(3), 510-522. 1993.

88. N. Hardavellas & I. Pandis. Intra-Query Parallelism. *In Encyclopedia of Database Systems Springer US.* pp. 1567-1568. 2009.

89. H. Pirahesh, C. Mohan, J. Cheng, T. Liu & P. Selinger. Parallelism in relational data base systems: architectural issues and design approaches. *In Proceedings of the second international symposium on Databases in parallel and distributed systems ACM.* (pp. 4-29). 1990.

90. R. Vein & P. Dale. Determinants and their applications in mathematical physics. *Springer Science & Business Media.* (Vol. 134). 2006.

91. K. Bailey & K. Curran, An evaluation of image based steganography methods, *Multimedia Tools and Applications,* vol. 30, pp. 55-88, 2006.

92. A. A. Gutub, Pixel indicator technique for RGB image steganography, *Journal of Emerging Technologies in Web Intelligence,* vol. 2, pp. 56-64, 2010.

93. F. A. Jassim, A novel steganography algorithm for hiding text in image using five modulus method, *arXiv preprint* arXiv:1307.0642, 2013.

94. M. Karim, A new approach for LSB based image steganography using secret key, *in 14th International Conference on Computer and Information Technology* (ICCIT 2011), pp. 286-291, 2011.

95. M. Khan, et al. A secure method for color image steganography using gray-level modification and multi-level encryption. *TIIS* 9.5: 1938-1962, 2015.

THESIS CONTRIBUTIONS

Journal Articles

1. *'LSB Image Steganography based on Block Matrix Determinant method'*, accepted for publication in KSII Transactions on Internet and Information Systems.
2. *'Threshold-based Steganography: A Novel Technique for Improved Payload and SNR'*, International Arab Journal of Information Technology; Vol. 13, No 4, 2016.
3. *'Ensuring Efficient Threshold-Based Image Steganography using Parallel Computing'*, under review.

Conference Paper

4. '*A Novel image Steganography Technique based on Similarity of Bits Pairs.'* IEEE 8th Control and System Graduate Research Colloquium (ICSGRC), Shah Alam, Malaysia; 08/2017

CURRICULUM VITAE

Danish Shehzad

Computer Engineering Department
Kadir Has University, Istanbul, Turkey
Tel: +90537-8380318
E-mail: danish.shehzad@khas.edu.tr

Areas of Interest

Strong interest in Information Security, Parallel Programing, High Performance Computing, Parallel Languages and Wireless Networks

Education

2014 - 2018	**PhD Computer Engineering,** Kadir Has University, Istanbul, Turkey (CGPA: 3.93/4.0)
2012 - 2014	**MS Computer Science,** Hazara University, Mansehra, Pakistan (CGPA: 3.78/4.0)
2006 - 2010	**BS Telecommunication & Networks,** (Honors) CIIT, Pakistan (CGPA: 3.25/4.0)

Experience

2016-2018	**Graduate Assistant,** Computer Engineering Department, Kadir Has University, Istanbul, Turkey
2014 - 2016	**Research Assistant** Computer Engineering Department, Kadir Has University, Istanbul, Turkey Project: (BIRTS) Brain Inspired Run Time Systems for Very Large Scale Brain Simulation *(Funded by TUBITAK)*