AN IMPROVED IMAGE STEGANOGRAPHY BASED ON LEAST SIGNIFICANT BIT MATCHING REVISITED (LSBMR) USING SOBEL EDGE DETECTION

BY

MARIAM OCHEJA

DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF PHYSICAL SCIENCE

AHMADU BELLO UNIVERSITY,

ZARIA, NIGERIA.

JANUARY, 2017

AN IMPROVED IMAGE STEGANOGRAPHY BASED ON LEAST SIGNIFICANT BIT MATCHING REVISITED (LSBMR) USING SOBEL EDGE DETECTION

BY

Mariam, OCHEJA

MSC/SCI/22070/2012-2013

A DISSERTATION SUBMITTED TO THE SCHOOL OF POSTGRADUATE STUDIES, AHMADU BELLO UNIVERSITY, ZARIA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF

MASTER OF SCIENCE DEGREE IN COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF PHYSICAL SCIENCE

AHMADU BELLO UNIVERSITY,

ZARIA, NIGERIA

JANUARY, 2017

DECLARATION

I declare that the work in this Dissertation entitled "AN IMPROVED IMAGE STEGANOGRAPHY BASED ON LSBMR USING SOBEL EDGE DETECTION" has been carried out by me in the Department of Computer Science under the supervision of Professor S.B. Junaidu and Professor A.A. Obiniyi. The information derived from the literature has been duly acknowledged in the text and a list of references provided. No part of this dissertation was previously presented for another degree or diploma at this or any other institution.

OCHEJA, Mariam ---------- ----------

Name of Student Signature Date

CERTIFICATION

This Dissertation entitled "AN IMPROVED IMAGE STEGANOGRAPHY BASED ON LSBMR USING SOBEL EDGE DETECTION" by MARIAM, OCHEJA (MSc/Sci/22070/2012-2013) meets the regulations governing the award of the degree of Master of Science of the Ahmadu Bello University, and is approved for its contribution to knowledge and literary presentation.

---------- Date----------

Professor S.B. Junaidu
Chairman, Supervisory Committee

---------- Date----------

Prof. A.A. Obiniyi
Member, Supervisory Committee

---------- Date----------

Professor S.B. Junaidu
Head of Department

---------- Date----------

Professor K. Bala
Dean, School of Postgraduate Studies

DEDICATION

This dissertation is dedicated to my beloved parent Alhaji Alhassan and Hajia Fatima Ocheja for their love for education and inspiration.

ACKNOWLEDGEMENTS

First and foremost, my profound gratitude goes to Almighty Allah the beneficent, the merciful for his kindness, love, guidance and protection throughout my M.Sc. Programme.

I would like to take this opportunity to extend my hearty gratitude to my guide and supervisor Professor Sahalu Balarabe Junaidu. It was my honor and pleasure to work with him and I cannot express how grateful I am that he gave me research freedom when I needed it. He was always able to create a friendly yet professional environment that motivated every student to work hard and think for themselves. Also I extend my sincere appreciation to my second supervisor Prof. Afolayan Ayodele Obiniyi, whose constant guidance and encouragement made the completion of my M.Sc. dissertation possible.

I am obliged to all the Lecturers of the Department of Computer Science for instilling in me the basic knowledge about the field that greatly benefitted me while carrying out the research and achieving the goal. Also, I acknowledge the support from the Non-academic Staff such as the Librarians and Laboratory Technicians.

This dissertation would not have been possible without the support of my family. I owe my sincerest gratitude to them for their endless support, for which my mere expression of thanks does not suffice. Most especially I am indebted, to my Parents, Alhaji Alhassan and Hajia Fatima Ocheja for being considerate, enduring and appreciative throughout my study. I appreciate the effort of my siblings-Aishat, Abubakar, Ibrahim, Larre, Muhammad and Aminat for their passionate love. I am very grateful to my little Nephew, Bashir Alhassan and his father, Alhassan Lukman for his immense support. I also, acknowledged my paternal and maternal relations for their fervent prayers. Their love and support made my stay so far away from home much easier.

I owe my sincerest gratitude to Abdulhakeem Ibrahim and Oyelade Olaide Nathaniel for their endless support even with their tight schedules.

To my fellow colleagues, for their relentless support in augmenting the value of work to mention but a few Habu Saratu, Abdulkadir Khadija, Emma N. Achi, Al-Amin Aliyu, Mayowa Oyetunji, Iwoh John, Abdulkarim Rukkya, Gambo Isma'il, Olalekan David, Murjanatu Abdullahi , Donna Bala, Labran Harirat, Abdulsalami O. Aminu and Aliyu Kufena.

Last, by no means least, I wish to offer my heartfelt thanks and gratitude to the kind people around me during the course of my work. They are: Mr. Friday Ugbede, Mr Chris Odonwodo, Mr. Abdul Ariola Malik for their constant prayers and support. I am indebted to the family of Mr. and Mrs. Daniel, and their children-Kuyet, Grace, Kish and Lucky Daniel for their accommodation, hospitality, kindness and love. Mr Titus and Family, Mrs. J.N. Edeoga, Chinedu Edeh, Abiola Daudu, and Abdullahi Salihu for their kindness.

I am grateful to all of those with whom I have had the pleasure to work with during the course of my Master Degree in Computer Science. May GOD bless you all!

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

ABBREVIATIONS, DEFINITIONS, GLOSSARIES AND SYMBOLS

ACRONYM	DEFINITION
2D-DCT	2-Dimensional Discrete Cosine Transform
BMP	BitMaP
BPP	Bit Per Pixel
DCT	Discrete Cosine Transform
EALSBMR	Edge Adaptive Least Significant Bit Matching Revisited
GIF	Graphical Interchange Format
HVS	Human Visual System
IDCT	Inverse Discrete Cosine Transform
IDE	Integrated Development Environment
JPEG	Joint Photographic Expert Group
LSB	Least Significant Bit
LSBMR	Least Significant Bit Matching Revisited
MSE	Mean Squared Error
Pixel (s)	Picture Element (s)
PNG	Portable Network Group
PSNR	Peak Signal to Noise Ratio
PRNG	Pseudo Random Number Generator
RGB	Red Green and Blue
RS	Regular Singular
VSL	Virtual Steganographic Laboratory

ABSTRACT

Image steganography is the science of hiding data for securing confidential communication and it is the most popular type of carrier to hold information. Many algorithms have been proposed to hide information into digital images. The least significant bit algorithm (LSB) as one of the algorithms proposed is widely used in message embedding. However, the robustness of the algorithm based on LSB is low. The hidden message is usually destroyed when some image operations like resizing, cropping and rotation are applied to the stego-image. To overcome this limitation, this work proposed an improved image steganography based on least significant bit matching revisited (LSBMR) using Sobel edge detection that withstands image operations like resizing, rotation and cropping. The proposed method employs 2-dimensional discrete cosine transformation (2D-DCT) to transform the detected edges of the cover image pixel value into its coefficient, embeds the secret message in the coefficients of the detected edges of the cover image which was implemented in Netbeans IDE. Experimental results produced better stego-image quality that is robust against multiple image operations such as resizing and cropping. The statistical steganalysis tools such as Virtual Steganographic Laboratory (VSL) and StegExpose cannot detect the presence of secret information in the stego-image. Also, the proposed system generated stego-image with Peak Signal to Noise Ratio (PSNR) that is an image quality metric of 68 decibels (dB) for 8000 bits of secret message as regards to the invisibility over the existing steganography technique.

CHAPTER ONE: INTRODUCTION

This chapter discusses the introductory part of this dissertation, which includes the background to the study, problem statements and research motivation, the research aim and objectives, research methodology, contribution to knowledge and finally the organization of the rest of the dissertation.

1.1 Background to the Study

Recently, people exchange information using the existing communication technologies such as the internet and huge volume of data transfer takes place via the plethora of services offered by the web. This information can be very sensitive and need to be protected against any attacker who tries to intercept them during the transmission stage. According to Ratnakirti *et al.,* (2013), data over internet may be stolen, intercepted, illegally modified or even destroyed by an adversary resulting in intellectual property rights infringement, data loss, data leakage and data damage. Transmitting top secret information cannot be solely relied on the existing communication channels because the technologies are vulnerable to attacks Osama, (2005) and exchanged information can be detected relatively easily. Therefore, it is vital to protect the privacy and confidentiality of top secret message during its transit through the internet. To preserve the privacy and confidentiality of important data over the internet it must be provided with a metaphorical envelope such that its contents are revealed only to the intended receiver Ratnakirti *et al.,* (2013) without arousing suspicions. Data hiding techniques such as steganography precisely aim at performing this task.

The steganography technique has been used many years ago to convey secret messages. For instance, Greek historian Herodotus was the first to document the usage of steganography to send messages (Aubrey, 1996). A slave was sent by his master to deliver a secret

message tattooed on his scalp. After the message was tattooed, the slave waited until his hair grew back and concealed the message. The most popular steganographic methods between the 13th and 16th century involved written text. One method used a mask, a paper with holes, shared between the sender and recipient. The mask was simply put over the text and the message was revealed. Francis Bacon realized that two different fonts for each letter can be applied to embed binary representations of messages. Holub, (2014), stated that "Brewster devised a very original technique in 1857, which was later used in several wars".

Security of most of the previously mentioned methods was achieved only by assuming ignorance of the adversary. This is sometimes pejoratively called security through obscurity. The adversary did not attempt any targeted attack in the sense of modern steganalysis, instead they trained spies and secret services to obtain the necessary information by other means (Fridrich, 2009).

Although steganography is an ancient subject, the modern formulation of it is often given in terms of the prisoner's problem proposed by Simmons (Patel and Gadhiya, 2015), where two inmates wish to communicate in secret to hatch an escape plan. All of their communication passes through a warden who will throw them in solitary confinement should he/she suspect any covert communication (Chandramouli *et al.,* 2003).

The warden, who is free to examine all communication exchanged between the inmates, can either be passive or active. A passive warden simply examines the communication to try and determine if it potentially contains secret information. If she suspects a communication to contain hidden information, a passive warden takes note of the detected covert communication, reports this to some outside party and lets the message through without blocking it. An active warden, on the other hand, will try to alter the

communication with the suspected hidden information deliberately, in order to remove the information (Anderson and Petitcolas, 1998). Figure 1.1 represents modern steganography. Generally steganography is known as “invisible” communication of hiding secret messages into digital cover-media such that attackers will not be aware of the existence of the hidden messages (Micheal and Herbert, 2011). It is a mechanism that completely differs from cryptography. In fact, in cryptography the information is modified but still can be seen in this unreadable format once sent over the networks, whereas in steganography the information is simply embedded into a digital support and cannot be noticed as long as the quality of the carrier is not deteriorated (Zohreh and Jihad, 2014).

Steganography hides information in a variety of multimedia carriers that include video clip, a digital image, an audio file or text called *cover object.* Once the information is embedded in any of the cover media it is called *stego-object*. If the cover is an image or video file, then the result of embedding the information in the cover is referred to as stego-image or stego-video respectively.

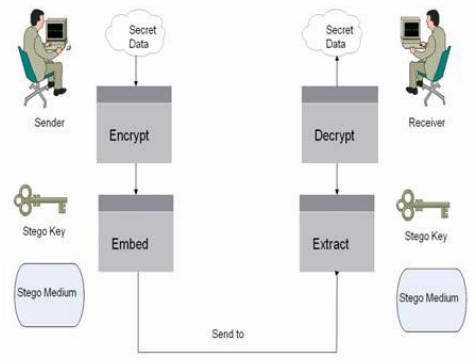


Figure 1.1: Modern Steganography System (Por *et al.*, 2008)

It is shown that images are excellent carriers to hide and exchange sensible information over networks (Rodrigues *et al.*, 2004). Many algorithms have been proposed recently to hide information into images and preserve their quality. In this dissertation, we focus on

image steganography algorithms because image used as a host object was observed to have low communication cost and availability of large number of redundant bits. An image consists of light luminance or pixels represented as an array of values at different points. A pixel consists of one byte or more. For example in 8-bit images each pixel consists of 1 byte (i.e., 8 bits). While each pixel in a 24-bit image is represented as three bytes representing the Red, Green and Blue (RGB) colors (Caldwell, 2003).

Image steganography has many applications, especially in today's modern, high-tech world. Most people on the internet have a concern about privacy and secrecy. For two parties, image steganography allows to communicate secretly (Kamred, 2014). For some morally-conscious people is allowed to safely whistle blow on internal actions (Sahar, 2015). Also, it allows for copyright protection on digital files using the message as a digital watermark. One of the other main applications for image steganography is for the high-level or top-secret documents transportation between international governments (Phad *et al.,* 2012). In medicine, medical practitioners can embed some information such as name, comments or diagnosis of the patient into their medical imagery and exams. Then medical images can be of different types such as embedding information into ECG images (Ibaida *et al.,* 2010). In military, not only the content of the communication but also the communication itself between agencies must be kept secret. Information hiding technique can be used when two or more agencies communicate via digital short radio (Jiang *et al.,* 2009). In smart id's, information of the person is embedded inside their image for confidential information (Flores-Escalante *et al.,* 2012). In remote sensing, information can be hidden into some site images to provide secret only to authorized users (Wang and Niu, 2008). In e-commerce, registration information can be hidden in electronic papers that can be used to identify authentication based on steganographic techniques (Wang and Ye,

2010). Also, it can be used in several areas such as: printers, database systems, human rights organization and correcting media transition errors.

There exist several embedding algorithms for image steganography both in spatial and transform domains. But most algorithms in the spatial domain are vulnerable to image processing operations such as cropping, resizing and rotation.

These problems, however, are of utmost importance; therefore there is need for continued improvement.

1.2 Problem Statement

Many research works have been conducted on spatial domain image steganography-based algorithms. Least significant bit (LSB) replacement in the spatial domain is a well-known steganographic method. In this embedding scheme, only the LSB plane of the cover image is directly overwritten with the secret bit stream according to a pseudorandom number generator (PRNG).

Subhedar and Mankar, (2014) stated that "spatial domain steganography schemes achieve high embedding capacities, but they are vulnerable to small modifications that may result from image processing operations such as cropping, rotation, scaling and resizing".

It was investigated that the robustness of the image steganography algorithm based on LSBMR using Sobel edge detection is very low. The hidden message was destroyed when some image operations such as cropping, resizing and rotation were applied to the stego-image.

Hence, the need to design and implement an improved image steganography based on LSBMR using Sobel Edge Detection that is robust against any image operations and works in transform domain.

1.3 Research Motivation

Security of information during transmission is a major issue in this modern era. All of the communicating bodies want confidentiality, integrity, and authenticity of their secret information. Since an unhidden coded message, no matter how unbreakable it is, will arouse suspicion.

In addition, due to the problems and challenges posed by the existing image steganographic algorithm, which is based on LSB in spatial domain specifically the work of (Zohreh and Jihad, 2014). Therefore, the need arises for an improved image steganography based on LSBMR using Sobel Edge Detection that is robust enough to withstand any image operations (like cropping, resizing and rotation) and operates in transform domain.

1.4 Research Aim and Objectives

The aim of this research is to develop an improved image steganography based on Least Significant Bit Matching Revisited (LSBMR) using sobel edge detection that is robust against image operations like rotation, resizing and cropping.

The objectives are to:

- **a.** carry out process analysis on image steganography techniques;
- **b.** design a modified image steganography based on LSBMR using Sobel Edge Detection;
- **c.** employ 2D-DCT transformation to the detected edges of the cover image;
- **d.** implement the proposed system;
- **e.** evaluate and compare the performance of the technique side by side the work of (Zohreh and Jihad, 2014) in relation to robustness, undetectability and invisibility.

1.5 Research Methodology

The following steps were taken in the course of this research to achieve the stated objectives.

a. Review of related literature on security, encryption, steganography and digital images was conducted with a view to get ideas on how to enhance the work of (Zohreh and Jihad, 2014).

b. Modify the image steganography based on LSBMR using Sobel edge detection to make it more robust and work in transform domain.

c. Enhance the technique by employing 2D-DCT transformation technique so as to embed the secret message into the coefficients of the cover image rather than into the image pixels directly.

d. Application of standard matrix to quantize the coefficients before embedding the secret bits into the resultant coefficients.

e. Implement the enhanced algorithm using open source technologies such as Java Programming Language and Netbeans Integrated Development Environment (IDE).

f. Evaluate and compare the proposed system with the work of (Zohreh and Jihad, 2014) in relation to robustness, undetectability and invisibility using standard statistical steganalysis and image quality metrics such as PSNR and MSE steganalysis.

1.6 Organization of the Dissertation

The dissertation is structured as follows:

Chapter One: The general introduction has been presented in this chapter.

Chapter Two: This chapter reviews the relevant literature on information security. An overview of the information hiding techniques using steganography is stated in details. Categories of steganography algorithms are described. An overview of digital images used in steganography is presented. State-of-art image algorithms are also presented in this chapter.

Chapter Three: The methodology of improved image steganography based on LSBMR using Sobel edge detection on images is demonstrated. The message embedding and extraction processes of the proposed algorithm are presented in detail.

Chapter Four: The research implementation strategy of the system integrating the different steganography techniques is presented in this chapter. We also present analysis of the experimental results obtained to evaluate the performance of the method. Comparative performance results between the existing and the enhanced systems are presented.

Chapter Five: This chapter presented the summary of the research, the conclusion and future work.

CHAPTER TWO: LITERATURE REVIEW

The purpose of this chapter is to review the relevant concepts that are related to the research work. It takes a look at information security, overview of information hiding techniques using steganography, categories of steganography algorithms, overview of digital images used in steganography and related works on image steganography based on LSB.

2.1 Information Security

Information or data is the wealth of any organization therefore security issues are top priority to an organization dealing with confidential data (Michael and Herbert, 2011). Information security evolved from the early field of computer security and it is the protection of information assets that use, store, or transmit information from risk. The critical characteristics of information, among them confidentiality, integrity, and availability (the C.I.A. triangle), must be protected at all times; this protection is implemented by multiple measures through policies, education training and awareness, and technology.

Security is protection from danger (Webster, 1831). In other words, protection against adversaries from those who would do harm, intentionally or otherwise is the objective. National security, for example, is a multi-layered system that protects the sovereignty of a state, its assets, its resources, and its people. Achieving the appropriate level of security for an organization also requires a multifaceted system. A successful organization should have the following multiple layers of security in place to protect its operations:

a. Physical security: This is to protect physical items, objects, or areas from unauthorized access and misuse.

b. Personnel security: To protect the individual or group of individuals who are authorized to access the organization and its operations.

c. Operations security: This is to protect the details of a particular operation or series of activities. Communications security: To protect communications media, technology and content (Z'aba and Maarof, 2006).

d. Network security: This is to protect networking components, connections, and contents.

e. Information security: This is the protection of information assets that use, store, or transmit information from risk through the application of policy, education and technology (Michael and Herbert, 2011).

2.2 Information Hiding

Recently with the rapid use of information in modern technology, information hiding methods received much attention from the research community in information security. This growth of information encourages researchers to develop security techniques and to keep data transmission between sender and receiver safer from attackers (Al-Shatanawi and El-Emam, 2015). The idea of data hiding or digital steganography was first introduced with the example of prisoner's secret message by Simmons in 1983 according to (Subhedar and Mankar, 2014).

Information hiding refers to embedding secret message into a digital medium. The secret message can be a simple text, an image, an audio or any object that can be presented by some number of bits. It is desired to embed the secret message in an unsuspected object. This object can come in several formats such as image, audio, video, file or any other types that can carry information without destroying it. It is then referred to as cover image, cover audio, and cover video respectively. Once the secret message is embedded into the cover

object it is called stego-object. After the stego-object is sent, the receiver should extract the message from the stego-object. Both the sender and receiver can agree on stego-key that is used at the extraction phase. The stego-key is used to control the hidden message from being recovered by eavesdropping. In addition, the receiver extracts the message based on the stego-key since it defines how the secret message is embedded (Subhedar and Mankar, 2014).

In this dissertation, secret data is referred to as secret message or covert message and will be used interchangeably.

Steganography, digital watermarking and covert channels are all fields related to information hiding as shown in Figure 2.1.

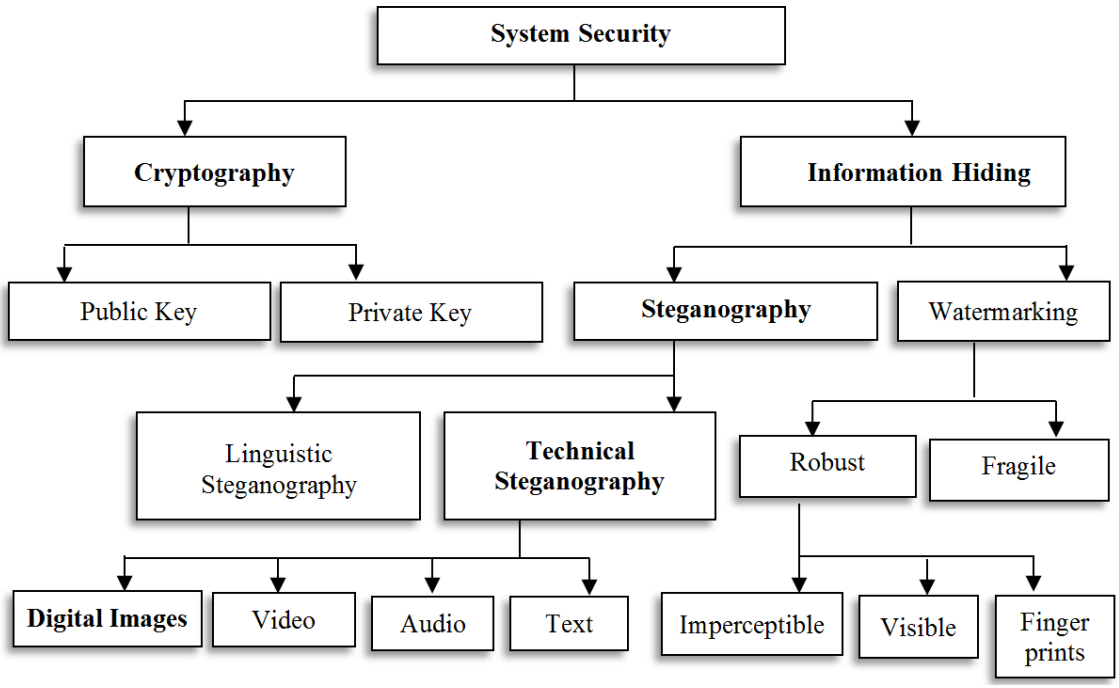


Figure 2.1: Taxonomy of Security System (Singh and Attri, 2015)

2.3 Cryptography

The science of encryption, known as cryptology, encompasses cryptography and cryptanalysis. Cryptography, which comes from the Greek words kryptos, meaning "hidden," and graphein, meaning "to write," is the process of making and using codes to secure the transmission of information (Z'aba and Maarof, 2006). Cryptanalysis is the process of obtaining the original message (called the plaintext) from an encrypted message (called the ciphertext) without knowing the algorithms and keys used to perform the encryption. Encryption is the process of converting an original message into a form that is unreadable to unauthorized individuals—that is, to anyone without the tools to convert the encrypted message back to its original format. Decryption is the process of converting the ciphertext message back into plaintext so that it can be readily understood (Michael and Herbert, 2011).

An illustration of the cryptography process is shown in Figure 2.2. It is clear that at the sender side the plain text is encrypted using a certain key. Then the cipher text is transferred through a communication channel to the recipient. Finally, at the receiver side the cipher text is decrypted using a key. This key may be the same or different from the sender key.

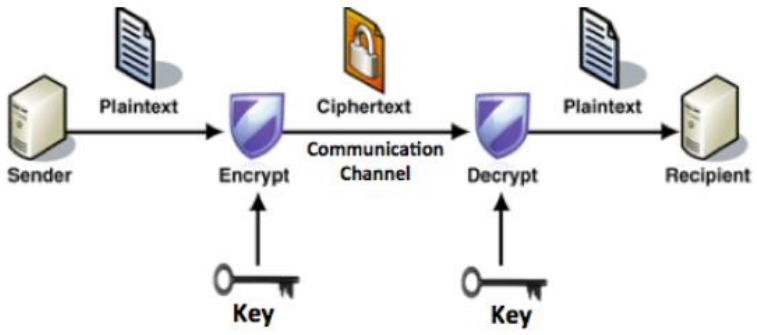


Figure 2.2: Encryption Process (Zohreh and Jihad, 2014)

Different algorithms and techniques exist to encrypt a plain text. Encryption algorithms are classified based on the type of transformation and keys. The use of key in different

algorithms is handled differently. In some algorithms the key should be agreed and distributed between two parties prior to the communication of encrypting the cipher text, while others do not have this restriction. Symmetric-key cryptography is a cryptography approach in which the encryption and the decryption algorithms use the same key such as Advanced Encryption Standard (AES). Public-key cryptosystems use two keys in which the sender uses a key different from that of the receiver. These keys are determined by some mathematical functions (Zohreh and Jihad, 2014).

For this research, the message was first compressed; using compression will decrease number of secret bits which leads to decrease number of modified pixels. The compression algorithm used was deflate and is one of the most commonly method adopting lossless compression technique and numerous applications use this method as it has a high performance (Yazdanpanah and Hashemi, 2011). Deflate uses combination of LZ77 and Huffman coding. The LZ77 algorithm uses the seen word as an entry in the dictionary. If a duplicate string occurs, then the second string is replaced by an index of dictionary of the duplicate string. The Huffman coding is a statistical probability for estimating the occurrence of all symbols. Then a Binary Tree is constructed according to probability size from downward to upward and encoding is conducted (Weimin, 2008). Afterward, AES encryption technique is used to encrypt the compressed text. AES is one of the most popular symmetric encryption algorithms. In symmetric encryption algorithm the same key is used for encryption and decryption. AES 128-bit data block-cipher operates on 4x4 byte matrix. Numbers of transformations are applied to convert the plaintext to ciphertext. According to (Thambiraja *et al.*, 2012) Four transformations are applied which are: SubBytes replaces one byte with another, ShiftRows shifts the rows cyclically with a specified offset, MixColumns is linear transformation that combines 4 bytes of each

column and KeyAddition applies bitwise XOR of the data block with round key. Encryption is used with the proposed steganography method to add another level of security. In order to ensure the security of the message, compression and encryption were both used.

2.3.1 Features of Cryptography

Adequate cryptography systems must achieve some goals to ensure high security. According to (Thambiraja *et al.*, 2012) the features of cryptography are summarized as follows:

a. Confidentiality: This is to ensure that the information is encrypted and can only be correctly decrypted by the authorized party. Confidentiality depends on how strong encryption algorithm is which is difficult to be broken. It is the assurance that no one other than the intended authorized party can read the information.

b. Authentication: Is the assurance that the received information is arrived from an authorized person not a false identity.

c. Integrity: This is the assurance that the transmitted information is not modified or altered by some unauthorized person. It ensures that the received secret message is not corrupted over communication channel.

d. Non-Repudiation: Is making sure that first party or the second party should not be able to deny transmission.

e. Access control: This is the last aspect of Cryptography that is used to prevent authorized parties from rejecting actions they performed.

2.4 Steganography

Steganography is a field in the domain of information hiding. The numbers of research works in this area have increased. With the boost of the Internet technology, steganography becomes a trend for exchanging information, and has existed many years ago. The word steganography (the art of secret writing) is derived from the Greek words steganos, meaning "covered" and graphein, meaning "to write." While steganography is technically not a form of cryptography, it is another way of protecting the confidentiality of information in transit (Michael and Herbert, 2011). It is done in a way that the existence of secret communication between two parties or more are hidden to attackers. The secret messages can be hidden in less suspicious digital object that is not detectable by human perceptibility. Then the person who possesses the stego-key can only extract the message from the stego-object at the target destination (Rodrigues *et al.*, 2004).

Figure 2.3 illustrates the general steganography model used to embed and extract a message. This example uses an image as the cover object to carry bits of the secret message. At the sender's side, the secret message is embedded into the cover image by using some embedding function and the stego-image can be parameterized by a stego-key. Then the stego-image is sent over a communication channel to the receiver. The communication channel can be any type of transmission technologies that exist such as the Internet. Then at the receiver's side the secret message is extracted using the extracting function. In addition, the extracting function uses the stego-key if it was used in the embedding phase (Cheddad *et al.*, 2010).

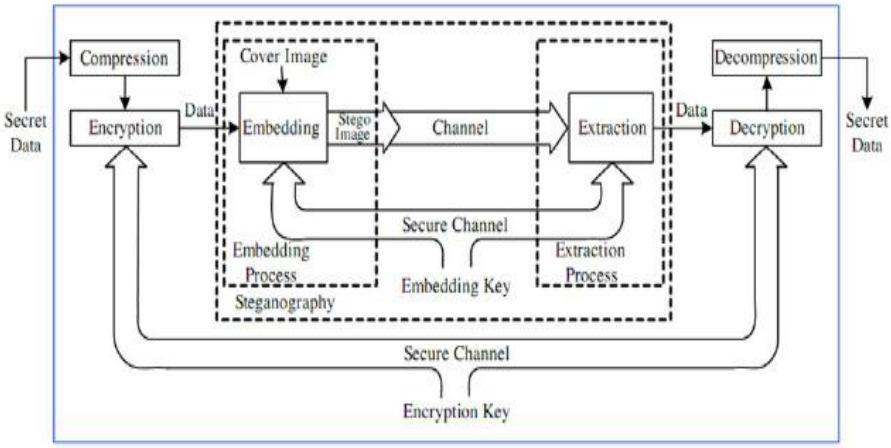


Figure 2.3: General Steganography Model (Subhedar and Mankar, 2014)

2.4.1 Features of Steganography

There are various features that characterize the advantages and disadvantages of a steganographic technique. Relative importance of each component depends on the application (Bender *et al.*, 1996). According to Zaid and Ahmad, (2015), for the stenographer it is important to show and analyze the relation between the three factors (capacity, robustness and security) to make them work together. This relation can be presented by the steganography triangle**.** Figure 2.4**,** represents a balance triangle each of its ribs specifies a factor associated with a steganographic method. So that for instance in order to improve capacity, you sacrifice security. It makes sense that the more embedding in an image the more probability that an observer will notice the degradation and suspect something is out of place. It is obvious that improving one factor will affect the other factors so that any steganography method must take care of the three factors at all times. Trying to keep the triangle as balanced as possible you have to change the other two elements.

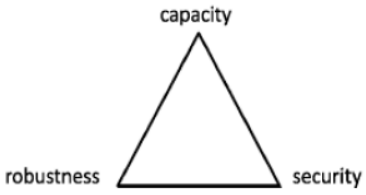


Figure 2.4: Steganography Triangle (Zaid and Ahmad, 2015)

The following paragraphs will explain each type of the properties that characterize steganography.

a. Security: Is one of the characteristics of steganography. In which the confidentiality is guaranteed by embedding the sensitive information in a way that is invisible and secured.

The authentication and identification are provided by steganography. To ensure identification between sender and receiver, a shared security key is used. The successful extraction of the embedded message ensures the authentication (Zohreh and Jihad, 2014). The embedded information cannot be removed beyond reliable detection by targeted attacks based on a full knowledge of the embedding algorithm and the detector (except a secret key), and the knowledge of at least one carrier with hidden message (Fridrich, 1999).

b. Payload (Embedding) capacity: This refers to the amount of secret message that a stego-image can carry before the distortions become noticeable. This is another significant criteria in steganography used to embed as much as information possible in a stego-object without degrading the object's quality. Many research works are conducted on the area of embedding more messages with fewer modifications (Hong and Chen, 2012). There is a tradeoff between the invisibility and payload capacity, such that as the capacity rate increases the invisibility decreases.

c. Robustness: This means after embedding, data should stay intact if stego-image goes into some operations such as cropping, rotation, resizing, scaling, filtering and addition of noise (Mehdi and Hussain, 2013). This robustness is the goal of this research work, which aims at showing that when a stego-image is generated from our algorithm it should be able

to withstand the image operations after embedding secret message to the cover image. That is, this measures the amount of alteration that the stego-image can take against attacks before destroying the secret information. Attacks can be either intentional or unintentional. The intentional attacks involve eavesdroppers trying to remove the embedded information or adding noises to the image to destroy the hidden message. However, unintentional attacks include manipulating the object by compression and conversion (Zohreh and Jihad, 2014). It is preferable for steganographic algorithms to be robust against either malicious or unintentional changes to the image.

The other features of steganography recognized in (Imaculate and Ashok, 2013) are: undetectability and invisibility.

d. Undetectability: This means that the existence of the secret information should be undetectable whenever the stego-image is analyzed. The output generated from the algorithm must be visually similar to the human eye. It is critical in any steganography system to provide undetectability feature. The steganography systems that use statistical analysis to detect the existence of secret messages in a stego-object are called steganalysis (Provos and Honeyman, 2003).

e. Invisibility: Is a mandatory property for the steganography's system. The changes in the cover image should be imperceptible to human visual system (HVS) characteristic. This means that the modification conducted to the cover-object should not be visible to the human naked eyes (Zohreh and Jihad, 2014). When a person views a cover image, he or she should be unable to distinguish the image with embedded information from an image without embedded information. The goal is that the before and after images appear identical (Sharmila and Shanthakumari, 2012).

2.5 Classification/ Categorization of Steganography

There are two ways to categorize steganography: either based on the techniques that are used in the process of embedding secret data into the carrier or based on the cover type that holds the secret information (Zohreh and Jihad, 2014).

2.5.1 Techniques Based on Algorithms

There are three major types of steganography techniques: injection, substitution and generation (Ashok *et al.*, 2010).

a. Injection-based Technique: This technique is known as 'insertion'. It involves injecting the secret message into the cover object. The secret message is hidden in an invisible part of the cover object. The disadvantage of this method is that the size of the stego file will be larger than the size of the original cover file, but the contents of the cover file will not be altered (Hassanein, 2014).

b. Substitution-based Technique: The substitution-based technique is very popular.

It overcomes the size enlargement problem of the cover image related to the injection technique. The most common substitution technique is 'least significant bits (LSB) replacing' technique used at the spatial domain, where the bits of the secret message substitute the LSB of the cover file. However, depending on the cover file and hiding algorithm used, substitution may result in degrading the quality of the cover file (Fridrich, 2009)

c. Generation-based Technique: Unlike both previous methods, this method does not need a cover file since it creates a cover file by itself specifically for the purpose of hiding the secret message. The properties of the generated cover file are usually dependent on the secret message structure (Fridrich, 2009).

In addition to these methods of information hiding presented, Kipper, (2004) further identifies six categories of information hiding techniques namely: substitution, transform domain, spread spectrum, statistical method, distortion, and cover generation.

Categorizing information hiding based on the hiding techniques used is one approach. An alternative approach is considering the type of the cover file that is used to carry the embedded information. This approach is presented in next sub-section.

2.5.2 Based on Cover Type

Almost all digital file formats can be used as cover file to contain hidden data or secret messages. However, the ability of such files to embed secret data depends on the availability of redundant or insignificant areas within these files. The redundant bits of a cover file are those bits that can be altered without the alteration being detected easily (Anderson and Petitcolas, 1998). Redundancy is defined as the bits of a cover object that provide accuracy far greater than necessary for the object's use and display. For example, image files can display 16 million different colours, while the human eye is only able to perceive about 10 million different colours (Owens, 2002). The file formats with a high degree of redundancy is preferable, since redundant bits can be substituted with secret information in an imperceptible and undetectable manner. Thus, in the case of interception by an unauthorized recipient, the appearance of cover files should not indicate the existence of hidden data. Many kinds of digital media such as text, image, audio, and video files can be used as cover files. The properties of cover files vary from one type to another depending on the redundancy created in the digital representation and the unique characteristics of the file format. These properties control how the secret data can be hidden in the digital representation of the cover files. Thus, most information hiding techniques are classified according to the cover file used (Cole, 2003).

2.6 Digital Images

An image is the most common type of digital media used for steganography (Anderson and Petitcolas, 1998). Digital images often have a large amount of redundant data and for this reason it is possible to hide message inside image file (Chandramouli and Memon, 2001). To a computer, an image is an array of numbers that constitute different light intensities at a number of points (Johnson and Jajodia, 1998). This numeric representation forms a grid and the individual points are referred to as picture elements or pixels. These pixels make up the image's raster data (Shamim and Hemachandran, 2012).

Figure 2.5 shows an image of a deer and zooming part of its area, you can notice the rectangular pixels. Value is assigned to each pixel to represent a color or gray level. Values are in binary code and formed of zeros and ones known as bits where the bits are stored in sequence, each pixel can be represented by a set of bits (Zohreh and Jihad, 2014).

For instance, if an image consists of black and white pixels, each pixel represents 8 bits. Then black pixels are represented as (00000000) while the white pixels are represented as (11111111). In a case whereby each pixel is represented by 4 bits then the black pixels are (0000) and the white pixels are (1111).

The number of bits used to represent each pixel determines bit-depth of an image. The bit-depth implies to the number of bits used in color schema to represent a gray-scale level or a color. As the bit-depth increases, the number of gray-scale level or color representation increases. For example if depth of an image is 4 bits then the total of 2^4 =16 colors can be displayed. This means each pixel composes of 4 bits. Each pixel can have any color value from 0 to 16. Also an image of 8-bit depth can be represented by 2^8 = 256 color. For example Figure 2.5 shows an image of 24-bit depth. Therefore each pixel is represented by a value from 0 to 2^{24}.

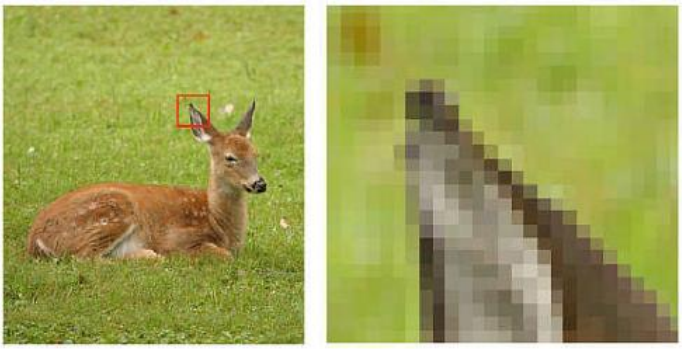


Figure 2.5: An Image Deer and Pixels of Part of the Image (Zohreh and Jihad, 2014)

Colour Image Formats are represented as Graphical Interchange Format (GIF) and Portable Network Group (PNG) and are examples of widely used file format based on palette. In addition Bitmap (BMP) can be stored as gray-scale palette with full RGB value. Tagged Image File Format (TIFF) is a flexible image format that may contain multiple images of different types in the same file through tags. Joint Photographic Experts Group (JPEG) is an image format that was approved as an international standard in 1994. JPEG is usually lossy, but may also be lossless and has become a popular format for image representation on the Internet (Knut, 2008).

Johnson and Jajodia, (1998) Suggested 24-bit image files to be used as cover images in steganography systems. Also, Zohreh and Jihad, (2014) observed that the 24 bit-depth (RGB model) images are common over the Internet. Colour images are preferred over other types because the shades change very gradually if changes are done to the pixel. So it may serve the best result for steganography purpose. In this dissertation, concentration was on colour images for the image steganography system especially GIF images of dimension 512 X 512.

2.7 Image Steganography

Coding secret messages in digital images are most widely used today. Image steganography is about exploiting the limited power of the HVS (Johnson and Jajodia, 1998). If any specific colour is viewed closely it has been observed that single digit modifications to the contribution level are imperceptible to the human eye (that is, a pixel with a value of (255, 255, 0) is indistinguishable from (254, 255, 0) in RGB colour representation (Shamim and Hemachandran, 2012). Figure 2.6 shows a general image steganography. Image steganography techniques can be classified into two broad categories: Spatial-domain based steganography and Transform-domain based steganography (Mohan and Singh, 2015). These methods are discussed in detail in forthcoming subsections.

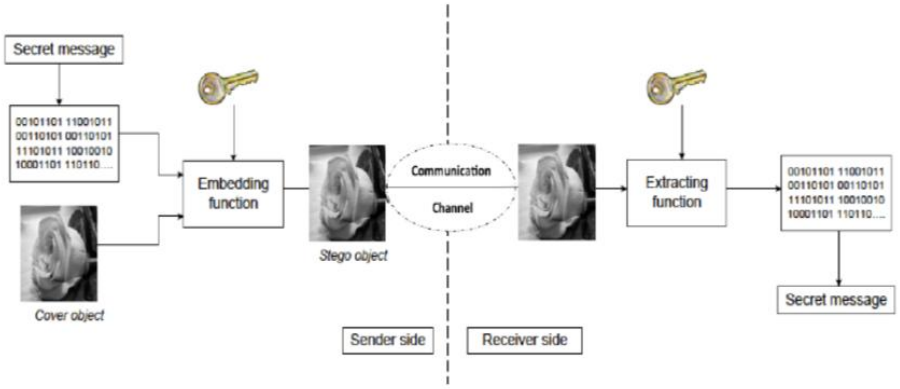


Figure 2.6: General Model of Image Steganography (Zohreh and Jihad, 2014)

2.7.1 Spatial Domain Based Image Steganography

In this method, the pixel value is directly modified for data hiding. Images in this domain are represented as rectangular grid of pixels or points of color where the human perception does not observe the image as a grid. Also known as substitution techniques, consists of simple techniques that create a covert channel in the parts of the cover image in which changes are likely to be imperceptible to the HVS (Shamim and Hemachandran, 2012). The various approaches to achieve embedding in spatial domain are shown in Figure 2.7.

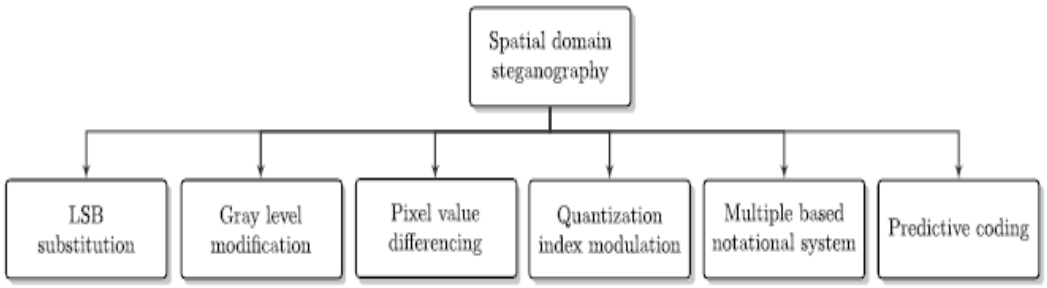


Figure 2.7: Spatial Domain Steganography Techniques (Subhedar and Mankar, 2014)

Least Significant Bit (LSB) Substitution is an embedding method based on the fact that the least significant bits in an image can be thought of as random noise, and consequently they become not responsive to any change on the image (Kharrazi *et al.*, 2004). Least significant bit (LSB) is the most commonly used type of insertion scheme used currently in digital steganography (Memon and Chandramouli, 2001). This method is probably the easiest way of hiding information in an image and yet it is surprisingly effective (Tiwari and Shandilya, 2010). The secret message is hidden by altering least significant bit in a certain layer of the image file. This change is so slight that the human eye may not notice it (Shamim and Hemachandran, 2012).

The LSB hides the message bits into the image pixels either in a sequential or randomized fashion. Then create a path for replacing the least significant bits of the image with the message bits. If the path is randomly generated then the pseudo random number generator PRNG is used (Rodrigues *et al.*, 2004). The PRNG should be seeded with some stego-key that is shared between the sender and receiver. In this way the message bits will be spread over the stego-image. The extraction phase is the inverse of the embedding phase. At the receiver side the path is created based on the stego-key. First the length of the secret message is recovered by retrieving the least significant bits of the pixels. Then the pixels

are traversed based on the path and least significant bit of each pixel is retrieved. This process of traversing all the pixels continues until reaching the end of the message length.

LSB is the lowest bit in a series of numbers in binary. Example in the binary number: 10110001, the least significant bit is at the far right 1. In other words, least significant bit of a pixel is the redundant bit which is the most right bit of a byte. From the following example embedding can be achieved by simply replacing LSB of the randomly selected pixel in the cover image with the secret message bit.

Let's assume that we want to embed the letter 'A' as the secret message into a 24-bit cover image. The binary value of 'A' is 10000011. Assume the three adjacent pixels of the image are the following:

(10110100 11010111 10001110)

(00011100 11110110 11010111)

(10001110 00011100 11100101)

After replacing the LSB of each byte of the cover image with the bits of the secret message, the following pixels of stego-image is acquired. Bits that have been changed because the cover image pixels did not match the message bits are italicized and unbold.

(1011010*1* 1101011*0* 10001110)

(00011100 11110110 1101011*0*)

(1000111*1* 0001110*1* 11100101)

The algorithm first selects a pixel (*xi*) sequentially. Then it checks whether the least significant bit of (*xi*) matches with the message bit (*mi*). If *LSB* (*xi*) = *mi* then no change otherwise LSB of *xi* is substituted with *mi*. Then it selects the next pixel and message bit and checks whether they match or not. This process continues until reaching the end of

secret message bits where all secret bits are embedded in the image (Karaman and Sagiroglu, 2012).

Another approach of LSB based steganography include adaptive LSB substitution based on brightness, edges and texture masking of the host image to estimate the number of LSBs for data hiding Yang *et al.*, (2009) such as Sobel Edge Detector.

In edge detection techniques, operators such as Sobel Edge Detector are used to conduct edge detection of cover image (Vincent and Folorunso, 2009). It is an orthogonal gradient operator and also a gray weighted algorithm of the adjacent points in two directions. It detects edges of the point according to its adjacent points. The secret message is then hidden in the detected edges. Sobel edge operator is convolved for the y and x direction as follows:

$$Gx = f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1) - \{f(x-1, y-1) + 2f(x-1, y) + f(x-1, y+1)\}$$

$$Gy = f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1) - \{f(x-1, y-1) + 2f(x, y-1) + f(x+1, y-1)\} \quad \ldots (1)$$

The gradient is computed for each point $f(x, y)$ as follows:

$$\text{Gradient} = |Gx| + |Gy| \quad \ldots (2)$$

The gradient of points in an image required to be estimated to identify the edges. It is based on the Sobel operator of 3x3 kernels convolved on the image to calculate approximate derivate of horizontal change and vertical change (Deng *et al.*, 2011). The partial derivatives of the formulas above need to be calculated for each pixel location. In practice, we often use small area template convolution to do approximation. G*x* and G*y* need a template each, so there must be two templates combined into a gradient operator (Elham *et*

al., 2013). The two 3×3 templates used by Sobel edge operator convolved for the y and x direction are as follows in Figure 2.8:

-1	0	+1
-2	0	+2
-1	0	+1

G_x

+1	+2	+1
0	0	0
-1	-2	-1

G_y

Figure 2.8: Sobel Edge Masks (Elham *et al.*, 2013)

Many works that hide messages in LSB based steganography exist, and some of them include LSB Matching Revisited (Mielikainen, 2006) and Edge Adaptive image steganography based on LSB Matching Revisited (Luo *et al.*, 2010).

2.7.2 Transform (Frequency) Domain Based Image Steganography

Images that are in the domain of transform space are more robust in terms of some image processing manipulation and lossy compression and less prone to attacks (Subhedar and Mankar, 2014). Images of this type are extensively transmitted over the Internet communication and JPEG is the most popular type of this format (Provos and Honeyman, 2003). To obtain the frequency domain representation, image transforms are used and are designed to possess two main properties: (a) Reduce image redundancy (b) Identify less important parts of image by isolating various frequencies in image. Frequency domain representation depicts that low frequencies correspond to significant image features and high frequencies represent less important image details.

This technique is highly robust against image compression, tampering, resizing, filtering and cropping. The hidden data cannot be lost with image manipulation (Mali *et al.*, 2012).

Various image transforms that can be employed for data embedding, which includes Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Haar Transform,

Hadamard Transform, Integer Transform, Contour let Transform, Ridgelet Transform and Ripplet Transform (Subhedar and Mankar, 2014).

2.8.2.1 Overview of 2D-DCT

Discrete Cosine Transform (DCT) as an orthogonal transform for digital image and signal processing that transforms signal or image values from the spatial domain to the frequency domain by separating the image into high, middle and low frequency components (Walia *et al.,* 2010). It has advantages such as high compression ratio, small bit error rate, good information integration ability, good synthetic effect of calculation complexity and low computation time (Kaur *et al.*, 2011). Therefore employing it in steganography won't add any vast complexity.

2D-DCT is based on two dimensional correlation of pixels (a pixel tends to resemble all its near neighbors not just those in its row). It hides the secret bits in significant parts of the cover file. The technique tries to encode message bits in the transform domain coefficients of the image. Data embedding performed in the transform domain is widely used for robust watermarking. Similar techniques can also realize large embedding capacity for steganography (Katzenbeisser and Petitcolas, 2000).

This research embedding method employs 2D-DCT on every 8×8 non-overlapping block of the detected image. This restriction is partly for simplicity of exposition and image compression, the DCT is typically restricted to this size. Rather than taking the transformation of the image as a whole, the DCT is applied separately to 8×8 blocks of the image (Watson, 1994). The technique involves forward DCT, quantization, zigzag scanning and inverse DCT.

a. Forward 2D-DCT: Before computing the DCT of the 8×8 block, its values are shifted from a positive range to one centered on zero. For an 8-bit image, each entry in the original block falls in the range 0, 255. The midpoint of the range (in this case, the value 128) is subtracted from each entry to produce a data range that is centered on zero, so that the modified range is (-128,127). This step reduces the dynamic range requirements in the DCT processing stage that follows.

The 2D-DCT function for an input image F and an output image *C*, given a two-dimensional 8×8 image $f(x, y)$, it's discrete cosine transform $C(u, v)$ is defined as:

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{8-1}\sum_{y=0}^{8-1} f(x, y) \cos\left[\frac{(2x+1)u\pi}{16}\right] \times \cos\left[\frac{(2y+1)v\pi}{16}\right] \quad \ldots (3)$$

Where: u is the vertical spatial frequency, for the integers $0 \leq u \leq 8$, v is the horizontal spatial frequency, for the integers $0 \leq v \leq 8$, $\alpha(u) = \begin{cases} \frac{1}{\sqrt{8}} \\ \sqrt{\frac{2}{8}} \end{cases}$ and $\alpha(v) = \begin{cases} \frac{1}{\sqrt{8}} \\ \sqrt{\frac{2}{8}} \end{cases}$ are normalizing scale factor to make the transformation orthonormal. x and y are variables ranging from 0 to 8-1 and 0 to 8-1, where u and v are the cosine coefficients for vertical and horizontal index respectively. $f(x, y)$ is the pixel value at coordinates (x, y) and $C(u, v)$ is the DCT coefficient at coordinates (u, v).

The DCT block F consists of 64 DCT coefficients. The top-left coefficients F(0,0) correlates to lower frequency of the original image block, which is called DC coefficient, also called constant components. As we move away from the F(0,0) in all directions the DCT coefficients correlate to higher and lower frequencies, where F(7,7) corresponds to the highest frequency. A sample DCT block is shown in Figure 2.9.

$$F = \begin{bmatrix} 162 & 40 & 20 & 72 & 30 & 2 & -1 & -1 \\ 30 & 108 & 10 & 32 & 27 & 5 & 8 & -2 \\ -94 & -60 & 12 & -43 & -31 & 6 & -3 & 7 \\ -38 & -83 & -5 & -22 & 3 & 5 & -1 & 3 \\ -31 & 17 & -5 & -1 & 4 & -6 & 1 & -6 \\ 0 & -1 & 2 & 0 & 2 & 2 & 8 & 2 \\ 4 & -2 & 2 & 6 & 8 & -1 & 7 & 2 \\ -1 & 1 & 7 & 6 & 2 & 0 & 5 & 0 \end{bmatrix}$$

Figure 2.9: Sample DCT Block (Deepika and Rita, 2014)

b. Quantization: DCT co-efficient is now ready for compression by quantization. A useful feature in the JPEG process in this step varying image compression and quality is obtainable through the selection of specific quantization table. The standard quantization matrix of JPEG uses quality factor (α) 50 as shown in Figure 2.10 of a.

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

a

$$F^Q = \begin{bmatrix} 10 & 4 & 2 & 5 & 1 & 0 & 0 & 0 \\ 3 & 9 & 1 & 2 & 1 & 0 & 0 & 0 \\ -7 & -5 & 1 & -2 & -1 & 0 & 0 & 0 \\ -3 & -5 & 0 & -1 & 0 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

b

$$F^D = \begin{bmatrix} 160 & 44 & 20 & 80 & 24 & 0 & 0 & 0 \\ 36 & 108 & 14 & 38 & 26 & 0 & 0 & 0 \\ -98 & -65 & 16 & -48 & -40 & 0 & 0 & 0 \\ -42 & -85 & 0 & -29 & 0 & 0 & 0 & 0 \\ -36 & 22 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

c

Figure 2.10: (a) The Quantization Table (b) Quantized DCT Block (c) Dequantized DCT Block (Deepika and Rita, 2014)

Quantization is achieved by dividing each element in the DCT coefficient block by the corresponding value in the quantization matrix, and the result is rounded to the nearest

integer. As eye is not able to discern the change in high frequency components so these can be compressed to larger extent. Lower right side components of quantization matrix are of high value so that after quantization high frequency components become zero (Singla and Syal, 2012). The quantized DCT coefficients $C^{Q}(u, v)$ is computed by:

$$C^{Q}(u, v) = \left[\frac{C(u,v)}{Q(u,v)}\right] \quad \ldots(4)$$

Where: $C(u, v)$ is the image transform coefficients and Q(u,v) is a 64-element quantization table. The quantized DCT block and dequantized DCT block are both shown in Figure 2.10 of b and c respectively.

c. Zigzag Scanning: As part of entropy coding which is a special form of lossless data compression involves arranging the image components in a "zigzag" order employing run-length encoding (RLE) algorithm that groups similar frequencies together, inserting length coding zeros, and then using Huffman coding on what is left into one dimensional array. Although the DCT coefficients have been decorrelated by DCT transform to some extent, DCT coefficients in the same block are still not independent, which is called as intra-block correlation (Tu and Tran, 2002). While neglecting the impact of block edge, the general trend in magnitude of the block coefficients in each block is non-increasing along zigzag scan order. After block DCT coefficients are arranged by zigzag scan pattern, dependencies among neighboring coefficients in both horizontal and vertical directions can be conveniently investigated (Zhiping and Maomao, 2009). For example, the sequence pairs (5,6) and (14,15) describe the correlations along horizontal direction, while the sequence pairs (2,3) and (10,20) describe correlations along vertical direction. The secret message was hidden in the quantized DCT coefficients of the cover image as shown in Figure 2.11.

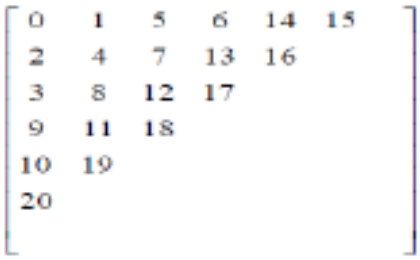


Figure 2.11: Zigzag Scan Order (Deepak and Rupali, 2012)

d. Inverse DCT (IDCT): After embedding the secret message, IDCT and dequantization is applied on the image to reconstruct the image and the output stego-image is generated (Anupam and Shiladitya, 2015). The IDCT is defined as:

$$f(x, y) = \frac{1}{4}\sum_{u=0}^{8-1}\sum_{v=0}^{8-1}\alpha(u)\alpha(v)f\ (x,\ y)\ \cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right] \quad \ldots(5)$$

For $x = 0 \ldots 8 - 1$ and $y = 0 \ldots 8 - 1$

For this research, the Sobel edge detection was employed first to the cover image. The resultant cover image was then subjected to 2D-DCT so as to convert it from spatial domain to frequency domain. Then the secret message was embedded in the coefficients of the detected cover image.

2.8 Steganalysis

Steganalysis is the science of detecting hidden information. The objective of steganalysis is to break steganography and the detection of stego image is the goal of steganalysis. Most of steganalysis algorithms rely on steganographic algorithms introducing statistical differences between the cover image and stego image (Hemang and Nehal, 2015). Different types of steganalysis include visual attack and statistical techniques. Statistical steganalysis

technique was adopted for our steganography system to analyse the rate of modification done to the cover image for invisibility and embedding purposes.

2.9 Related Work

There are numerous emerging approaches used to hide secret data, image steganography techniques that have been developed with LSB approach were reviewed and reported. Mielikainen, (2006) proposed the LSB matching revisited algorithm (LSBMR). The LSBMR uses two pixels; first pixel (x_i) is used to embed the secret message bit (m_i) and the binary relationship between both pixels value x_i and x_{i+1} is used to embed another message bit (m_{i+1}). The relationship between both pixels is based on the following binary function:

$$f(x_{i-1}, x_{i+1}) = \text{LSB}\left(\left\lfloor \frac{x_i}{2} + x_{i+1} \right\rfloor\right) \quad \ldots(6)$$

The embedding procedure for a unit composes of pair of cover image pixel (x_i, x_{i+1}) and message M bits (m_i, m_{i+1}) as input. Then after embedding (m_i, m_{i+1}) into (x_i, x_{i+1}), the algorithm produces the stego- pixels (y_i, y_{i+1}) as output. The algorithm of LSBMR for embedding a unit of two consecutive pixels is depicted in Figure 2.12. The pairs of pixels are selected randomly by using PRNG seeded with a shared stego-key.

The algorithm checks if the first message bit (m_i) matches the LSB pixel (x_i) of the first cover image, then the stego pixel $y_i = x_i$ (x_i remains unchanged), otherwise the stego pixel $y_{i+1} = x_{i+1}$ (x_{i+1} remains unchanged). In case m_i = LSB (x_i) and m_{i+1} does not match the binary function $f(x_{i-1}, x_{i+1})$ then $y_{i+1} = x_{i+1} \pm 1$. The algorithm either increases or decreases by one based on even- and odd- valued regions. In addition it would not introduce the LSB approach asymmetry property. The extraction process of the secret message from the stego-image is the opposite operation of embedding. At the beginning, units consisting of two pixels are selected randomly based on PRNG using the same stego-key used to embed the

secret message. Then for each unit along the order, two bits are retrieved. The LSB of the first pixel is extracted as first secret bit (m_i) and then the LSB of binary relation between both pixels is extracted as second secret bit (m_{i+1}). The stego-image produced by embedding secret message using this technique has better invisibility compared to the LSB algorithm. It resists the steganalysis attacks such as RS-analysis that exploit the asymmetry property of the stego-image. The security is maintained since LSBMR selects pixels using stego-key like LSB algorithm. However, the algorithm does not take into consideration the difference between a pair of pixels and does not consider the relationship between the message size and the content of cover image. As mentioned in (Anderson and Petitcolas, 1998) not all pixels are suitable to be modified. The LSBMR embedding algorithm was used as the embedding algorithm for this research.

Input: a pair of cover image pixels x_i, x_{i+1} and two bits of secret message m_i, m_{i+1}

Output: a pair of stego-image pixels y_i, y_{i+1}

Case 1: if $(m_i = LSB(x_i))$ & if $(m_{i+1} \neq f(x_i, x_{i+1}))$
$(y_i, y_{i+1}) = (x_i, x_{i+1} \pm 1)$

Case 2: if $(m_i = LSB(x_i))$ & if $(m_{i+1} = f(x_i, x_{i+1}))$
$(y_i, y_{i+1}) = (x_i, x_{i+1})$

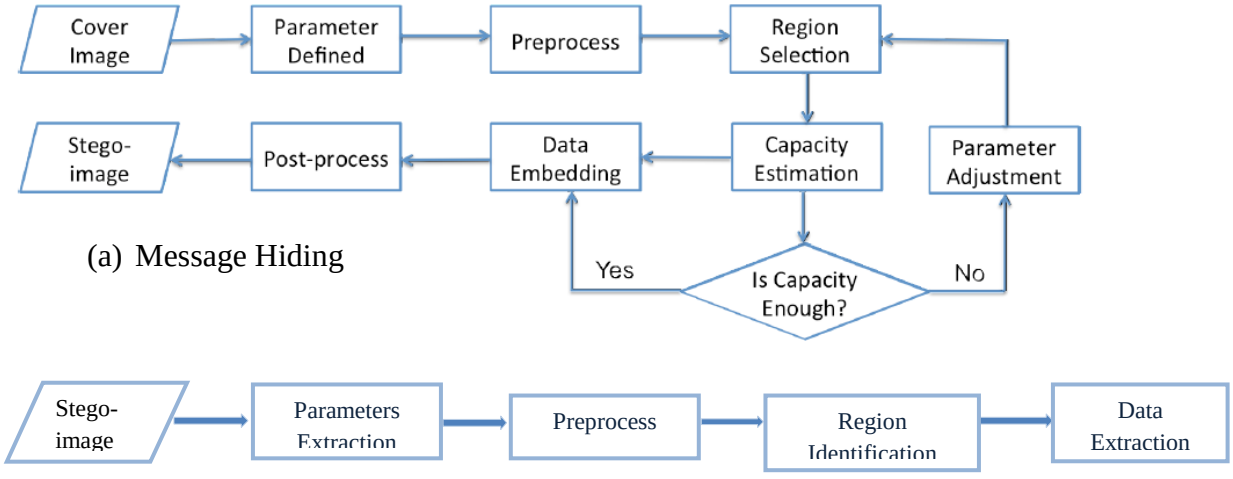
Case 3: if $(m_i \neq LSB(x_i))$ & if $(m_{i+1} = f(x_i - 1, x_{i+1}))$
$(y_i, y_{i+1}) = (x_i - 1, x_{i+1})$

Case 4: if $(m_i \neq LSB(x_i))$ & if $(m_{i+1} \neq f(x_i - 1, x_{i+1}))$
$(y_i, y_{i+1}) = (x_i + 1, x_{i+1})$

return (y_i, y_{i+1})

Figure 2.12: LSBMR Embedding Algorithm

Luo *et al.*, (2010) uses pixel value differencing approach to select regions and LSBMR as data hiding algorithm based on the message size and the content of the cover image. First a preprocessing of the image is conducted such as dividing cover image into number of non-overlapping blocks *Bz* and rotating each by some degree. Another parameter threshold T is initialized for region selection. T determines the units of two consecutive pixels to be selected. Where the units are selected whose absolute difference of the two consecutive pixels are greater than or equal T. T is calculated based on the message size and the content of the cover image. The LSBMR is adopted as data hiding technique of secret bits into the selected units. Then adjustment of the units, whose difference drop less than T, is performed. This adjustment is important for correct extraction of the message bit. Post-processing is conducted to obtain the stego-image by rotating back the *Bz* blocks. Finally, Bz, T and length of the secret message are embedded into the stego-image. Figure 2.13 depicts the schema for data embedding and extraction.



(a) Message Hiding

(b) Message Extraction

Figure 2.13: (a) The Data Embedding (b) Extracting Process of Edge Adaptive based on LSBMR (Luo *et al.*, 2010)

The inverse operation is taken to recover the secret message at the receiver side. The most significant property of this technique is that it uses the edges of an image to embed information first and leaves the smooth areas depending on the capacity of the secret message. In addition the maximum capacity that can be achieved is 1 bit per pixel and the security is high since two keys were used. However, the edges are determined either only in horizontal or in vertical direction. It does not take into account neighboring pixels to detect the edges. In our proposed work, we considered the neighboring pixels to detect the edges. But, made use of the coefficient of the detected edges of the cover image for embedding of secret messages.

Zohreh and Jihad, (2014) proposed edge adaptive based on LSBMR using Sobel operator which uses Sobel Edge Detection technique to get edges. Then these edges are manipulated for embedding purpose. The flow diagram is depicted in Figure 2.14.

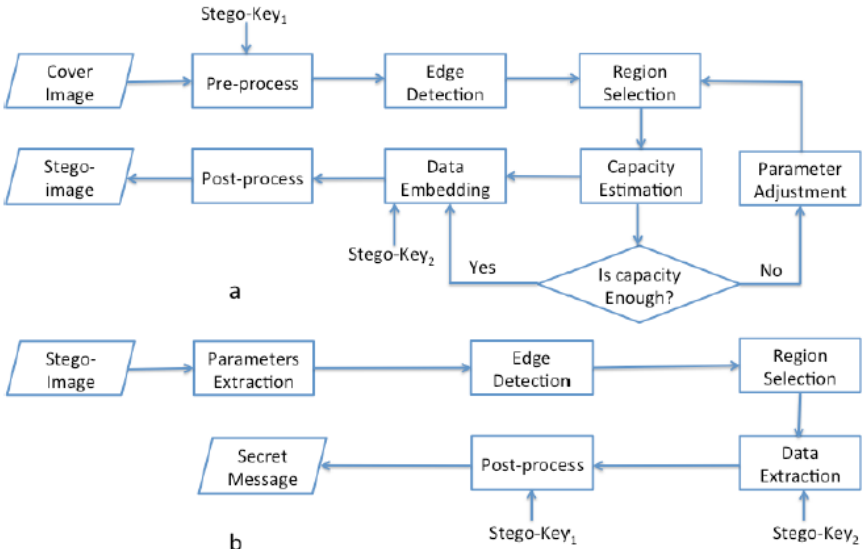


Figure 2.14: (a) Data Hiding Process (b) Data Extraction Process (Zohreh and Jihad, 2014)

The main advantage of the edge adaptive based on LSBMR using Sobel operator is that it first selects locations at sharper edges for data hiding based on the secret message size. The RS-steganalysis technique cannot detect the presence of secret information of the stego-

images. However, the robustness of this technique is low, since it works in spatial domain. The hidden information was lost and could not be recovered from stego-images when they undergo image operations such as cropping, rotation and resizing.

Hence, in this dissertation, we proposed an improved image steganography algorithm based on LSBMR using Sobel Edge Detection. The new technique has been applied to work in transform domain so as to improve the robustness of image steganography.

2.10 Gap in Literature

Although, spatial domain steganography schemes achieve high embedding capacities they are vulnerable to any small modifications that may result due to image processing operations such as cropping, rotation, scaling and resizing. The embedding of the secret message of the existing steganography algorithms were done in spatial domain.

Hence, this research extends the work of (Zohreh and Jihad, 2014) by improving on the robustness of the technique through the application of 2D-DCT to the detected edges of the cover image so as to produce a stego-image that can withstand image operations such as resizing, rotating and cropping and for the method to work in transform domain.

CHAPTER THREE: PROPOSED DESIGN OF IMAGE STEGANOGRAPHY BASED ON LSBMR USING SOBEL EDGE DETECTION

This part of the research presents the methods applied in our research in order to achieve the stated objectives. First part provides the readers with an opportunity to know about the approach of the study and the reasons behind the selection of the research method. Also, it presents the algorithm of the improved image steganography based on LSBMR using Sobel edge detection.

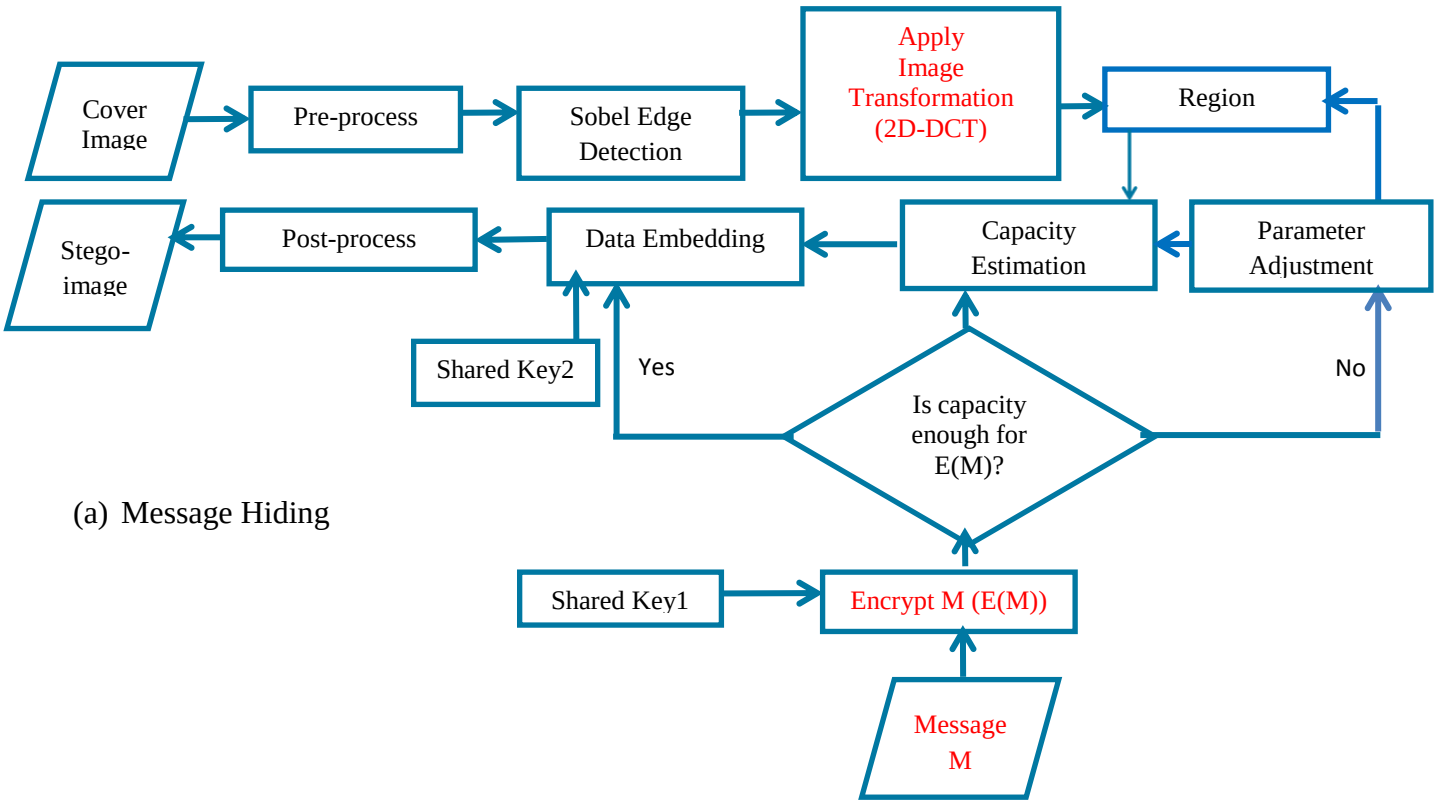
3.1 Improved Image Steganography based on LSBMR using Sobel Edge Detection

The improved image steganography based on LSBMR using Sobel Operator uses Sobel's edge detection technique to get edges and transform the detected edges into its co-efficient using 2D-DCT. These coefficients are manipulated for embedding purpose. For lower embedding rates the middle frequencies are used for holding information. The goal of this method is to preserve the statistical and visual features of the cover image and obtain a stego-image that is robust against image operations such as rotation, resizing and cropping. Selecting units for data hiding depends on the secret message and the co-efficient of the cover image content. Thus, the new technique is to provide better resistance against steganalysis process and image operations.

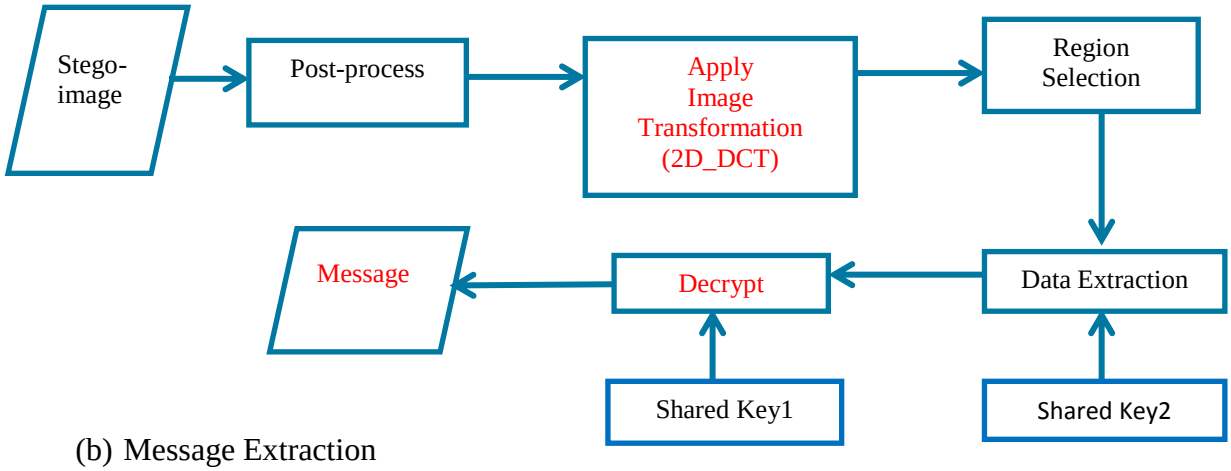
The next section described the details of the method.

3.2 Proposed System Flow Diagram

The flow diagram of the system is depicted in Figure 3.1. Figure 3.1(a) shows schema of the message hiding and (b) extraction process in parts. Figure 3.2 shows the combined detailed block system for message hiding and extraction process both on sender and receiver side.



(a) Message Hiding



(b) Message Extraction

Figure 3.1: Flow Diagram of the Proposed System (a) Message Hiding and (b) Message Extraction Process

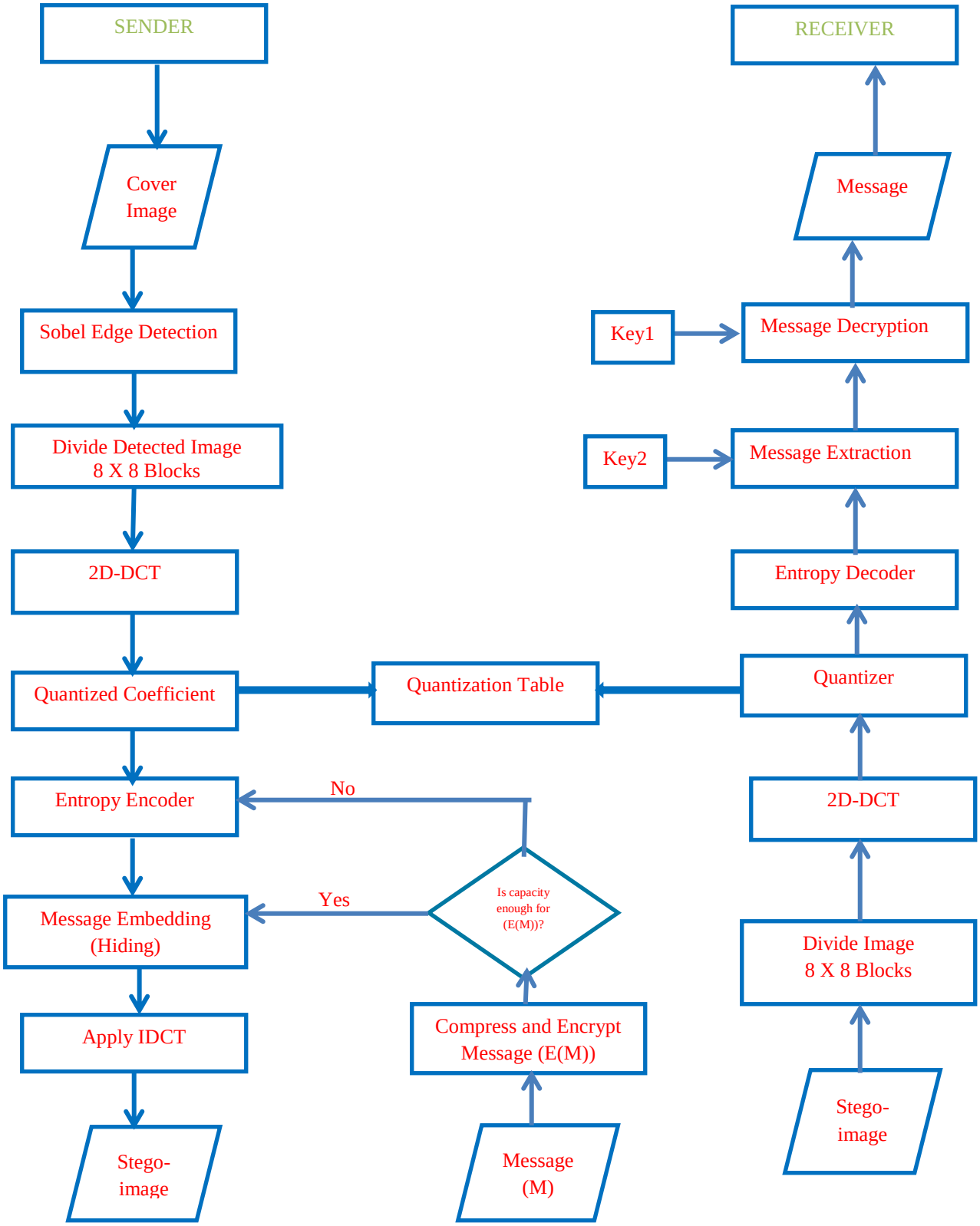


Figure 3.2: Combined Detailed Block Diagram of the Proposed System

The explanation of the system flow diagrams of figure 3.1 and 3.2 are stated in the next sub sections of 3.31 and 3.3.2.

3.2.1 Pseudocode for Proposed Message Hiding (Embedding) Process

The embedding algorithm is given as follows:

Input: An M×N size cover image and message to be hidden.

Output: Stego image.

Step1. First the text is compressed and encrypted using AES symmetric encryption. The key used for encryption at the sender side is the same as the key used at the receiver side which serves as stego-key1. Compression and encryption are used to reduce the amount of data required to be hidden in a cover image and to increase security respectively. The compression and encryption will only be used for implementing the application. In the experimental and result analysis this step was ignored.

Step2. The image is divided into 3x3 non-overlapping block. Subject it to Sobel edge detection based on the template for each center point of the 3x3 block of pixels of Figure 2.8 of Chapter Two.

Step3. Divide detected cover image into 8x8 blocks of pixels.

Step4. Shift the block from left to right, top to bottom, by subtracting 128 and apply 2D-DCT on each block of pixels using equation (3) in sub section of 2.8.2.1 of Chapter Two to transform it to its coefficients.

Step5. Perform quantization on the resultant coefficients of each block using equation (4), sub section 2.8.2.1 of Chapter Two.

Step6. Entropy encoder: carry out zigzag scan and RLE using Huffman algorithm to convert the quantized coefficients into one dimensional (1-D) array.

Step7. Calculate and replace the DC coefficients with encrypted message using the embedding algorithm that is LSBMR embedding algorithm of equation (6), section 2.10 of Chapter Two.

Step8. Perform Inverse DCT on each block using equation (5) of sub section 2.8.2.1 of Chapter Two after converting 1-D zigzag scan array back to 8 X 8 blocks of pixels.

Step9. Then combine all the blocks to form stego image. The stego-image is then produced.

3.2.2 Pseudocode for Proposed Message Extraction Process

To extract the message, the inverse operation is taken to recover the secret message at the receiver side. The stego-image is received in transform domain. Now stego image is divided into 8×8 blocks and DCT is performed on each block. Then scan the DCT block in zigzag way and extract the embedded data. Extraction algorithm is given as follows:

Input: Stego image.

Output: Message.

Step1. Read and divide the stego-image into 8x8 non-overlapping blocks of pixels.

Step2. Read bits-by-bits to get the coefficients.

Step3. Perform entropy decoder by dezigzag scanning to convert 8×8 blocks of the coefficients

Step4. Calculate the DC coefficients and extract the secret message using equation (6), section 2.10 of Chapter Two until it gets to the last pair of the coefficients.

Step5. Each block is dequantized using the quantization matrix.

Step6. Perform Inverse DCT (IDCT) on each block.

Step7. Shift the block by adding 128 to each value in the block and merge all the blocks to get the image.

Step8. Finally, decompress and decrypt the secret bits to obtain message and display it on screen.

3.2.3 Proposed Algorithm for Message Hiding (Embedding) Process

Input: An M×N size cover image and message to be hidden.

Output: Stego image.

1. Image img = get image
2. Let n = width (img)

   m = length (img)
3. Let x be message to hide
4. Comp msg =Deflate (x)
5. Encrypt msg = AES (encrypt deflate x) Shared Key1
6. Blocksimg [n] $\longleftarrow \lfloor img \rfloor$ 3X3 using Sobel …(1)
7. Detected image $\longleftarrow \lfloor blockimg[n] \rfloor_{8X8}$
8. For each block in detected image

   »Shift block

   ↓ ↓ Shift block

   Apply 2D-DCT on each block $C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{8-1}\sum_{y=0}^{8-1} f(x, y)$

   $\cos\left[\frac{(2x+1)u\pi}{16}\right] \times \cos\left[\frac{(2y+1)v\pi}{16}\right]$ …(3)
9. Apply Quantization to co-efficient $\longleftarrow C^{Q}(u, v) = \left[\frac{C(u,v)}{Q(u,v)}\right]$ …(4)
10. Compute DCT coefficients ⟵ Huffman encoding
11. (I –D) array ⟵ Zigzag scan

12. Embed DCT coefficients with encrypt deflate x ⟵ LSBMR $\left(\left\lfloor \frac{x_i}{2} + x_{i+1} \right\rfloor\right)$

    Shared key2 …(6)

13. Convert 1-D array to 8X8 block

14. For each block

    Perform IDCT ⟵ $f(x, y) = \frac{1}{4}\sum_{u=0}^{8-1}\sum_{v=0}^{8-1}\alpha(u)\alpha(v)f(x, y)\cos\left[\frac{(2x+1)u\pi}{16}\right]\cos\left[\frac{(2y+1)v\pi}{16}\right]$ …(5)

    tempBlock ⟵ block

15. Stego-image⟵ tempBlock

16. Output stego-image

3.2.4 Proposed Algorithm Message Extraction Process

Input: Stego-image

Output: Cover image and message.

1. Input Stego-image
2. Blocksimg[n] ⟵ $\lfloor stego - image \rfloor 8X8$

   »Shift block

   ↓ ↓ Shift block
3. Apply 2D-DCT on each block $C(u, v) = \alpha(u)\alpha(v)\sum_{x=0}^{8-1}\sum_{y=0}^{8-1} f(x, y)\cos\left[\frac{(2x+1)u\pi}{16}\right] \times \cos\left[\frac{(2y+1)v\pi}{16}\right]$ …(3)
4. Apply dequantization to co-efficient ⟵ $C^{Q}(u, v) = \left[\frac{C(u, v)}{Q(u, v)}\right]$ ...(4)
5. Compute DCT coefficients ⟵Huffman decoding

6. (I –D) array ⟵ dezigzag scan

7. Extract msg ⟵ LSBMR $\left(\left\lfloor \frac{x_i}{2} + x_{i+1} \right\rfloor\right)$ Shared key2 …(6)

8. For each block in blocksimg

   Perform I DCT ⟵ $f(x, y) = \frac{1}{4}\sum_{u=0}^{8-1}\sum_{v=0}^{8-1}\alpha(u)\alpha(v)f(x, y)\cos\left[\frac{(2x+1)u\pi}{16}\right]$

   cos $\left[\frac{(2y+1)v\pi}{16}\right]$ …(5)

   («Block)

   tempBlock ⟵ block

   Cover image ⟵ tempBlock

9. Uncomp msg ⟵ deflate (decrypt x)

10. Decrypt msg ⟵ AES (deflate x, shared key1)

11. Display msg and cover image

3.3 System Development Tools

The software requirement for the effective development and implementation of this system is as follows: a) Java Programming Language b) Netbeans IDE

The hardware requirements needed to run the software are: a) A Pentium IV CPU, 1.8 GHz processor with 512 MB memory b) A 40 GB hard drive capacity c) A graphic adaptor with screen resolution of 800 * 600 pixels and 32 bit quality d) A CD/DVD rewritable drive.

CHAPTER FOUR: IMPLEMENTATION, RESULTS AND ANALYSIS

This chapter provides the proposed system implementation with screenshots for hiding and extracting secret messages. The system provides easy to use interface. It also presents all the experiments conducted to evaluate the proposed system and results of the comparative analysis obtained from the research.

4.1 Implementation Detail

The system Improved Image Steganography based on LSBMR using Sobel Edge detection is considered to be a standalone application where it does not require using neither a database nor web server. It is expected to perform two main operations: hiding and extracting messages. The system was implemented on Windows 7 Ultimate Operating System that has processor of 2.27 GHz Intel Core i3 with memory 2 GB 350 MHz DDR3.

4.2 Graphical User Interface of the Proposed System

The Improved Image Steganography system has the following functions. The application allows users to open an image, save an image and enter message. Users (sender) can browse for an image, choose between two techniques to embed secret message inside it. Then save the resulting stego-image. Finally, the receiver can extract secret message from the stego-image using one of the two steganographic methods as can be seen in Figure 4.1.

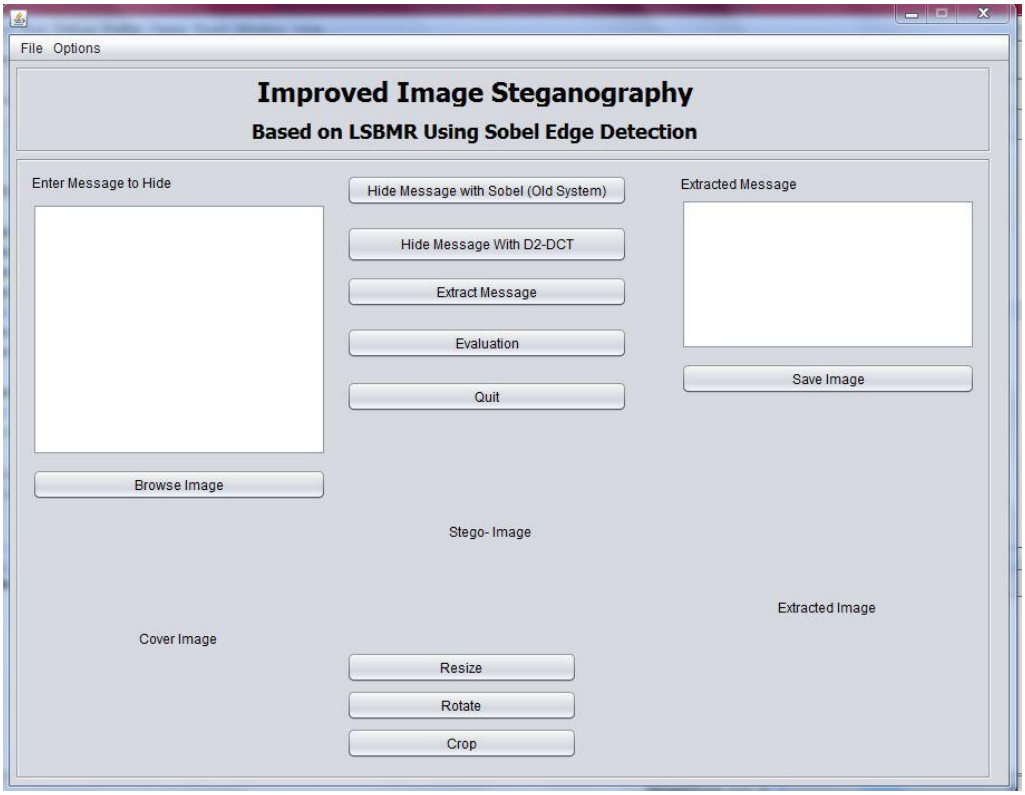


Figure 4.1: The GUI of Improved Image Steganography based on LSBMR Application

4.2.1 Message Hiding

The following description is used to embed a secret message in a cover image. The user has to first type the secret message in the text area in the top left of the application. Select an image by clicking on browse Image which will let the user browse the computer to select an image of type GIF as cover image. Then click on one of the two techniques to be used for conducting steganography. For instance, the user clicks on the Hide Message with 2D-DCT button to hide secret message inside selected cover image. The stego-image is produced and displayed under the label Stego-Image. Finally the stego-image is saved by the user as stego-im.gif. Figure 4.2 depicts the snapshot for message hiding with one of the techniques.

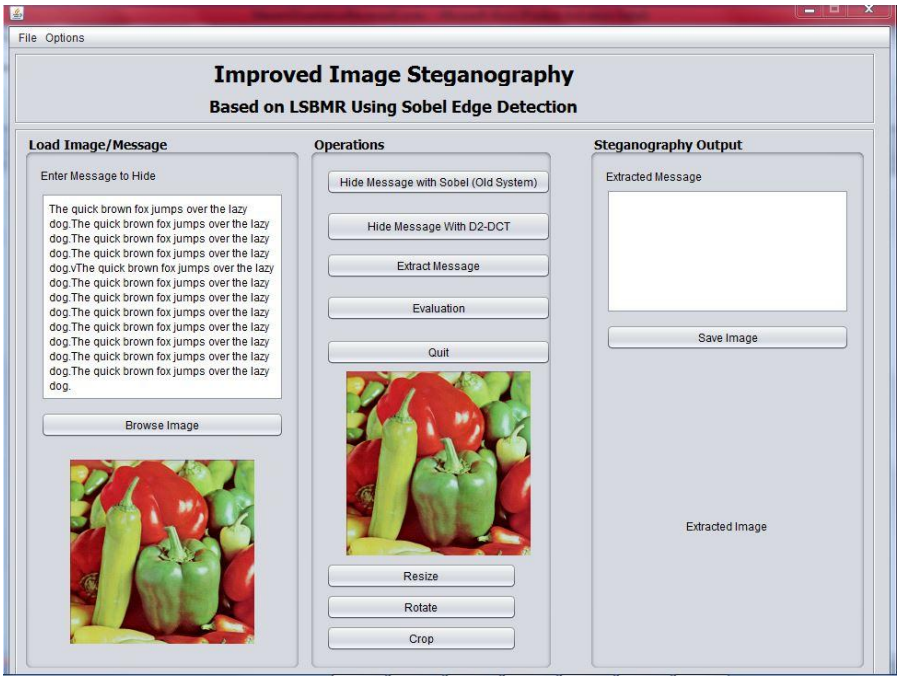


Figure 4.2: A Snapshot for Message Hiding

4.2.2 Message Extraction

On the other hand, the following description is on how to extract a secret message from a stego-image. First the user must click on The Extract Message. Then the user has to browse to select the stego-image from the location where it was saved. Then double click on the location, select the image that is stego-im.gif that was produced in the embedding phase. Click on open, for the secret message to be extracted. The extracted secret message is displayed at the top right corner of the application while the image appears at the right bottom end. Figure 4.3 depicts the snapshot for message extraction using one of the techniques.

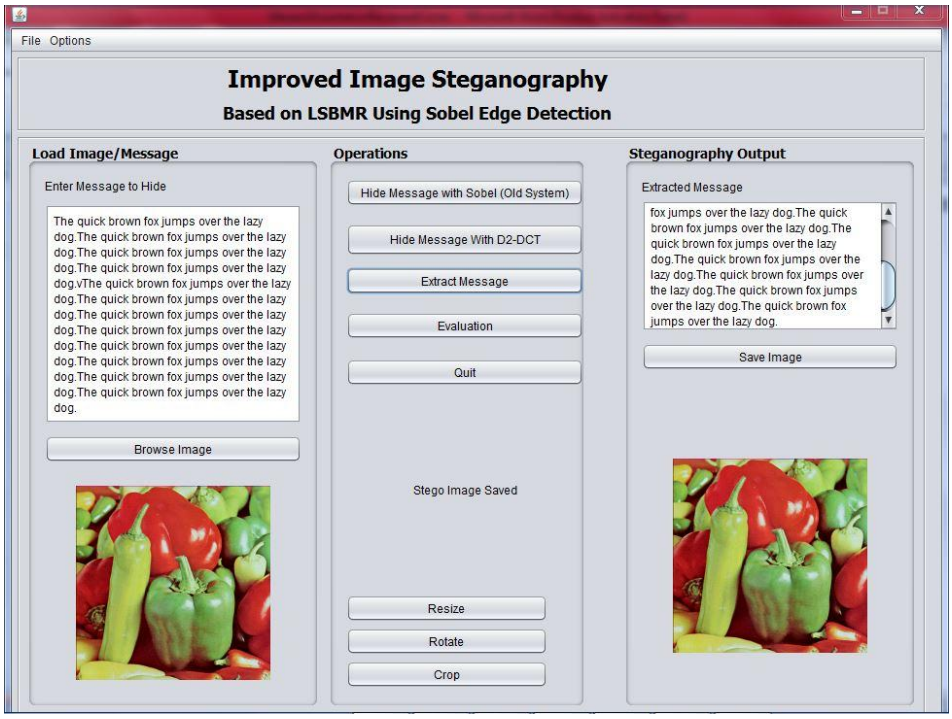


Figure 4.3: A Snapshot for Message Extraction

4.3 Experimental Results and Analysis

This section presents all the experimental dataset conducted to evaluate the effectiveness of the proposed system. The performance of the system was tested with different images and compared with the work of Zohreh and Jihad, (2014) in line with the stated objectives of our research. The comparative performance and security of the designed system was based on the following features such as robustness, undetectability, embedding (payload) capacity and invisibility. In addition, the experimental results are presented in tables and graph.

4.3.1 Dataset

This sub-section describes the dataset of images that were used for the experiment. The proposed technique employed dataset of 200 different standard colour images taken from "*The USC-SIPI Image Database"* that supports research in image processing and comparative image analysis. The dataset contains GIF image format, different texture,

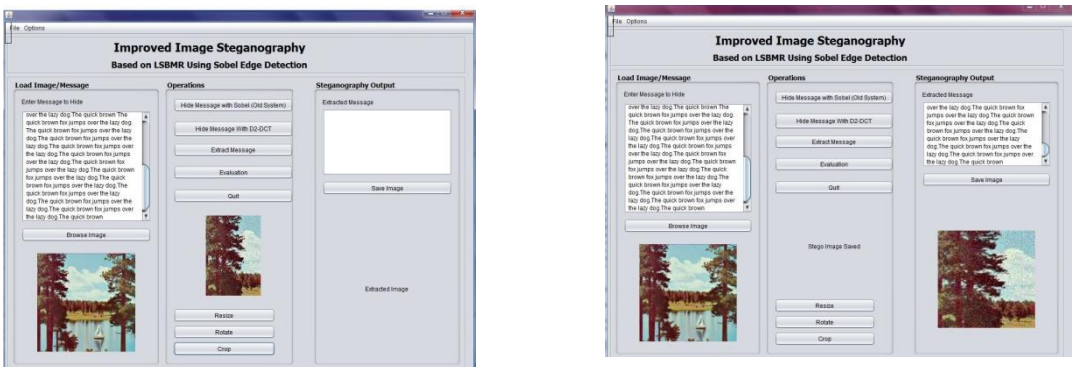
contents and contrast such as edgy and smooth standard color images of the same dimension 512×512 (Allan, 1981) . The embedding capacity used for the experiment was adopted from the work of (Zohreh and Jihad, 2014).

4.3.2 Robustness

The main advantage of the proposed method improved image steganography based on LSBMR using Sobel operator is that it selects the middle frequencies (coefficients) part of the detected edges for data hiding for the stego-image to withstand image operations such as resizing, rotation and cropping.

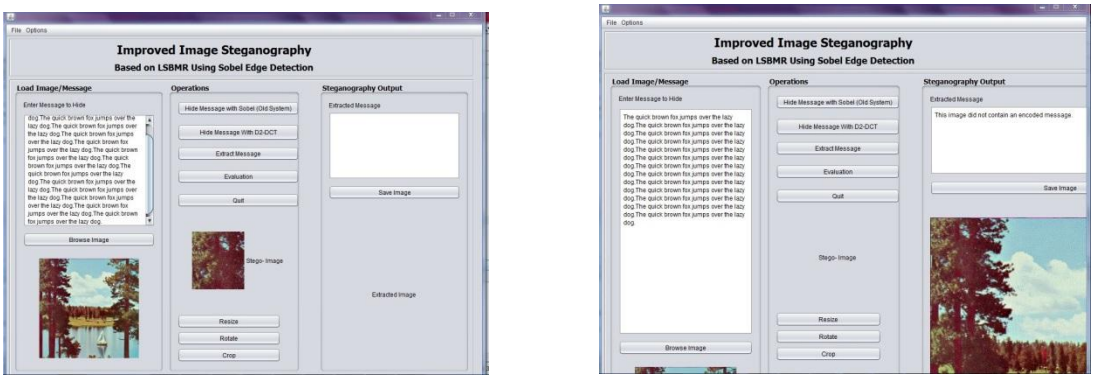
To test whether the hidden message can be recovered after image manipulations, some experiments were conducted. The image operations were applied on the generated stego images from the existing and proposed systems based on the following:

a. Cropping: Figure 4.4i shows the snapshot for cropping operation with the proposed technique: (a) Cropped Stego-image and (b) Recovered Message from Cropped Stego-image. Figure 4.4ii shows the snapshot for cropping operation with the existing technique: (a) Cropped Stego-image and (b) Message Lost from Cropped stego-image. Therefore, the proposed system was able to retrieve the hidden message from the stego-image when cropped to a height and width of 9cm respectively. Because the secret message was hidden in the middle frequencies of the scanned DCT coefficients of the detected cover image whereas with the existing system, the hidden message was lost.



a b

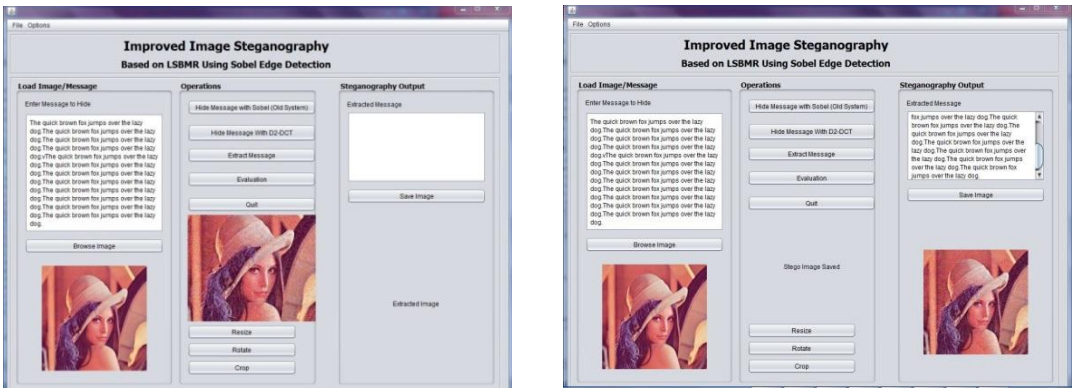
Figure 4.4i: Proposed Technique (a) Cropped Stego-image (b) Recovered Message from Cropped Stego-image



a b

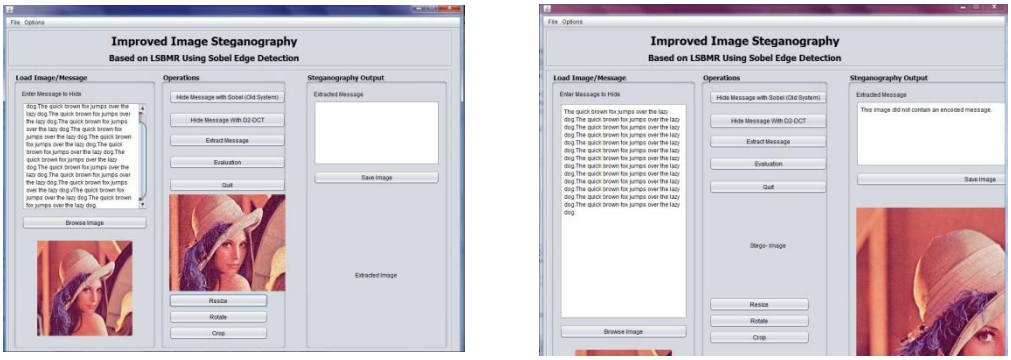
Figure 4.4ii: Existing Technique (a) Cropped Stego-image (b) Message Lost from Cropped Stego-image

b. Resizing: Figure 4.5i shows the snapshot for resized operation with proposed technique: (a) Resized Stego-image and (b) Recovered Message from Resized Stego-image. Figure 4.5ii shows the snapshot for resized operation with the existing technique: (a) Resized Stego-image and (b) Message Lost from Resized Stego-image. It can be concluded that the proposed system was able to retrieve the hidden message from the stego-image when the size was doubled by 0.5%, but for the existing system, the hidden message was lost.



a b

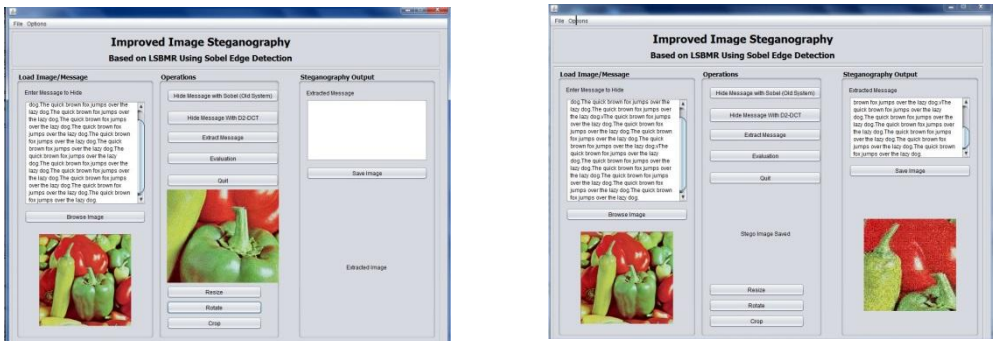
Figure 4.5i: Proposed Technique (a) Resized Stego-image (b) Recovered Message from Resized Stego-image



a b

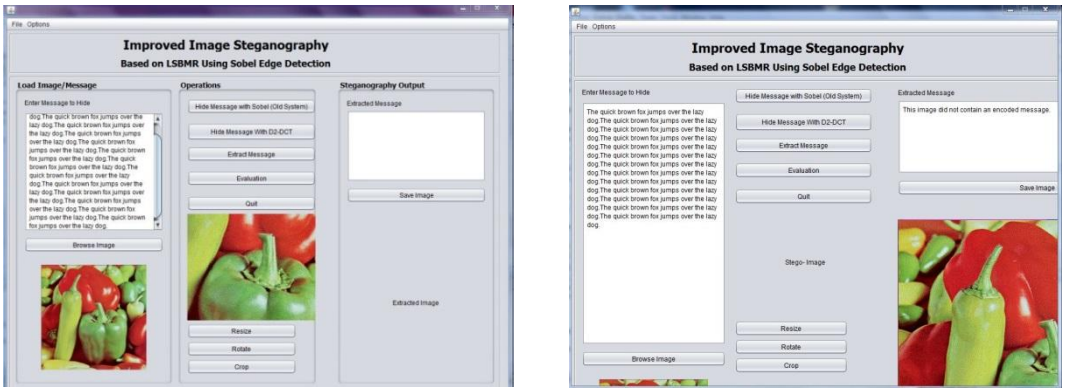
Figure 4.5ii: Existing Technique (a) Resized Stego-image (b) Message Lost from Resized Stego-image

c. Rotation: Figure 4.6i shows the snapshot for rotated operation with proposed technique: (a) Rotated Stego-image and (b) Recovered Message from Rotated Stego-image. Figure 4.5ii shows the snapshot for rotated operation with the existing technique: (a) Rotated Stego-image and (b) Message Lost from Rotated Stego-image. The proposed system was able to retrieve the hidden message from the stego-image rotated at angle of 60^{o} to the right, for the existing system, the hidden message was lost.



a b

Figure 4.6i: Proposed Technique (a) Rotated Stego-image (b) Recovered Message from Rotated Stego-image



a b

Figure 4.6ii: Existing Technique (a) Rotated Stego-image (b) Message Lost from Rotated Stego-image

4.3.3 Invisibility

To assure the quality of the image is preserved and invisibility feature is maintained a quantitative test experiment was conducted. Most popular measurement used in steganography to evaluate quality of the stego-image is Peak Signal-to-Noise Ratio (PSNR)

and Mean Square Error (MSE). The next section gives the explanation of the image qualities.

4.4.3.1 PSNR and MSE

PSNR is the ratio between the modified image (stego-image) and original cover image. It is used for calculating the observable deformation that occurs in stego images after intentionally embedding secret data. The PSNR is calculated in terms of decibels (dB). The higher the value of PSNR, the more the stego image is correlated with original cover image and vice versa (Muhammad *et al.*, 2015). Stego images with PSNR less than 30dB represent low quality. PSNR must strive for 40dB or higher values in order to fulfill the favourable demands of modern steganographic systems (Na and Kim, 2014).

PSNR is defined by using Mean Squared Error (MSE) that calculates the error between cover image and distorted stego image. In simple words MSE indicates average amount of modifications to pixels. Given width x height (m x n) of a cover image C and stego-image S, the MSE is defined as follows: $MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[C(i,j) - S(i,j)]^2$ … (6)

$$PSNR = 10 * \log_{10}\left(\frac{MAX_C^2}{MSE}\right) \quad \ldots (7)$$

$$= 20*log_{10}\left(\frac{MAX_C}{\sqrt{MSE}}\right)$$

$$= 20*log_{10}(MAX_C) - 10*log_{10}(MSE)$$

Where: *m* represents the numbers of rows of pixels of the image and *n* represents the number of columns of pixels of the image that is (image dimensions), *i* represents the index of that row and *j* represents the index of that column that is (loop counters), *C* is cover

image, S is stego image, and MAX_C^2 is the maximum pixel intensity that exists in the cover image.

The MAX of the cover images used for testing and conducting experiment is 255.

Since each image's pixel of the 200 images is 8 bit-depth. As the PSNR increase, it means the quality is increasing. So the higher PSNR value the better image quality and less difference between the stego-image and cover image.

In this part, we present the comparison of the proposed and existing techniques. The average PSNR of the work of (Zohreh and Jihad, 2014) and the proposed method for 200 images was calculated. The results are presented in Table 4.1 with the average modification rate of the 200 stego-images using the two steganographic techniques. The modification rate is calculated as number of modified pixels over the image size. As mentioned all of the images used for testing are of size 512*512.

Table 4.1: Average PSNR and MSE of 200 Stego-images

Capacity (bits)	Steganographic Techniques	Average PSNR	Average Modification Rate
8000	Existing System (Zohreh and Jihad, 2014)	67.42	0.01150
	Proposed Technique	68.13	0.01317
16000	Existing System (Zohreh and Jihad, 2014)	64.42	0.02294
	Proposed Technique	55.82	0.17742
24000	Existing System (Zohreh and Jihad, 2014)	62.66	0.03438
	Proposed Technique	54.70	0.22624
32000	Existing System (Zohreh and Jihad, 2014)	61.38	0.04581
	Proposed Technique	53.81	0.27751

From Table 4.1, the proposed technique achieved PSNR of 68dB when the embedding capacity was 8000 bits with higher modification rate of 0.01317 and lower rate of 0.27751

when the capacity of the secret message was increased to 32000 bits. This is attributed to fact that the embedding of the secret bits was done at the middle of the (frequencies) coefficients of the detected cover image pixels, for the stego-image to withstand image operations like rotation, resizing and cropping. The existing system has lower modification rate and PSNR of 61 for 32000 bits of secret message because the secret bits are hidden in the detected edges of the cover image pixels. But the quality of the stego-images generated by the proposed system is still preserved and invisibility feature is also maintained since PSNR is above 40dB. Therefore, the invisibility of the stego-images generated with the proposed system is preserved. The results displayed in Table 4.1 were used to generate Figure 4.7 which depicts the graph of average PSNR of the two steganographic techniques for different embedding capacity and Figure 4.8 which also depicts the graph for modification rate that is MSE of the two steganographic techniques for different embedding capacity.

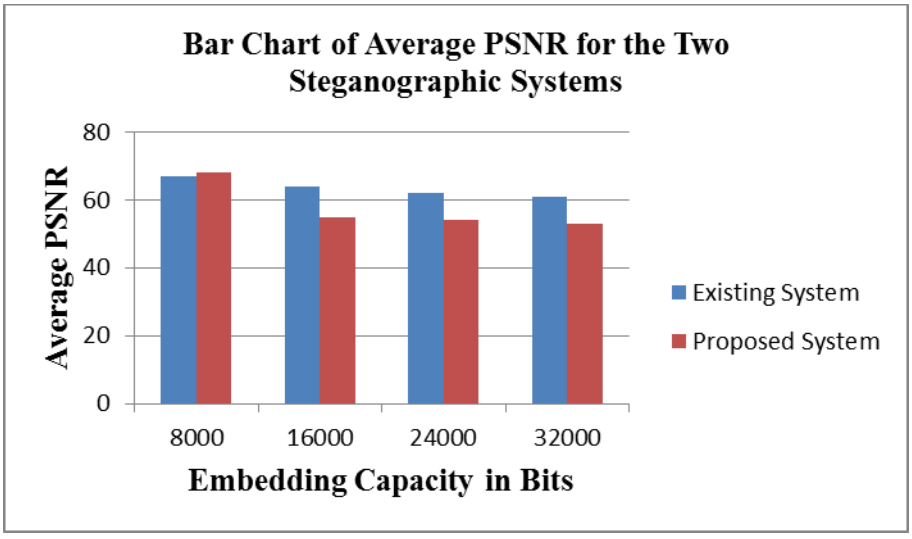


Figure 4.7: Bar Chart of Average PSNR using two Steganographic Techniques for Different Capacity

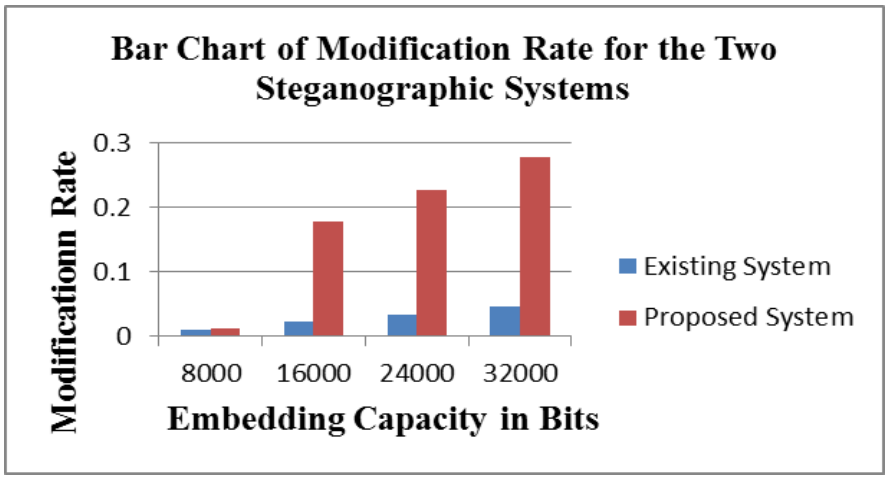


Figure 4.8: Modification Rate of the Two Steganographic Methods for Different Embedding Capacity

4.3.4 Undetectability

Hidden messages in stego-images can be detected using statistical attack techniques. Regular Singular (RS), Chi Square, Primary Set and Fusion Mean attacks were investigated in this section.

4.4.4.1 Statistical Steganalysis

After the messages are embedded in the stego-image, the stego-image should have similar statistical property with the original image. Some techniques called steganalysis are developed to detect whether an image is stego-image or not. Statistical steganalysis was used to test the resistance of our proposed method. First, Virtual Steganographic Laboratory (VSL) tool (Forczmanski and Wegrzyn, 2008) for digital Images used for only RS steganalysis of the hidden messages in stego-images and is licensed under GNU GPLv3 license was used. The second tool used for analyzing undetectability test of the proposed and existing systems is StegExpose. The tool combined intelligently different steganalytic methods such as Chi-square, Primary Sets and Fusion Mean (Boehm, 2014). The detection of hidden messages was performed on stego-images generated from both techniques. Table

4.2 shows the result summary of the sample images and statistical steganalysis of the tools applied.

Table 4.2: Result of five Samples of Stego-images Generated from the Two Steganographic Techniques and Statistical Steganalysis of the tools.

File name	Secret message size in bytes	RS Analysis	Primary Sets	Chi Square	Sample Pairs	Fusion (mean)	Detectability Status?
BaboonStego.gif.png (Existing System)	1116	0.05517	0.00742	0.0	0.01548	0.01952	False
BaboonStego.gif (Proposed System)	1116	0.05496	0.00782	0.0	0.01606	0.01971	False
Boat.gif.png (Existing System)	5484	0.21230	0.17175	0.0	0.11285	0.12423	False
BoatStego.gif (Proposed System)	5484	0.10946	0.16701	0.0	0.11276	0.12231	False
House2Stego.gif.png (Existing System)	3153	0.14000	0.13365	0.0	0.01854	0.07304	False
House2Stego.gif (Proposed System)	3153	0.14648	0.15072	0.0	0.01085	0.07701	False
LenaaStego.gif.png (Existing System)	4099	0.17338	0.21109	0.0	0.01499	0.09986	False
LenaStego.gif (Proposed System)	4099	0.1618	0.1871	0.0	0.0240	0.0932	False
PepperStego.gif.png	2566	0.11170	0.04605	0.0	0.06650	0.05606	False

(Existing System)							
PepperStego.gif (Proposed System)	2566	0.11419	0.05193	0.0	0.06187	0.05699	False

From Table 4.2, the VSL and StegExpose could not detect the hidden information inside the stego-image. The steganalysis tools assumed a mean of 0.19 as the threshold value to classify the status either true or false. A stego-image is classified as clean (False) if its statistical value is below the threshold value meaning it does not have hidden information. Stego-image is classified as not clean (True) if its statistical value is above the threshold value meaning the image has embedded secret message as the detectability status. This can be concluded that the statistical steganalysis techniques in the steganalysis tools cannot detect the presence of secret information in the stego-images generated from both techniques that is Image Steganography using Sobel Operator and the Improved Image Steganography using Image Transformation.

4.3.5 Security

The two techniques have high security that is Image steganography based on LSBMR using Sobel Edge Detection and the Improved Image Steganography based on LSBMR using Sobel Edge Detection. But Improved Image steganography has higher security because the secret message was hidden in the coefficients of the detected cover image. Table 4.3 displays the security features of both systems.

Table 4.3: Security Features of the Steganographic Techniques

Techniques	Message Compression	Message Encryption	Domain
Existing System (Zohreh, 2014)	Deflate Algorithm	AES	Spatial
Proposed System	Deflate Algorithm	AES	Transform

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Summary

In this dissertation, Improved Image Steganography based on LSBMR Using Sobel Edge Detection was designed. Although there have been many researches on image steganography, most of the existing algorithms have high embedding capacities, but they are vulnerable to small modifications that may result from image processing operations such as cropping, rotation, scaling and resizing. The improved system was designed with the objective of improving the robustness of the technique to work in transform domain so that the stego-images do not suffer any act of image manipulations. The proposed system was implemented and evaluated using image steganography performance metrics and statistical steganalysis technique and the results proved to be robust than the base technique that was extended.

5.2 Conclusion

In line with the objectives of the research, the following has been achieved.

a. An Improved Image Steganography Based on LSBMR using Sobel Edge Detection has been designed.

b. The proposed technique employs image transformation technique specifically 2D-DCT to detected edges of the cover image to provide a better stego-image that can withstand image operations such as cropping, resizing and rotation.

c. This technique was successfully implemented in Java and Netbeans IDE environment.

d. Experimental results conducted showed that the proposed technique produced better stego-image quality that can withstand multiple image operations such as rotation, resizing

and cropping in relation to robustness and PSNR of 68dB for 8000 bits of secret message with regards to the invisibility.

5.3 Recommendation

This section presents some suggestion to extend the technique.

a. The method can achieve its minimal time of processing by reducing the compression process to have lower time complexity.

b. Furthermore, future research should focus on applying the Improved Image Steganography based on LSBMR using Sobel edge detection on higher bit order instead of the least significant bit.

c. Finally, the proposed technique was applied on colour images. Thus, it is suggested to extend it to be used on other steganographic cover objects such as video or audio. For example, the Improved Image Steganography based on LSBMR using Sobel edge detection can be applied on frames of the cover video. Then extensive evaluation should be then conducted to witness the effectiveness of the technique on video frames.

5.4 Research Contributions to Knowledge

The main contributions of this work are in two phases:

a. An improved image steganography system based on LSBMR using Sobel edge detection that is robust and works in transform domain was designed.

b. The experimental results of this research produced better stego-image quality that can withstand multiple image operations such as rotation, resizing and cropping in relation to robustness and PSNR of 68dB for 8000 bits of secret message as regards to the invisibility.

REFERENCES

Al-Shatanawi, O. M. and El-Emam, N. N. (2015). A New Image Steganography Algorithm Based on MLSB Method with Random Pixels Selection. *International Journal of Network Security and Its Applications, 7*(2), 37-53.

Allan, W. (1981). *The University of Southern California-Signal and Image Processing Institute (USC-SIPI) Image Database*. Retrieved May 1, 2016, from http://sipi.usc.edu/research/sipi-image-database.html.

Anderson, R. and Petitcolas, F. (1998). On the Limits of Steganography. *Institute of Electrical and Electronics Engineers Journal on Selected Areas in Communication, 16*(4), 474-481.

Anupam, M. and Shiladitya, P. (2015). A Novel Approach of Image Based Steganography Using Pseudorandom Sequence Generator Function and DCT Coefficients. *International Journal of Computer Network and Information Security, 3*(6), 42-49.

Ashok, J., Raju, Y., Munishankaraiah, S. and Srinivas, K. (2010). Steganography: An Overview. *International Journal of Engineering Science and Technology, 2*(10), 5985-5992.

Aubrey, D. S. (1996). *Herodotus: The Histories.* London: Penguin Books.

Bender, W., Gruhl, D., Morimoto, N. and Lu, A. (1996). Techniques for Data Hiding. *International Business Machine Systems Journal, 35*(3), 313-336.

Boehm, B. (2014). *StegExpose: A Tool for Detecting LSB Steganography.* England: Univerity of Kent.

Caldwell, J. C. (2003). Steganography. Crosstalk the Journal of Defense Software Engineering. Retrieved February 18, 2016, From http://thesai.org/.../Paper_33/-improvisation_of_Security_Aspect_of_Steganographic/html.

Chandramouli, R. and Memon, N. (2001). Analysis of LSB based Image Steganography Techniques. *International Conference on Image Processing, Journal of Institute of Electrical and Electronics Engineers, 3*, 1019-1022.

Chandramouli, R., Kharrazi, M. and Memon, N. (2003). Image Steganography and Steganalysis: Concepts and Practice. *Proceedings of the 2nd International Workshop on Digital Watermarking, Journal of Institute of Electrical and Electronics Engineers, 3*, 35-49.

Cheddad, A., Condell, J., Curran, K. and Mc Kevitt, P. (2010). Digital Image Steganography: Survey and Analysis of Current Methods. *Journal of Signal Processing, 90*(3), 727-752.

Cole, E. (2003). Hiding in Plain Sight: Steganography and the Art of Covert Communication. *Communications of the Association of Computing Machine Journal, 47*, 76-82.

Deepak, S. and Rupali, S. (2012). Data Security Using LSB and DCT Steganography In Images. *International Journal of Computational Engineering Research, 2*(2), 359-364.

Deepika, B. and Rita, C. (2014). An Improved DCT Based Steganography Technique. *International Journal of Computer Applications, 102*(14), 46 – 49.

Deng, C., Ma, W. and Yin, Y. (2011). An Edge Detection Approach of Image Fusion Based on Improved Sobel Operator. *In 4th International Congress on Image and Signal Processing, Journal of Institute of Electrical and Electronics Engineers, 3*, 1189-1193.

Elham, J. M., Ahmed, J. M., Zainab, J. M., Gaillan, H. A., Iman, M. K. and Yasser, A. A.-K. (2013). Design Study Sobel Edge Detection. *International Journal of Application on Innovation in Engineering and Management, 2*(12), 248-253.

Flores-Escalante, J., Pérez-Díaz, J. and Gómez-Cárdenas, R. (2012). Design and Implementation of an Electronic Identification Card. *Journal Of Applied Research and Technology, 7*(3), 20-37.

Forczmanski, P. and Wegrzyn, M. (2008). Virtual Steganographic Laboratory for Digital Images. *In Information Systems Architecture and Technology: Journal of Information Systems and Computer Communication Networks*, 163-174.

Fridrich, J. (1999). *Applications of Data Hiding in Images.* New York: Center for Intelligent Systems, Suny Binghamton.

Fridrich, J. (2009). *Stegangoraphy in Digital Media- Principles, Algorithms and Applications.* USA: Cambridge University Press.

Hassanein, M. S. (2014). *Secure Digital Documents Using Steganography and QR Code.* London: Brunel University.

Hemang, A. P. and Nehal, G. C. (2015). Secured and Robust Dual Image Steganography: A Survey. *International Journal of Innovative Research in Computer and Communication Engineering, 3*(1), 30-37.

Holub, V. (2014). *Content Adaptive Steganography – Design and Detection.* New York: Binghamton University.

Hong, W. and Chen, T. S. (2012). A Novel Data Embedding Method using Adaptive Pixel Pair matching. *Information Forensics and Security, Institute of Electrical and Electronics Engineers Transactions, 7*(1), 176-184.

Ibaida, A., Khalil, I. and Al-Shammary, D. (2010). Embedding Patients Confidential Data in ECG Signal for Healthcare Information Systems. *In Annual International Conference of the Institute of Electrical and Electronics Engineers, Engineering in Medicine and Biology Society*, *3*, 3891-3894.

Imaculate, R. S. and Ashok, R. M. (2013). Adaptive Pixel Pair Matching based Steganography for Audio files. *In International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System*, *5*(2), 1-5.

Jiang, L., Rui, L., Zhanxin, Y. and Yahui, H. (2009). The Research on the Digital Short Radio Covert Communication Based on Audio Signal Information Hiding Technology. *In International Conference on Management and Service Science*, 1-4.

Johnson, N. F. and Jajodia, S. (1998). Exploring steganography: Seeing the unseen. *Journal of Institute of Electrical and Electronics Engineers*, *31*(2), 26-34.

Kamred, U. S. (2014). A Survey on Image Steganography Techniques. *International Jorunal of Computer Applications, 97*(18), 10-20.

Karaman, H. B. and Sagiroglu, S. (2012). An Application Based on Steganography. *In Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining, Journal of Institute of Electrical and Electronics Engineers*, *3*, 839-843.

Katzenbeisser, S. and Petitcolas, F. (2000). *Information Hiding Techniques for Steganography and Digital Watermarking.* Norwood-MA: Artech House.

Kaur, B., Kaur, A. and Singh, J. (2011). Steganographic Approach for Hiding Image in DCT Domain. *International Journal of Advances in Engineering and Technology, 2*(3), 72-78.

Kharrazi, M., Sencar, H. T. and Memon, N. (2004). Image Steganography: Concepts and Practice. *Journal of World Scientific Publishing Company*, *1*(49), 1-31.

Kipper, G. (2004). *Investigator's Guide to Steganography.* Retrieved May 1, 2016, from http://www.crcpress.com/Kipper/html.

Knut, M. M. (2008). *Numerical Algorithms and Digital Representation.* Oslo: Oslo University Press.

Luo, W., Huang, F. and Huang, J. (2010). Edge Adaptive Image Steganography Based on LSB Matching Revisited. *Journal of Institute of Electrical and Electronics Engineers Transactions on Information Forensics and Security, 5*(2), 201-214.

Mali, S., Patil, P. and Jalnekar, R. (2012). Robust and Secured Image-Adaptive Data Hiding. *Journal of Institute of Electrical and Electronics Engineers Transactions on Digital Signal and Image Processing, 22(2),* 314–323.

Mehdi, H. and Hussain, M. (2013). A Survey of Image Steganography Techniques. *International Journal of Advanced Science and Technology, 54*, 113-124.

Memon, N. and Chandramouli, R. (2001). Analysis of LSB Based Image Steganography Techniques. *Journal of Institute of Electrical and Electronics Engineers, Proceedings on Image Processing, 3*, 1019-1022.

Michael, W. E. and Herbert, M. J. (2011). *Principles of Information Security, 4th Ed.* Kennsaw University: Cengage Learning Press.

Mielikainen, J. (2006). LSB Matching Revisited. *Institute of Electrical and Electronics Engineers*, *13*(5), 285-287.

Mohan, S. and Singh, S. (2015). Information hiding with LSB based Image Steganography. *International Journal of Innovative Science, Engineering and Technology, 2*(4), 710-713.

Muhammad, K., Jamil, A., Haleem, F. and Zahoor, J. (2015). A Secure Method for Color Image Steganography using Gray-Level Modification and Multi-level Encryption. *Journal of Korean Society for Internet Information Transactions on Internet and Information Systems, 9*(5), 1938-1962.

Na, T. and Kim, M. (2014). A Novel No-Reference PSNR Estimation Method with Regard to Deblocking Filtering Effect in H. 264/AVC Bitstreams. *Journal of Institute of Electrical and Electronics Engineers Transactions on Circuits and Systems for Video Technology, 24*, 320-330.

Osama, K. (2005). Retrieved June 25, 2015, from google.com: http://cs.uccs.edu/~cs591/studentproj/projF2005/okhaleel/doc/C3S.ppt

Owens, M. (2002). A discussion of covert channels and steganography. Retrieved August 10, 2015, from http://www.sans.org/rr/information /Security/ Reading/ Room/whitepapers/covert/678.php.

Patel, Z. and Gadhiya, S. (2015). A Survey Paper on Steganography and Cryptography. *Research Hub – International Multidisciplinary Research Journal, 2*(5), 1-5.

Phad, V. S., Bhosale, R. S. and Panhalkar, A. R. (2012). A Novel Security Scheme for Secret Data using Cryptography and Steganography. *International Jorunal of Computer Network and Information Security*, *2*, 36-42.

Por, L., Lai, W., Z. Alireza, Z., Ang, T., M.T. Su, M. and Delina, B. (2008). StegCure: A Comprehensive Steganographic Tool using Enhanced LSB. *Journal of World Scientific and Engineering Academy Society Transactions on Computers*, *7*(8), 1309-1318.

Provos, N. and Honeyman, P. (2003). Hide and Seek: An Introduction to Steganography. *Institute of Electrical and Electronics Engineers Security and Privacy, 1*(3), 32-44.

Ratnakirti, R., Anirban, S. and Suvamoy, C. (2013). Chaos based Edge Adaptive Image Steganography. *ScienceDirect Journal and International Conference on Computational Intelligence: Modeling Techniques and Applications*, *10*, 138-146.

Rodrigues, J. M., Rios, J. R. and Puech, W. (2004). SSB-4 System of Steganography using bit 4. *In 5th International Workshop on Image Analysis for Multimedia Interactive Services, Journal of Institute of Electrical and Electronics Engineers*,*5*(2), 1-5.

Sahar, A. E. (2015). A Comprehensive Image Steganography Tool using LSB Scheme. *International Jorunal of Image, Graphics and Signal Processing*, *6*(2), 10-18.

Shamim, L. A. and Hemachandran, K. (2012). High Capacity Data Hiding using LSB Steganography and Encryption. *International Journal of Database Management Systems, 4*(6), 57-68.

Sharmila, B. and Shanthakumari, R. (2012). Efficient Adaptive Steganography for Color Images. *Association for Information and Communication Technologies Journal on Image and Video Processing*, *2*(3), 387-392.

Singh, S. and Attri, V. K. (2015). State-of-the-Art Review on Steganographic Techniques . *International Journal of Signal Processing, Image Processing and Pattern Recognition*, *8*(7), 161-170.

Singla, D. and Syal, R. (2012). Data Security Using LSB and DCT Steganography In Images. *International Journal Of Computational Engineering Research, 2*(2), 359-364.

Subhedar, M. S. and Mankar, V. H. (2014). Current status and key issues in image: A survey. *Journal of Computer Science Review (ScienceDirect)*, *13*(C), 95-113.

Thambiraja, E., Ramesh, G. and Umarani, D. R. (2012). A Survey on Various most Common Encryption Techniques. *International Journal of Advanced Research in Computer Science and Software Engineering, 2*(7), 226-233.

Tiwari, N. and Shandilya, M. (2010). Evaluation of Various LSB based Methods of Image Steganography on GIF File Format. *International Journal of Computer Applications, 6*(2), 1-4.

Tu, D. and Tran, T. (2002). Context Based Entropy Coding of Block Transform Coefficients for Image Compression. *Journal of Institute of Electrical and Electronics Engineers Transaction on Image Processing, 11*(11), 247-259.

Vincent, O. R. and Folorunso, O. (2009). A Descriptive Algorithm for Sobel Image Edge Detection. *Proceedings of Informing Science and IT Education Conference,* 98-107.

Walia, E., Jain, P. and Navdeep. (2010). An Analysis of LSB and DCT based Steganography. *Global Journal of Computer Science and Technology, 10*(1), 4-8.

Wang, X. and Niu, R. (2008). Steganograhpy for High-Resolution Color Remote Sensing Image. *Journal of In Institute of Electrical and Electronics Engineers, Computer Science and Software Engineering, 3,* 709-712.

Wang, X. and Ye, J. (2010). Information Hiding Technology in Electronic Notes System. *In International Conference on E-Business and E-Government, 2*(1), 1627-1630.

Watson, A. (1994). Image Compression Using the Discrete Cosine Transform. *Mathematica Jorunal, 4*(1), 81-88.

Webster. (1831). *Security.* Retrieved May 1, 2016, from Merriam-Webster Dictionary: http://www.merriam-webster.com/dictionary/Security.

Weimin, W. H. (2008). Improvable Deflate Algorithm. *In 3rd Institute of Electrical and Electronics Engineers Conference on Industrial Electronics and Applications*, *1*(2), 1572-1574.

Yang, H., Sun, X. and Sun, G. (2009). A High-Capacity Image Data Hiding Scheme using Adaptive LSB Substitution. *Journal of Radio Engineering, 18,* 509-516.

Yazdanpanah, A. and Hashemi, M. R. (2011). A Simple Lossless Preprocessing Algorithm for Hardware Implementation of Deflate Data Compression. *In 19th Iranian Conference on Electrical Engineering*, *Institute of Electrical and Electronics Engineers,3*(2), 1-5.

Z'aba, M. R. and Maarof, M. (2006). A Survey on the Cryptanalysis of the Advanced Common Encryption Techniques. *In International Symposium and Exhibition on Geoinformation, International Journal of Advanced Research in Computer Science and Software Engineering*, *10*(3), 97-102.

Zaid, A. O. and Ahmad, T. A. (2015). A Survey on Digital Image Steganography. *The 7th International Conference on Information Technology*, *5*(1), 109-115.

Zhiping, Z. and Maomao, H. (2009.). Steganalysis for Markov Feature of Difference Array in DCT Domain. *Proceedings of Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, *7,* 581-584.

Zohreh, A.F. and Jihad, M. A. (2014). Image Steganography Based on LSBMR using Sobel Edge Detection. *Institute of Electrical and Electronics Engineers*, *3*(2), 141-145.

APPENDIX 1: SAMPLE PROGRAM CODING

```
package MainOperations;

/**
 *
 * @author OCHEJA MARIAM
 */
public class MariamsThesisForm extends javax.swing.JFrame implements MouseListener, MouseMotionListener{
    /**
     * Creates new form MariamsThesisForm
     */
    public MariamsThesisForm() {
        initComponents();
        setResizable(false);
        setLocationRelativeTo(null);
    }
    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {
```

```
jPanel1 = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
jPanel3 = new javax.swing.JPanel();
jButton1 = new javax.swing.JButton();
jLabel3 = new javax.swing.JLabel();
jScrollPane1 = new javax.swing.JScrollPane();
jTextArea1 = new javax.swing.JTextArea();
jLabel4 = new javax.swing.JLabel();
jPanel4 = new javax.swing.JPanel();
jButton3 = new javax.swing.JButton();
jButton4 = new javax.swing.JButton();
jButton5 = new javax.swing.JButton();
jButton6 = new javax.swing.JButton();
jButton7 = new javax.swing.JButton();
jButton8 = new javax.swing.JButton();
jLabel7 = new javax.swing.JLabel();
jButton10 = new javax.swing.JButton();
jPanel5 = new javax.swing.JPanel();
jLabel5 = new javax.swing.JLabel();
jScrollPane2 = new javax.swing.JScrollPane();
jTextArea2 = new javax.swing.JTextArea();
jButton2 = new javax.swing.JButton();
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```