TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

THESIS NO: 070/MSI/613

SECURED CRYPTO STEGANO DATA HIDING USING IMPROVED LEAST SIGNIFICANT BIT SUBSTITUTION AND ENCRYPTION

By

Sanjita Lamichhane

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

NOVEMBER, 2015

SECURED CRYPTO STEGANO DATA HIDING USING IMPROVED LEAST SIGNIFICANT BIT SUBSTITUTION AND ENCRYPTION

By

Sanjita Lamichhane

Thesis Supervisor

Prof. Dr. Subarna Shakya

A thesis

submitted in partial fulfillment of the requirements for the degree of

Master of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering

Institute of Engineering, Pulchowk Campus Tribhuvan University

Lalitpur, Nepal

November, 2015

COPYRIGHT

The author has agreed that the library, Department of Electronics and Computer Engineering, Institute of Engineering, Pulchowk Campus, may make this thesis freely available for inspection. Moreover the author has agreed that the permission for extensive copying of this thesis work for scholarly purpose may be granted by the professor(s), who supervised the thesis work recorded herein or, in their absence, by the Head of the Department, wherein this thesis was done. It is understood that the recognition will be given to the author of this thesis and to the Department of Electronics and Computer Engineering, Pulchowk Campus in any use of the material of this thesis. Copying of publication or other use of this thesis for financial gain without approval of the Department of Electronics and Computer Engineering, Institute of Engineering, Pulchowk Campus and author's written permission is prohibited. Request for permission to copy or to make any use of the material in this thesis in whole or part should be addressed to:

Head

Department of Electronics and Computer Engineering

Institute of Engineering, Pulchowk Campus

Pulchowk, Lalitpur, Nepal

TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certify that they have read and recommended to the Department of Electronics and Computer Engineering for acceptance, a thesis entitled **"Secured Crypto Stegano Data Hiding Using Improved Least Significant Bit Substitution and Encryption "**, submitted by **Sanjita Lamichhane** in partial fulfillment of the requirement for the award of the degree of "**Master of Science in Information and Communication Engineering**".

____

Supervisor: Prof. Dr. Subarna Shakya
Institute of Engineering, Pulchowk Campus
Department of Electronics and Computer Engineering

____

External Examiner: Er. Mahesh Singh Kathayet
Head, Computer Directorate
Police Headquarter, Nepal Police

____

Committee Chairperson: Dr. Dibakar Raj Panta
Head
Department of Electronics and Computer Engineering
Institute of Engineering, Pulchowk Campus

Date: 4th November, 2015

DEPARTMENTAL ACCEPTANCE

The thesis entitled **"Secured Crypto stegano Data Hiding Using Improved Least Significant Bit Substitution and Encryption"**, submitted by **Sanjita Lamichhane** in partial fulfillment of the requirement for the award of the degree of "**Master of Science in Information and Communication Engineering**" has been accepted as a bonafide record of work independently carried out by her in the department.

Dr. Dibakar Raj Pant

Head of the Department

ACKNOWLEDGEMENT

I would like to express my sincere and cordial thanks to distinguished persons who helped in myriad of ways to bring my thesis to this level. I would like to express my profound gratitude and sincere thanks to my thesis supervisor **Prof. Dr. Subarna Shakya** for his continuous supervision, guidance and suggestions throughout the thesis.

I would like to express my sincere gratitude to the **Department of Electronics and Computer Engineering**, **Institute of Engineering** for the suitable platform and cooperation. I would like to extend my sincere thanks for providing me with all the essential co-operation, valuable suggestions for choosing the thesis topic "**Secured Crypto Stegano Data Hiding Using Least Significant Bit Substitution and Encryption**". I am very grateful to **Dr. Dibakar Raj Pant**, Head of Department for his suggestions and support throughout the thesis. My heartfelt acknowledgement goes to **Dr. Surendra Shrestha,** Master's degree program coordinator for his guidance and cooperation. Likewise, I am very grateful to my respected teachers for their guidance and support. Finally, I would like to express my heartfelt thanks to my family and friends who always encouraged and supported me.

ABSTRACT

Communication has always been an integral part of our existence. With the rapid advancement in Internet and networking technologies during the recent years, communication and information exchange have become much easier and faster but at the same time the issues related to data security and confidentiality have become a major concern of time. To cater to this need of information security, a number of hidden and secret communication techniques such as cryptography, steganography and watermarking have been developed. These are the practical means to provide security services and are becoming a powerful tools in many applications for information security. Cryptography scrambles a message so it cannot be understood whereas the Steganography hides the message so that it cannot be seen. In this method we first encrypt a message using an algorithm based on Fibonacci series or Rijndael cryptographic algorithm and then embed the encrypted message inside an image using Least Significant Bit (LSB) substitution method. Digital images often have a large amount of redundant data and for this reason it is possible to hide message inside image file. Image steganography is about exploiting the limited power of the human visual system where we hide information in the least significant bit of the image data. This embedding method is based on the fact that the least significant bit in an image can be thought of as random noise, and consequently they become less responsive to any change on the image. The combination of these two methods will enhance the security of the data embedded. This combinational methodology will satisfy the requirements such as capacity, security and robustness for secure data transmission over an open channel.

Key Terms: Plain text, Cipher text, Encryption, Decryption, Fibonacci number, Key, Cover Image, Stego Image, Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE)

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	**Full Form**
HVS	Human Visual System
LSB	Least Significant Bit
MSB	Most Significant Bit
RGB	Red Green Blue
DES	Data Encryption Standard
AES	Advance Encryption Standard
SSL	Secured Socket Layer
RSA	Rivest Shamir Adi
MSE	Mean Square Error
PSNR	Peak Signal to Noise Ratio

TABLE OF CONTENTS

CHAPTER ONE: INTRODUCTION

1.1. Background

From the dawn of civilization to the highly networked societies, information exchange has always been an important part of our lives. In the current digital era, the rapid escalations in digital multimedia and network have paved ways for people around to acquire, utilize and share multimedia information. Methods of communication today include radio communication, telephonic communication, network communication mobile communication etc. All these methods and means of communication have played an important role in our lives, but in the past few years, network communication, especially over the internet, has emerged as one of the most powerful methods of communication with an overwhelming impact on our lives. With the growth of computer network, security of data has become a major concern and thus data hiding technique has attracted people around the globe. Such rapid advances in communications technology have also given rise to security threats to individuals and organizations. Since, the information that could benefit or educate groups (or individual) can also be used against such groups (or individual).Hence forth, the information security has evolved as an important and urgent issue not only for individuals but also for business and governments.

1.2 Security attacks

The data is transmitted from source to destination which is the normal flow of data as shown in figure. But the intruders might hack the network in order to access or modify the original data. These types of attacks are formally known as security attacks.

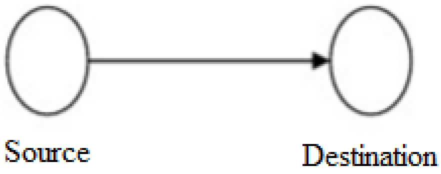


Figure 1.1: Normal data flow

An intruder can disrupt this normal flow by implementing the different types of techniques over the data and network in following ways. They are:

- Interruption
- Interception
- Modification
- Fabrication

Interruption:

Interruption is an attack by which an intruder can interrupt the data before reaching the destination. This type of attack usually destroys the system asset and makes the data unavailable or useless.

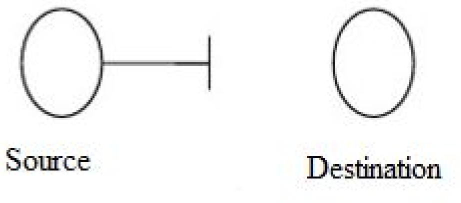


Figure 1.2: Interruption

Interception:

When the network is shared that through a local area network and is connected to Wireless LAN or Ethernet it can receive a copy of packets intended for other device. On the internet, the determined intruder can gain access to email traffic and other data transfers. This type of attack shows the effect on confidentiality of data.

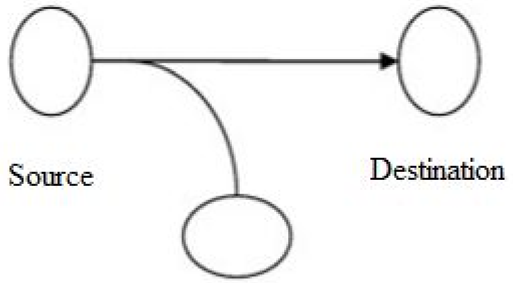


Figure 1.3:Interception

Modification:

This refers to altering or replacing of valid data that is needed to send to destination. This type of attacks is done usually by unauthorized access through tampering the data. It shows effect on the integrity of the data.

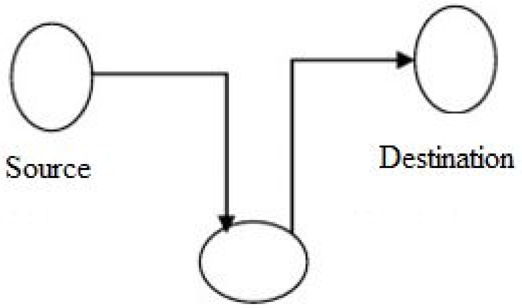


Figure 1.4: Modification

Fabrication:

In this type, the unauthorized user places data without the interface of source. An unauthorized person inserts the unauthorized objects by adding records to the file or insert spam messages etc. This type of attack affects on the authenticity of message.

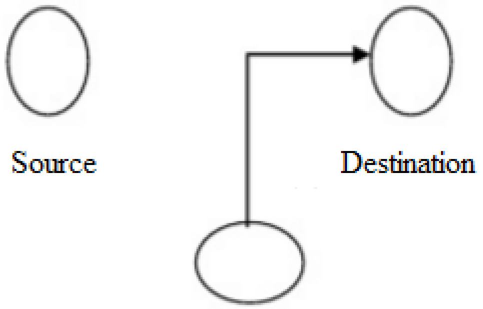


Figure 1.5: Fabrication

1.3 Cryptography

Cryptography is the science of using mathematics to encrypt and decrypt data. Cryptography enables us to store sensitive information or transmit it across insecure networks (like the Internet) so that it cannot be read by anyone except the intended recipient. In other words, Cryptography is the science that protects data by transforming it into a digital form which is not discernible to an attacker without the secret key [1].The term is derived from the Greek language krytos means secret and graphos means writing. Encryption is the actual process of applying cryptography. Its purpose is to ensure privacy by keeping the information hidden from anyone for whom it is not intended. Decoding is the reverse of encoding. It is the transformation of encrypted data back into some intelligible form. Cryptography is popularly known as the study of encoding and decoding private messages. In cryptography, encryption processes are used in transforming information using an algorithm to make it unreadable to anyone except those possessing special knowledge, usually referred to as a key.

Much of cryptography is math oriented and uses patterns and algorithms to encrypt messages, text, words, signals and other form of communication. Cryptography has many uses, especially in the areas of espionage, intelligence and military operations. Today, many security systems and companies use cryptography to transfer information over the Internet. Some of this encryption is highly advanced. However, even simple encryption techniques can help uphold the privacy of any person.

1.4 Cryptographic Goals

Authentication: It is the process of providing proof of identity of the sender to the recipient so that the recipient can be assured that the person sending the information is who and what he or she claims to be.

Privacy/confidentiality: It is the process of keeping information private and secret, so that only the intended recipient can understand the information.

Integrity: It is the method to ensure that information is not tampered with during its transit or its storage on the network. Any unauthorized person should not be able to tamper with the information or change the information during transit.

Non-repudiation: It is the method to ensure that information cannot be disowned. Once the non-repudiation process is in place, the sender cannot deny being the originator of the data.

1.5 Steganography

Cryptography and Steganography are two popular methods of sending secret information in a secured way. One hides the existence of the message and the other distorts the message itself. These are well known and widely used techniques that manipulate messages in order to cipher or hide their existence respectively. Cryptography is the art and science of protecting information from undesirable individuals by converting it into a form non-recognizable by its attackers while stored and transmitted [1]. Steganography is the art and science of communicating in a way which hides the existence of the communication [2]. Cryptography scrambles a message so it cannot be understood whereas the Steganography hides the message so that it cannot be seen.

Steganography techniques are used to address digital copyrights management, protect information, and conceal secrets [1]. Data hiding techniques provide an interesting challenge for digital forensic investigators. Data is the backbone of today's communication. To ensure that data is secured and does not go to unintended destination, the concept of data hiding came up to protect a piece of information [3]. Digital data can be delivered over computer networks with little errors and often without interference. The Internet provides a communication method to distribute information to the masses. Therefore, the confidentiality and data integrity are required to protect against unauthorized access and use [4]. Steganography and cryptography are two different information hiding techniques, where we transform the message so as to make it meaning obscure to a malicious people who intercept it. Steganography relies on hiding message in unsuspected multimedia data and is generally used in secret communication between acknowledged parties [5]. The technique replaces unused or insignificant bits of the digital media with the secret data. The concept is to embed the hidden object into a significantly larger object so that the change is undetectable by the human eye. All digital file formats can be used for steganography, but the formats those are with a high degree of redundancy are more suitable [6]. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. The most popular cover objects used for steganography are digital images. Digital images often have a large amount of redundant data, and this is what steganography uses to hide the message. Cryptography merely obscures the integrity of the information so that it does not make sense to anyone except the creator and the recipient [7].

Cryptography assures privacy whereas Steganography assures secrecy [9]. Steganography and cryptography are both used to ensure data confidentiality. However, steganography differs from cryptography in the sense that the cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret [8]. Thus, with cryptography anybody can see that both parties are communicating in secret. Steganography hides the existence of a secret message in such a way that nobody can see that both parties are communicating in secret.

1.6 Image Steganalysis

Steganalysis is the breaking of steganography and is the science of detecting hidden information [14]. The main objective of steganalysis is to break steganography and the detection of stego image. Almost all steganalysis algorithms depend on steganographic algorithms introducing statistical differences between cover and stego image. Steganalysis are of three different types:

- Visual attacks: It helps to discover the hidden information by separating the image into bit planes for further more analysis.
- Statistical attacks: It may be passive or active. Passive attacks involves with identifying presence or absence of a secret message or embedding algorithm used. Active attacks is used to investigate embedded message length or hidden message location or secret key used in embedding.
- Structural attacks: The format of the data files changes as the data is embedded helps in identifying the characteristic structure changes which helps us to find the presence of image/text file.

1.7 Problem Definition

In modern era, unauthorized access of information is increasing day by day due to this we need to secure information and this can be done using cryptography or steganography technique. Steganography and Cryptography are parallel data security techniques. In fact steganographic system can implement cryptographic data security. With cryptography we can protect the message but not hide its existence [7].Steganography pay attention to the degree of invisibility while cryptography pays attention to the security of the message. There are many image steganography and cryptography algorithms that has been already developed such as Least Significant Bit (LSB), Random Scattered (RS), Most Significant Bit (MSB) but LSB is most frequently used because it simply inserts the bit of secret message with the least significant bit of image utilizing the fact that the least significant bits in an image can be thought of as random noise, and consequently become less responsive to any change on the image which cannot be detected by human eye.

1.8 Objectives

The main objectives of the thesis are:

- to design and develop an efficient image steganography algorithm which is made more secure with the help of cryptographic algorithms based on Fibonacci series and Rijndael cryptographic algorithm
- to provide resistance against various visual and statistical attacks

1.9 Applications

Cryptography along with steganography are best known as a way of keeping the contents of a message secret. Confidentiality of network communications is of great importance for ecommerce and other network applications. Cryptography has many uses, especially in the areas of espionage, intelligence and military operations for the purpose of sending confidential data. However, the applications of cryptography along with steganography go far beyond simple confidentiality. In particular, it allows the network business and customer to verify the authenticity and integrity of their transactions. Today, many security systems and companies use cryptography along with steganography to transfer information over the Internet. Some of this encryption is highly advanced. However, even simple encryption techniques can help uphold the privacy of any person.

CHAPTER TWO: LITERATURE REVIEW

Encryption and related technologies are widely and frequently used as a means of ensuring that the information is secure, and their importance has been growing with the increasingly widespread utilization of the Internet. The use of encryption can be traced to as far back as about 3000 B.C., during the Babylonian Era. Encryption technologies evolved as they were used in military and political settings, but as a result of the recent widespread use of the Internet and the dramatic increase in the amount of information people come into contact in their daily lives, the settings in which encryption technologies are applied and implemented have increased, and they are now used all around us in our daily lives.

Hieroglyphics (pictograms used in ancient Egypt) inscribed on a stele in about 3000 B.C. are considered the oldest surviving example of encryption. Hieroglyphics were long considered impossible to ever read, but the discovery and study of the Rosetta Stone in the 19th century was the catalyst that made it possible to read hieroglyphics [6].The Caesar cipher, which appeared in the 1st century B.C., was so named because it was frequently used by Julius Caesar, and it is a particularly prominent method of encryption among the great number of encryption methods that emerged during the long history of encryption. The Caesar cipher method of encryption involves replacing each of the letters of the alphabet in the original text by a letter located a set number of places further down the sequence of the letters in the alphabet of the language. The sender and receiver agree in advance to replace each letter of the alphabet in the text by a letter. Substitution ciphers are a well-known encryption method, and they are the most commonly used encryption method in the history of encryption. An encryption method that involves rearranging the sequence of characters according to a specific rule is referred to as a “substitution cipher”. The modern encryption machine called “Enigma” made it possible to apply the substitution cipher method with a higher level of sophistication [2]. Cryptography became more popular during the Middle Ages as encryption technologies became increasingly sophisticated based on the knowledge acquired during efforts to decrypt classical encryptions and the invention of new encryptions. The increased diplomatic activity during this time led to an increase in need to convey confidential information, which led to the frequent use of encryption. With the advancement of communication technology, encryption and decryption came to be actively performed during World Wars [2].

In general, there are three types of cryptographic schemes typically used to accomplish the cryptographic goals: secret key (or symmetric) cryptography, public-key (or asymmetric) cryptography and hash functions. In all cases, the initial unencrypted data is referred to as plaintext. It is encrypted into cipher text, which will in turn (usually) be decrypted into usable plaintext. The process of encryption and decryption of information by using a single key is known as single key cryptography or symmetric key cryptography. In symmetric key cryptography, the same key is used to encrypt as well as decrypt the data. The main problem with symmetric key algorithms is that the sender and the receiver had to agree on a common key. Public key cryptography technique is based on a combination of two keys: secret keys and public key. It is also known as asymmetric encryption. In this process, one key is used for encryption, and the other key is used for decryption. This process is known as asymmetric cryptography because both the keys are required to complete the process, and these two keys are collectively known as the key pair. In asymmetric cryptography, one of the keys is freely distributable and this key is called the public key which is used for encryption. Hence, this method of encryption is also called public key encryption. The second key is the secret or private key and is used for decryption. However, it still has not been possible to find any one way function that realizes asymmetrical ciphering, involving the use of different keys for ciphering and decrypting. The theory of this public key encryption method has been applied in practice in the form of the "RSA Cipher" [3]. Three researchers at the Massachusetts Institute of Technology, Ronald L. Rivest, Adi Shamir, and Leonard M. Adlemen, devised the mathematical method that was used to make the concept of a public key a reality, proposed by Diffie and Hellman [7]. This public-key cipher is called the "RSA Cipher", with "RSA" being the initials of the last names of the three researchers who devised the mathematic method. The RSA cipher method utilizes prime factorization.

The public key cryptosystem is an extremely convenient system for exchanging keys to decrypt encryptions with a certain party or parties alone via the Internet. In other words, even though public keys are available to anyone on the Internet, to which any number of people have access, because it is difficult to decrypt the secret key within any reasonable time, for all practical purposes, the public-key cryptosystem can be viewed as a dramatic solution to the problem of distributing the key that had been a source of difficulty since ancient times. However, transmission time for documents encrypted using public key cryptography is significantly slower than symmetric cryptography [5]. The key sizes must be significantly larger than symmetric cryptography to achieve the same level of protection. Public key cryptography is susceptible to impersonation attacks. SSL is a protocol that was proposed by Netscape Communications and incorporated into Netscape Navigator, which made it possible for secure communications between a web server and a client [6]. The characteristics of SSL include the issuing of an electronic certificate that authenticates the identity of a server (web server or mail server), and is used for verification by the client before starting an SSL communication to ensure that it has explicitly indicated the communication is being initiated with the correct server[8]. It also prevents data interceptions or leaks by encrypting subsequent communications.

The DES cipher uses a 56-bit key, and since the number of combinations for 56-bit keys is 2 to the power of 56, which is roughly 70 quadrillion, it was considered nearly impossible to decrypt. Ultimately, however, it was decrypted in 1994. Modern encryptions have gradually become more susceptible to decrypting because of the recent significant improvements in the computational capacity of computers. The use of digital signature came from the need of ensuring the authentication. In addition, the signature assures that any change made to the data that has been signed is easy to detect by the receiver. Hash functions, also called message digests and one way encryption, are algorithms that use no key. Instead, a fixed-length hash value is computed based upon the plaintext that makes it impossible for either the contents or length of the plaintext to be recovered. Hash algorithms are typically used to provide a digital fingerprint of a file's content often used to ensure that the file has not been altered by an intruder or virus. Hash functions are also commonly employed by many operating systems to encrypt passwords.

The Advanced Encryption Standard (AES), also known as Rijndael is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001.It is based on the Rijndael cipher developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen. It has been adopted by the U.S. government and is now used worldwide. RIJNDAEL is based on a design principle known as a substitution-permutation network, combination of both substitution and permutation, and is fast in both software and hardware. Zhi and Fen [8] proposed method of LSB image steganography, which used a method called detection of random LSB in which secret message was inserted in selected part of image randomly not in fixed or predefined manner due to this steganalysis became difficult. Zhang *et al.*[7] introduced a new method of LSB steganalysis which is based on statistical distribution of pixel difference in spatial domain which can be done on high resolution images. Based on the difference of zero and non-zero values of pixels and also finds the error which is used to determine the steganographic features. It also uses Laplacian distribution. As we know that pixels are highly correlated to each other in image and zero, non-zero values occur frequently. If change in some neighbor's pixel value occurs then it may slightly change the intensity level of colours. Li *et al.* [9] proposed LSB Information Hiding algorithm which could Lift wavelet transform image. Furthermore, made the objective evaluation of image quality by the PSNR and normalized cross correlation coefficient. Achieving the purpose of information hiding with the secret bits of information to replace the random noise, using the lowest plane embedding secret information to avoid noise and attacks, utilized redundancy to enhance the sound embedded in the way nature to be addressed[10]. Results showed that the proposed algorithm has a very good hidden invisibility, good security and robustness for a lot of hidden attacks.

CHAPTER THREE: RESEARCH METHODOLOGY

3.1 Model

An image is the most common type of digital media used for steganography. Digital images often have a large amount of redundant data and for this reason it is possible to hide message inside image file. Image steganography is about exploiting the limited power of the human visual system where we hide information in the least significant bit (LSB) of the image data [11]. This embedding method is based on the fact that the least significant bits in an image can be thought of as random noise, and consequently they become less responsive to any change on the image.

Steganography also can be implemented to cryptographic data so that it increases the security of this data [12]. In this method we first encrypt a message using an algorithm based on Fibonacci series and RIJNDAEL cryptographic algorithm then embed the encrypted message inside an image using LSB embedding method. The combination of these two methods will enhance the security of the data embedded. This combinational methodology will satisfy the requirements such as capacity, security and robustness for secure data transmission over an open channel.

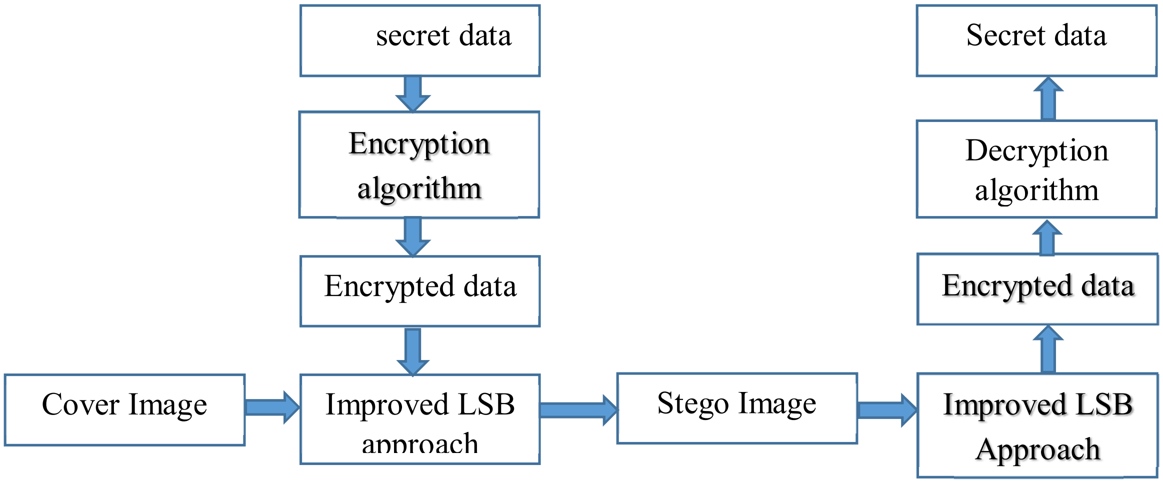


Figure 3.1: Block Diagram of system

3.2 Method I: Encryption Method based on Fibonacci series

In the proposed method, the original message called plain text is converted into cipher text by using a key and Fibonacci numbers generated. The algorithm being used can produce a different output each time it is used, based on the key selected. The obtained cipher text is converted into decimal numbers, and these symbols are transmitted to the recipient through an unsecured communication channel. Since the message is encrypted in two levels, it is hidden from others and makes the decryption process more difficult for any intruders.

3.2.1 Plain Text to Cipher Text

The conversion of a plain text to cipher text can be explained through an example. Let us consider a message to be encrypted and send through an unsecured channel is "IOE PULCHOWK". Each character is replaced with another character based on the Fibonacci number and security key chosen. Any one character is chosen as a first security key to generate cipher text. The characters in the cipher text depend on the security key chosen and the Fibonacci numbers generated. The algorithm being used can produce a different output each time it is used, based on the key selected. For instance let the first security key chosen be "k".

Plain text: IOE PULCHOWK
Security Key: k
Characters: k l m n o p q r s t u v w x y z a b c d e f g h i j k l m n o p q r s t u v w x ….
Fibonacci Series: 1 2 3 5 8 13 21 34 55 89 144 233.........and so on.
Cipher Text: klmorwermuxi
The character set follows a round robin method and the character which falls in accordance with the Fibonacci number will be taken as the character in the cipher text. Since the selection of the character depends on the Fibonacci number, it provides more security for the system, and any unknown person cannot decode the message easily.

3.2.2 Cipher Text to Decimal Numbers

Cipher text to decimal numbers In the second level of security, the required decimal numbers are obtained by the sum of the ASCII code of each character obtained from the cipher text and the ASCII code of the equivalent character in the original message. Hence, the secret data is converted into a set of decimal numbers. Decimal Number Vector thus obtained from previous example:

[180 187 178 143 194 204 177 181 181 196 207 180]

These decimal numbers are sent to the recipient through an unsecured communication channel. By looking at the numbers no unknown persons can identify what it is and the message cannot be retrieved unless the retrieval procedure is known.

3.2.3 Decryption

Decimal Number Vector : [180 187 178 143 194 204 177 181 204 177 181 181 196 207 180]

Security Key: k

Characters**:** **k l m** n **o** p q **r** s t u v **w** x y z a b c d **e** f g h i j k l m n o p q **r** s t u v w x ….

Fibonacci Series: 1 2 3 5 8 13 21 34 55 89 144 233.........and so on.

Cipher Text: klmorwermuxi

Now the ASCII values of the individual characters in cipher text is subtracted from the obtained decimal number vector in order to get the original text.

180-107(k) =73(I)

187-108(l) =79(O)

178-109(m) =69(E)

143-111(o) =32( )

194-114(r) =80(P)

204-119(w) =85(U)

177-101(e) =76(L)

181-114(r) =67(C)

181-109(m) =72(H)

196-117(u) =79(O)

207-120(x) =87(W)

180-105(i) =75(K)

Finally, the performance of this algorithm is compared with the standard Rijndael algorithm.

3.3 Method II: RIJNDAEL Cryptographic Algorithm

The Advanced Encryption Standard (AES), also referenced as Rijndael is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST).It is based on a design principle known as substitution permutation network. It has a fixed block size of 128 bits, and a key size of 128,192 or 256 bits. If the key size is 128 bits RIJNDAEL perform 10 rounds, if the key size is 192 bits it performs 12 rounds and if the key size is 256 bits it performs 14 rounds [13].Before applying the algorithm to the data, the block and key sizes must be determined. The standard encryption uses RIJNDAEL - 128 where both the block and key size are 128 bits. The four stages are as follows:

3.3.1 Sub Bytes Transformation: It is a non-linear substitution step in which each byte is replaced with another according to the entries in a lookup table called an S-box. An Sbox is a one to one mapping for all byte values from 0 to 255[15]. The S-box is used to change the original plain text in bytes to cipher text. Using plain text for the next steps of the algorithm would make it more vulnerable so a byte substitution in the form of the S-box was used. The original bytes are transformed by using the multiplicative inverse and an affinity matrix to make the cipher text resistant to algebraic attacks. In matrix form, the affine transformation element of the S-box can be expressed as:

$$\begin{bmatrix} b_0' \\ b_1' \\ b_2' \\ b_3' \\ b_4' \\ b_5' \\ b_6' \\ b_7' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}.$$

The figure below illustrates the effect of the SubBytes () transformation on the State.

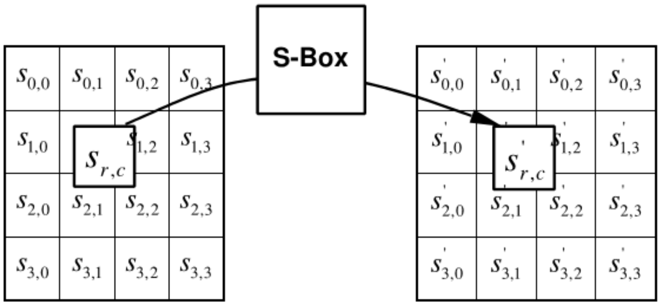
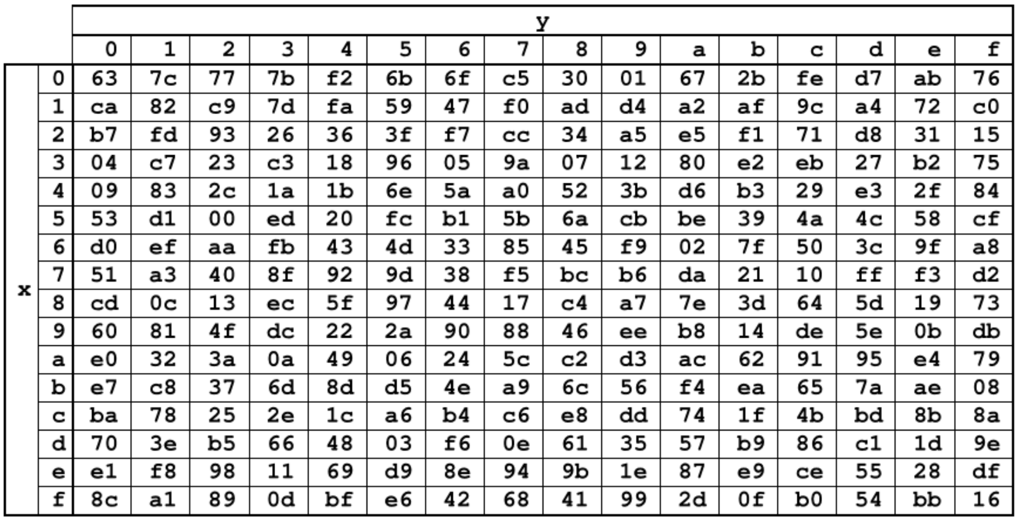

S-Box
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{r,c}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s'_{0,0}$ $s'_{0,1}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{r,c}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,1}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,1}$ $s'_{3,2}$ $s'_{3,3}$

Figure 3.2: S-box transformation

The S-box used in the SubBytes () transformation is presented in hexadecimal form in figure below. For example, if $s_{11}=$ {53}, then the substitution value would be determined by the intersection of the row with index ‘5’ and the column with index ‘3’. This would result in s_{11}' having a value of {ed}.



		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Source: AES Proposal: Rijndael

Figure 3.3: S-box substitution values for the byte xy (in hexadecimal format)

3.3.2 Shift Rows: It is a transposition step in which each row of the state is shifted cyclically at a certain number of steps. . The rows are shifted x number of bytes to the left where x is the row number [14]. This means row 0 will not be shifted, row 1 will be shifted 1 byte to the left, row 2 will be shifted 2 bytes to the left, and row 3 will be shifted 3 bytes to the left and so on.

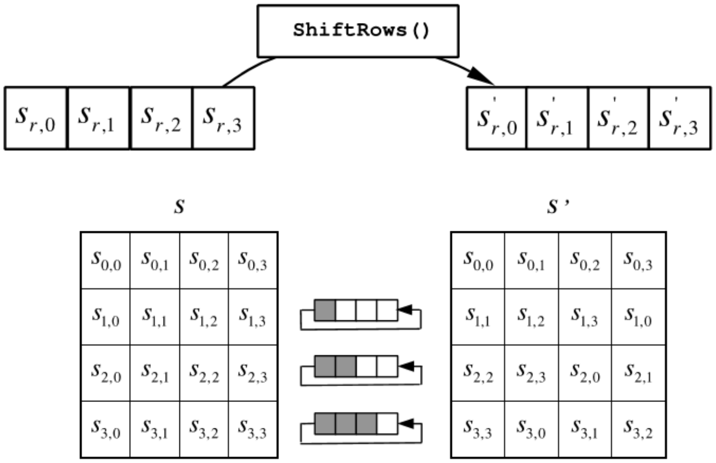

ShiftRows()
$s_{r,0}$ $s_{r,1}$ $s_{r,2}$ $s_{r,3}$
$s'_{r,0}$ $s'_{r,1}$ $s'_{r,2}$ $s'_{r,3}$
s
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
s'
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,1}$ $s_{1,2}$ $s_{1,3}$ $s_{1,0}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,3}$ $s_{3,0}$ $s_{3,1}$ $s_{3,2}$

Figure 3.4: Shift Rows transformation

3.3.3 Mix Columns: After applying the S-box and shifts to the state, the operation of a Mix Column is used. In this step a mixing operation is operated on the columns of the state, combining the four bytes in each column. It is equivalent to the matrix multiplication.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

As a result of this multiplication, the four bytes in a column are replaced by the following.

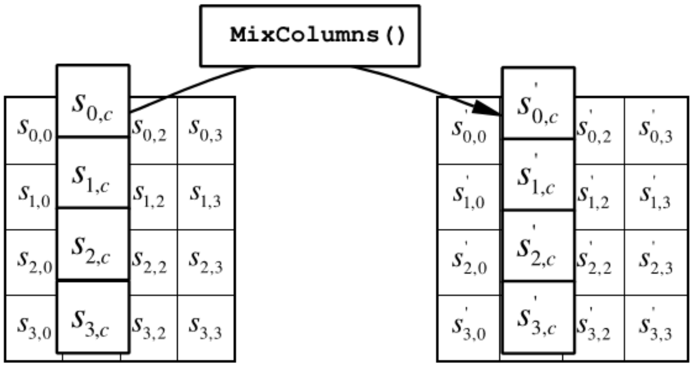


Figure 3.5: Mix Column transformation

The above figure illustrates the Mix Columns() transformation which is equivalent to the following arithmetics.

$$s'_{0,c} = (\{02\} \bullet s_{0,c}) \oplus (\{03\} \bullet s_{1,c}) \oplus s_{2,c} \oplus s_{3,c}$$

$$s'_{1,c} = s_{0,c} \oplus (\{02\} \bullet s_{1,c}) \oplus (\{03\} \bullet s_{2,c}) \oplus s_{3,c}$$

$$s'_{2,c} = s_{0,c} \oplus s_{1,c} \oplus (\{02\} \bullet s_{2,c}) \oplus (\{03\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{03\} \bullet s_{0,c}) \oplus s_{1,c} \oplus s_{2,c} \oplus (\{02\} \bullet s_{3,c}).$$

3.3.4 Add Round Key: At a basic level the Rijndael algorithm uses a number of rounds to transform the data for each block [16]. The output of Add Round Key fully depends on the key or the password specified by the user. In this stage a subkey, which is the same size as the state, is computed from the main key using Rijndael's Key Schedule. This process consists of three parts: Rotate, Rcon, and SubBytes. The first part is to rotate or to shift the bytes that form the keyword 8 bits to the left, which is similar to what happens to the second row in Shift Rows. The second part is to apply sub-operation called Rcon. And the third part is to perform Rijndael's S-Box. Thus obtained key is X-ORed with the output from the mix column step to get the cipher text. Finally, we get the cipher text.

For example let's encrypt the text "Fun Cryptography" which consists of exactly 16 characters. Converting each character to its respective ASCII code and assigning all bytes into a state yields,

$$\begin{bmatrix} 70 & 117 & 110 & 32 \\ 67 & 114 & 121 & 112 \\ 116 & 111 & 103 & 114 \\ 97 & 112 & 104 & 121 \end{bmatrix} = \begin{bmatrix} 01000110 & 01110101 & 01101110 & 00010000 \\ 01000011 & 01110010 & 01111001 & 01110000 \\ 01110100 & 01101111 & 01100111 & 01110010 \\ 01100001 & 01110000 & 01101000 & 01111001 \end{bmatrix}$$

SubBytes then scramble the above state to

$$\begin{bmatrix} 01011010 & 10011101 & 10011111 & 10110111 \\ 00011010 & 01000000 & 10110110 & 01010001 \\ 10010010 & 10101000 & 10000101 & 01000000 \\ 11101111 & 01010001 & 01000101 & 10110110 \end{bmatrix}$$

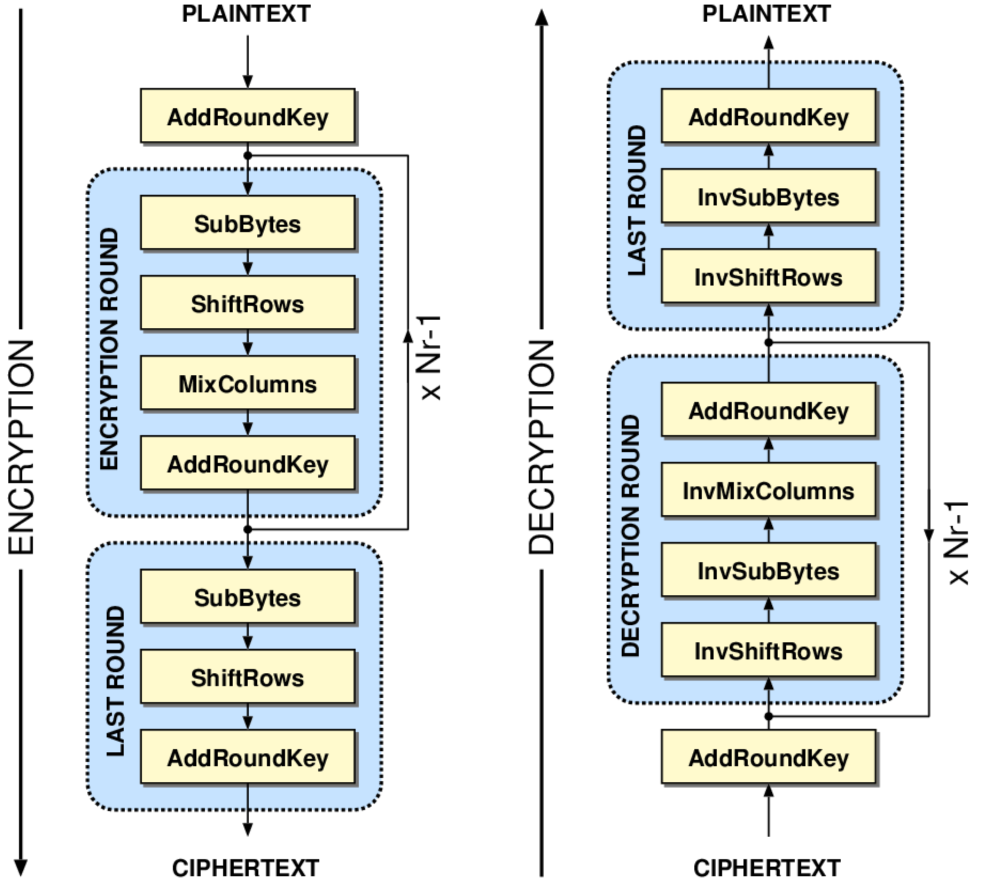
followed by Shift Rows

$$\begin{bmatrix} 01011010 & 10011101 & 10011111 & 10110111 \\ 01000000 & 10110110 & 01010001 & 00011010 \\ 10000101 & 01000000 & 10010010 & 10101000 \\ 10110110 & 11101111 & 01010001 & 01000101 \end{bmatrix}$$

and lastly Mix Columns,

$$\begin{bmatrix} 11100111 & 00011000 & 00100100 & 01110000 \\ 00101010 & 10101011 & 00111001 & 01100011 \\ 00010101 & 01100101 & 11110111 & 10100111 \\ 10101011 & 11110110 & 00000011 & 10100100 \end{bmatrix} = \begin{bmatrix} 231 & 24 & 36 & 112 \\ 42 & 171 & 57 & 99 \\ 21 & 101 & 247 & 167 \\ 171 & 246 & 3 & 164 \end{bmatrix}$$

Now, thus obtained output is X-ORed with the cipher key entered in order to obtain the required cipher text.



Source: www.iis.ee.ethz.ch

Figure 3.6: Rijndael algorithm

3.3.5 Decryption

For decryption all layers must actually be inverted. However, as we will see, it turns out that the inverse layer operations are similar to the layer operations used for encryption i.e., the Byte Substitution layer is replaced by Inv Byte Substitution layer, the Shift Rows layer is replaced by Inv Shift Rows layer and so on.

- Inverse Mix Column Sublayer- After the addition of the subkey, the inverse Mix Column step is applied to the state (again, the exception is the first decryption round). In order to reverse the Mix Column operation, the inverse of its matrix must be used.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

As a result of this multiplication, the four bytes in a column are replaced by the following:

$$s'_{0,c} = (\{0e\} \bullet s_{0,c}) \oplus (\{0b\} \bullet s_{1,c}) \oplus (\{0d\} \bullet s_{2,c}) \oplus (\{09\} \bullet s_{3,c})$$

$$s'_{1,c} = (\{09\} \bullet s_{0,c}) \oplus (\{0e\} \bullet s_{1,c}) \oplus (\{0b\} \bullet s_{2,c}) \oplus (\{0d\} \bullet s_{3,c})$$

$$s'_{2,c} = (\{0d\} \bullet s_{0,c}) \oplus (\{09\} \bullet s_{1,c}) \oplus (\{0e\} \bullet s_{2,c}) \oplus (\{0b\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{0b\} \bullet s_{0,c}) \oplus (\{0d\} \bullet s_{1,c}) \oplus (\{09\} \bullet s_{2,c}) \oplus (\{0e\} \bullet s_{3,c})$$

- Inverse Shift Rows Sublayer

  In order to reverse the Shift Rows operation of the encryption algorithm, we must shift the rows of the state matrix in the opposite direction.

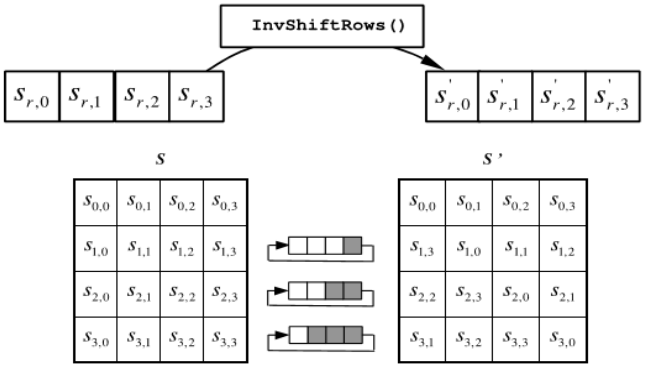
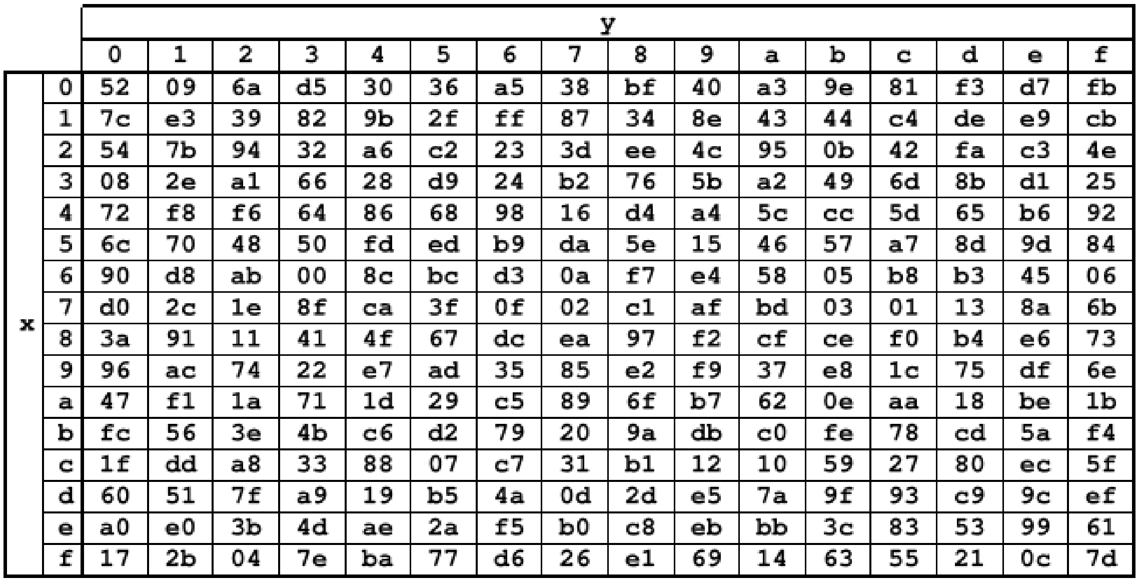

InvShiftRows()
$s_{r,0}$ $s_{r,1}$ $s_{r,2}$ $s_{r,3}$
$s'_{r,0}$ $s'_{r,1}$ $s'_{r,2}$ $s'_{r,3}$
s
s'
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,3}$ $s_{1,0}$ $s_{1,1}$ $s_{1,2}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,1}$ $s_{3,2}$ $s_{3,3}$ $s_{3,0}$

Figure 3.7: Inverse Shift Rows Operation

- Inv Sub Bytes ()-It is the inverse of the byte substitution transformation, in which the inverse S- box is applied to each byte of the State.



		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
	1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
	2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
	3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
	4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
	5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
	6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
	7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
	8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
	9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
	a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
	b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
	c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
	d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
	e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
	f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

Source: AES Proposal: Rijndael

Figure 3.8: Inverse S box

3.4 Data Embedding Procedure

The encrypted message to be hidden is converted into its ASCII equivalent character and subsequently into binary digit. For an example if the character “t” is an encrypted character of the message then ASCII value for “t” is 116 and binary value for it is 1110100. As image comprises of pixel contribution from red, green and blue components and each pixel has numbers from the colour components (for 24-bit bitmap image each of red, green and blue pixel has 8 bit).At 8 bit of the colour number, if we change least significant bits, our visual system cannot detect changes in pixel and thus it is possible to replace message bits with image pixel bit. So we embed the encrypted data into least significant bits of colour. If we change the LSB in a byte of an image, we either add or subtract one from the value it represents [2].In order to hide the encrypted message, data is first converted into byte format and stored in a byte array. The message is embedded into the LSB position of each pixel. Suppose our original pixel has bits: (r7 r6 r5 r4 r3 r2 r1 r0, g7 g6 g5 g4 g3 g2 g1 g0, b7 b6 b5 b4 b3 b2 b1 b0).In addition, our encrypted character (bytes) has some bits: (c7 c6 c5 c4 c3 c2 c1 c0).Then we can place the character bits in the least significant of selected pixel, next character bits in the next lowest pixel, and so on. (r7 r6 r5 r4 r3 r2 r1 c2, g7 g6 g5 g4 g3 g2 g1 c1, b7 b6 b5 b4 b3 b2 b1 c0).

If we take an example of pixel (225,107,100) represented in binary form (11100001, 01101011, 01100100) into which to embed message character “a” having bit 01100001(ASCII value 97), then we can obtain New pixel as (224, 106,101) represented in binary form (11100000, 01101010, 01100101).Here we can notice that a pixel value of (225,107,100) is changed to a new pixel value of (224,106,101). And this change is not visible to human vision. The embedding process operates over the image, and embeds the message character into cover image pixel by pixel at a time. Once all the message characters are embedded into the cover image, the target character (*) represented in bit by 101010, is inserted in the pixel of the cover image immediately next to the one containing the last input character of the message. The target character is a special symbol and is known as Terminator Character. Because it is the last character that is embedded and after embedding the target character (101010), insertion process stops from next row onwards. This helps the decoding process to stop extracting of data from stego image by informing that the target character (*) signifies the end of the message.

3.5 Data Retrieval Process

In the image based steganography the secret message is extracted from the stego image. The recipient inputs the stego image to the extraction algorithm, which outputs the secret message. The method of retrieval of message from the stego image is called steganalysis. In extracting encrypted message, the process opens the stego image file and read the RGB colour of each pixel. The LSBs of each pixel of stego image is extracted. As in the embedding process a Terminator (Target) Character is placed in the message which is the last character to signify the end of embedding of data. When the binary representation of the Terminator character is found the extraction process stops. The purpose of putting this character is that in the process of retrieving the message, the extraction algorithm may take extra bits. The bits of the LSB are retrieved and placed in the array. Then content of the array converts into decimal value that is actually ASCII value of encrypted message. Each 8 bit from the array is converted into character. The message retrieved is then decrypted using the cryptographic algorithms.

3.6 Improved LSB Algorithm

In LSB Algorithm, only least significant bit is replaced by information bit but in Improved LSB, least significant bit is not directly changed. Here we introduce a secret key to protect the hidden information.

Cover image + secret key + secret information = stego image

Step 1: Take a cover image and divide it into three matrices (Red, Green and Blue).

Step 2: Get the secret key and convert it into 1D array of bit stream.

(Secret key and Red matrix are used only for decision making to allocate the secret information bits into either Green matrix or Blue matrix).

Step 3: Perform XOR operation between each bit of secret key with the each LSB of Red matrix. (The resulting XOR value decides whether to hide information on Green matrix or the Blue matrix.)

Step 4: If the XOR value is 1 then replace the LSB of Blue matrix by the first bit of secret information. If the XOR value is 0 then the LSB of Green matrix is replaced by the first bit of secret information and it is continued as until all the bits are hidden.

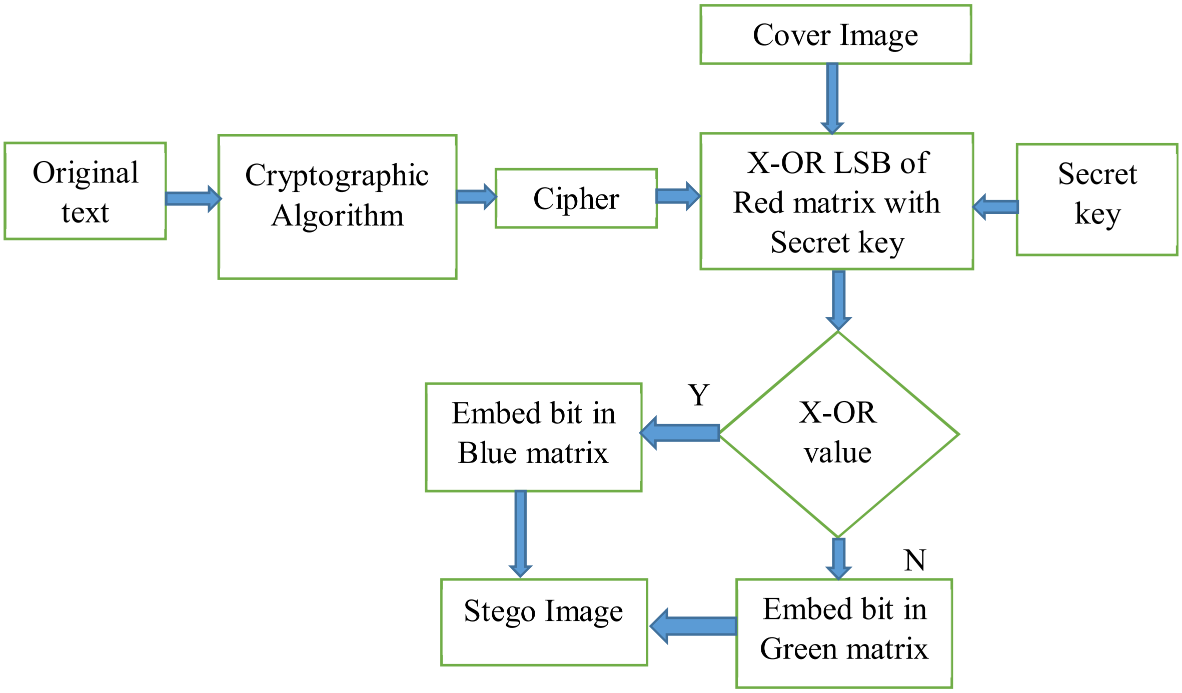


Figure 3.9: Flowchart of Improved LSB Algorithm

3.7 Message Extraction Process

To recover the hidden information, the stego image is divided into three matrices (Red, Green and Blue). Then we have to know the secret key. If the XOR value is 1 then the hidden bit can be found at LSB of Blue matrix. And if the XOR value is 0 then the hidden bit can be found at LSB of Green matrix. This bit is picked and stored into a 1D array and it is continued till all the secret bits are extracted. Now, we get the cipher text which is to be decrypted using the cryptographic algorithms discussed above.

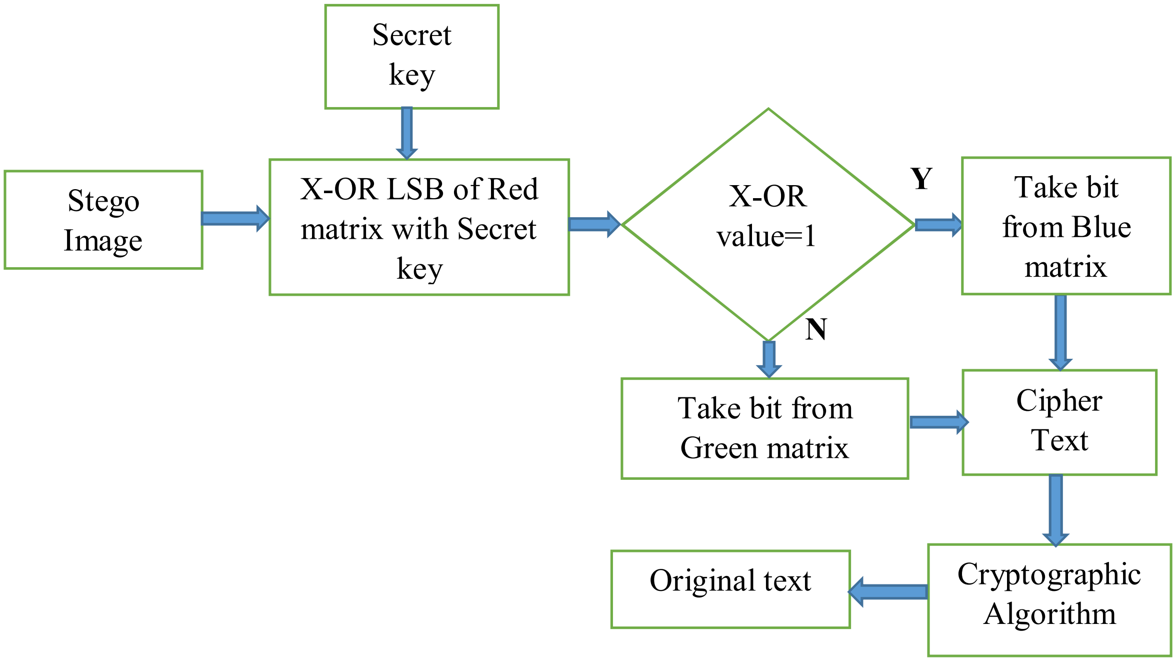

Secret key
Stego Image
X-OR LSB of Red matrix with Secret key
X-OR value=1
Y
Take bit from Blue matrix
N
Take bit from Green matrix
Cipher Text
Cryptographic Algorithm
Original text

Figure 3.10: Flowchart of Message Extraction Process

Usually, the invisibility of the hidden message is measured in terms of the Peak Signal-to-Noise Ratio. To analyze the quality of the stego image with respect to the original, the measure of PSNR is employed,

$$\mathrm{PSNR} = 10\log_{10}\frac{255^2}{\mathrm{MSE}}$$

Where mean square error (MSE) is a measure used to quantify the difference between the cover Image *I* and the stego (distorted) image *I*'. If the image has a size of M * N then

$$MSE = \frac{1}{M \times N}\sum_{i=1}^{M} . \sum_{j=1}^{N}[I(i,j) - I'(i,j)]^2$$

The method is also evaluated based on image parameters like entropy, mean and standard deviation to check the impact on image in case of replacement of bits.

CHAPTER FOUR: RESULTS, ANALYSIS AND COMPARISIONS

Experimental data:

Algorithm	Data size (bits)	Encryption Time (microseconds)	Decryption Time (microseconds)
Based on Fibonacci series	32	1.414	1.481
Rijndael		25.721	35.261
Based on Fibonacci series	64	5.65	5.712
Rijndael		74.966	104.128
Based on Fibonacci series	96	11.313	11.526
Rijndael		126.013	174.291
Based on Fibonacci series	128	22.627	22.812
Rijndael		174.922	244.768
Based on Fibonacci series	160	45.254	45.761
Rijndael		225.031	315.811
Based on Fibonacci series	192	90.509	90.618
Rijndael		274.339	386.211
Based on Fibonacci series	224	181.01	181.071
Rijndael		324.068	454.128
Based on Fibonacci series	256	362.038	362.186
Rijndael		376.027	525.261
Based on Fibonacci series	288	724.077	724.921
Rijndael		424.068	595.826

Table 4.1: Encryption/Decryption time for cryptographic algorithms

4.1 Data analysis:

The Encryption/Decryption time for the algorithms has been computed using MATLAB. It has powerful built-in routines that enable a very wide variety of computations. It also has graphics commands that are easy to use which makes the visualization of results immediately available. Specific applications are collected in packages referred to as toolbox. From these data, it was found that the encryption/decryption time for the algorithm based on Fibonacci series increases exponentially whereas that of Rijndael algorithm increases linearly. So, it's better to use cryptographic algorithm based on Fibonacci series for smaller texts. However for larger text messages, Rijndael cryptographic algorithm is preferred.

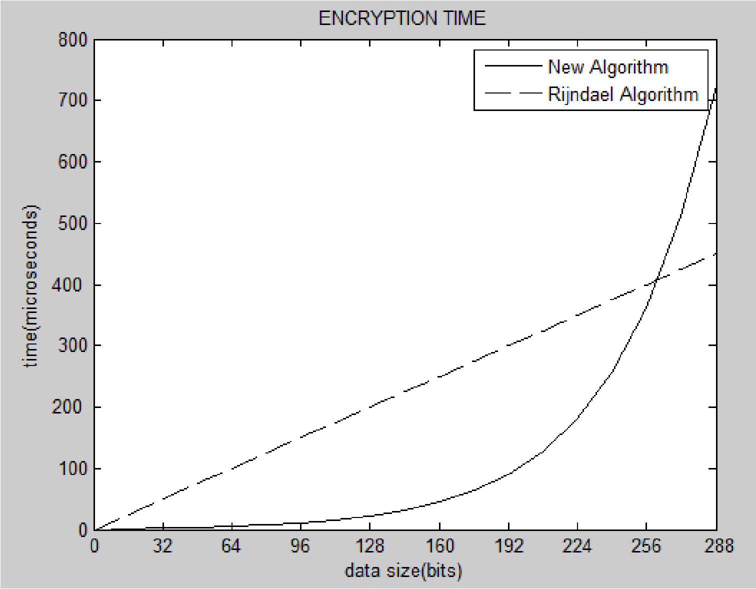


Figure 4.1: Comparative analysis of encryption time of two algorithms

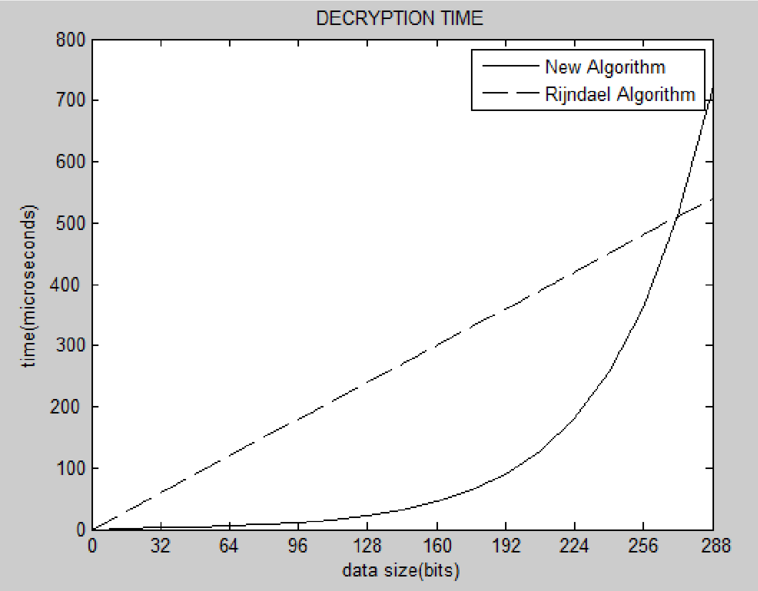
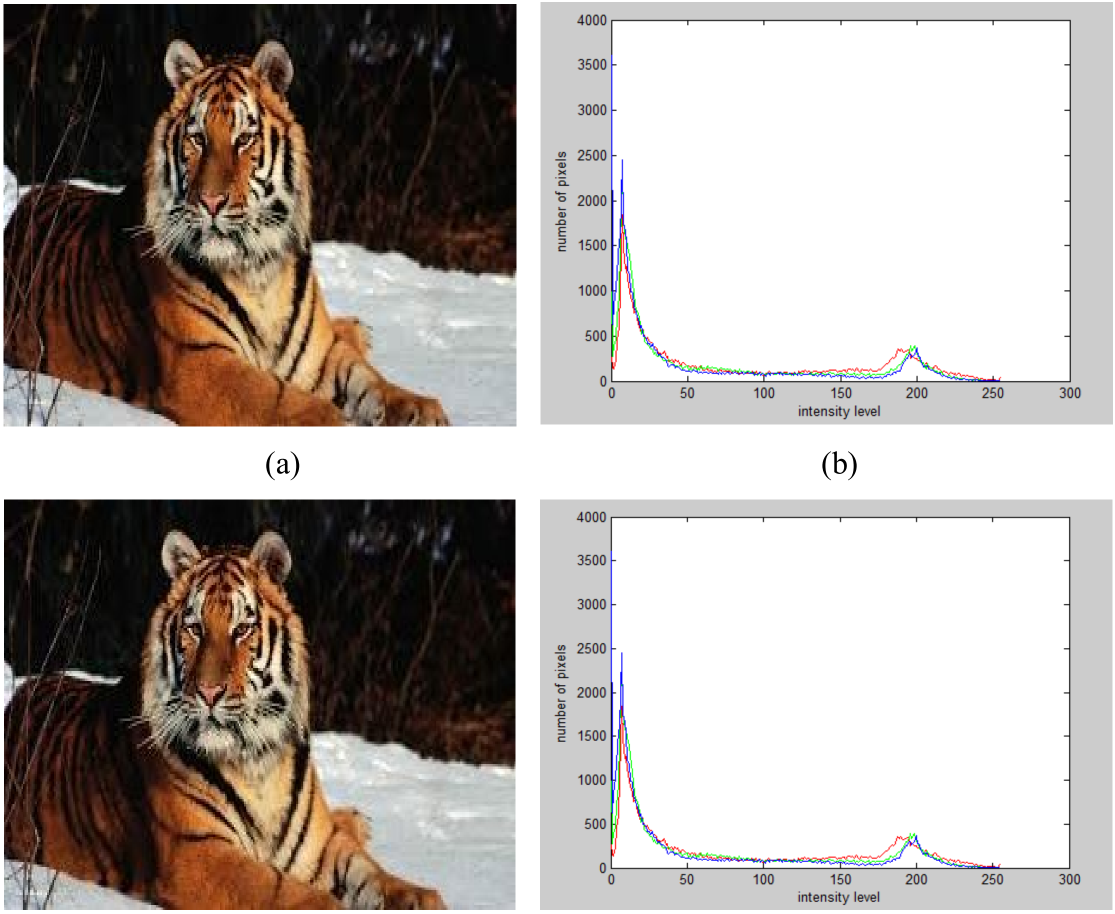


Figure 4.2: Comparative analysis of decryption time of two algorithms

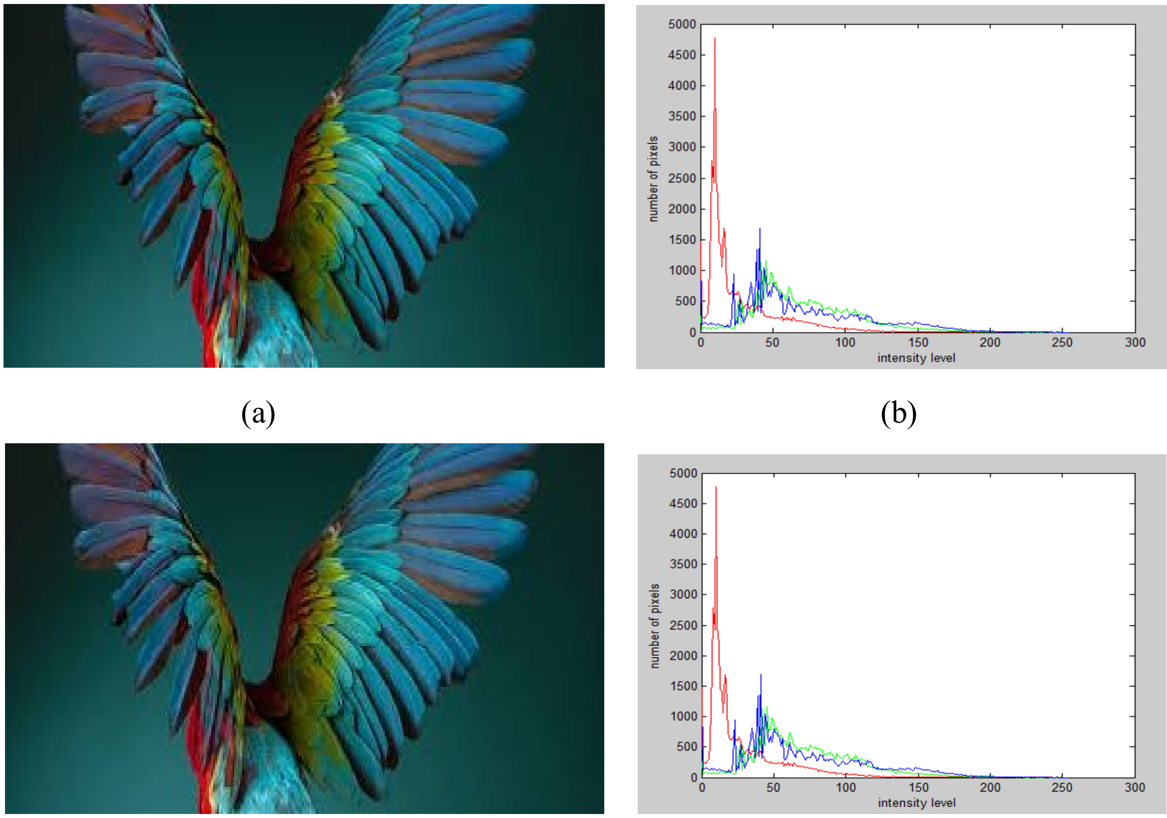
4.2 Images with their histograms of individual RGB Channels



(a) (b)

(c) (d)

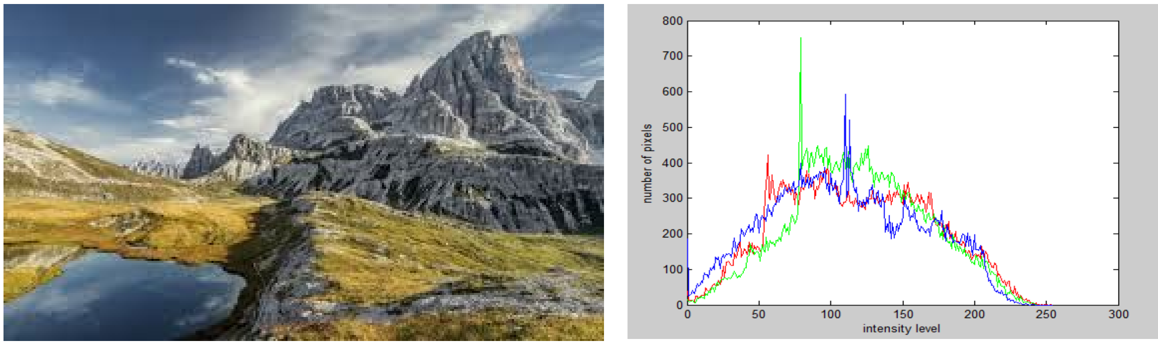
Figure 4.3: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image



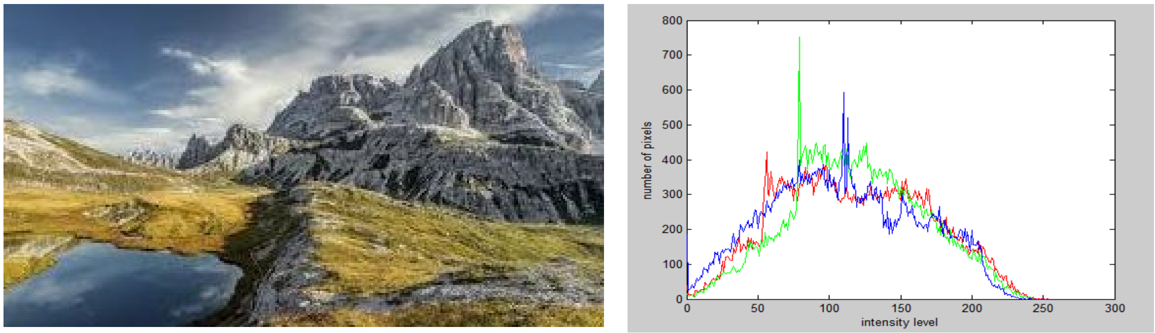
(a) (b)

(c) (d)

Figure 4.4: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image

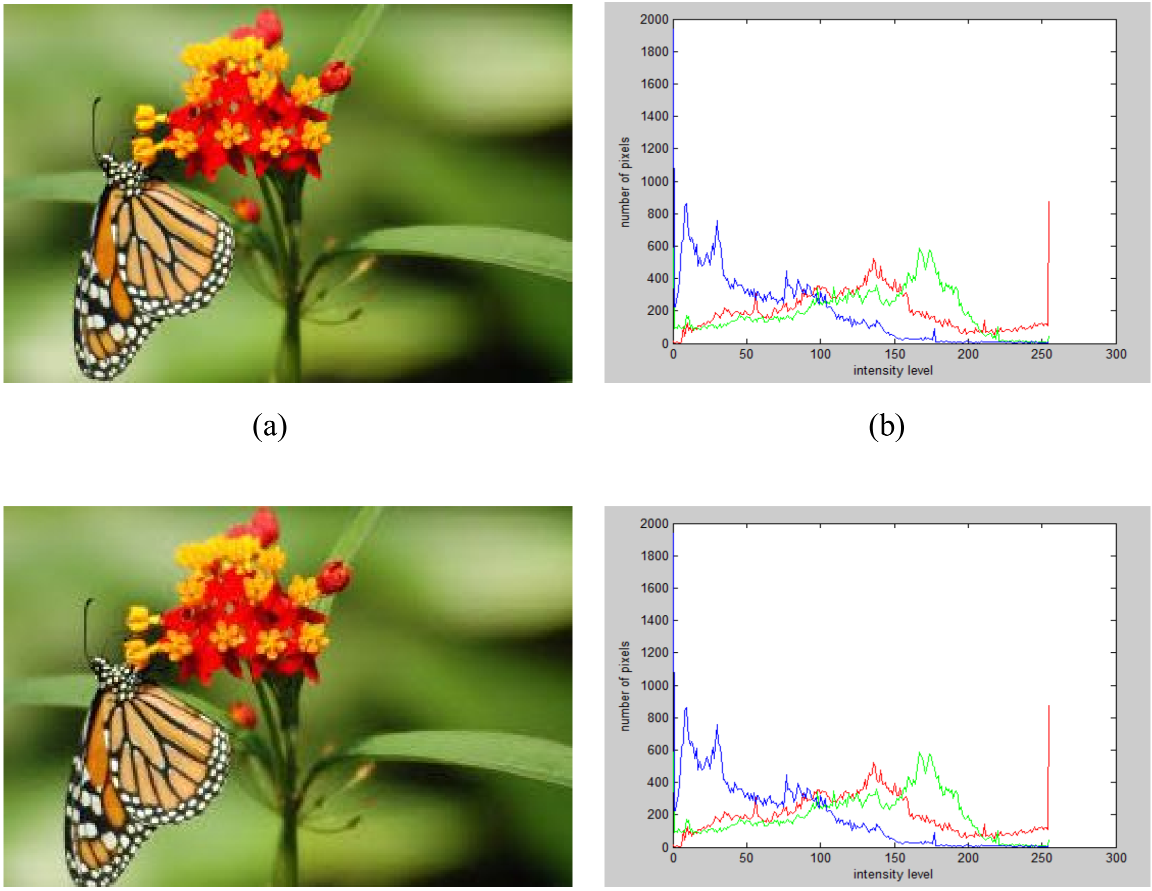


(a) (b)



(c) (d)

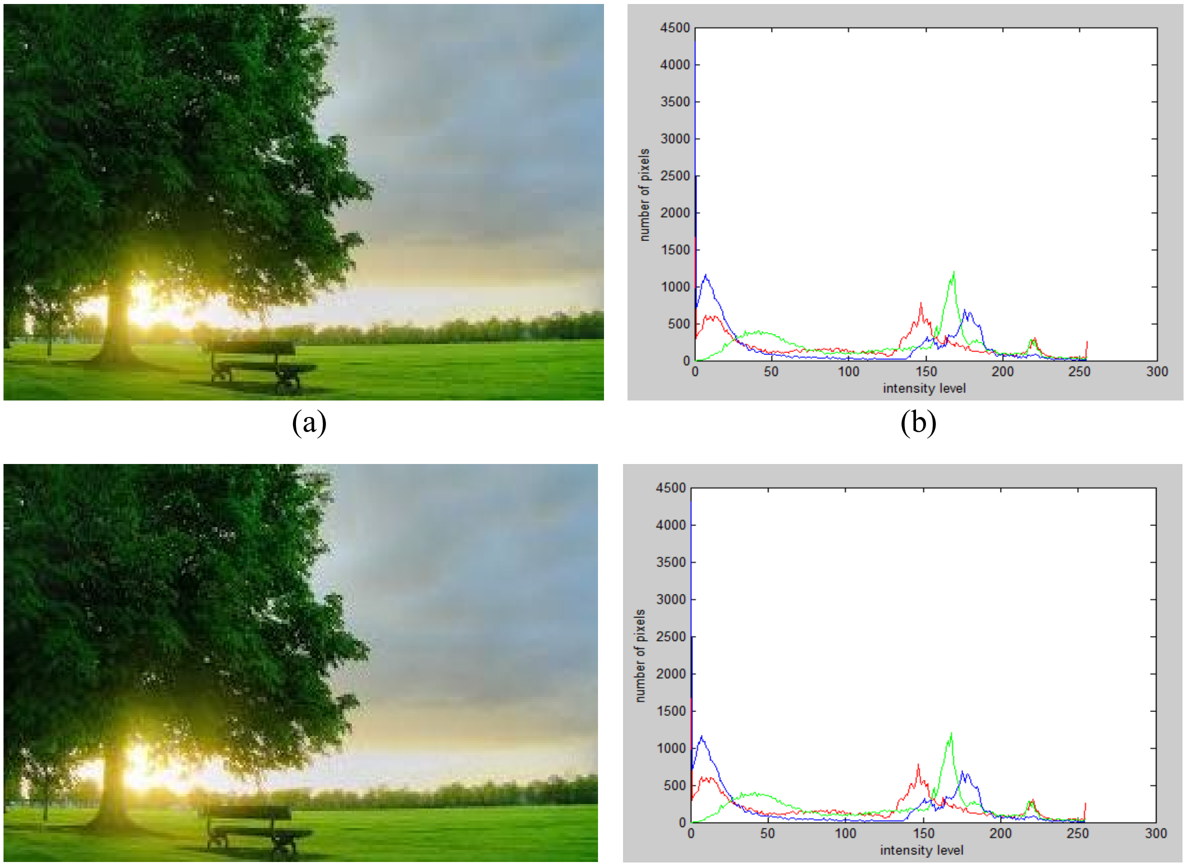
Figure 4.5: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image



(a) (b)

(c) (d)

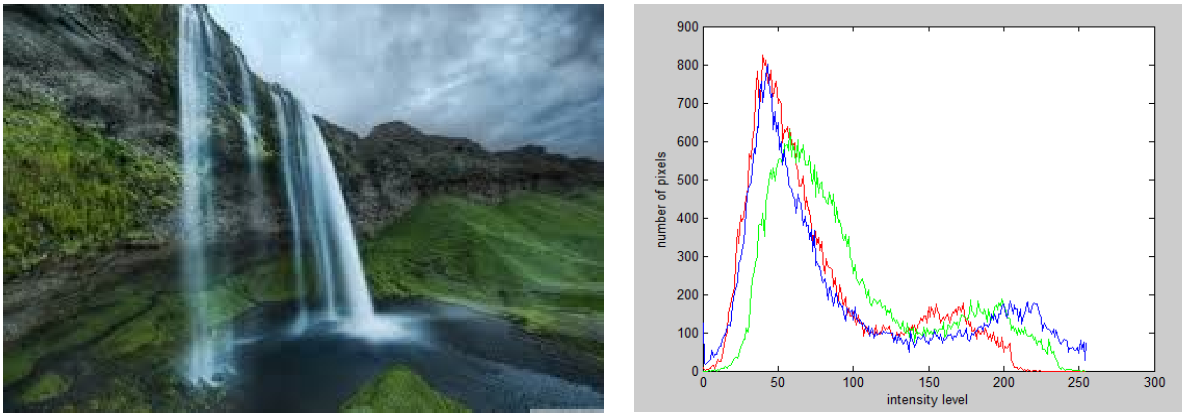
Figure 4.6: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image



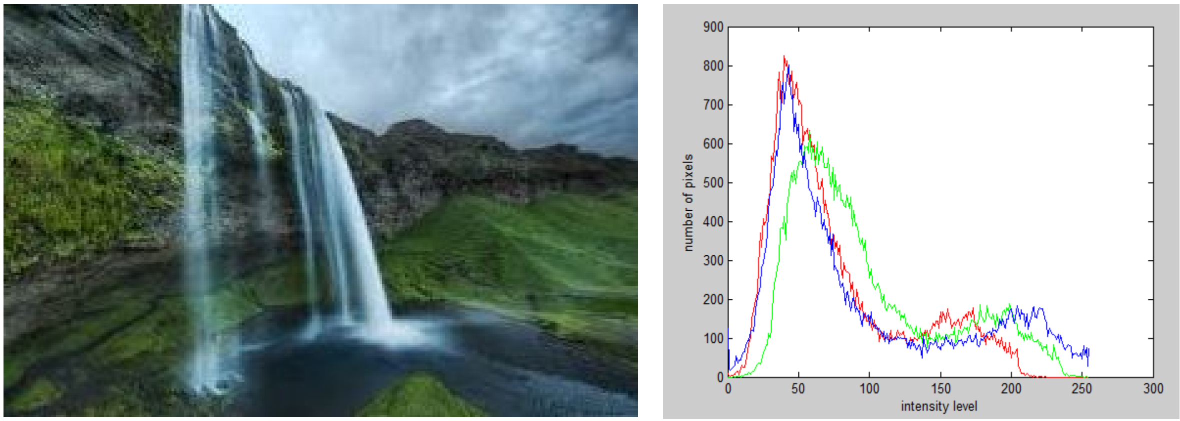
(a) (b)

(c) (d)

Figure 4.7: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image

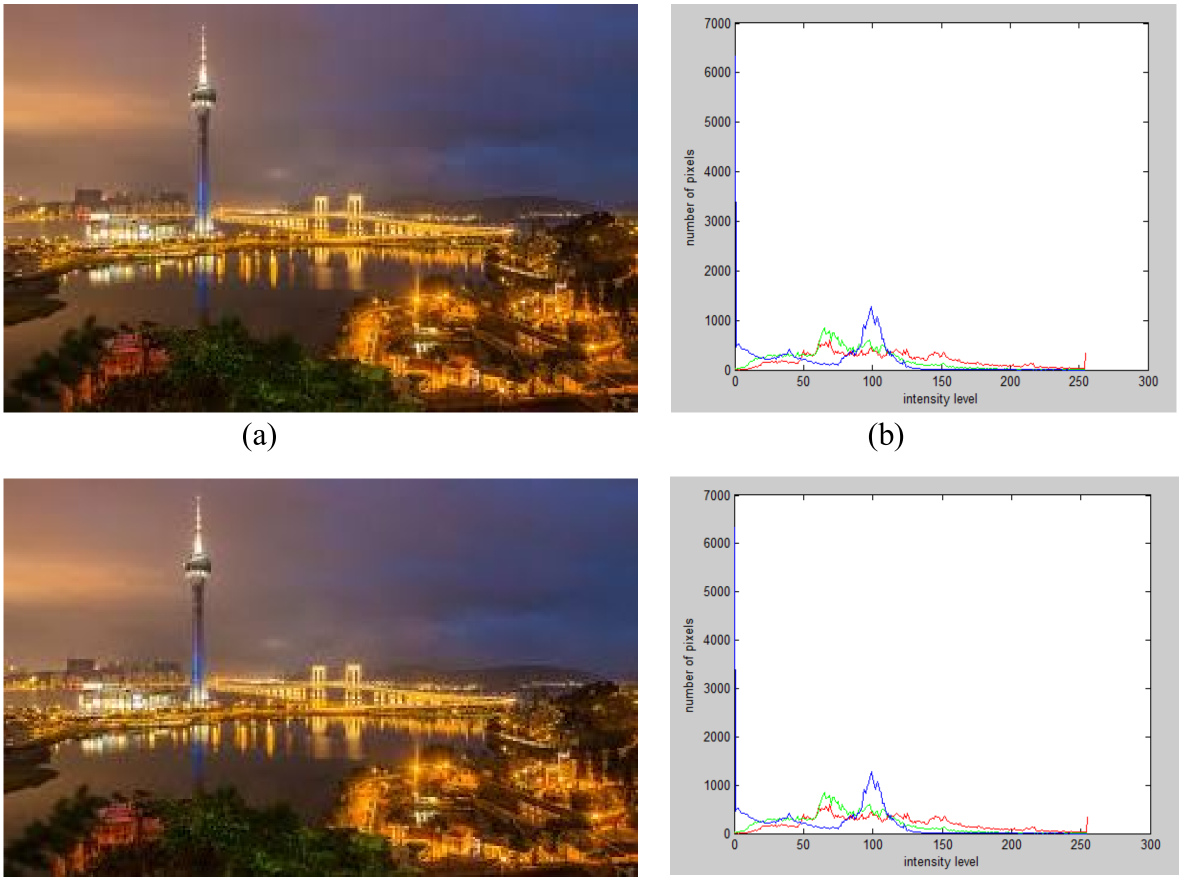


(a) (b)



(c) (d)

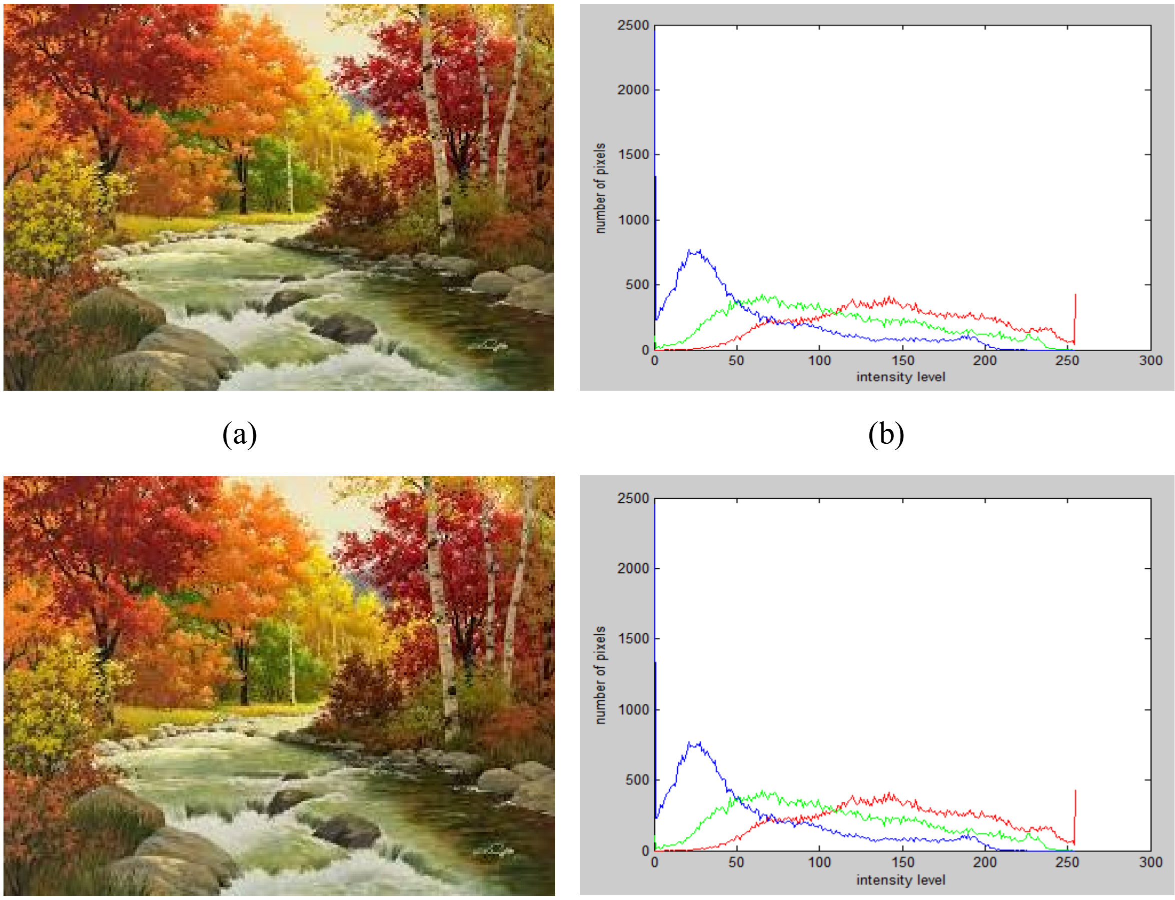
Figure 4.8: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image



(a) (b)

(c) (d)

Figure 4.9: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image



(a) (b)

(c) (d)

Figure 4.10: (a) Original Image (b) Histogram of Original Image (c) Stego Image (d) Histogram of Stego Image

4.3 Comparisions

Digital images often have a large amount of redundant data and for this reason it is possible to hide message inside image file. Image steganography is about exploiting the limited power of the human visual system where we hide information in the least significant bit of the image data. This embedding method is based on the fact that the least significant bit in an image can be thought of as random noise, and consequently they become less responsive to any change on the image. The result also showed the negligible difference in the original image and stego image. The experimental results performed on hiding 2 KB of useful data on 50KB image size showed that compared with Least Significant Bit Substitution, Improved Least Significant Bit substitution has lower MSE and higher PSNR. This shows that Improved Least Significant Bit substitution is an improvement over simple Least Significant Bit substitution.

Images	LSB substitution		Improved LSB substitution	
	MSE	PSNR	MSE	PSNR
Figure 4.2.1(Image 1)	0.0041	72.0236	0.0036	72.6236
Figure 4.2.2(Image 2)	0.0021	74.9132	0.002	75.0126
Figure 4.2.3(Image 3)	0.0032	73.1631	0.0029	73.4362
Figure 4.2.4(Image 4)	0.0048	71.2673	0.0039	72.2136
Figure 4.2.5(Image 5)	0.0052	71.0106	0.00427	71.8233
Figure 4.2.6(Image 6)	0.0025	74.132	0.0019	75.3167
Figure 4.2.7(Image 7)	0.0057	70.5316	0.0052	70.9624
Figure 4.2.8(Image 8)	0.0025	74.2362	0.0015	76.236

Table 4.2: Comparative analysis of LSB Substitution and Improved LSB Substitution

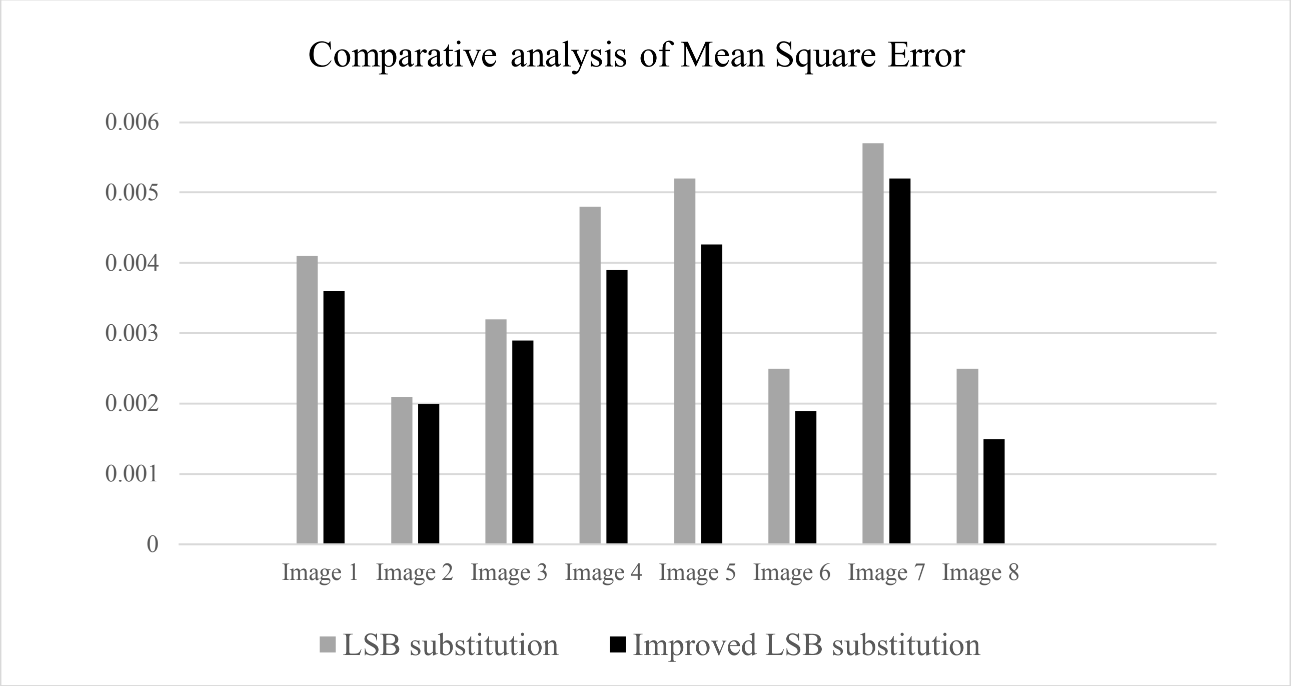


Figure 4.11: Comparative analysis of Mean Square Error

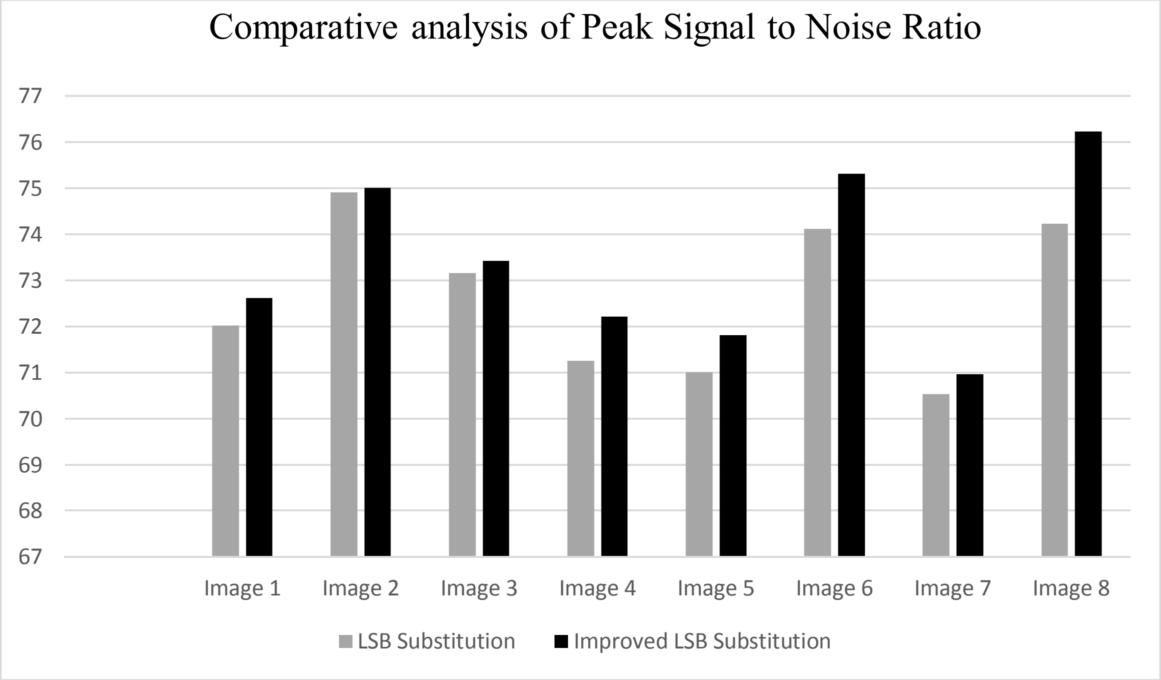


Figure 4.12: Comparative Analysis of PSNR

4.4 Results and Discussions

The experimental results showed that the method is an effective way to integrate the secret information within an image as it is very difficult for the unauthorized users to identify the changes in stego image. The goal of steganography is to keep its information undetectable. Generally secret information is stored into the specific position of Least Significant Bit of a cover image which may be detected by using various retrieval methods. The main intention of image steganography is to ensure the security of secret information. For security purpose, the addition of secret key ensures the security of secret information. The insertion of secret information is totally controlled by the secret key which decides the appropriate position of hidden information. Secret key and Red matrix are used only for decision making to allocate the secret information bits into either Green matrix or Blue matrix which gives negligible difference in the overall appearance of an image. The use of secret key and red matrix for decision making further enhances the security of data. Hiding secret information in blue matrix or green matrix further increases the complexity in extracting information and thus protects useful data from being used by intruders. The use of the secret key gives a way to secure the information from illegal user. This increases the complexity level in retrieving the hidden information. So, by using a secret key, we can increase the security level of the hidden information in improved LSB based image steganography. With the negligible difference in original image and stego image, the embedded information can be transmitted securely. The experimental results performed on hiding 2 KB of useful data on 50KB of image showed that compared with Least Significant Bit Substitution, Improved Least Significant Bit substitution has lower MSE and higher PSNR. This shows that Improved Least Significant Bit substitution is an improvement over simple Least Significant Bit substitution.

CHAPTER FIVE: EPILOGUE

5.1 Conclusion

Cryptography has evolved from an ancient science to an important area of research to secure communications. By the use of cryptography along with steganography for secure communication, we can safe guard ourselves from being compromised by those who could steal our information. Hence, an efficient image steganography algorithm has been made more secure with the help of cryptographic algorithms based on Fibonacci series and Rijndael cryptographic algorithm. The combination of these two methods will enhance the security of the data embedded. With the negligible difference in original image and stego image, the embedded information can be transmitted securely. The increase in complexity level in retrieving information further enhances the security of secret data in improved LSB substitution. Compared with Least Significant Bit Substitution, Improved Least Significant Bit substitution has lower MSE and higher PSNR. This shows that Improved Least Significant Bit substitution is an improvement over simple Least Significant Bit substitution. Hence, this combinational methodology provide resistance against various visual and statistical attacks.

5.2 Limitations and future enhancements

Data hiding techniques have been widely used for transmission of secret messages for a long time. Ensuring data security is a big challenge for computer users. Since the LSB of image is prone to error, the system can be made more secure by introducing various error recovery techniques. Likewise, for hiding secret information, larger image size is required which can be minimized by replacing other bits at the cost of effective appearance on an image. One of the limitations in symmetric cryptography is the difficulty in transmitting the secret key. However, cryptography assures privacy whereas Steganography assures secrecy. Even though both methods provide security, to add multiple layers of security it is always a good practice to use Cryptography and Steganography together. The method can be defined as undetectable, strong and secured communication of data related to the multimedia image.

6. References

[1] Raphael, J., Sundaram, V. “Secured communication through Fibonacci series and Unicode symbols”. *International journal of scientific and engineering research*, Volume 3, Issue 4, 2012, 1-5.

[2] Gangwar, A. Shrivastava, V. “Improved RGB-LSB Steganography Using Secret Key”,*International Journal of Computer Trends and Technology-Volume 4, Issue2-2013*

[3] Patel, K., Vishwakarma, S., “Triple Security of Information Using Steganography and Cryptography”. *International journal of Emerging Technology and Advance Engineering, 2013*

[4] Caldwell, J., “Steganography using the technique of orderly changing pixel component”, *International Journal of Computer Applications,* Vol.58, No.6, 2014.

[5] C. P. Sumathi and T. Santanam, “A Study of Various Steganographic Techniques Used for Information Hiding”, *International Journal of Computer Science & Engineering Survey*, 2013

[6] M. A. Ahmad et al., “Achieving Security for Images by LSB and MD5”, *Journal of Advanced Computer Science and Technology Research,* 2012

[7] T. Zhang and X. Ping, Reliable detection of LSB steganography based on the difference image histogram, IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 3, pp.545-548 April 2003.

[8] Li Zhi, Sui Ai Fen., “Detection of Random LSB Image Steganography” *The IEEE 2003 International Symposium on Persona1,lndoor and Mobile Radio Communication Proceedings, 2004.*

[9] Li, C., Xu, W., Meng, L., Liu, B., Wang, Y. and Wu, L. ; “ Realization of a LSB Information Hiding algorithm Based on Lifting Wavelet Transform Image”, *International Conference on Mechatronic Science, Electric Engineering and Computer,* pp.1015-1018, 2011.

[10] A. Joseph Raphael and Dr. V. Sundaram, 2011. "Secured crypto stegano communication through unicode symbols". *World of computer science and information technology journal*, Volume 1.

[11] Fabien, A. P., Petitcolas, F. A. P., Anderson, R. J. and Kuhn, M. G. "Information Hiding: A Survey", *Proceedings of the IEEE*, Vol. 87, No. 7, pp. 1062-1078, 1999.

[12] Provos, N. & Honeyman, P.; "Hide and Seek: An introduction to steganography", *Security and Privacy*, Vol.1, pp.32-44, 2003.

[13] Shirali-Shahreza, S. and Shirali-Shahreza M.; "Steganography in Textiles", *4th International Conference on Information Assurance and Security,* pp.56-61, 2008.

[14] Singh, K. M., Singh, L. S., Singh, A. B. and Devi, K. S.; "Hiding Secret Message in Edges of the Image", *International Conference on Information and Communication Technology* (ICICT), pp.238-241, 2007.

[15] Rahate, N. D. and Rothe, P. R.; "Data Hiding Technique for Security by using Image Steganography", *International Conference on Industrial Automation and Computing* (ICIAC),pp. 33-36, 2014.

[16] J. Daemen and V. Rijmen, *AES Proposal: Rijndael*, AES Algorithm Submission, September 3, 1999.

APPENDIX

Figures used for cover images