بسم الله الرحمن الرحيم

Sudan University of Science and Technology
College of Graduate Studies

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Information Technology

Secure audio Steganography using modified LSB and DCT

إخفاء البيانات في الصوت باستخدام خوارزميتي البت الأقل اهمية المعدله و تحويل جيب التمام المتقطع

Prepared By:
Mohammed alfath ahmed ismail
Supervised By:
Dr. Faisal Mohamed Abdallah Ali

Sep 2021

آیه

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيمِ

اَللّٰهُ لَآ اِلٰهَ اِلَّا هُوَ الْحَيُّ الْقَيُّومُ

لَا تَأْخُذُهُ سِنَةٌ وَلَا نَوْمٌ لَهُ مَا فِي السَّمٰوٰتِ وَمَا فِي الْاَرْضِ مَنْ ذَا الَّذِي يَشْفَعُ عِنْدَهُ
اِلَّا بِاِذْنِهِ يَعْلَمُ مَا بَيْنَ اَيْدِيهِمْ وَمَا خَلْفَهُمْ وَلَا يُحِيطُونَ بِشَيْءٍ مِنْ عِلْمِهِ اِلَّا بِمَا شَآءَ
وَسِعَ كُرْسِيُّهُ السَّمٰوٰتِ وَالْاَرْضَ وَلَا يَئُودُهُ حِفْظُهُمَا وَهُوَ الْعَلِيُّ الْعَظِيمُ

Dedications

This research is dedicated to:

The sake of Allah, my Creator and my Master, my great teacher and messenger, Mohammed (May Allah bless and grant him) who taught us the purpose of life, Sudan University of Science and Technology, my second magnificent home, my great parents, who devoted themselves helped to get this point My beloved brothers and sisters, my beloved family Who always can be with me. And my bosom friends who encourage and support me, all the people in my life who touch my heart I dedicate this research.

Acknowledgements

*Praise is to **Allah**, the almighty for having guided me at every stage of my life. I would like to thank my research supervisor **Dr**. **Faisal Mohamed Abdullah Ali for** giving me the opportunity to work with him and guiding and helping me throughout this research and other courses. Also, l would like to thank **my colleagues** for their encouragement and insightful comments, and special thanks to **H. Abdullah awed***

Abstract

In a world of digital technology, maintaining the security of the secret data has become a big challenge. One way to achieve this is to encrypt the message before it is sent. But encryption draws the attention of third parties, which may cause the third party to seek breaking the encryption and detecting the original message. Another way is steganography, steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message. In this thesis a concept for performing hidden secret data, called secure audio Steganography using modified Least Significant Bit (LSB) and Discrete Cosine Transform (DCT), was presented. Which is consist of two stenographic methods utilized respectively. Two-levels of stenography have been applied; the first level is called (the lower-level), and it has been applied using DCT. In this level three different plain texts are used as a secret data, and the cover is gray scale image, the output from the first level is a stego image. The second level is called (the Upper-level); it has been applied using modified Least Significant Bit. In this level Wav file has been used as a cover media and embeds (the gray scale image output from the first level) as a secret data.
After implementation of the proposed method, many experiments have been conducted. Different sizes of plain texts and different sizes of cover have been experimented. Finally comparative analysis had been applied to experiments results and illustration to strength and weakness points of proposed system against existing systems is obtained.

المستخلص

في عالم التكنولجيا الرقمية أصبح الحفاظ علي أمن البيانات السرية تحدي كبير.احدي الطرق المستخدمة هي تشفير الرسالة قبل ارسالها ، ولكن التشفير قد يلفت انتباه طرف ثالث ،و هذا قد يتسبب في السعي الي اكتشاف الرسالة الاصلية.هنالك طريقة اخري هي إخفاء المعلومات. إخفاء المعلومات هو فن وعلم كتابة الرسائل المخفية.

في هذا البحث، تم تطبيق مفهوم اخفاء البيانات السرية،يسمي ب إخفاء البيانات في الصوت باستخدام خوارزميتي البت الأقل اهمية المحسنه و تحويل جيب التمام المتقطع والذي يحتوي علي مستويان ويتم تطبيقهم علي التوالي. المستوي الاول يسمي بالمستوي الادني(ويتم تطبيقه عن طريق تحويل جيب التمام المتقطع (DCT) لإخفاء الصور، البيانات السرية عبارة عن نص عادي باللغة الانجليزية يتم إخفائها في صور غير ملونة(رماديه).

المستوي الثاني يسمي بالمستوي الاعلي(وتم تطبيقه باستخدام خوارزمية البت الأقل اهمية المحسنه للاخفاء في الصوت . في هذا المستوي يتم استخدام صوت من نوع wav كغطاء يتم فيها اخفاء الصورة الناتجة من المستوي الأولي و الناتج يكون عبارة ملف صوتي(Stego audio).
هنالك العديد من التجارب تم إجرائها بعد الإنتهاء من تطبيق النظام المقترح ، وهذه التجارب تم إجرائها بإستخدام أحجام مختلفة من الرسائل السرية و أيضا أحجام مختلفة من الغطاء التي يتم الإخفاء فيه.و اخيرا تم عمل مقارنه بين النظام المقترح و الانظمه الحاليه و توضيح نقاط القوه و الضعف بينهما.

List of content

List of Tables

s

List of figures

List of Abbreviation

DCT	: Discrete Cosine Transformation
LSB	: least significant bit
HVS	: human visual system
2D	: two dimension
RGB	: Red, Green and Blue
LCG	: linear congenital generator
3R	: three Red
3G	: three Green
2B	: two Blue
ECA-BM	: enhancing the hiding capacity of audio steganography based on block mapping
MSB	: most significant bit
Hseq	: hopping sequence

CHAPTER ONE
INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 Background

The increasing rate of usage of the internet and the revolution that occurred in digitization of information. The overall structure of modern communication is changed. The revolution in software industry and semiconductor industry made it feasible that hardware as well as software are more user-friendly and flexible and enables consumers to communicate multimedia data. Peoples are now able to transmit large multimedia files through broadband connection. Security of data to transmit is high concern in today's communication system. Steganography and encryption are a techniques of providing data security.
Steganography is the art and science of hiding information such that its presence cannot be detected. The secret information is hidden in some cover file and then transmitted. The cover file can be an image, Audio file, text file, video file, etc.

1.2 Problem statement

Systems that use only one level of Steganography are usually more vulnerable .Due to the fact that they lack the complexity to keep the data secure. Furthermore the most commonly used Steganography algorithm which is the Normal (LSB) algorithm is proved to be weak and the secret data is easy to retrieve.

1.3 Objectives

The main objectives of this study are

1. Add more complexity to the Steganography process through applying it in two levels.
2. Try to balance the capacity of embedded data (plain text) and the changed Pixels value.
3. Study and analyze the obtained results with existing systems

1.4 Research questions

1. How the proposed method helps in hiding the secret information and Protect it from unauthorized disclosure?
2. How to use two levels of steganography with different techniques to Hide the secret information?
3. How to extract the image from audio and extract secret information (text) from image?

1.5 Research methodology

We will apply two levels of steganography. The first level Discrete Cosine Transformation (DCT) will be used. The proposed technique does not affect the quality of the stego image and is more secure than the conventional LSB techniques. This technique does not affect the image quality and the embedded secret information can be recovered with the help of reverse algorithm.
The second level will use modified LSB which hides the image file (first level output) on audio file cover.

1.6 Thesis layout

Chapter one gives introduction and brief history about the steganography. Recently Literatures review, related works and defining the types of steganography will be explained in chapter two. Chapter three Research Methodology will be explained in details. The implementation of the proposed algorithm and discussion of the results appears in chapter four and finally Chapter five contains the conclusion, recommendations and future work.

CHAPTER TWO

LITERATURE REVIEW AND RELATED WOR

CHAPTER TWO

LITERATURE REVIEW AND RELATED WORK

2.1 Introduction

Now a day, a lot of applications are Internet-based and in some cases, it is desired that the communication be made secret. There are two techniques are available to achieve this goal. One is cryptography, where the sender uses an encryption key to encrypt the message, this encrypted message is transmitted through the insecure public channel, and decryption algorithm is used to decrypt the message. The reconstruction of the original message is possible only if the receiver has the decryption key. The second method is steganography, where the secret message is inserted in another medium. Steganography is the art of hiding information through original files in such a manner that the existence of the message is unknown. The term steganography is coming from Greek word Steganos, which means, "Covered Writing". The original files can be referred to as cover text, cover image, or cover audio. After inserting the secret message, it is referred to as stego-medium. A key is used for hiding process to restrict detection and/or recovery of the embedded data. While cryptography protects the content of messages, steganography hides the message so that intermediate persons cannot see the message. (Mandal 2012)

Steganography differs from cryptography. The purpose of cryptography is to secure communications by changing the data into a form that cannot be understand. Steganography techniques, on the other hand, hide the existence of the message itself, which makes it difficult for a third person to find out where the message is. Sometimes sending encrypted information may draw attention, while invisible information will not.
Accordingly, cryptography is not the good solution for secure communication; it is only part of the solution.(Mandal 2012)
Both techniques can be used together to better protect information. In this case, even if steganography fails, the message cannot be recovered because a cryptography technique is used as well. The cracking of steganographic messages is called steganalysis. The purpose of steganalysis is to identify the information and determining that whether or not they have hidden messages encoded into them and if possible, extract the hidden information. There are many types of steganography methods. We are going to take a short look at different

steganography methods. Fig. 1 below shows the different categories of file formats that can be used for steganography techniques.(Mandal 2012)

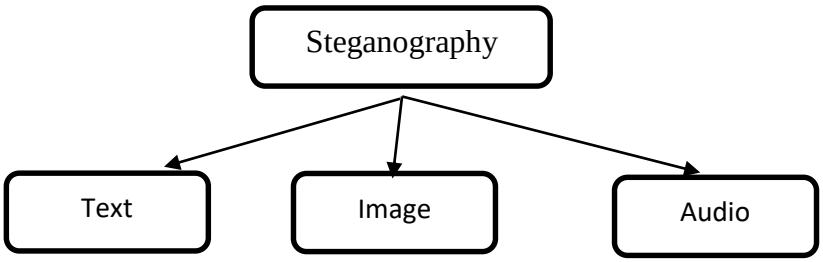


Fig. 2.1 steganography categories

2.2 Image steganography

This Steganography technique is more popular in recent year than other steganography possibly because of the flood of electronic image information available with the advent of digital cameras and high-speed internet distribution. It can involve hiding information in the naturally occurred noise within the image. Most kinds of information contain some kind of noise. Noise refers to the imperfections inherent in the process of rendering an analog picture as a digital image. In Image steganography we can hide message in pixels of an image. An image stenographic scheme is one kind of stenographic systems, where the secret message is hidden in a digital image with some hiding method(Kaur and Behal 2014)

2.2.1 Spatial domain techniques

There are many versions of spatial steganography, all directly change some bits in the image pixel values with the hiding data. A simple approach for embedding information in cover image is using Least Significant Bits (LSB). The simplest steganography techniques embed the bits of the message directly into least significant bit plane of the cover image in a deterministic sequence. Modulating the least significant bit does not result in human-perceptible difference because the amplitude of the change is small. To hide a secret message inside an image, a proper cover image is needed(Hariri, Karimi et al. 2011). Because this method uses bits of each pixel in the image, it is necessary to use a lossless compression format, otherwise the hidden information will get lost in the transformations of a lossy compression algorithm. When using a 24-bit color image, a bit of each of the red, green and blue color components can be used, so a total of 3 bits can be stored in each pixel. For example, the following grid can be considered as 3 pixels of a 24-bit color image, using 9 bytes of memory.(Hariri, Karimi et al. 2011)

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

When the character A, which binary value equals 10000001, is inserted, the following grid results:

(00100111 11101000 11001000)
(00100110 11001000 11101000)
(11001000 00100111 11101001)

In this case, only three bits needed to be changed to insert the character successfully. On average, only half of the bits in an image will need to be modified to hide a secret message using the maximal cover size. The result changes that are made to the least significant bits are too small to be recognized by the human visual system (HVS), so the message is effectively hidden. As you see, the least significant bit of third color is remained without any changes. It can be used for checking the correctness of 8 bits which are embedded in these 3 pixels. In other words, it could be used as "parity bit".(Hariri, Karimi et al. 2011)

2.2.2 Frequency Domain Technique

In frequency domain, images are first transformed and then the message is embedded in the image. When the data is embedded in frequency domain, the hidden data resides in more robust areas, spread across the entire image, and provides better resistance against statistical attacks. There are many techniques used to transform image from spatial domain to frequency domain. The most common frequency domain method usually used in image processing is the 2D discrete cosine transform. In this technique the image is divided into 8×8 blocks and DCT transformation on each block is performed. DCT arranged the pixel of image according to their frequency value. The data bits are embedded in the low frequency coefficients of DCT. (Singla and Syal 2012)

2.3 Audio Steganography

Audio steganography is focused in hiding secret information in an innocent cover audio file or signal securely and strongly. By embedding secret information using an audio signal as a cover medium, the very existence of secret information is hidden away during communication. This is a serious and vital issue in some applications such as battlefield communications and banking transactions.in a computer-based audio steganography system, secret messages are embedded in digital sound. The secret message is embedded by slightly altering the binary

sequence of a sound file, some of audio steganography methods are described below(Shrivastava and Patidar)

2.3.1 Echo hiding

This method adds an echo to the main audio signal to embed information within an audio file. Three echo parameters that are important to embedding data: decay rate, offset and initial amplitude. Just one bit of information can be encoded, if only one echo is created from the original signal. It becomes more difficult for the human ear to separate among the two signals as the delay decrease between the echo and the original audio. Furthermore, offset to show the binary message is varied. A binary one shows by the first offset, and binary zero shows by a second offset value. Just one information bit can be encoded, if just one echo from the original signal is created. Thus, before starting the encoding process the original signal is broken into the blocks. The blocks are combined together when the encoding process is finished to produce the last signal(Kaur and Behal 2014).

2.3.2 Standard LSB algorithm

In the scope of steganography, LSB algorithm is one of the best methods to analyze information hiding. In this method least significant of binary sequences of each digitized audio sample is replaced with the secret message binary equivalent LSB coding permits to a large amount of information to be encoded by replacing a binary message with the least significant bit of each sampling point. For instance, for embedding the sample value of '1000001' which is equivalent to letting 'A' into an audio signal that each sample are introduced with 16 bits, after that, least significant bit of 7 sequential samples are substituted with each bit of the binary equivalent of the letter 'A'(Kaur and Behal 2014).

2.4 Multi-Level Steganography

In multilevel steganography, at least two stenographic methods are utilized simultaneously, in such a way that one method (called the upper-level) serves as a carrier for the second one (called the lower-level). Such a relationship between two (or more) information hiding solutions has several potential benefits. The most important is that the lower-level method stenographic bandwidth can be utilized to make the steganogram unreadable even after the detection of the upper-level method: e.g., it can carry a cryptographic key that deciphers the steganogram carried by the upper-level one., thus possibly making the stenographic communication harder to detect.(Frączek, Mazurczyk et al. 2012)

2.5 Tools

The programming language used in the implementation of a multilevel data hiding scheme is MATLAB.MATLAB (R2015a) is also used to evaluate the results of the proposed method by calculating the PSNR and MSE of image and audio files.

2.6 Evaluation Parameters

2.6.1 Peak Signal to Noise Ratio (PSNR)

The PSNR is abbreviated as the peak signal-to-noise ratio. Basically, it is the ratio of two terms. Generally it is represented as

$PSNR$=10 log10 (MAX$f\sqrt{MSE}$)

The maximum error between the original image and modified image is obtained in terms of PSNR. PSNR and the quality of image are directly proportional to each other. It means higher the PSNR, image quality is also high. To calculate PSNR first we want to check mean square error i.e. MES. With the MES the PSNR is represented as

MSE= 1mn Σ Σ| |n−10m−10 (i,)−(i,j) | | 2

The output is in decibels (dB) among two images. For the quality comparisons purpose this ratio is used. It is the comparison between original image and modified image or compressed image(Aldabagh 2020)

2.6.2 Mean Square Error (MSE)

Mean square error defines or gives the performance as per the mean of the squared errors. It is the network function. Basically the cumulative difference in between original image and modified image is known as MES. Another way we define it as square of pixels difference values between original image and modified image. Smaller value of MES represents the improvement in quality of image. It reduces the error. Mathematically it is represented as,

MSE= 1mn Σ Σ| |n−10m−10 (i,)−g(i,j) | | 2

Where M*N size of image i.e. no of rows and columns of original input image.(Aldabagh 2020)

2.7 Related works

This part describes several related works in image steganography and audio steganography using different techniques.

2.7.1 In (Rajput and Chavan 2018)G. G. Rajput, Ramesh Chavan presented Improved LSB based Image Steganography using Run Length encoding and Random Insertion technique for Color Images, in this paper a novel method for secret message hiding in color images is proposed. The message is encoded by extracting the RGB components of a color image. Run length encoding is performed on the data and insertion of the data in least significant bits (LSB) of the pixel is guided by linear congruential generator (LCG). A 3R-3G-2B LSB pattern is recommended for insertion of the data making the information more secure without bringing any significant distortions to the original image.
In RGB cube model, a pixel in a color image possesses three components; Red (R), Green (G), and Blue (B). Each component comprises of 8 bits. These R, G, and B components (channels) can be treated as independent bytes and LSB substitution can be applied. In simplest LSB substitution, it means 3 data bits can be hidden in one pixel. However, it is not wise to implement in this form, since such approach is vulnerable to attacks for secret message retrieval. The method proposed is described below.

Hiding the Secret Message (Data Hiding)

The cover image is a color image with 24 bits per pixel described in RGB color space. The secret text message is binarized and stored as stream of bits. Run length encoding is performed on the stream of bits. Angular transformation is performed on the cover image and the three channels, R, G, and B, respectively, of the cover image are extracted and Run Length Encoded data is inserted in the LSBs of the pixels of the channels in the following pattern: 3 LSBs of R channel, 3 LSBs of Green channel and 2 LSBs of Blue channel- a total of 8 bits are used per color pixel. However, the choice of pixel is based on linear congruential generator (LCG). Given a seed, LCG generates a sequence of pseudo random numbers which are taken as pixel positions in the channels and the sequence is followed to insert the secret data in LSBs positions in pattern specified. The number of pixels used for inserting the data is recorded in the last pixel of the cover image. After the insertion, reverse angular transformation is performed to generate the final stego-image

Step 1. Read the cover medium i.e., color image.
Step 2. Read the secret message (text), perform run length encoding and then binarize.
Step 3. Compare size of binarized secret data against size of cover image to ensure

that the cover image is not distorted after embedding. For example, for true image 24bit of size 20 × 20 pixels, (8 bits/ pixel) 3200bits of binaries data can be embedded using LSB technique.

Step 4. A sequence of random positions is generated using LCG method with a choice of seed value. These positions represent the pixel positions in the channels of color image.

Step 5. Starting from the first random position of pixel, insertion of data is performed in 3R-3G-2B pattern

Step 6. The number of pixels used for inserting is written in LSB of the last pixel of the image.

Step 7. Reverse angular transformation is performed to retain original position of the cover.

Step 8. Output the stego image

Secret Message Retrieval

The process of retrieving the secret message from stego-image is presented below.

Step 1. Read the stego image.

Step 2. Using stego key (seed value), generate the sequence of random numbers representing the position of the pixels used for inserting text in RGB channels. Following the pixel positions, read the data bits in 3-3-2 pattern and store it in the array. The number of pixels to read is known from the data embedded in last pixel of the stego- image.

Step 3. Perform run-length decoding on the extracted bits.

Step 4. Output the secret message.

2.7.2 In (Ali, Mokhtar et al. 2017)Ahmed Hussein ali and loay Baghdad presented enhancing the hiding capacity of audio steganography based on block mapping which is an audio hiding scheme using fractal coding and chaotic LSB to improve the efficiency of the audio data hiding. Fractal coding is employed to find a mapping between the secret and cover blocks in order to decrease the amount of the secret data and improve the hiding capacity. Chaotic LSB is adopted as an embedding technique for two reasons, first to enhance the security of ECA-BM and second to maintain the fidelity of the stego. The experimental results exhibit the relation among block length, hiding capacity, the fidelity of stego and retrieved files. The achieved hiding capacity is 80 % of the cover file with maintaining the fidelity of the stego and reconstructed secret file 69.3 and 38.9 dB respectively. Like other steganography techniques, ECA-BM comprises embedding and extraction processes as shown below

Embedding Process:

This process is run on the sender side. It begins with the loading of the secret and cover file then, splitting the data from the header. The cover samples are partitioned into overlapped blocks while the secret samples into non-overlapped blocks. Each block has N samples; N depends on the ratio between the size of the cover to the secret file and the number of bits required for each IFS code. Because of its simplicity and requiring less
Computation time, fixed partitioning is used. Mean and variance for each secret and cover block are calculated using Eq. At the end, each secret block is encoded to a set of IFS coefficients by matching it with all cover
Blocks by obtaining the most similar cover block with minimum error using Eq. The IFS coefficients for each secret block consist of optimal domain position, scale, symmetry, and range mean. The binary sequence of the IFS is embedded chaotically using the secret key in the cover samples and the 1-LSB in each cover sample is
Modified for hiding the secret bits.

Extraction Process:

The extraction process is quite simple and straightforward. Throughout this process, the particular receiver will collect the IFS coefficients using the secret key. The LSB bits of the selected stego-file samples are gathered to recreate the IFS coefficients in the same order as in embedding process. The retrieved coefficients are used to reconstruct the secret blocks using Eq. then the reconstructed secret file is created.

2.7.3 In(Divya and Reddy 2012) S.S. Divya, M. Ram Mohan Reddy presented hiding text in audio using multiple LSB steganography and provide security using cryptography with 2 novel approaches of LSBs of audio samples for data hiding. These methods check the MSBs of the samples, and then number of LSBs for data hiding is decided. In this way, multiple and variable LSBs are used for embedding secret data. These proposed methods remarkably increase the capacity for data hiding as compared to standard LSB without causing any noticeable distortion to the data.

Using MSB algorithm:

This method considers the value of the MSB of the digitized samples of cover audio for data hiding. Representation of the embedding procedure. The steps for data embedding and data retrieval as follow.

Table 2.1 steps for data embedding and data retrieval

MSB bit	No of LSBs needed for data embedding
0	6
1	7

Steps for Data Embedding:

1. Read the cover audio signal.
2. Write the text in an in file to be embedded. Convert it into a sequence of binary bits.
3. Every message bit from step 2 is embedded into the variable and multiple LSBs of the samples of the digitized cover audio cover.
4. For embedding purpose, the MSB of the cover sample is checked. As shown in above table.
If MSB is „0" then use 6 LSBs for data embedding.
If MSB is „1" then use 7 LSBs for data embedding.
5. The modified cover audio samples are then written to the file forming the stego object.

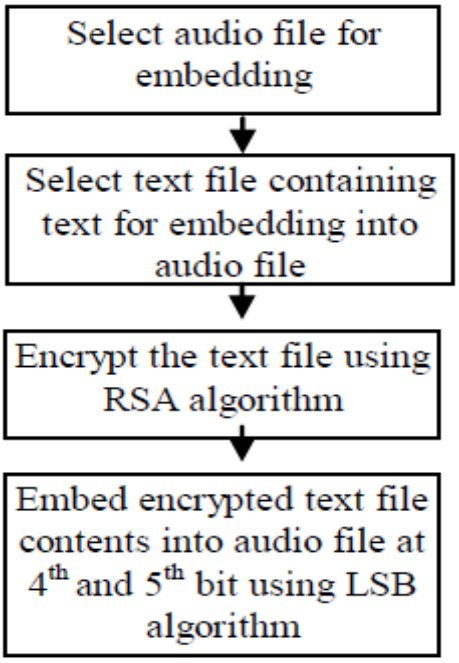
Steps for Data Retrieval:

1. Read the stego object.
2. Retrieval of message bits is done by checking the MSB of the samples.
If MSB is „0" then use 6 LSBs for data retrieve.
If MSB is „1" then use 7 LSBs for data retrieve.
3. After every such 16 messages bits retrieved, they are converted into their decimal equivalents and finally the secret audio signal reconstructed.

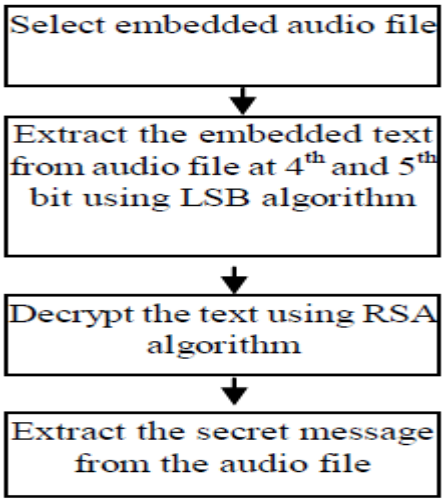
2.7.4 in (Padmashree and Venugopala 2012) Represented the complete working of the audio steganography process of embedding the encrypted secret message using public key cryptographic algorithm, RSA into the 4th and 5th layers of the audio file. In the sender side, the text file which has to be embedded into an audio file is encrypted using public key cryptographic algorithm, RSA. The cipher text obtained is then embedded in the 4th AND 5th LSB bit using one of the Steganography algorithms, LSB algorithm. The resultant audio file contains the secret message embedded into it. On the receiver side, the embedded audio file is selected to extract the secret message. The secret message is decrypted using RSA decryption method and the secret messages are compared before embedding and after embedding. Also, comparisons are made based on PSNR of both original

audio file and embedded audio file, to indicate that less noise intrusion even after changing the 4th and 5th LSB bit of the original wave

Steps for embedding



Steps for extracting



The results of the experiment conducted by changing the 4th and 5th LSB bit with different data have been tabulated below

Table 2.2 SNR/PSNR Values for Same Audio File with Varying Text Content Sizes

	Audio File Duration : 60sec		
File Name	**Size (Bytes)**	**SNR**	**PSNR**
Text1	103	-6.84E-09	16.66214532
Text2	100	-6.80E-09	16.66214532
Text3	75	-6.88E-09	16.66214532
Text4	50	-6.85E-09	16.66214532
Text5	25	-6.84E-09	16.66214532

2.7.5 in (Chandran and Bhattacharyya 2015) This paper deals with hiding text in an image file using Least Significant Bit (LSB) based Steganography, Discrete Cosine Transform (DCT) based Steganography and Discrete Wavelet Transform (DWT) based steganography. The LSB algorithm is implemented in spatial domain in which the payload bits are embedded into the least significant bits of cover image to derive the stego-image whereas DCT & DWT algorithm are implemented in frequency domain in which the stego -image is transformed from spatial domain to the frequency domain and the payload bits are embedded into the frequency components of the cover image. Comparative analysis of LSB based, DCT based & DWT based steganography has been done on basis of parameters like PSNR, MSE, Robustness
& Capacity on different images and the results are evaluated.

Table 2.3 PSNR/MSE values of LSB technique

Cover Image	PSNR(dB)	MSE(dB)
Jet	52.7869	.58505
Baboon	53.7558	.52329

Table 2.4 PSNR/MSE values of DCT technique

Cover image	PSNR(dB)	MSE(dB)
Jet	55.6473	.420896
Baboon	58.3766	.30740

Table 2.5 PSNR/MSE values of DWT technique

Cover image	PSNR(dB)	MSE(dB)
Jet	44.76	1.4741
Baboon	44.96	1.4405

Table 2.6 Parameters analysis of LSB & DCT & DWT Methods

Features	LSB	DCT	DWT
Invisibility	Low	High	High
Payload capacity	High	Medium	Low
Robustness against image manipulation	Low	Medium	High
PSNR	Medium	High	Low
MSE	Medium	Low	High

In conclusion, published research on steganography has focused on various techniques for enhancing the hiding of secret data in cover media. However, it did not consider adding one more layer of security protect information from detection by attackers. Therefore this thesis uses more than one technique to increase security of concealment and to strengthen the protection of hidden information. While reading all these research papers, we understand that
Spatial domain techniques are very simple and easy to detect and DCT method is complex and little bit lossy but it provides higher security than LSB.

CHAPTER THREE

METHODOLOGY

CHAPTER THREE

METHODOLOGY

3.1 Overview

This chapter describes the proposed method (Secure audio Steganography using modified LSB and DCT) and explains the diagrams that clarify the proposed method. level one (first level) is a DCT based Image steganography, a gray scale image is used as a cover image with a secure data (texts) converted to long bit-stream before concealing, while level two (second level) modified LSB is used Which embed image file in audio cover. The overall process is described in figure 3.1

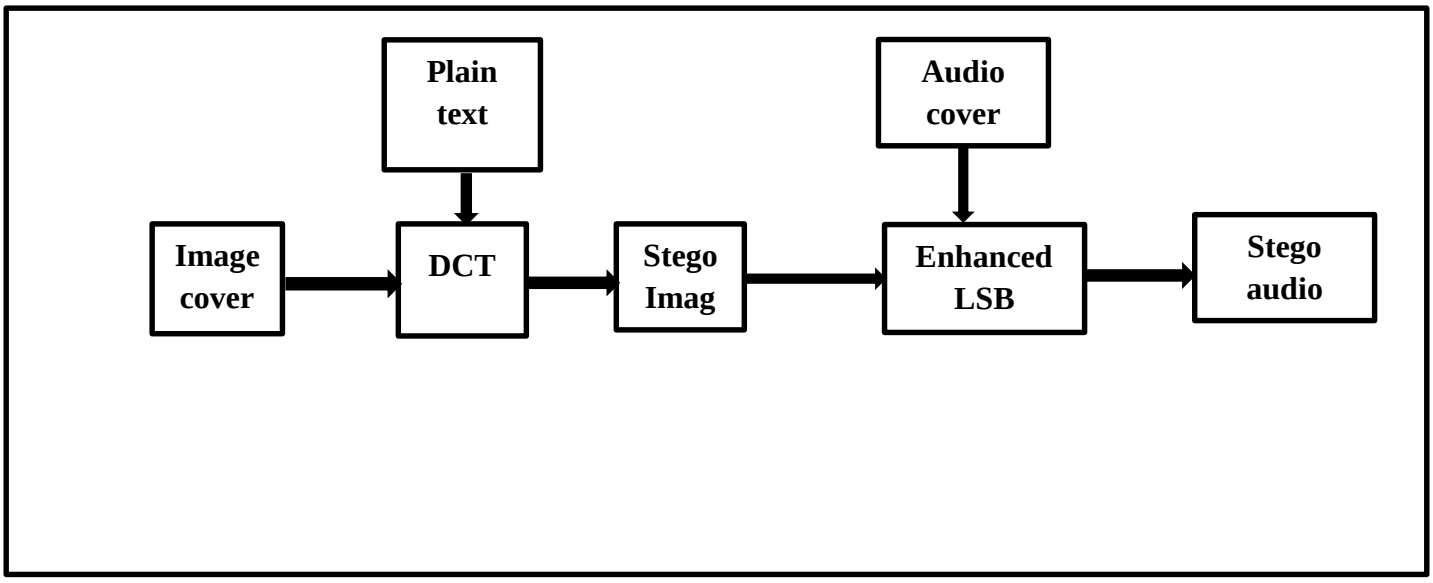


Figure: 3.1 the overall process of proposed method

3.2 Proposed Method

The proposed method is using a multilevel data hiding scheme, the proposed work involves a double Steganography (LSB of DCT and modified LSB), and in the first level combination of frequency domain by means of DCT and LSB technique of spatial domain steganography has been used to hide data. In the first stage of DCT a Two-dimensional DCT converts the image block from spatial domain to

frequency domain and then data bits are embedded (plain Text) by altering LSB of DCT coefficients. The output from level one is stego image, the stego image will be converted into binary and will work as input in level two.
Level two conceals the stego image in audio file, the stego image is concealed in this audio file using modified LSB based audio steganography and the output of this level is stego audio file.

3.2.1 Embedding process using LSB of DCT (first level)

The embedding procedures includes transformation of an image representation into a frequency representation, by grouping the pixels into 8 × 8-pixel blocks and transforming the pixel blocks into 64 DCT. DCT is used in steganography as- Image is broken into 8×8 blocks of pixels. Working from left to right, top to bottom, the DCT is applied to each block. Each block is compressed through quantization table to scale the DCT coefficients and message is embedded in DCT coefficients the details of embedding algorithm are mentioned below.

Step 1. Input the cover image of size MxN
Step 2. Input the payload (Text).
Step 3. Convert the plain text to Binary representation for later embedding.
Step 4. Divided image into non-overlapping blocks of 8 ×8 blocks, each block will do the same process individually.
Step 5. Apply DCT to each block.
Step 6. Quantize the DCT coefficient by using Quantization table.
Step 7. Apply Huffman coding for each block
Step 8. Embed the plain text to least significant bits of the quantized DCT coefficients.
Step 9. Apply Huffman decoding for each block
Step 10. Dequantize the DCT coefficient by using Dequantization tables.
Step 11. Apply IDCT to each block.
Step 12. Stego image has created.

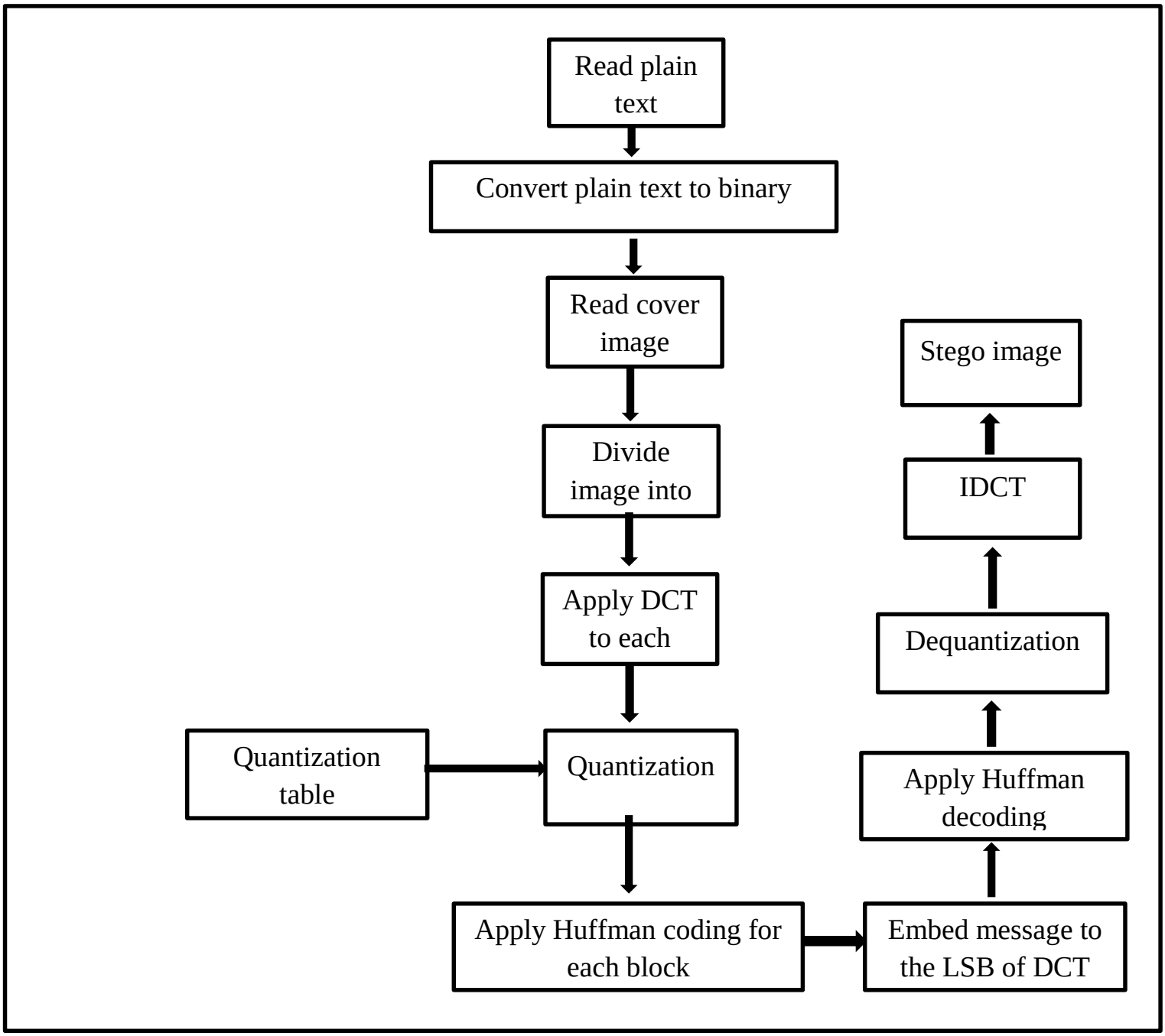


Figure: 3.2. Block diagram of LSB-DCT steganography (Embedding module)

3.2.2 Extracting process using LSB of DCT (first level)

The extracting procedure includes the transformation of spatial domain to the frequency domain utilizing a DCT algorithm then extracting step takes place and finally reverse transform from frequency domain to spatial domain is carried out. The extracting algorithm retrieves the payload from the stego image, where it is transformed into a different color scale and is split to non-overlap 8 ×8 blocks, then, transformed into the frequency domain and quantized. Finally, the payload is extracted, then the stego image is transformed back to the spatial domain and dequantizated. The details of extracting algorithm are shown below.

Step 1. Input the stego image of size MxN
Step 2. Divided the stego image into non-overlapping blocks of 8 ×8 blocks.
Step 3. Apply DCT to each block.
Step 4. Quantize the DCT coefficient by using Quantization tables.
Step5. Extract the plain text from the least significant bit of the quantized DCT coefficients in each block.
Step 6. Dequantize the DCT coefficient by using Dequantization tables.
Step 7. Apply IDCT to each block.
Step 8. Extracted plain text

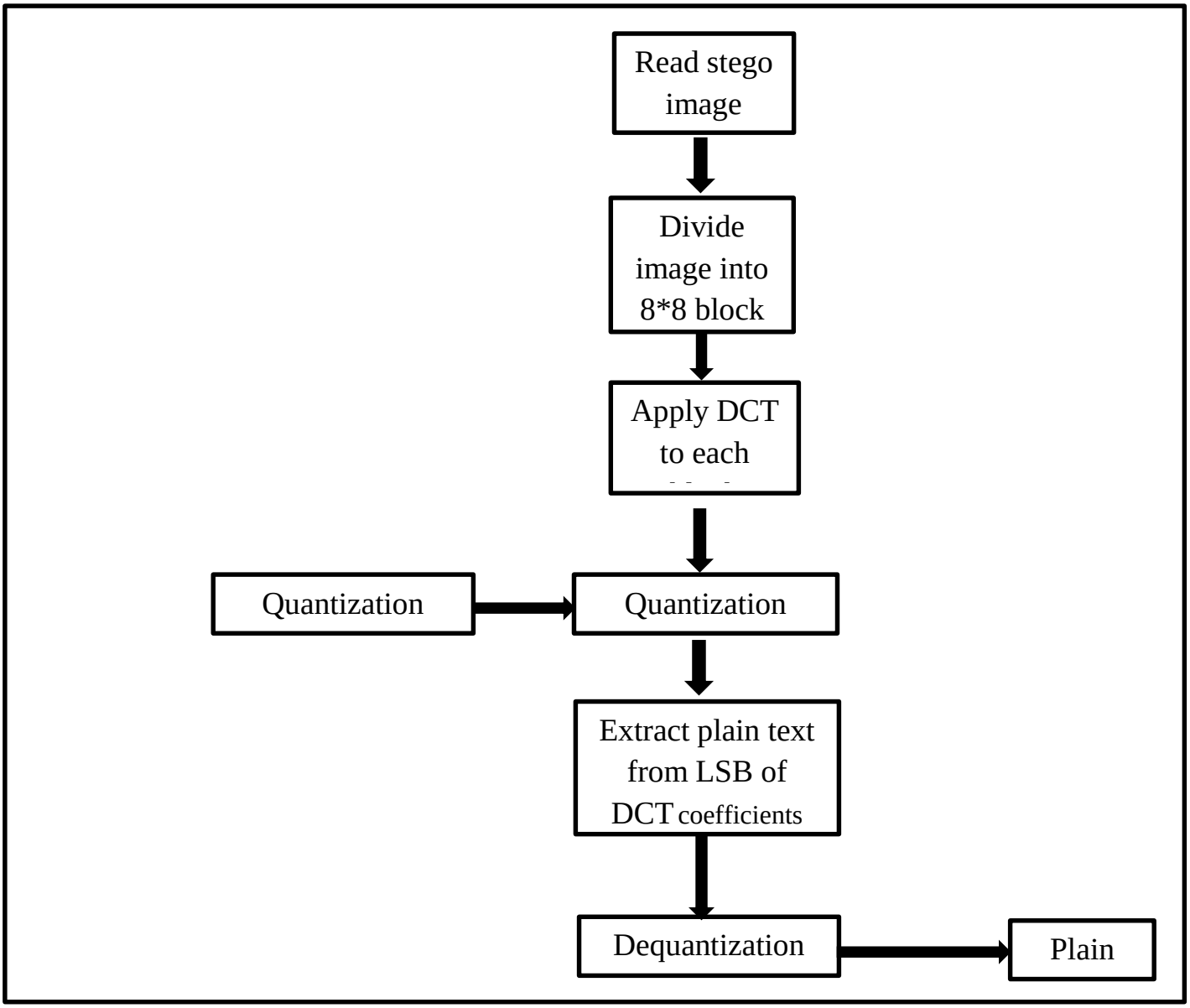


Figure: 3.3. Block diagram of LSB-DCT steganography (Extracting module)

3.2.3 Modified LSB algorithm (second level)

In this part, Stego image will be embedded into a cover audio, which is done by Changing a bit of the sample in every four samples of audio samples. The secret message is embedded randomly in audio samples based on Hseq (hopping sequence) which is an equation that generate a random position to conceal image bits in audio samples. In this embedding technique bitwise operators (bit_or and bit_and) is performed. The data is selected on the basis of bit which is to be embedded, the random position of the audio samples is customized or kept unchanged. The advantage of this technique is that this increases the capability of the cover audio by 8 times greater and provides strong encryption. Modified LSB is described below.

Preprocessing

Step 1. First read image, convert it into one dimensional array.
Step 2. Convert one dimensional. Array to bit array.
Step 3. Create bits array to hold bits to be embedded.
Step 5. Create hopping sequence (random position to conceal data)
Step 6. Read sound file.
Step 7. Convert sound array to bit array.

Embedding

Step 1. Obtain embed location by utilizing hopping sequence.
Step 2. Take the bit from bits' array (image bits).
Step 3. If bit value is 1, use bit_or function.
Step 4. Else use bit_and function.
Step 5. Modify audio samples

Extracting

Step 1. Create Array for holding bits which are embedded to sound.
Step 2. Take embed location from hopping sequence.
Step 3. Extract bits of image embedded to the audio.
Step 4. Stores all bits extracted from the sound into a different array.
Step 5. Image construction.
Step 6. Write extracted image.

Table 3.1 example for data embedding using bitwise (bit-or and bit-and) at random positions

Secret message	Value of audio before embedding	Value of audio after embedding	Action if message bit is 1	Action if message bit is 0
0	0110110100101101	0110110100101101		No Change
1	1001011010010111	1001011010010111		
0	1101000111001110	1101000111001110		
1	0110111010111111	0110111010111111		
0	1110100101110111	1110100101111111	Flip LSB	
0	0110110100101101	0110110100101101		
1	0110111010111111	0110111010111111		
1	1001011010010111	1001011010010111		

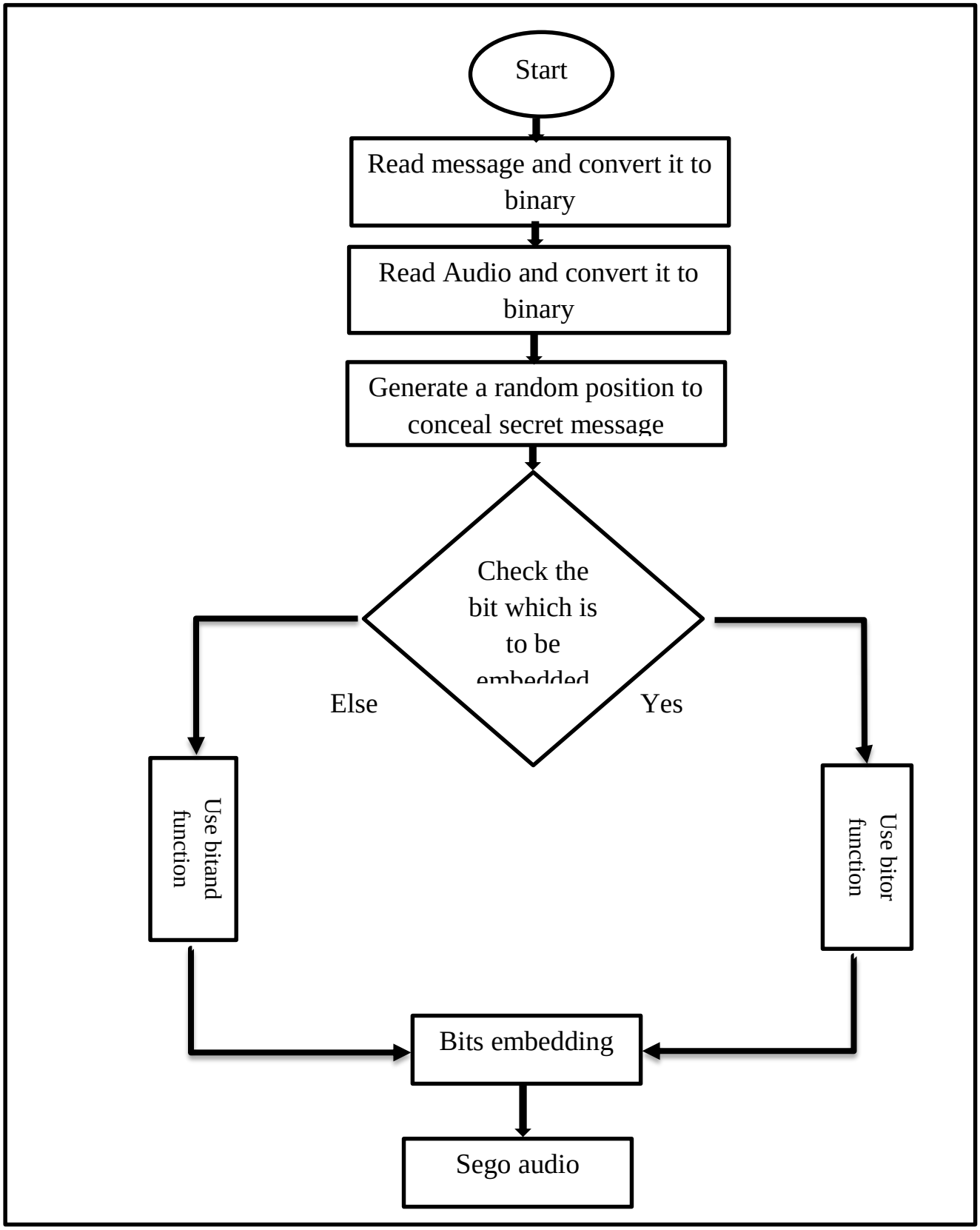


Figure: 3.4 Data embedding in audio sample

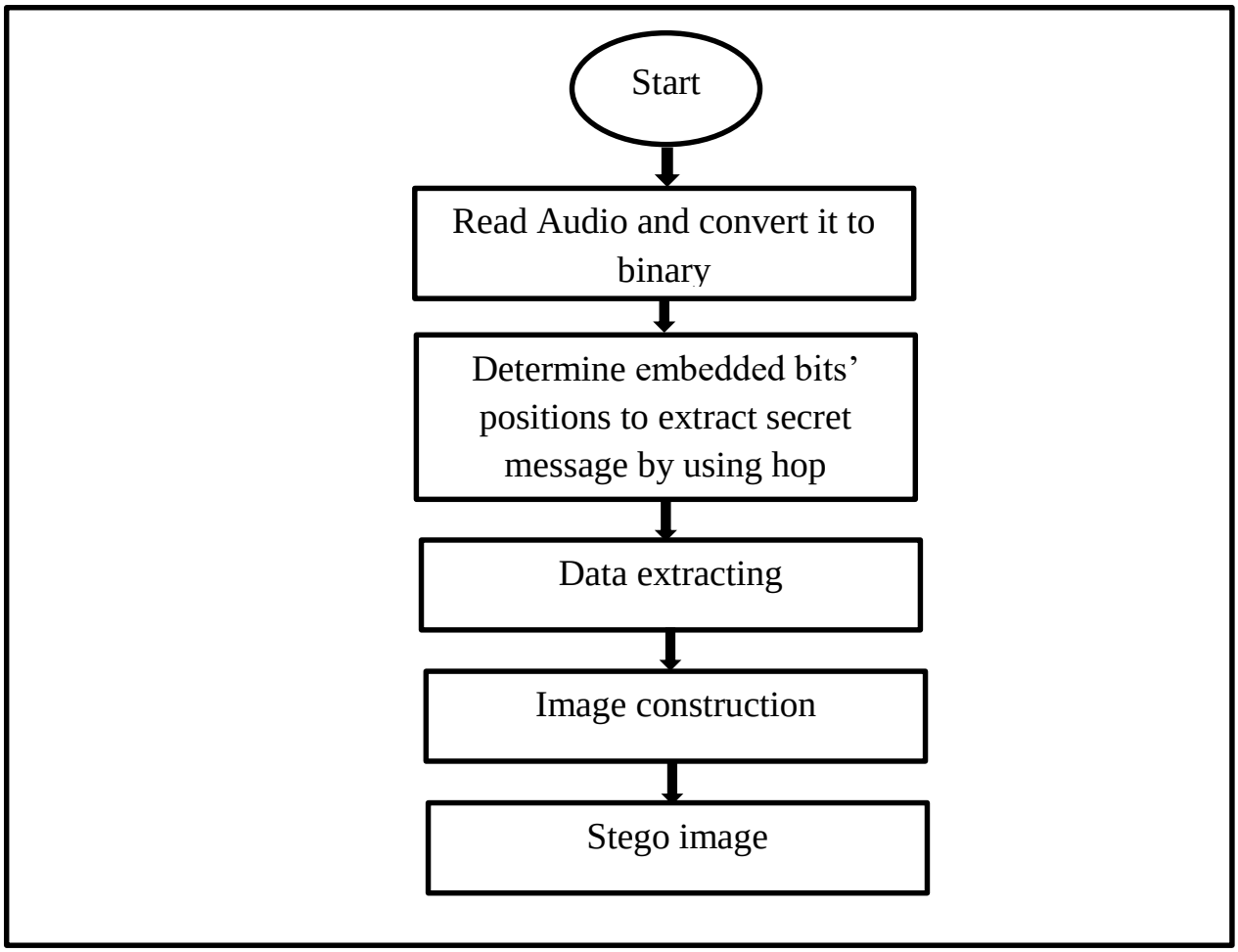


Figure: 3.5 Data extracting in audio sample

CHAPTER FOUR

IMPLEMENTAIONS AND RESULTS

CHAPTER FOUR

IMPLEMETAIONS AND RESULTS

4.1 Introduction

The proposed method used an efficient and secure schema to hide secret text in image and secret image in audio respectively. This section illustrates the results of applying the proposed method to hide different messages with different sizes on image file and hide different secret images on different audios file and measure the accuracy of the resulted outputs using Peak Signal to Noise Ratio and Mean Squared Error measurements.

4.2 Used plain texts

There are three different plain text with different sizes have been used to be embedded in image file in the first level of steganography, Figure 4.1, Figure 4.2 and Figure 4.3 below show example of the secret messages used.

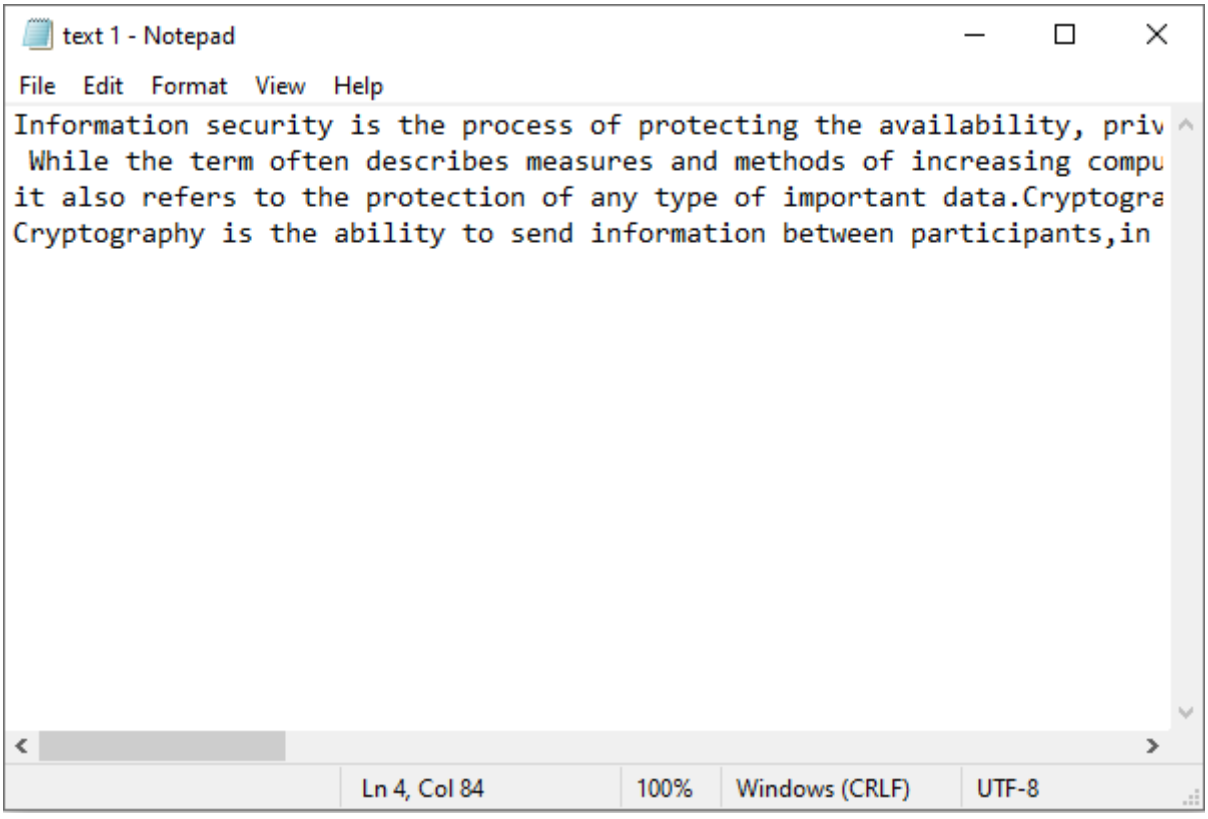


Figure 4.1: plain text 1

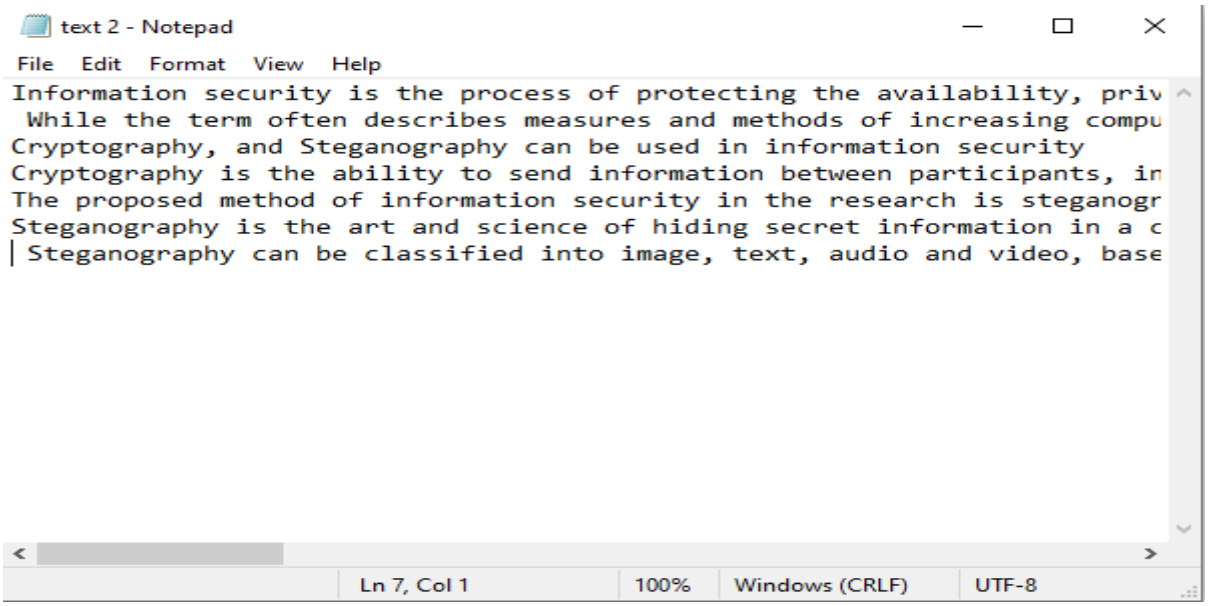


Figure 4.2: plain text 2

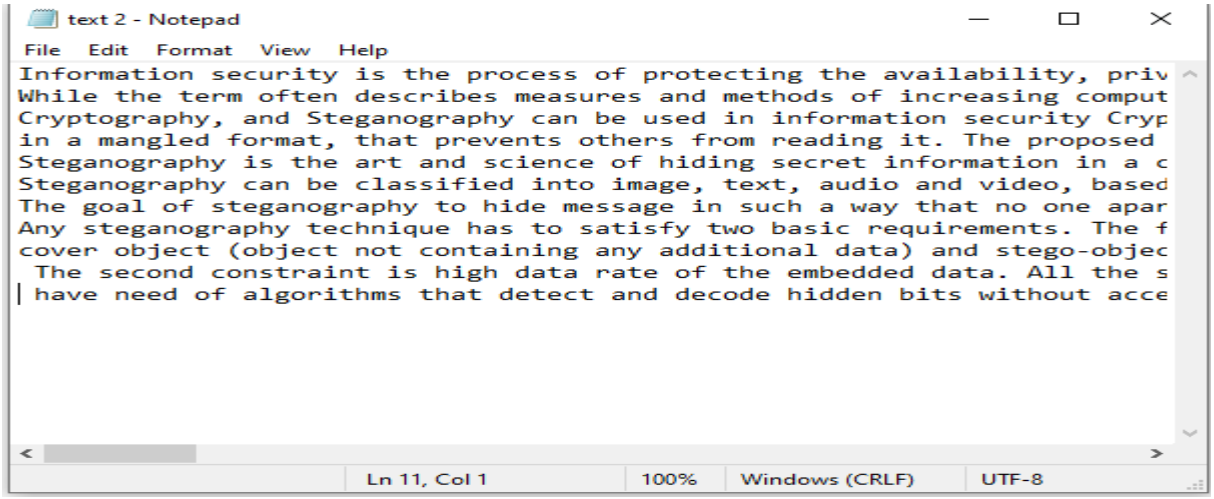


Figure 4.3: plain text 3

The sizes of the plain texts used is shown in table 4.1 below

Table 4.1: plain texts File Sizes

message file	message file size (in bytes)
Plain text 1	452
Plain text 2	962
Plain text 3	1,640

4.3 Experimental Results

4.3.1 First level (DCT)

After the first level (level one DCT) is applied to the messages shown on table 4.1 above, the output is three images files (Lena 1.BMP, Lena 2.BMP, and Lena 3.BMP) that concealing three different plain texts. The cover image file used is lena.bmp image file and it concealed different three secret message at time, the size of the image file was 263,222 bytes. The following Figures shows the image file with three different plain texts (before and after embedding with histogram).

After the first level (level one - DCT) is applied to the above plain texts the output is three stego images. Each image concealing one of the secret messages. The first cover image is the Lena 1 and is concealing (plain text 1) as secret data, Figure 4.5 shows the Lena 1stego image, the size of the stego image is 263,222 bytes. The second cover image is Lena 2 image and is concealing (plain text 2) as secret data, the size of the stego image is 263,222 bytes Figure 4.9 shows the Lena 2_stego image, Finally the third stego image is the Lena 3, in this image the maximum capacity of the secret message is embedded (plaint text 3) the output is shown in Figure 4.13 and its size is 263,222 bytes.

Figure 4.4: before embedding plain text 1

Figure 4.5: After embedding plain text 1

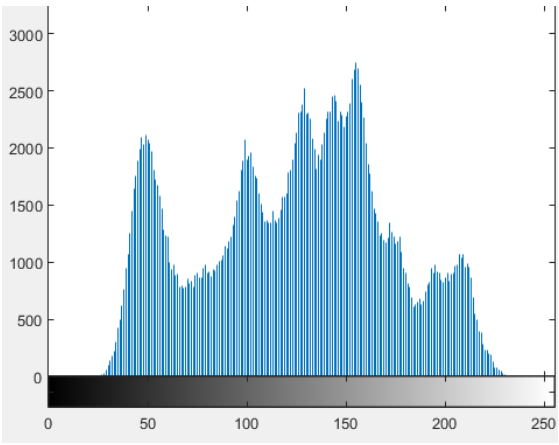


Figure 4.6: before embedding plaint text 1(histogram)

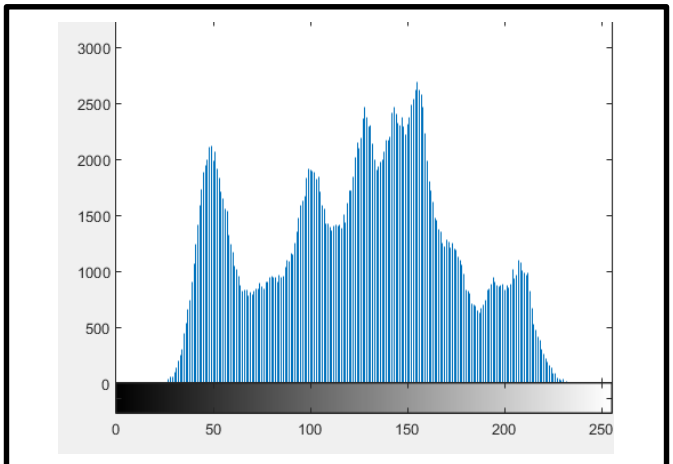


figure 4.7 after embedding plain text 1 (histogram)

Figure 4.8: before embedding plain text 2

Figure 4.9: After embedding plain text 2

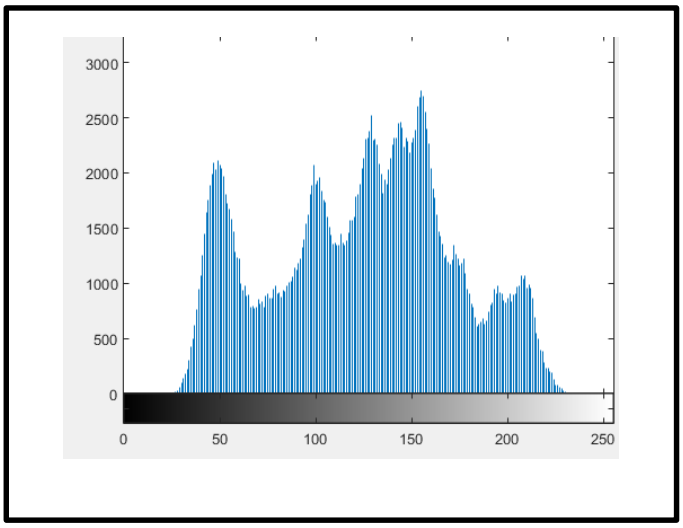
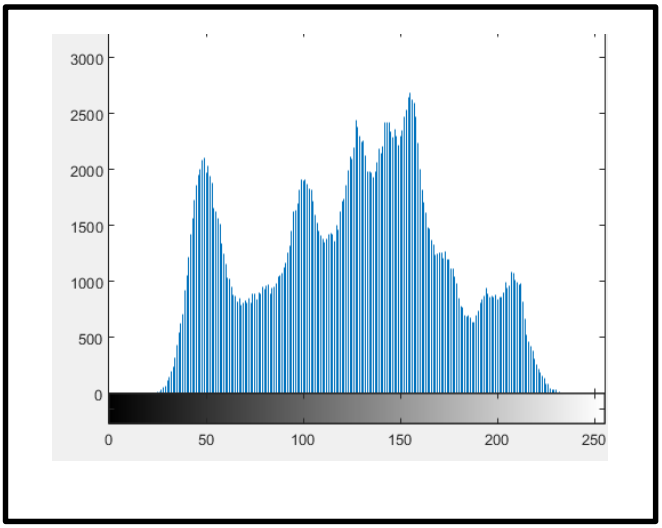


Figure 4.10: before embedding plain text 2(histogram)



4.11: after embedding plain text 2(histogram)

Figure 4.12: before embedding plain text 3

Figure 4.13 after embedding plain text 3

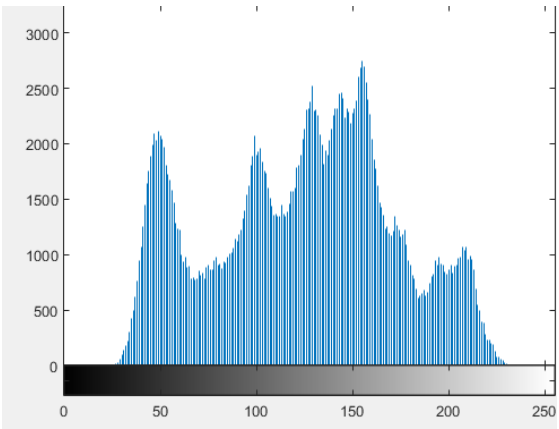


Figure 4.14: before embedding plain text 3(histogram)

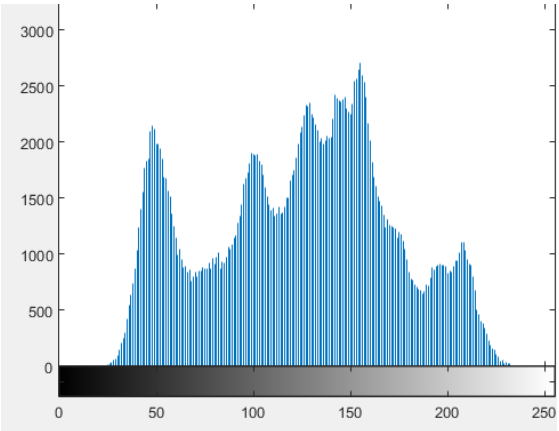


figure 4.15 after embedding plain text 3 (histogram)

Table 4.2 the experiment results of first level outputs and contains the PSNR

and MSE values of stego images above. Figure 4.16 is a Diagram showing its PSNR values

Table 4.2 shows the experiment results of Lena _stego images

Secret message	Secret message size (in bytes)	Cover image	Cover image size (bytes)	PSNR	MSE
Message 1	542	Lena.bmp	263,222	43.562	2.86
Message 2	962	Lena.bmp	263,222	42.939	3.30
Message 3	1,460	Lena.bmp	263,222	41.856	4.24

4.3.2 Second level (modified LSB)

In the second level (level two modified LSB based audio steganography) two wav files with different sizes have been used as a cover. The first audio file is test 1.wav with 187 sec of audio length and test2.wav with 368 sec of audio length used as shown in figure 4.19 and 4.20, the secret data to be embedded in this cover audio files are the stego images (Lena 2 and Lena 3) which were the outputs of the upper level. Firstly, the stego image (Lena 2) in figure 4.21 is used as secret data and is concealed in the two cover audio files respectively. Secondly the stego image (Lena 3) in figure 4.22 is used as secret data and is concealed in the two cover audio files respectively.

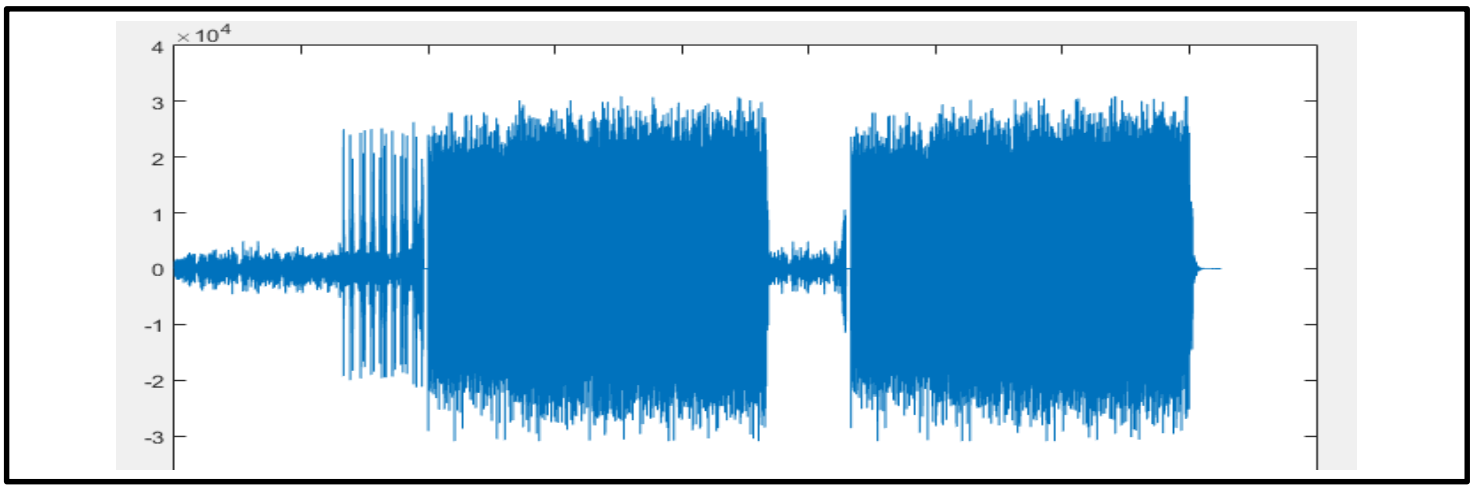


Figure 4.16 test 1 original audio file

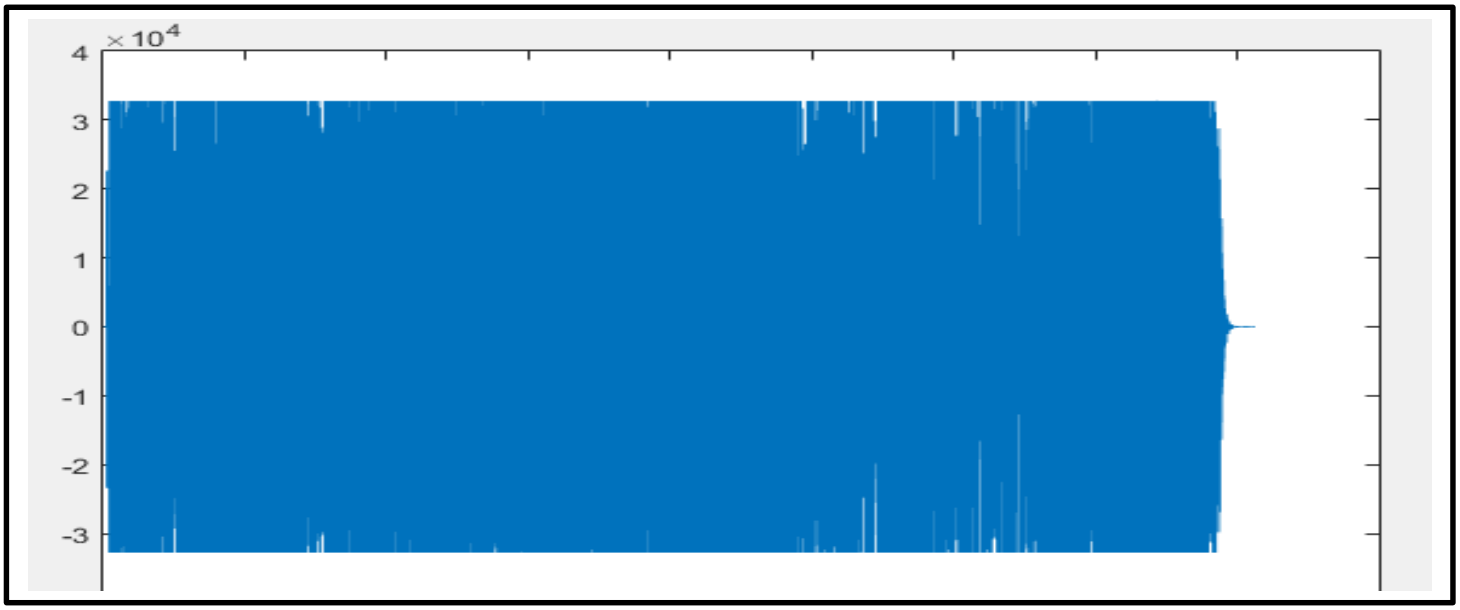


Figure 4.17 test 2 original audio file

Figure 4.18 Lena 2 as secret data

figure 4.19 Lena 3 as secret data

After the second level (level two modified LSB) is applied to the stego images shown on figures 4.18 and 4.19 above, the output was four audio files. Each audio

file concealing one of the two secret images at time. The first cover audio file is the test1 wave file and it concealed the first Lena 1 and the second Lena 2 secret messages, the size of the audio file was 16,501,158 bytes. Figure 4.20 shows the original file wave (test1). Figure 4.21 shows stego wav file with Lena 2 image file embedded on it and figure 4.22 shows Lena 2 extracted image.

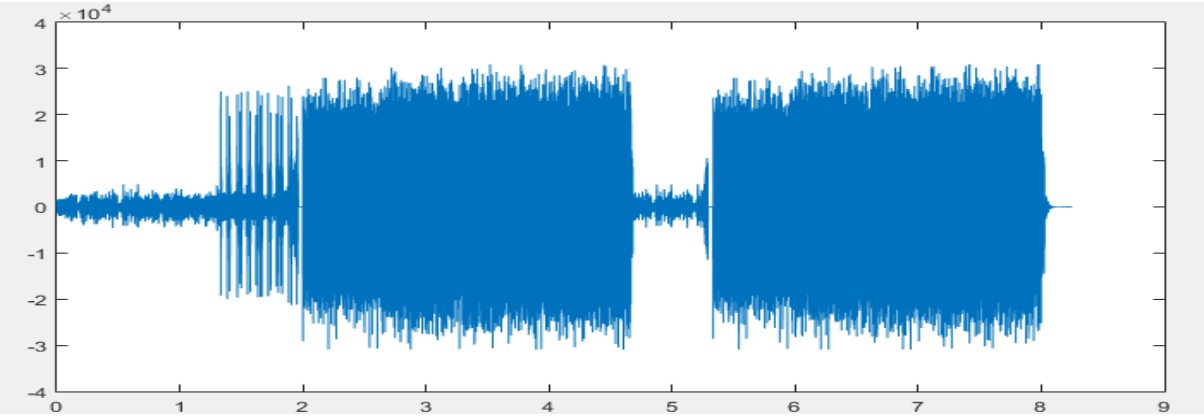


Figure 4.20 the original test 1 audio file before embedding

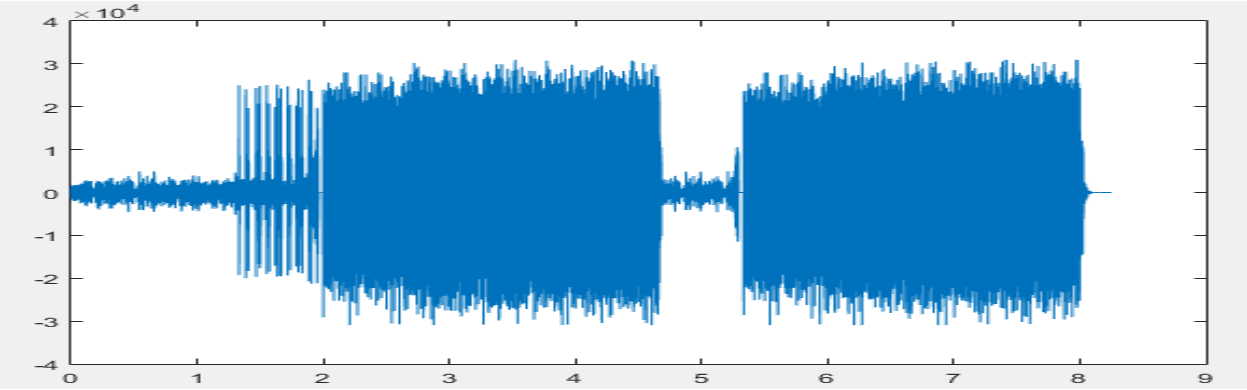


Figure 4.21 test 1 stego audio file

Figure 4.22 Lena 2 extracted image

Figure 4.23 shows the original file wave (test1). Figure 4.24 shows stego wav file with Lena 3 image file embedded on it and figure 4.25 shows Lena 3 extracted image.

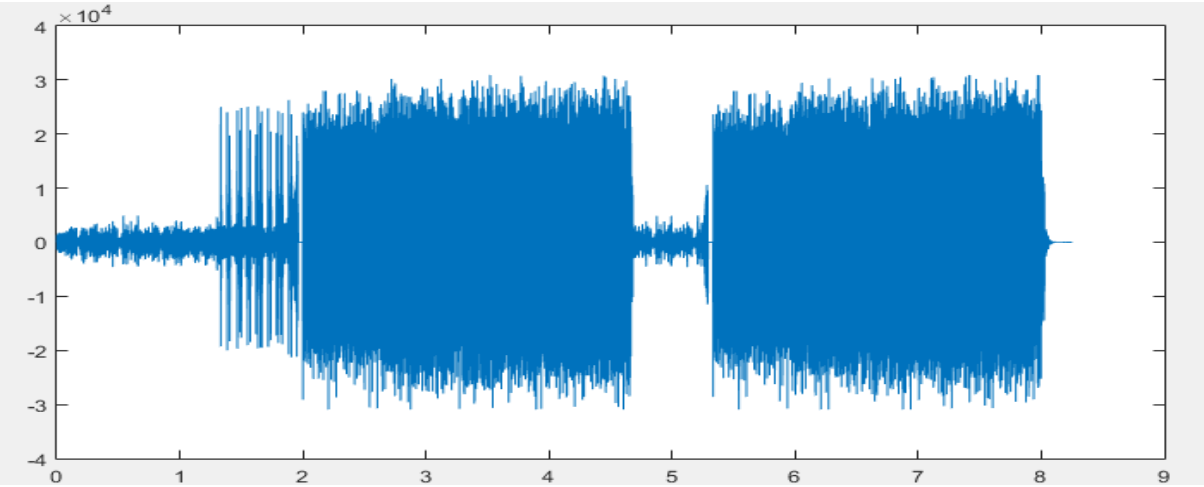


Figure 4.23 the original test 1 audio file before embedding

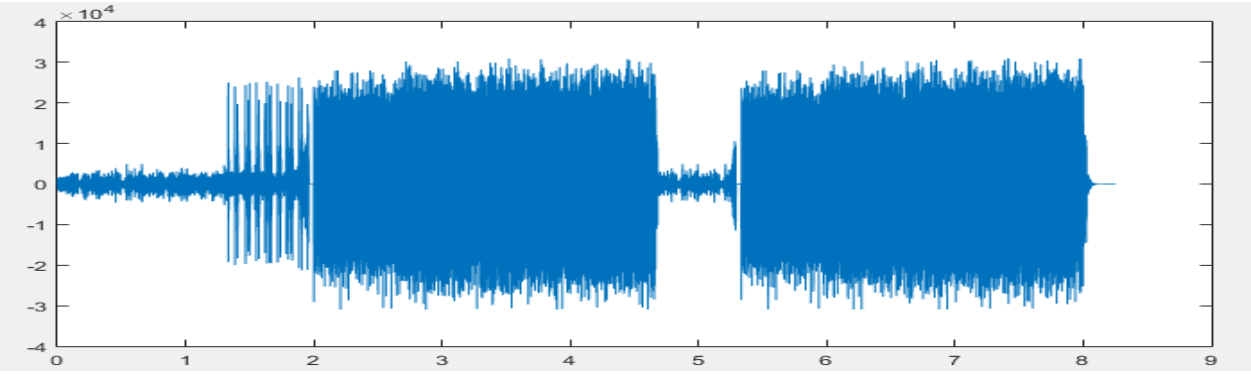


Figure 4.24 test 1 stego audio file

Figure 4.25 Lena 3 extracted image

Figure 4.26 shows the original file wave (test2). Figure 4.27 shows stego wav file with Lena 2 image file embedded on it and figure 4.28 shows Lena 2 extracted image.

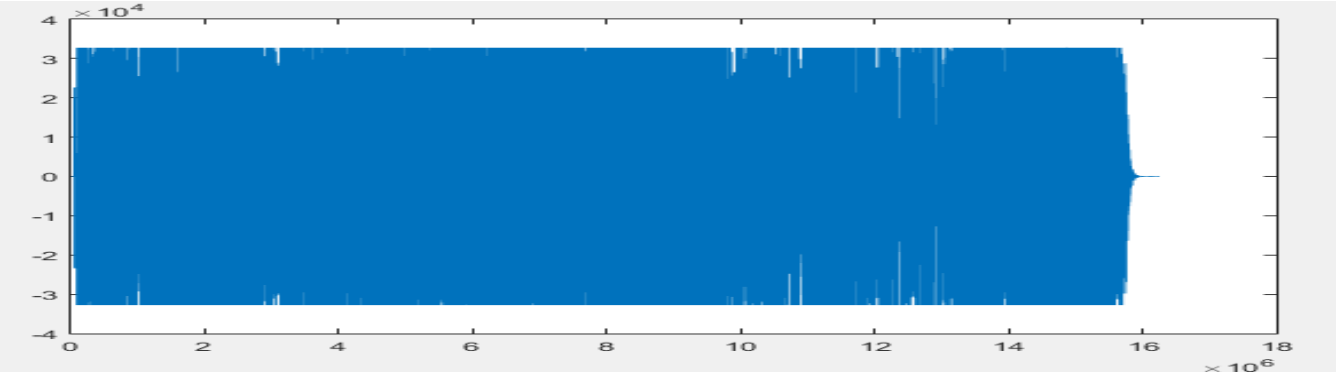


Figure 4.26 test 2 original audio file before embedding

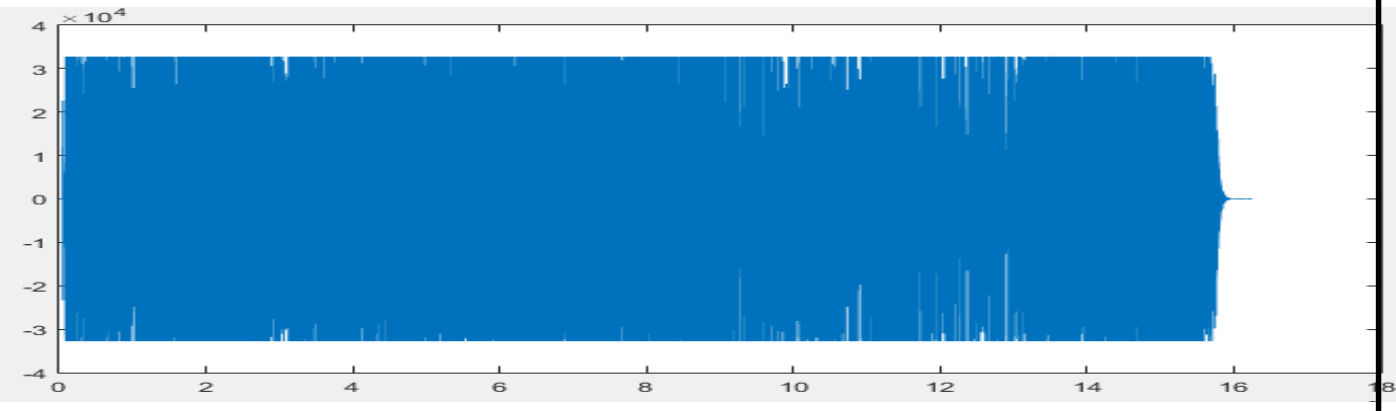


Figure 4.27 test 2 stego audio file

Figure 4.28 Lena 2 extracted image

Figure 4.29 shows the original file wave (test2). Figure 4.30 shows test 2 stego wav file with Lena 3 image file embedded on it and figure 4.31 shows Lena 3 extracted image.

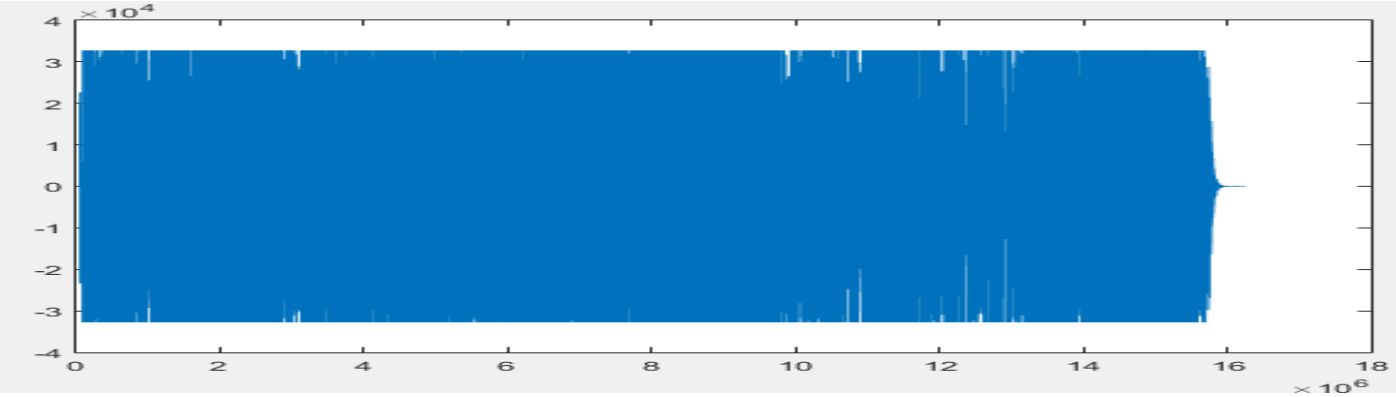


Figure 4.29 test 2 original audio file before embedding

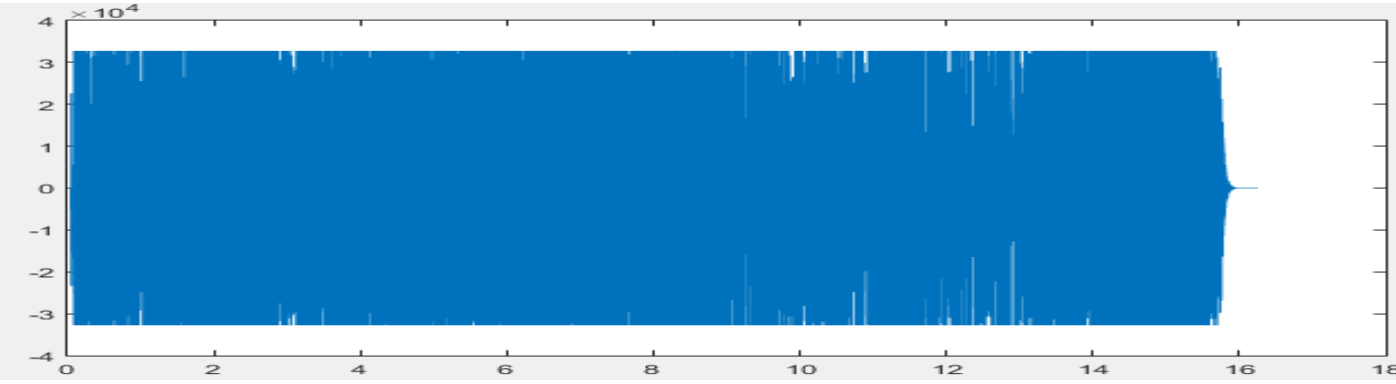


Figure 4.30 test 2 stego audio file

Figure 4.31 Lena 3 extracted image

Table 4.3 shows the experimental results of audio files (test 1, test 2) and contains the PSNR and MSE values of stego audios above

Table 4.3 the experiment results of cover audio files

Audio file cover	Secret message	Size of secret message in bytes	PSNR	MSE
Test 1	Lena 2	263,222	35.208691	9.0684
	Lena 3	263,222	35.207328	9.0712
Test 2	Lena 2	263,222	38.374412	4.5923
	Lena 3	263,222	38.375424	4.5912

4.4 Discussion

This section is dedicated to evaluating the performance of Secure Audio Steganography using modified LSB and DCT schema in terms of hiding capacity and the fidelity of the cover and reconstructed secret files. The files that are used in the experiment are Lena image with 512 x 512 dimension, 263,222 bytes of size and test 1, test 2 audio files the selected audio files are with the following specifications: WAVE (.wav) mono format with 16 bits per sample and a sampling rate of 44100 Hz with different sizes. The test is conducted to inspect the impact of the DCT and modified LSB on the performance.

Two metrics are utilized to assess the performance of the Secure Audio Steganography using modified LSB and DCT schema which is Peak Signal to Noise Ratio (PSNR) and Mean error square respectively(MSE).
After the completion of all experiments, the results showed that, as the file size of the secret embedded message increases, there is variation in the PSNR values. The PSNR values decreases as the file size increases. This simply means, as more files are embedded, the quality of the cover will be low and that will compromise the security of the system.

CHAPTER FIVE

CONCLUSION AND FUTURE WORKS

CHAPTER FIVE

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

The main objective of proposed method is increasing the security of these roads and to protect information from detection by attackers. The more complex the method of concealment and the more steps followed, the more cynical the method and harder for the attackers to break it and access to hidden information. This method uses more than one technique to increase security of concealment and to strengthen the protection of hidden information.

The proposed method is two levels of steganography, level one is applied by using discrete cosine transformation (DCT), and it has been applied to enhance the performance of basic spatial domain algorithms and increase algorithms strength because the basic existing systems has considerably low robustness against attacks, in this level gray scale image (with bmp extensions) have been used as a cover image and it's conceal a secure data (plain texts) files to generate a stego images.

Level two has been applied using (modified LSB) algorithm to add another level of security to proposed method, in this level audio files (with wav extensions) has been used as a cover and embed (outputs from level one) as a secure data and generate audio stego files.

Measuring the performance of proposed algorithm has been applied using many experiments and calculate values of each experiment, the first value is Peak signal to noise ratio (PSNR) , this ratio is used as a quality measurement between two Files, the second measurement value is Mean Squared Error is the average squared difference between original files

And a modified files (stego files).

Finally the proposed method has potential benefits, as it may enhance the confidentiality of the secret information by using two level of steganography in one the system and add more complexity to the steganography process through applying it in two levels.

5.2 Future Work

- Adding encryption algorithm to first level to encrypt the plain text before encoding to increase security.
- Adding compression algorithm to compress the stego images (level one outputs) after level one and before level two to enhance the performance of the proposed method.

- Increase the System functionality to hide all other data types like video and audio not only text data and images.
- Trying to enhance the performance of algorithms in both levels to increase the system capacity.

REFRENCES

Aldabagh, G. M. T. K. (2020). "Proposed algorithm using image in multi level text steganography." Journal of Southwest Jiaotong University **55**(1).

Ali, A. H., et al. (2017). "Enhancing the hiding capacity of audio steganography based on block mapping." Journal of Theoretical & Applied Information Technology **95**(7): 1441-1448.

Chandran, S. and K. Bhattacharyya (2015). Performance analysis of LSB, DCT, and DWT for digital watermarking application using steganography. IEEE International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO).

Divya, S. and M. R. M. Reddy (2012). "Hiding text in audio using multiple LSB steganography and provide security using cryptography." International journal of scientific & technology research **1**(6): 68-70.

Frączek, W., et al. (2012). "Multi-level steganography: Improving hidden communication in networks." Journal of Universal Computer Science (J. UCS) **18**(14): 1967-1986.

Hariri, M., et al. (2011). "An introduction to steganography methods." World Applied Programming **1**(3): 191-195.

Kaur, N. and S. Behal (2014). "Audio Steganography Techniques-A Survey." Int Journal of Engineering Research and Applications ISSN: 2248-9622.

Kaur, N. and S. Behal (2014). "A Survey on various types of Steganography and Analysis of Hiding Techniques." International journal of engineering trends and technology **11**(8): 388-392.

Mandal, P. C. (2012). "Modern Steganographic technique: A survey." International Journal of Computer Science & Engineering Technology (IJCSET) **3**(9): 444-448.

Padmashree, G. and P. Venugopala (2012). "Audio Stegnography and Cryptography: Using LSB algorithm at 4th and 5th LSB layers." International Journal of Engineering and Innovative Technology **2**(4).

Rajput, G. and R. Chavan (2018). "Improved LSB based Image Steganography using Run Length encoding and Random Insertion technique for Colour Images." World Scientific News **112**: 180-192.

Shrivastava, S. and M. K. Patidar "A Modified Approch Audio Stagnography Based On Technique LSB Coding." International Journal of Engineering and Applied Sciences **2**(5).

Shrivastava, S. and M. K. Patidar "A Modified Approch Audio Stagnography BasedOn Technique LSB Coding." International Journal of Engineering and Applied Sciences **2**(5): 257924.Singla, D. and R. Syal (2012). "Data security using LSB & DCT steganography in images." Int. J. Of Computation Engeneering Research **2**(2): 359-364.