REPUBLIC OF IRAQ
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
AL-QADISIYAH UNIVERSITY
COLLEGE OF COMPUTER SCIENCES AND IT
MULTIMEDIA DEPARTMENT

STEGANOGRAPHY DATA IN IMAGE BY USING LSB ALGORITHM

2019-2018

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة القادسية
كلية علوم الحاسبات وتكنولوجيا المعلومات
قسم الوسائط المتعددة

اخفاء بيانات مشفرة في صورة بأستخدام خوارزمية الأخفاء في البت الأخير

2019-2018

((وَإِنِّي مُرْسِلَةٌ إِلَيْهِمْ بِهَدِيَّةٍ فَنَاظِرَةٌ بِمَ يَرْجِعُ الْمُرْسَلُونَ (35) فَلَمَّا جَاءَ سُلَيْمَانَ قَالَ أَتُمِدُّونَنِ بِمَالٍ فَمَا آتَانِيَ اللَّهُ خَيْرٌ مِمَّا آتَاكُمْ بَلْ أَنْتُمْ بِهَدِيَّتِكُمْ تَفْرَحُونَ (36) ارْجِعْ إِلَيْهِمْ فَلَنَأْتِيَنَّهُمْ بِجُنُودٍ لَا قِبَلَ لَهُمْ بِهَا وَلَنُخْرِجَنَّهُمْ مِنْهَا أَذِلَّةً وَهُمْ صَاغِرُونَ (37)))

سورة النمل

REPUBLIC OF IRAQ
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
AL-QADISIYAH UNIVERSITY
COLLEGE OF COMPUTER SCIENCES AND IT
MULTIMEDIA DEPARTMENT

GRADUATION PROJECT REPORT

STEGANOGRAPHY DATA IN IMAGE BY USING LSB ALGORITHM

by

Mustafa Hussein Mohammed **Ahmed Hussein Abd Zaid**

Faid Raqib Mohammed Rasheid **Kawther Hassan Hafaz**

ADVISER
ASSISTANT LECTURE. ALI HAKEM

AL-QADISIYAH, 2019

جمهورية العراق
وزارة التعليم العالي والبحث العلمي
جامعة القادسية
كلية علوم الحاسبات وتكنولوجيا المعلومات
قسم الوسائط المتعددة

تقرير مشروع تخرج البكلوريوس

اخفاء بيانات مشفرة في صورة بأستخدام خوارزمية الأخفاء في البت الأخير

اعداد

مصطفى حسين محمد
فيد رقيب محمد رشيد
احمد حسين عبد زيد
كوثر حسن حفاز

اشراف
م.م. علي حاكم جبر

القادسية 2019

REPUBLIC OF IRAQ

AL-QADISIYAH UNIVERSITY

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

COLLEGE OF COMPUTER SCIENCES AND IT

MULTIMEDIA DEPARTMENT

STEGANOGRAPHY DATA IN IMAGE BY USING LSB ALGORITHM

A project submitted by Mustafa Hussein Mohammed,Ahmed Hussein Abd Zaid, Faid Raqib Mohammed Rasheid, and Kawther Hassan Hafaz , in partial fulfilment of the requirements for the degree of BACHELOR'S OF COMPUTER SCIENCE, is approved by the committee in Department of Multimedia.

Project Adviser

Assistant Lecturer. Ali Hakem

Al-Qadisiyah University

Approved By the Examining Committee

ACKNOWLEDGEMENTS

First, I would like to thank God for giving me strength and patience to work on this thesis. It would not have been possible without God guidance and support.

I would like to thank my supervisor, Assist. Lecture. Ali Hakem, for the opportunity to work under his guidance, for the suggestions and advices on how to perform a master thesis; without his guidance and present support, this master thesis would not have been possible. Under his guidance, I could learn on working methods and improve my performance, which allows me to perform this thesis.

Also, I want to thank all my family fhather ,mother, brothers, and sisters,

who were a great source of encouragement and motivation.

April 2019

Authors

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

ABSTRACT

STEGANOGRAPHY DATA IN IMAGE BY USING LSB ALGORITHM

Ahmed Hussein Abdzaid **Faid Raqib Mohammed Rashid**
Kawthar Hassan Haffaz **Mustafa Hussein Mohammed**

ADVISER
ASSISTANT LECTURE. ALI HAKEM

B.Sc. Project Report
DEPARTMENT OF MULTIMEDIA

The need to hide messages and to secretly or efficiently pass information has been around since ancient times. Today digital media; including pictures, sound files, movies, etc.; provides a rich environment to hide more information than meets the eye of a casual observer by exploiting unused or redundant data bits. This information can be used for clandestine messages to secret agents but can just as easily be used as a compression method to send additional messages within a single file, thereby reducing the amount of data to be sent. For this project we explore two methods for encoding/storing a message within the RGB pixels of a cover image without visual distortion. The methods a message is broken down to individual components, converted into 8-bit binary values, encrypted using a simple symmetric Exclusive OR (XOR) encryption key, and then encoded in the

cover image by changing the least significant bit of the pixel value. The method implemented sequentially stores the message starting with the top left pixel and then encodes the message from top to bottom and left to right..

Keywords: Information Scurity , Data Steganography , Least Segnificant Bit Algorithm

اخفاء البيانات المشفرة بأستخدام خوارزمية الأخفاء في البت الأخير

مصطفى حسين محمد احمد حسين عبد زيد
فيد رقيب محمد رشيد كوثر حسن حفاز

المشرف
م.م.علي حاكم جبر

تقرير مشروع تخرج البكلوريوس
قسم الوسائط المتعددة

نحتاج اخفاء الرسائل المهمة وارسالها عبر الوسائط الغير امنة . تعتبر اليوم الوسائط المتعددة مثل الصور والفديو والصوت بيئة مهمة لأخفاء المعلومات في داخل بياناتها . هذه المعلومات يمكن ان تعتبر مهمة لذلك يمكن مقارنتها بمعلومات اخرى لغرض كشفها لذلك من الضرورة اخفائها داخل ملفات امنة وارسالها . في هذا التقرير استخدمنا طريقتين الاولى تشفير الرسالة والثانية اخفائها داخل ملف صور ملونة . لتشفير الرسالة قبل اخفائها نحولها الى 8 بت ثم نشفرها بأستخدام طريقة التشفير المتناظر وبأستخدام الدالة (xor) , ثم نقوم بأخفاء تلك الرسالة المشفرة داخل بكسلات الملف الحامل للرسالة . في هذه الطريقة قمنا بأختيار البكسلات تتابعياً من الاعلى الى الاسفل ومن اليمين الى اليسار .

كلمات مفتاحية : امنية المعلومات , اخفاء البيانات , خوارزمية اخفاء البيانات في البت الأخير

جامعة القادسية
كلية علوم الحاسوب وتكنلوجيا المعلومات / قسم الوسائط المتعددة

CHAPTER 1

INTRODUCTIONS

1.1 Introduction

In computer Scinces the steganography is the science and art of concealing writing or messages. For as long as there have been secrets, there has been a need for people to hide their secrets. Steganography is often confused with cryptography, though they are actually two separate fields. Cryptography is the science of making a message unreadable without a password or key. Steganography however, deals with hiding the fact that there even is a secret message at all. Steganography is a very important field in today's world, due to the lack of privacy in the modern era. Steganography allows people to communicate without the scrutiny of others, because nobody will even know there is a secret message encoded.

Steganography has a long history going back to ancient times. Before the invention of computers, people used many different mediums in order to hide a message. One such medium used to conceal messages was in other people. The Ancient Greeks used to shave the head of a messenger and tattoo a message on his head, and when his hair grew back, he was dispatched. Another method using people was to write a message on silk, and wrapping it in wax and having a messenger swallow it. The message would then be retrieved at a later date. Both of these methods assumed that the message was not time sensitive. Another way messages were concealed was inside

ordinary items, such as in the bunghole of a beer barrel. The Spartans were warned of an attack by Xerxes, when a messenger delivered a blank folding tablet with a message written under the wax [1, 2].

World War II was a big time for steganography, and many steganographic techniques were created and employed. The Nazis invented microdots, which were essentially microfilms the size of a period on a piece of paper. Each microdot could contain pages of text or pictures when viewed under high magnification.

1.2 Objective of the Project

In this work, we design system for hid data (text or image) within caver image in order to transmit secretly data or efficiently information in unsecure channel.

1.3 Motivation

In the modern age, steganography has evolved to deal with the advent of digital media. With the prevalence of digital cameras, and the easy access that comes with the internet, there is a plethora

of media in which to hide information. Steganography has become very popular in the last few decades. One reason for this, is that many governments do not allow any cryptography in their digital media, and so in order to send a private message, the secretive method of steganography is required. It is also becoming more popular due to international safety reasons. After September 11th, many people were concerned that terrorist organizations were sending secret messages over the internet and through videos. Whether a person is trying to send or detect hidden messages, steganography is now a very important field of study [2].

CHAPTER 2

STEGANOGRAPHY ALGORITHM

2.1 Component of Steganography System

The steganography hides different types of data within a cover file. The resulting stego file also contains hidden information, although it is virtually identical to the cover file. Steganography exploit human perception; human senses are not trained to look for files that have information hidden inside of them, although there are programs available that can do what is called Steganalysis (Detecting use of Steganography).Fig.2.1 [1] shows the block diagram of a secure steganographic system. Input messages can be images, texts, video, etc. The components of steganographic system are:

- **Secret Message**: The secret message or information to hide.
- **Cover File/ Digital Medium**: The data or medium which concealed the secret message.
- **stego file**: A modified version of cover that contains the secret message.
- **key**: Additional secret data that is needed for the embedding and extracting processes and must be known to both, the sender and the recipient
- **Steganographic Method**: A steganographic function that takes cover, secret message and key as parameters and produces stego as output.
- **Inverse of Steganographic Method**: A steganographic function that has stego and key as parameters and produces secret message as output. This is the inverse of method used in embeding process in the sense that the result of the extracting process is identical to the input of the embedding process.

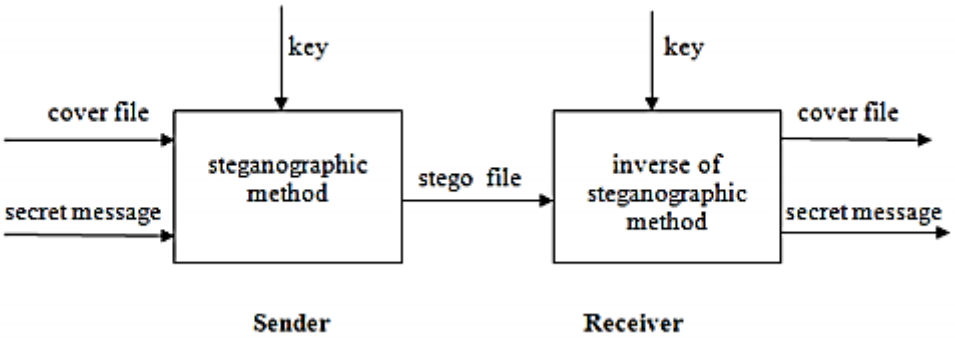

Figure 2.1. A generic Steganography System

2.2 File Formats Used as Carrier in Steganography

I. The four main categories of file formats [4] that can be used as carrier in steganography are: I. Text II. Images III. Audio/ Video IV. Protocol

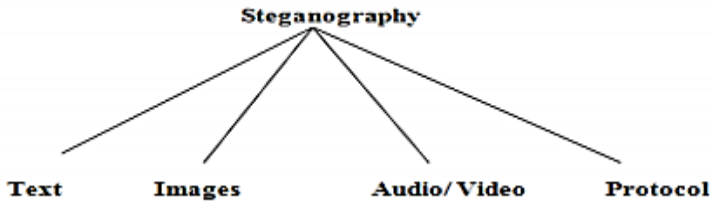

Figure 2.2: Different carriers in steganography [4]

II. **Text steganography:** Hiding information in text is the most important method of steganography. The method was to hide a secret message in every nth letter of every word of a text message. After booming of Internet and different type of digital file formats it has decreased in importance. Text stenography using digital files is not used very often because the text files have a very small amount of redundant data .

III. **Image steganography**: Images are used as the popular cover objects for steganography. A message is embedded in a digital image through an embedding algorithm, using the secret key. The resulting stego image is send to the receiver. On the other side, it is processed by the extraction algorithm using the same key. During the transmission of

stego image unauthenticated persons can only notice the transmission of an image but can‟t guess the existence of the hidden message.

IV. **Audio steganography:** Audio stenography is masking, which exploits the properties of the human ear to hide information unnoticeably. An audible, sound can be inaudible in the presence of another louder audible sound .This property allows to select the channel in which to hide information.

V. **Protocol steganography**: The term protocol steganography is to embedding information within network protocols such as TCP/IP. We hide information in the header of a TCP/IP packet in some fields that can be either optional or are never used.

CHAPTER 3

LEAST SIGNIFICANT BIT ALGORITHM

3.1 Principle of Embedded Data in Last Bit

In order to hide a message within a digital message, we take advantage of the least significant bit within each pixel of an image. Each pixel is made up of three eight-bit integers that store the value of the color in each image. For example, 255 red, 255, green, and 0 blue makes the color

yellow. By replacing the least significant bit in each of these color values, it is possible to hide a secret message, bit by bit, without changing the color values too much, as illustrated in Example bellow.

(Unaltered least significant bit. Encoded least significant bit.)

	Red Pixel Value	Green Pixel Value	Blue Pixel Value
Unaltered:	1111 1111	1111 1111	0000 0000
Encoded:	1111 1111	1111 1110	0000 0001
Message:	1	0	1

For all its complexity and refinement, the human eye has a hard time determining subtle differences between color values. For example, in Figure 2 the left and right halves of the image are comprised of the color values 50 and 51 respectively (in an unsigned integer 8 format where values range between 0 – 255), but we would be hard pressed to note the difference.

Steganography looks to exploit this limitation by storing a message within the pixel color values of a cover image.

3.2 Least Significant Bit Embedded Algorithm

Encoding hidden messages using steganography follows the basic process outline in Figure 3.1 and algorithm 1.

LSB Algorithm 1

Input :- File,Secret Text or image

Output:- Stego Image Steps of encoding process:-

[1] During encoding first text file is selected .

[2] Then Image file is selected in which text is to hide

[3] Split the secret text to insert in Image and then it get hidded using least significant bit insertion technique.

[4] Hash code is used to find position for LSB insertion and also embed data within the frame. It has some password to hide data .

[6] Afterwards places splited secret text characters 3 bit in red pixel,3 bit in green pixel,2 bit in blue pixel and stego frame will be formed.

The message is first analyzed to determine whether it is a text or image message type. The message type and dimensions of the message (overall length for text and height/width values for an image) form an 8-bit Header that is used to reconstruct the message during the Decoding Process.

The Header is concatenated to the beginning of the message and this new combined message is encrypted using a simple symmetric Exclusive OR

(XOR) encryption key, which follows the bit logic outlined in Table and Example bellow.

Encoding		**Decoding**	
Plaintext:	10101010	Ciphertext:	10100101
Encryption Key:	00001111	Encryption Key:	00001111
Ciphertext:	10100101	Recovered Plaintext:	10101010

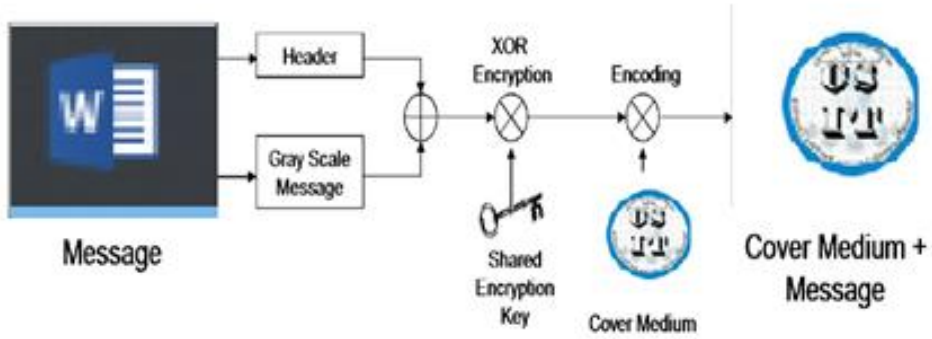

Figure 3.1 Least Significant Bit Embedded Algorithm

From Example above we can clearly see that we need to use the same encryption key during the decryption process if we want to successfully recover the original message. Unfortunately, this symmetric encryption key is a shared secret that must be sent separately from the encrypted message to ensure it remains uncompromised and secure.

Finally, the encrypted Header and Message is encoded onto the Cover Medium Image's least significant bits using one of two methods.

3.3 Least Significant Bit Extract Data Algorithm

Extraction and recovering the hidden message follows the basic process outlined in Figure 3.2 and algorithm 2. This process reverses the effects of the encoding process and reveals the secret message to an authorized user.

LSB Algorithm 2

Decoding Process For Extracting Secret Information:-

Input :- Stego Image

Output :- Secret Text or Image Steps of decoding process :-

[1] During decoding or extracting the data from stego Image first video file selected.

[2] These stego Image will be applied to extract hidden data from frame.

[3] Here the same password is used to decode the data as it is known to intended receiver.

[4] In this way secret message will be displayed on text bit and it is extracted easily.

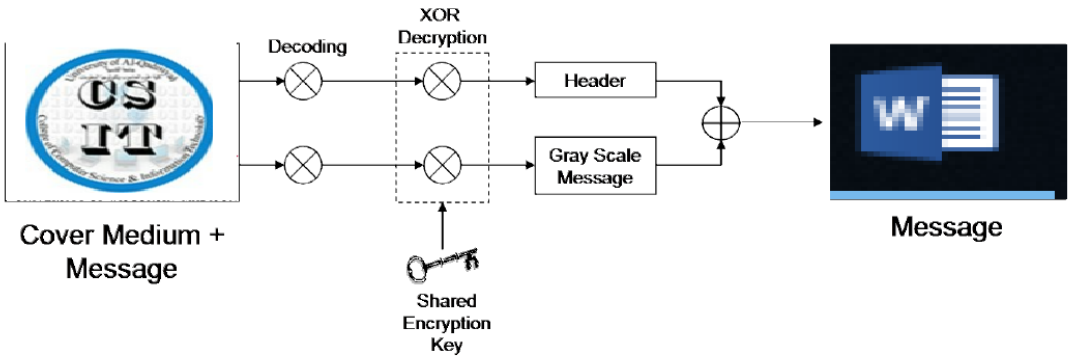

Figure 3.1 Least Significant Bit Extract Data Algorithm

Firstly, the encrypted Header values are recovered from the Cover Medium Image and decrypted using the same XOR encryption key used during the encoding process. The Header dimension value (length for text message; width times height for image message) is used to determine the stop value for the recovery algorithm, thereby reducing the complexity and speeding up the recovery process. Next, the program recovers the entire encrypted Message from the Cover Medium Image by using the Header Dimensions to determine when to stop. The recovered Message is decrypted using the same XOR encryption key used during encoding. Finally, if the message type is an image, the program uses the height/width dimensions from the recovered Header to reconstruct the original image from the decrypted message values for display.

CHAPTER 4

PEPOESED SYSTEM AND RESULT DESICCATION

4.1 Preposed System

For our project we implemented Sequential and Decoder functions using MATLAB script files. Our functions follow the code outline provided in [1], but also include message transposition and XOR encryption for enhanced security. We also developed a rudimentary user-input function that allows a user to encode and decode their own hidden messages without preloading variables in MATLAB. Sequential steganography is relatively simple to implement and provides the most basic level message hiding capabilities. Sequential, like the name implies, begins at some point and then follows some set pattern to encode the message across the least significant bits of the Cover Medium Image. The stegancoder function begins with the top left pixel of the Cover Medium Image and proceeds to encode the message down the columns of the image from left to right, like is shown in Figure 4.1.

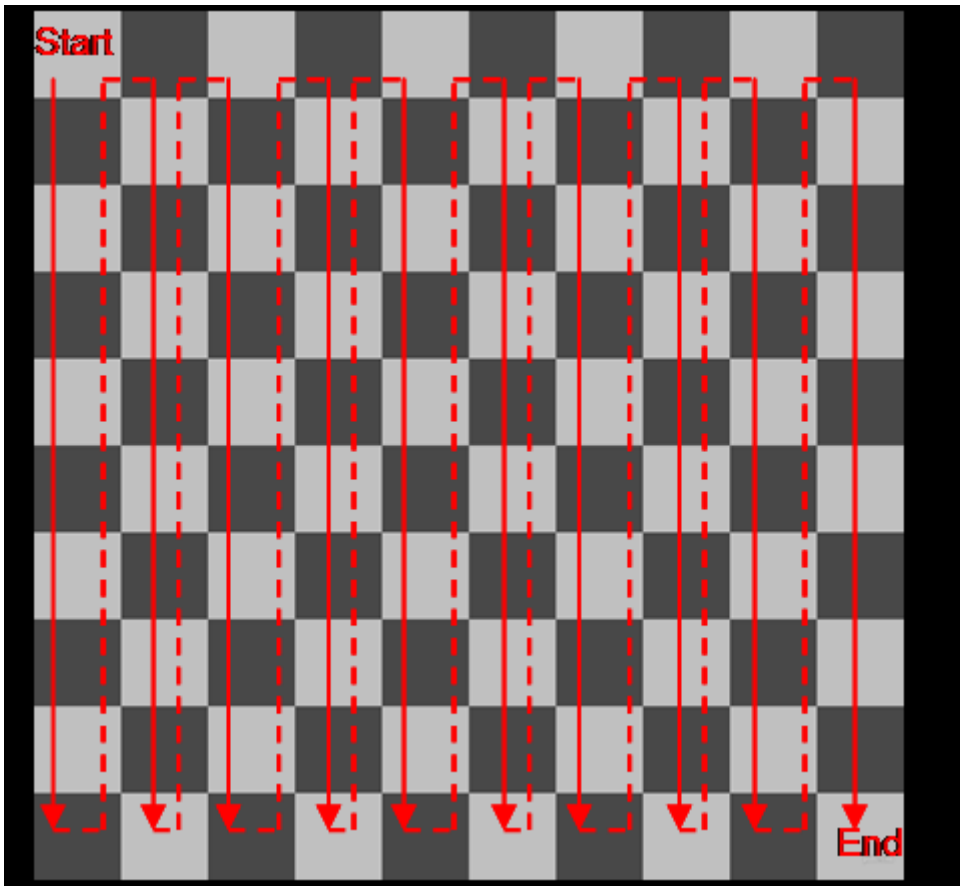

Figure 4.1 Sequential Encoding Method.

After the message has been analyzed, the Header added, and the new message encrypted; each value of the message is converted from a value in the range of 0-255 and is represented in its 8-bit binary equivalent, which we will call a message word. For example, the message word for the number 15 would be represented as 00001111.

Next, to enhance message security we implement a layer of scrambling, called transposition, during the encoding process. Each bit of every message word is stored in the following pattern: RGBBGRRG; where R, G, and B mean the message value is stored in the Red, Green, and Blue Channels of the next available pixel. This means that every message word uses three pixel's worth of Red and Green Channel values while only require two pixel's worth of Blue Channel values. To implement this pattern the stegancoder and stegandecoder functions use a set of horizontal and vertical

counters to determine what the next available pixel location is for each color channel. After a bit is encoded the counter(s) are increased to indicate the next available pixel until the entire message has been encoded.
To recover the message the stegandecoder function uses the same process to reverse the encoding. Red, Green, and Blue Channel counters are initialized and used to recover the first 64

values that constitute the Header. These values are rearranged into the proper order (by reversing the RGBBGRRG encoding order) and decrypted using the same XOR encryption key used during the encoding process. The stegandecoder function then uses the dimension data stored in the recovered Header information to determine the amount of decoding cycles needed to fully recover the Message data. It is important to note that the Red, Green, and Blue Channel counters are NOT reset before the entire process is repeated to recover, decode, decrypt, and prepare the Message from the Cover Medium Image. The final output of the stegandecoder function is a fully decoded message that can be displayed using MATLAB or written out as a text or bitmap file.

4.2 System Interface

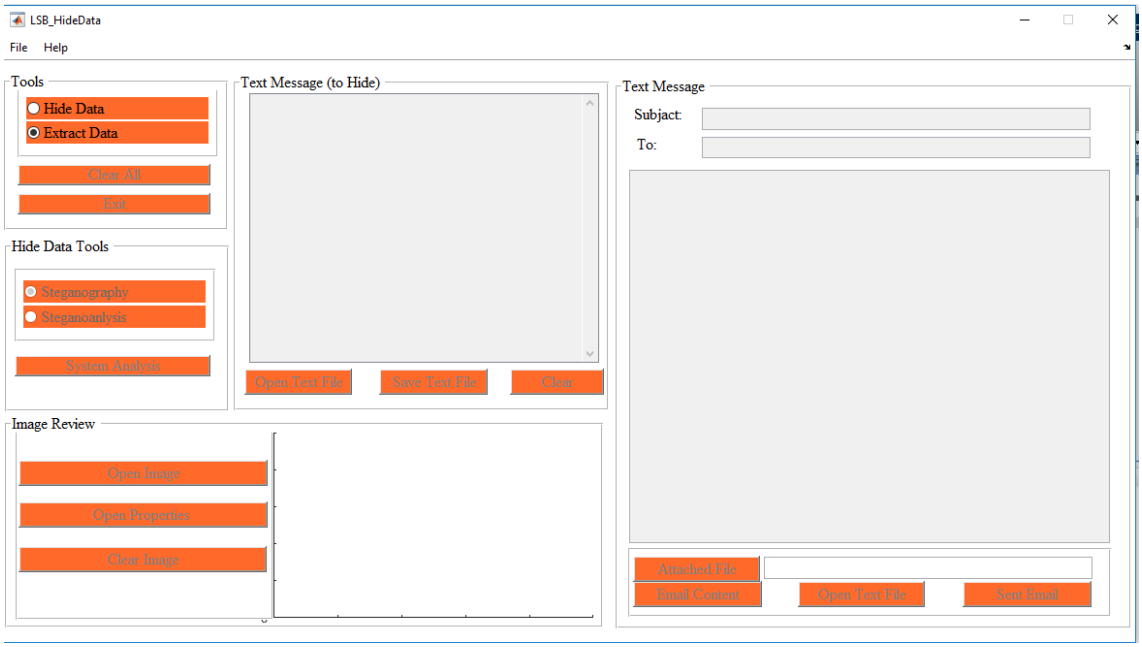

Figure 4.1 Main interface

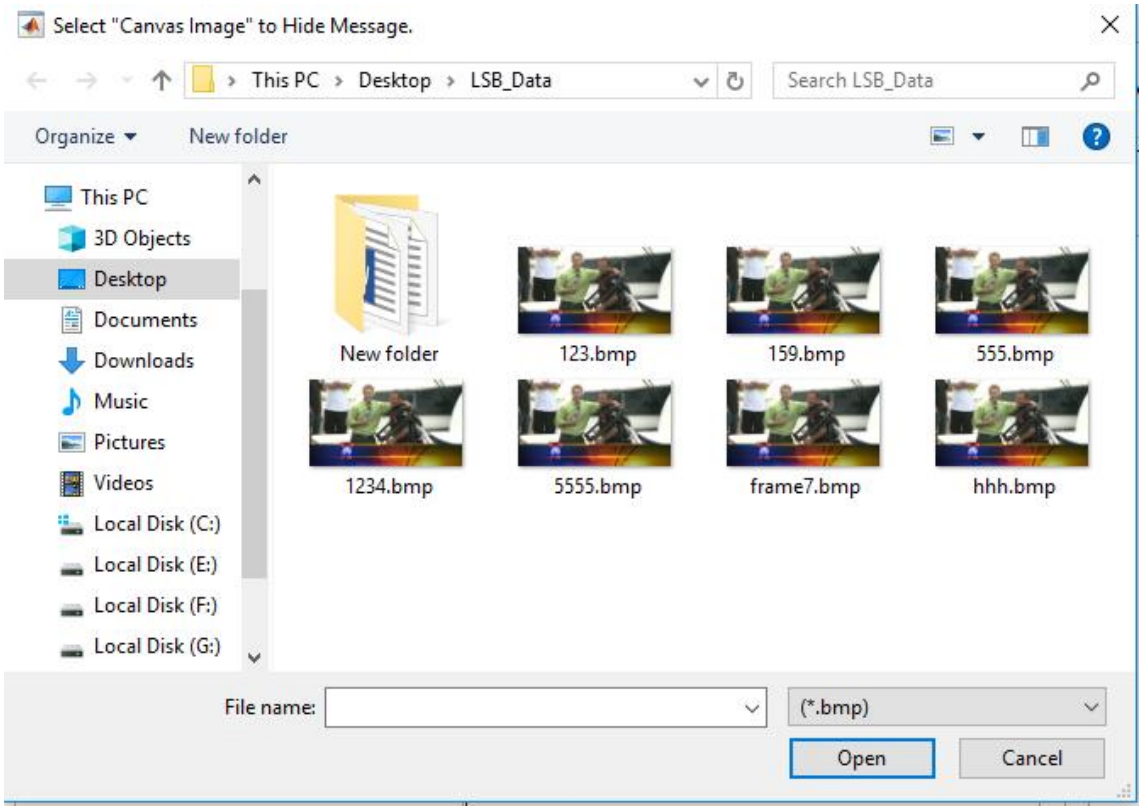

Figure 4.2 Browse

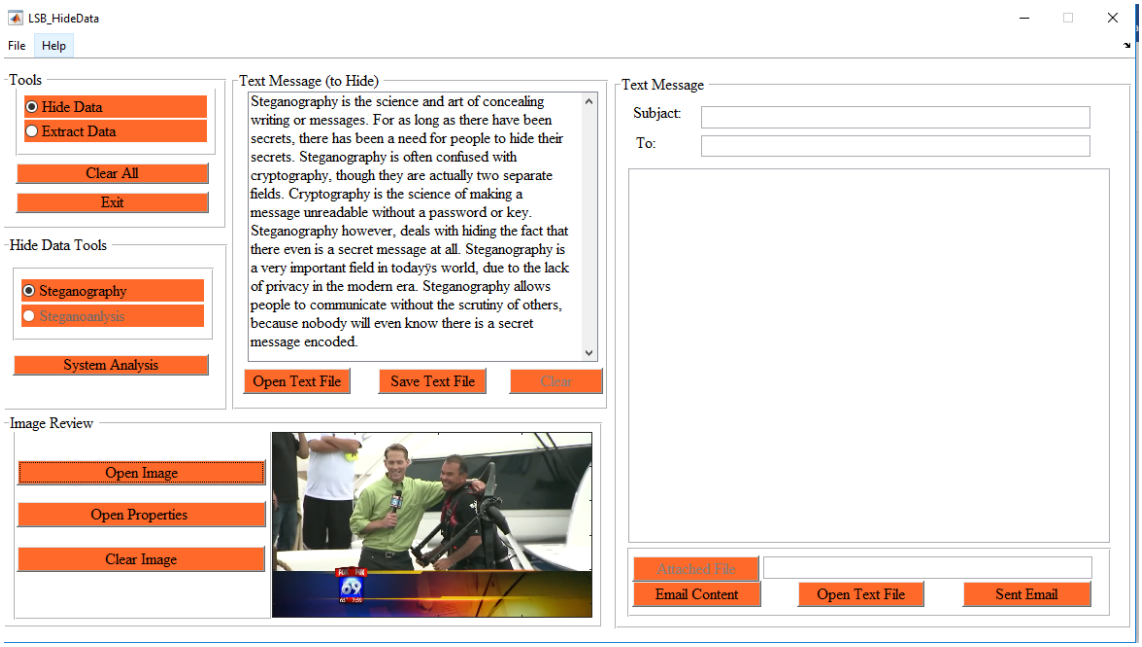

Figure 4.3 interface after execute

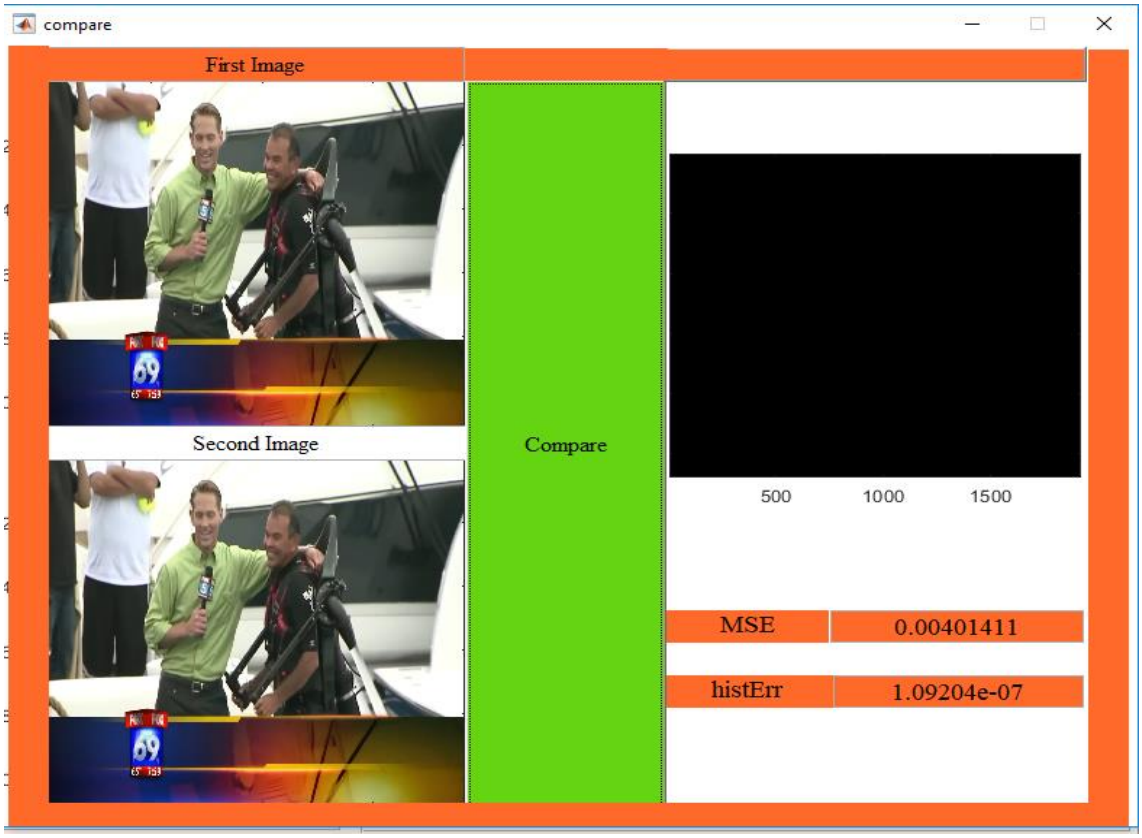

Figure 4.4 compression between two images

4.3 Steganalysis - Detecting Hidden Messages

Even though the human visual system is unable to differentiate between subtle color differences or changes, steganographic messages are still detectable. Steganographic messages are typically encoded by altering the least significant bit of a pixel color value in a specific order or pattern, leaving them vulnerable to statistical analysis tools which can be used to detect and provide information about messages hidden within cover media.

For messages Sequentially Encoded, where the message starts at the top left corner of an image and proceeds to the next pixel until the message is completely encoded, histogram analysis can be completed to identify the presence and length of the message hidden in an image. In Figure 9, the first chapter of Lewis Carroll's The Hunting of the Snark was Sequentially Encoded into the image in (b) leaving no visual indication that anything has been changed, but when we examine the Discrete Cosine Transform (DCT) Histogram of the images we can clearly see that the message has altered the distribution of the DCT coefficients from a normal distribution [1].

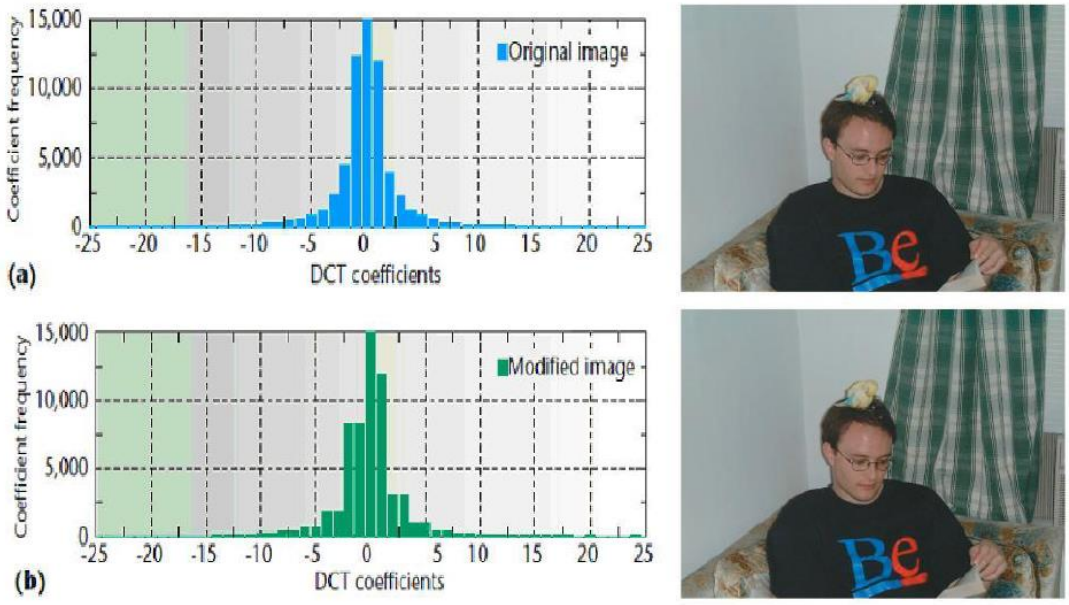

Figure 9: Histogram Analysis of Sequentially Encoded Message [1].

(a) DCT Coefficients Histogram (left) and Image (right) without message.

(b) DCT Coefficients Histogram (left) and Image (right) with Sequentially Encoded message.

CHAPTER 5

CONCLUSIONS AND FUTURE WORKs

Hidden messages remain an important and evolving science facilitating the secure transmission of information. Steganographic techniques and processes exploit detection limitations in the human visual system to store messages in underutilized/redundant bits used by digital media. Our project implements relatively simple Sequential Encoding and Decoding techniques using MATLAB for the functions. Converting these functions into a more efficient and web-friendly format is a natural extension of this work and would provide an interesting comparison. Hidden messages remain an important and evolving science facilitating the secure transmission of information. Steganographic techniques and processes exploit detection limitations in the human visual system to store messages in underutilized/redundant bits used by digital media. Our project implements relatively simple Sequential and Pseudo-Random Encoding and Decoding techniques using MATLAB for the functions. Converting these functions into a more efficient and web-friendly format is a natural extension of this work and would provide an interesting comparison.

Even though steganography makes it difficult to detect the presence of a hidden message, steganalysis uses statistical analysis tools and processes to identify and recover message data from a cover image. Another area for continued research would be to analyze the effectiveness and relative security provided by our functions by using available steganalysis tools, such as Outguess. It would be interesting and useful to provide users with a relative security level score or value during the encoding process, so the user could determine how safe their message is.

Ultimately humanities' need to securely communicate makes steganography an evolving and relevant field for continued research and development

In Future used video format other than media format to hide data in most significant bit

REFERENCES

[1] N. Provos, P. Honeyman, Hide and Seek: An Introduction to Steganography, IEEE Computer Security 2003, <http://www.citi.umich.edu/u/provos/papers/practical.pdf>.

[2] J. C. Judge, Steganography: Past, Present, Future, SANS Institute, <http://www.sans.org/reading_room/whitepapers/stenganography/steganography-past-present-future_552>

[3] Wikipedia, Microdot, <http://en.wikipedia.org/wiki/Microdot>

[4] S. Singh, The Code Book, Anchor Books, 2000, ISBN: 0385495323.