

DIGITAL IMAGE WATERMARKING USING LSB METHOD

**By
MAULIDIN**

UNIVERSITI UTARA MALAYSIA

2011

DIGITAL IMAGE WATERMARKING USING LSB METHOD

A project submitted to the
Dean of Awang Had Salleh Graduate of School Arts and Sciences
In partial fulfillment of the requirement for the degree
Master of Science (Information and Communication Technology)
Universiti Utara Malaysia

By
MAULIDIN

© MAULIDIN, 2011. All rights reserved.

PERMISSION TO USE

In presenting this project in partial fulfillment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this project in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor or, in their absence by the Dean of the Graduate School. It is understood that any copying or publication or use of this project or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my project.

Requests for permission to copy or to make other use of materials in this project, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok
Kedah Darul Aman.

ABSTRACT

Nowadays, the transmission of data is very important and it has been carried out by using digital media (Internet, e-mail). There are some problems that may occur in the transmission of digital media data which is associated with data security level to be sent. Base on that condition, the thought of doing data concealment are emerge. Digital image is one of the most common media used by the community. Digital watermarking is one of cryptography methods which used to hide data into digital image so that the data sent cannot be identified by the irresponsible people. Data encryption is used to improve the security of data to be sent. Internet as the world's largest network system that connects nearly all computer worlds, making all the computers in the world is increasingly easy to exchange data. There are so many digital data falsification has occurred. Digital falsification of data aimed to succeed the counterfeiting syndicate these hidden plans. the falsification problem of digital data will from be resolved by applying the watermarking using LSB method (Least Significant Bit).

Keywords: Digital Watermarking, LSB method

ACKNOWLEDGEMENTS

First and foremost, I would like to thank to Allah SWT for his bless and mercy who has guided me in completing this final report. Then I would like to thank to my supervisor of this Master Project, Assoc. Prof. Hatim Mohamad Tahir for the valuable guidance and advice. He inspired me greatly to work in this master Project. His willingness to motivate me contributed tremendously to my master project. I do believe that without his kind guidance, the master project might not complete as it is intended to be. Also my sincere appreciation to Ms. Syahida Binti Hassan who gives some guidance was a great impact on my studies and all the lecturers who were instrumental in my quest for knowledge.

I would like to give a sense of appreciation to my family (father, mother, Abi, Umi, Mimi, Wati, Riska, Faris, Fathan, and Salma) in encouraging me to further my studies and also for my sweet heart Indah Pratiwi Putri for her support, encouragement, and help to complete this master project. Finally, i would like to express my appreciations to all my friends, colleagues, superiors and everyone who were involved directly or indirectly in this Master Project.

TABLE OF CONTENTS

PERMISSION TO USE	iv
ABSTRACT.....	v
ACKNOWLEDGMENTS	vi
TABLE OF CONTENTS.....	vii
LIST OF FIGURES	ix
LIST OF TABLES	xi
CHAPTER I INTRODUCTION	
1.1. Background	1
1.2. Problem Statements.....	5
1.3. Research Questions	6
1.4. Research Objectives	6
1.5. Scope of Project	7
1.6. Significance of Project	7
1.7. Organization of Project	8
1.8. Summary	8
CHAPTER II LITERATURE REVIEW	
2.1. Information Hiding.....	9
2.2. Watermarking Concept	13
2.3. Watermarking Technique.....	26
2.4. Summary	34

CHAPTER III RESEARCH METHODOLOGY

3.1. Research Definition.....	35
3.2. Design Definition.....	36
3.3. Research Design Technique.....	36
3.4. Conclusion.....	57
3.5. Summary.....	57

CHAPTER IV FINDING AND RESULT

4.1. Digital Image File.....	59
4.2. Text File.....	59
4.3. Image File with Hidden Text Message.....	60
4.4. LSB analysis in Application.....	60
4.5. Decryption and Encryption Process Analysis of Text Messages.....	61
4.6. Size Analysis Data Hidden.....	62
4.7. Digital Image Watermark Testing.....	63
4.8. Summary.....	86

CHAPTER V CONCLUSION AND RECOMENDATIONS

5.1. Conclusion.....	87
5.2. Research Contribution.....	88
5.3. Limitation and Recommendation.....	88

REFERENCES.....	89
-----------------	----

APPENDIX.....	93
---------------	----

LIST OF FIGURES

Figure 2.1: Information Hiding.....	12
Figure 2.2: The general scenario for information hiding.....	13
Figure 2.3: Type of watermarking.....	17
Figure 2.4: LSB watermark system (Embedding).....	28
Figure 2.5: LSB watermark system (Extracting).....	28
Figure 3.1: Steps in Waterfall Model.....	36
Figure 3.2: Use Case Diagram.....	42
Figure 3.3: Activity diagram for encryption.....	44
Figure 3.4: Activity diagram for decryption.....	45
Figure 3.5: Flow Chart for encryption.....	47
Figure 3.6: Flow Chart for decryption.....	48
Figure 3.7: Basic Flow that show overall encryption step.....	49
Figure 3.8: Collaboration Diagram show overall encryption.....	50
Figure 3.9: Sequence diagram show overall decryption.....	51
Figure 3.10: Collaboration Diagram show overall decryption.....	51
Figure 3.11: Sequence Diagram of Exception Flow.....	52
Figure 3.12: Collaboration Diagram of Invalid Image Exception.....	52
Figure 3.13: Sequence Diagram of Large Text Message Size.....	53
Figure 3.14: Collaboration Diagram of Large Text Message Size.....	53
Figure 3.15: Sequence Diagram of watermarked image type not BMP.....	54
Figure 3.16: Collaboration Diagram of watermarked image type not BMP.....	54
Figure 3.17: Class Diagram.....	55
Figure 3.18: Software interface design.....	56
Figure 4.1: Plain image in Jpeg file format.....	64
Figure 4.2: Plain image which is already converted to binary as a raw data.....	64

Figure 4.3: Plain image in BMP file format	65
Figure 4.4: Plain image which is already converted to binary as a raw data.....	65
Figure 4.5: Typing text	66
Figure 4.6: Notepad text file.....	67
Figure 4.7: Notepad text file using WinHex.....	67
Figure 4.8: WordPad text file	68
Figure 4.9: WordPad text file using WinHex	68
Figure 4.10: Load plain Jpeg image.....	71
Figure 4.11: Jpeg watermark result with typing text message.....	71
Figure 4.12: Jpeg image with txt text file.....	72
Figure 4.13: Jpeg watermark result with txt text file	73
Figure 4.14: Jpeg image with rtf text file.....	73
Figure 4.15: Jpeg watermark result with rtf text file.....	74
Figure 4.16: Load plain BMP image.....	74
Figure 4.17: BMP watermark result with typing text message.....	75
Figure 4.18: BMP image with txt text file	75
Figure 4.19: BMP watermark result with txt text file	76
Figure 4.20: BMP image with rtf text file.....	76
Figure 4.21: BMP watermark result with rtf text file.....	77
Figure 4.22: Text message decryption in Jpeg.....	78
Figure 4.23: Txt file text message decryption in Jpeg	78
Figure 4.24: Rtf file text message decryption in Jpeg.....	79
Figure 4.25: Text message decryption in BMP.....	79
Figure 4.26: Txt file text message decryption in Jpeg	80
Figure 4.27: Txt file text message decryption in Jpeg	80
Figure 4.28: Enlarging manipulation step of computer.jpeg	81
Figure 4.29: Enlarging manipulation step of uumLogo.bmp.....	81
Figure 4.30: Narrow manipulation step of computer.jpeg	82
Figure 4.31: Narrow manipulation step of uumLogo.bmp	82
Figure 4.32: Rotation manipulation step of computer.jpeg	83
Figure 4.33: Rotation manipulation step of uumLogo.bmp.....	83
Figure 4.34: Cropping manipulation step of computer.jpeg	84
Figure 4.35: Cropping manipulation step of uumLogo.bmp	84

LIST OF TABLES

Table 3.1:	Gant Chart	38
Table 3.2:	Functional Requirements	40
Table 3.3:	Non-Functional Requirements	40
Table 4.1:	Raw data images	63
Table 4.2:	LSB raster process computer.jpeg	69
Table 4.3:	LSB raster process uum.BMP	69
Table 4.4:	Result of manipulation test	85

CHAPTER I

INTRODUCTION

This chapter introduces a brief description of this study. A general overview of the field of this work, problem statement, the objective and the scope of this study has been presented.

1.1. Background

Computer has been rapidly used in almost all aspects of life. Based on the history of modern computer it's begun with two separate technologies automated calculation and programmability, but no single device can be identified as the earliest computer, partly because of the inconsistent application of that term. The word of computer represents a tool which carried out calculations, or computations, and the word which is continued with the same meaning until the middle of the 20th century. From the end of the 19th century onwards, the word began to take on its more familiar meaning, describing a machine that carries out computations (Rojas, et al, 2000).

Computer is a tool used to process the data according to procedures that have been formulated. Computer as digital processing equipment is currently owned by almost every family. Nowadays, computers are also widely used as a medium of entertainment and communication sharing. Using computer, people share their

information as a digital data to others. Digital data is a form of electronic media where data is stored in digital form.

By looking at development of computers and knowledge in digital processing, the data in digital form significantly increase being used; those things are caused by the computer that is currently developing electronic equipment that use and process data in digital form (binary). The usage of digital data in the form of text, sounds, images and video extremely rapidly with computers elements, especially in the term of coupled with technological developments in the world network of computers called the Internet.

Electronic information sharing brings challenges to a digital data authenticity. The challenges occur because the digital data is very easy to copy, modify and spread. Based on this condition, the rules of original access control will be lost. In this process, the ownership of files, the relationship between the users may change, it requires the security model to support the mutable attributes and implement security file sharing throughout the file life cycle.

In the past one decade, there has been a phenomenal increase in the use and circulation of information in digital multimedia formats for various purposes. Today, many paintings, photographs, newspapers, books, music etc., are available over the Internet in one or other multimedia format (Shi & Lv, 2010). Therefore, protection and ownership of such data from illegal use and replication is a major concern of the owners.

There are many ways that have been taken to provide security or protect digital data of all the changes, such as: header marking, visible marking, encryption

and copy protection; but all these ways have their respective weaknesses. There are several ways that has done to keep the originality or creativity in their work, such as (Wang, X & Ye, J, 2010):

- a. Header marking; providing information or information as a mark of authenticity (Originality) in the header of a digital data.
- b. Visible marking; to give signs of originality (originality) in digital data explicitly.
- c. Encryption; encode digital data into another representation that is different from the original representation (but can be restored to original form) and requires a key from the holder of the work of the digital image to return to the original representation.
- d. Copy protection; provide protection to digital data by limiting or by providing protection in such a way that digital data cannot be duplicated.

The increasing of necessity to protect intellectual property rights of such digital content has led to considerable research on the direction. Digital watermark is one of suitable approach, which has been widely used to protect the copyright of digital images. In order to strengthen the intellectual property right of a digital image, a trademark of the owner could be selected as a watermark and embedded into the protected image (Shi & Lv, 2010). Digital watermarking system should be set up on the basis of application and its technology is mainly applied to copyright protection, pirated copies tracing, image authentication and copy control. Among them, copyright protection is the major application as well as the motive and goal of digital watermarking technology (Li, Y, et al, 2010).

The image embedded with watermark is called a watermarked image. Then watermarked image could be published, the owner can prove the ownership of a

suspected image by retrieving the watermark from the watermarked image. According to the retrieved results, we can determine the ownership of the suspected image (Barreto, et al, 2002; Celik, et al, 2001). Therefore, watermark usually used to identify copyright holders and ensure proper payment of royalties. This technique allows attachment of information to digital media.

Security of a watermarking system is measured in terms of the difficulty in destroying its watermarks. Current literature often refers to this measure as the robustness of a watermarking system to distinguish it from the related notion of security of a steganographic system. The latter is measured in terms of the difficulty in detecting the existence of covert communication. Destruction of its watermarks is not the only way to defeat a watermarking system; it is also necessary to impose restrictions on the structure of watermarks to resist interpretation attacks, which seek to confuse true ownership by embedding an equally valid watermark (Cheng, P, et al, 2006; Tran, N, 2002).

Watermarking that will be discussed in this project aims to insert information in the form of text messages in a digital image which results also in the form of digital image data as well. And one of the goals watermarking election as the title of this project is that digital data (in this case is a digital image) specific watermarking method can be applied in order to protect the authenticity of the digital image data by inserting a number of text messages into digital image data.

In this project, the researcher also try to solve the security problem in the way of bring out different idea using Least Significant Bit (LSB) in watermarking technique. LSB use statistical method of bit manipulation in digital data, especially in digital image. LSB algorithm is embedding in watermark into the least important

part of the primary image, and the watermark can be extracted from the primary image.

A LSB is a simple approach to be implemented when we intent to compare pixel value differencing. As referencing specific bits within a binary number, it is common to assign each bit specifically with bit number, ranging from zero upwards to one less than the number of bits in the number. However, the order system is used for this assignment may be in same direction. Both orderings are used in different context, which is why LSB is often used to designate the units bit instead of a bit number, which has the potential for confusion. By extension, the least significant bits (plural) are the bits of the number closest to, and including, the LSB.

1.2. Problem Statements

In this day, computer technology growth increased rapidly. All people can do the communication and sharing of information via the Internet. There are so many digital data falsification has occurred. Digital falsification of data aimed to succeed the counterfeiting syndicate these hidden plans. Property rights or copyright is a policy that indicates the data or images belong to the owner. This policy is only given to the creator or owner of any real data or images that have been officially registered. Through this policy, the owner will get loyalty. Those things will happen in condition of data or images will be made copies and sold without their consent, they would not have suffered losses due to loyalty to the sale.

Based on the condition above, the falsification problem of digital data will be resolved by applying the watermarking technique using LSB (Least Significant Bit).

Type of withholding information that would be done by applying these methods are the concealment of information in the form of text messages on a digital image, in order to see how far the process of embedding a text message information affecting the original image after manipulation of the LSB is given in terms of changes in size and protection of data as a text message embedded data in digital technology advances. With this system, each image will from some of the embedded watermarks belong to the true owner of the property to protect that image. Embedded watermark is invisible type. The advantages of this watermark is difficult to be identified, so level of modification made by other users can be reduced. Meanwhile, the extraction process was performed to detect and remove the watermark.

1.3. Research Questions

- i. How to insert information in the form of text messages in digital images using watermarking applications?
- ii. How to analyze the authentication of digital images?

1.4. Research Objectives

There are several research objectives that need to be achieved:

- i. To hide the information in the form of text messages on the digital image data using LSB.
- ii. To identify text messages contents which hidden in the digital image using LSB.

1.5. Scope of Project

The scope of this project includes analysis, design, and implementation phases of the application in order to embed secret information to hide the form of text messages or sentences which later merged with the original digital image data. However inserting an image is one alternative as a prevention way to protect the authenticity of digital images. This insertion steps are aimed to strengthen the protection of media data duplication from the digital image. Media data used is a bitmap (BMP) and JPEG / JPG. Programming language used to create applications through Microsoft Visual Basic version 6.0 to create a software watermark. This project is form of case study of digital image watermarking using the Least Significant Bit (LSB) insertion.

1.6. Significance of Project

In this research project, I try to analyze the extent to which the protection of digital image data can be implemented. The implementation of the protection of digital images by using an application intended to hide the information data in the form of text messages in digital images. It was tested using the methodology applied by LSB insertion technique in watermarking method. The significance of this research generally to ensure the best shape of his way to understand how the power of text message content hidden in digital image data.

1.7. Organization of Project

This section explains general overview for each chapter. This study is classified by five chapters; introduction, literature review, methodology, finding and result, conclusion and recommendation.

Starting with the introduction an overall idea of this study will be gathered in the readers mind or personal perspective. Explaining some terminologies that have been used in this study will make it easy to understand this work.

Based on literature reviews that written in chapter two, the main concept of watermarking and their method have been clarified. Chapter three presents the methodology that has been applied in this study, which adopted from waterfall model. The experiments applied in this study can be found in chapter four. The other way, a conclusion and recommendation are presented in chapter five.

1.8. Summary

The research elements of this study successfully addressed in this chapter based on the existing recharging problem. This chapter also describes motivation factors that lead to the selection scope of study. It also explain the existing problem that need to solve, research question, objective of the study, as well as its significant to the real world situation. These elements are important as it ignites the implementation of the project. The next chapter deals with literature reviews which elaborate to related works that have been established in the different fields.

CHAPTER II

LITERATURE REVIEW

A crucial element of all research degrees is the review of relevant literature. So important is this chapter that its omission represents a void or absence of a major element in research (Afolabi, 1992). According to Bourner (1996) there are good reasons for spending time and effort on a review of the literature before embarking on a research project.

This chapter presents the literature review based on related works that have been done by the past researchers and books. All the issues reviewed will be used as the guide in order to fulfill all of the research purpose and requirement. The detailed explanation for literature review is discussed below.

2.1. Information Hiding

Information hiding technology discusses how to make certain information hidden in public information, then through a public information transmission to transmit hidden information. Information hiding must be considered that a variety of hiding information in the experience of a variety of environments and a variety of operations still has the immune ability. Information hiding must be considered a threat to the normal operation of the information, so that confidential information on

these operations has immunity. Information hiding technology is currently the anti-counterfeiting technology and information security technology for a new direction, and it can realize information hiding and tracking of new technology in an open network environment. So far, information hiding technology in the digital works of intellectual property protection, business transactions in the notes anti-counterfeiting, anti-counterfeiting and other printing products have a wide range of applied research (Wang, X & Ye, J, 2010).

Information hiding technology in accordance with the different processing objects is divided into overlapping like the technology, digital watermarking technology and for the sound technology. Domestic and foreign companies and academia, digital watermarking has been committed to applied research, the main focus on addressing the digital copyright protection issue, a lot of options (Wang, X & Ye, J, 2010).

Information hiding is an emerging research area which encompasses applications such as copyright protection for digital media, watermarking, fingerprinting, steganography, and data embedding. In particular, watermarking is now a major activity in audio, image, and video processing, and standardization efforts for JPEG-2000, MPEG-4, and digital video disks are underway (Moulin, P & O'Sullivan, J, A, 2003)

Information technologies are used for the processing of valuable information more and more nowadays. Information processing, from information technologies point of view, consist of information storage, transmission, evaluation and interpretation of data. However, these data represent valuable information and they have to be protected in the following ways: a) only authorized persons should have

access to these data, b) only authentic data should have been processed, c) there should be a method how to find out who have made, changed or removed these data, d) data should be confidential, e) data should not be denied, when they are needed (Ridzon, et al, 2010).

Generally, information hiding including four stages: preprocessing, embedding, passing, and extracting. In order to insure every stage safety, encryption algorithm must be introduced at the stage of preprocessing, and at the stage of embedding, algorithm of hiding information such as based wavelet used. At passing stage, communicating should be taken stealthily, so that the stage of passing is safety, too. So the hiding strategy will form a safety system, with which both the content of information and the manner information received and transferred can be hidden. So, hiding communication could be built (Zhang & Li, 2009).

Zhao clarifies the main principles for hiding information in digital documents. In using randomness and inconspicuousness to hide information, a reliable security level could be reached. Digital watermarking is a fairly new research area and combines studies and results from other research areas, such as digital signal processing, communications, compression, information theory, and cryptography (Seitz, J, 2005).

Information hiding is defined as a nearly invisible embedding of information within various host data sets such as audio, text, and digital images. It is highly multi disciplinary field that combines multimedia data (audio, text, image, etc) and signal processing with cryptography, communication theory, coding theory, signal compression and the theory of visual perception. (Kazakeviciute, G & Rosenbaum, 2001). The various information hiding areas can be classified as given in figure 2.1.

The hidden message may have no relationship to the carrier image in which it is embedded (this is the case in covert, secure communication), or the message may supply important information about the carrier image, such as copyright notices, authentication information, captions, date and time of creation, serial number of the digital camera that took the picture, information about image contain and access to image and many more.

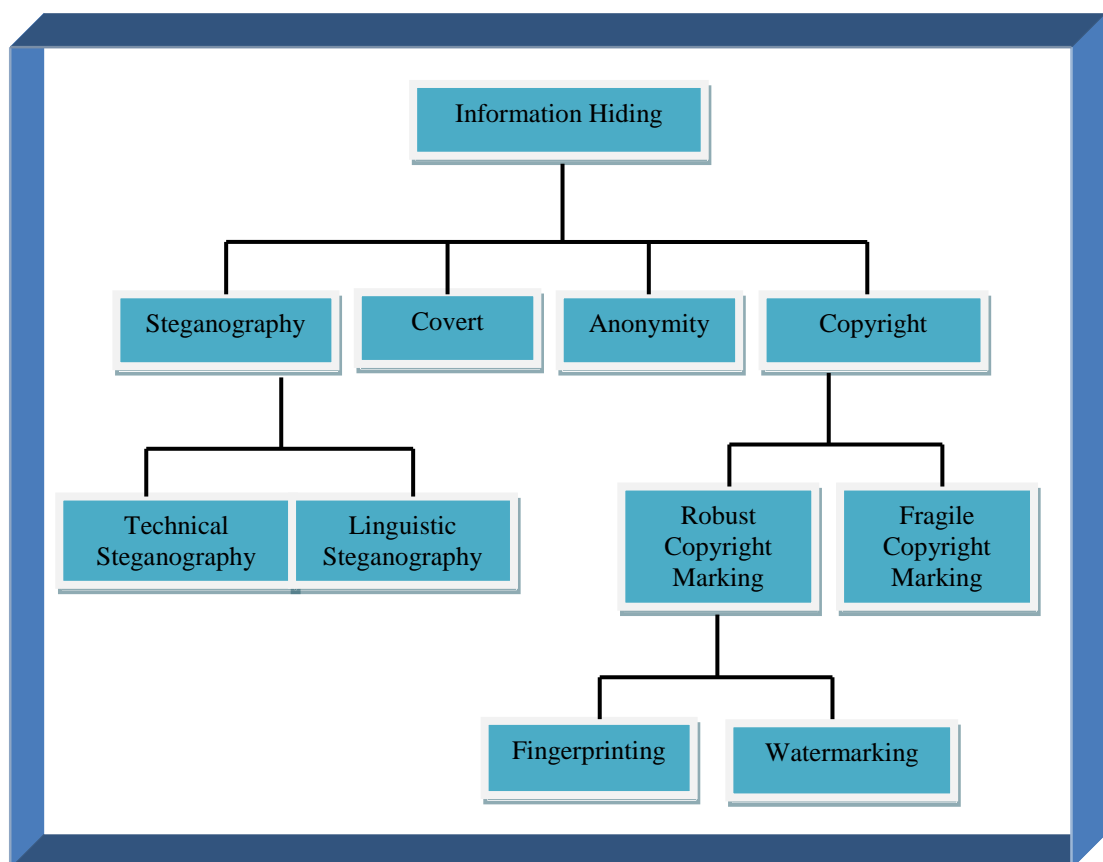


Figure 2.1: Information Hiding (Kazakeviciute, G & Rosenbaum, 2001)

The most general scenario for information hiding process shown as the following (Figure 2.2):

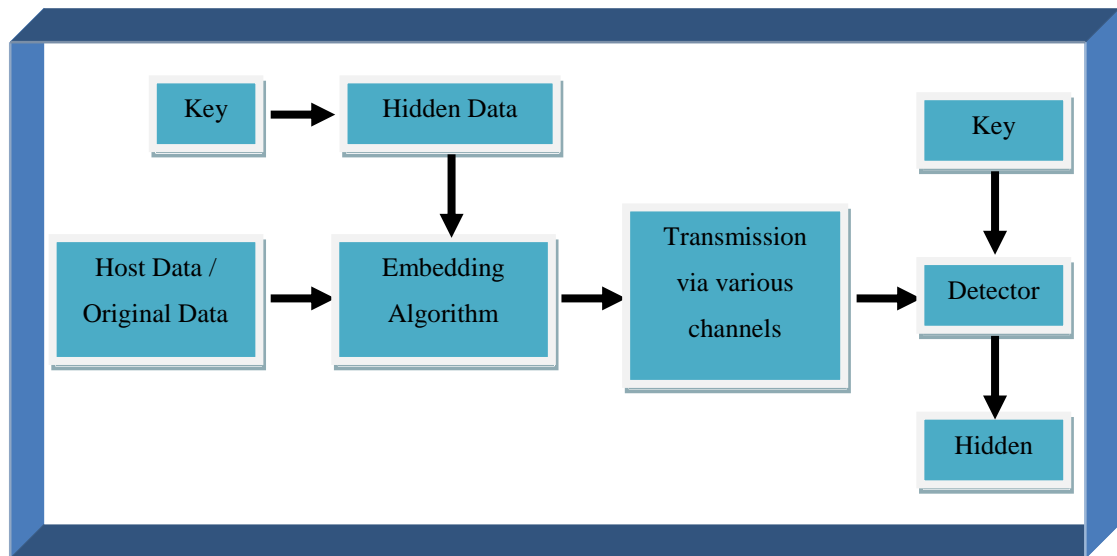


Figure 2.2: The general scenario for information hiding (Fridich J, 1998)

Each of the information hiding technique consists of: (1) the embedding algorithm and (2) a detector function. The embedding algorithm is used to hide secret information inside a cover (or carrier) data; the embedding process is protected by a keyword so that only those who possess the secret keyword can access the hidden message. The detection function is applied to a (possibly modified) carrier and returns the hidden secret message. The key used to extract the hidden message must be the same as the key used to embed the hidden data.

2.2. Watermarking Concept

2.2.1. History Watermarking

A watermark is made by impressing a water-coated metal stamp or dandy roll onto the paper during manufacturing. These watermarks were first introduced in Bologna, Italy, in 1282; however the dandy roll was invented in 1826 by John Marshall. Watermarks have been used by papermakers to identify their product, and

also on postage stamps, currency, and other government documents to discourage counterfeiting. In France, they were introduced during World War II by the Vichy regime, and counterfeited by people such as Adolfo Kaminsky. The invention of the dandy roll revolutionised the watermark process and made it much easier for a company to watermark their paper (Meggs, P, 1998).

Although the art of papermaking was invented in China over one thousand years earlier, paper watermarks did not appear until about 1288, in Italy. The marks were made by adding thin wire patterns to the paper molds. The paper would be slightly thinner where the wire was and hence more transparent. The meaning and purpose of the earliest watermarks are uncertain. They may have been used for practical functions such as identifying the molds on which sheets of papers were made, or as trademarks to identify the paper maker. On the other hand they may have represented mystical signs, or might simply have served as decoration

By the eighteenth century, watermarks on paper made in Europe and America had become more clearly utilitarian. They were used as trademarks, to records the date the paper was manufactured, and to indicate the sizes of original sheets. It was also about this time that watermarks began to be used as anti counterfeiting measures on money and other documents (Cox, et al, 2008).

The terms of watermark seem to have been coined near the end of the eighteenth century and may haven derived from the German term *wassermark* (though it could also be that the German word is derived from the English. The term is actually a misnomer, in that water is not especially important in the creation of the mark. It was probably given because the marks resemble the effects of water on paper.

William Congreve, an Englishman, invented a technique for making color watermarks by inserting dyed material into the middle of the paper during papermaking. The resulting marks must have been extremely difficult to forge, because the Bank of England itself declined to use them on the grounds that they were too difficult to make. A more practical technology was invented by another Englishman, William Henry Smith. This replaced the fine wire patterns used to make earlier marks with a sort of shallow relief sculpture, pressed into the paper mold. The resulting variation on the surface of the mold produce beautiful watermarks with varying shades of gray.

Another idea that played important role in several wars in the nineteenth and twentieth century's was originality proposed by Brewster (1857). He suggested hiding messages by shrinking was made possible by the technology developed by French photographer Dragon during the Franco-Prussian War (1870-1871).

2.2.2. Watermarking Development

Digital watermarking is a technology that protects the integrity or copyright of digital works. Using of random redundant data in digital media files, the watermark to be incorporated, not easily to be detected but it can be determined. Watermark is usually not visible or nondetect, and with the original data (such as images, audio, video and so on) closely and hide in them, becoming an inseparable part of the source data without impairing the perceptual quality of that source (Yu, Y. H, et al, 2007).

Digital watermarks are primarily used for ownership and author rights protection and copy prohibition. Digital watermarking represents embedding of information within the content of the image. Embedded information should be undetectable by human visual system but has to be detectable by a detector, which is used in the watermark extraction or detection process (Furht, et al, 2005).

Digital watermarking is described as a viable method for the protection of ownership rights of digital audio, image, video and other data types. It can be applied to different applications including digital signatures, fingerprinting, broadcast and publication monitoring, authentication, copy control, and secret communication (Seitz, J, 2005).

Digital watermarking is an imperceptibly altering a work in order to secure and protect the copyright or ownership of the source and owner from unauthorized copying. It embeds into a background texture certain messages that are invisible on original documents but appear clearly on the copies made thereof (Anan, et al, 2007). A digital watermarking can indicate the copyright owner, identify the buyer or provide additional information of digital content, and embed the information into digital images, digital audio and video sequence. Applied of watermarking technique also include invisible ink, microdots, arrangement of words, digital signatures, hidden paths and wide spectrum of communication (Jiang, X, 2010).

Digital watermarking means embedding information into digital material in such a way that it is imperceptible to a human observer but easily detected by computer algorithm. A digital watermark is transparent, invisible information pattern that is inserted into a suitable component of the data source by using a specific

computer algorithm (Dittmann, 2000; Hartung & Kutter, 1999; Katzenbeisser & Petitcolas, 2000; Petitcolas, Anderson, & Kuhn, 1999).

Digital watermarks are signals added to digital data. Watermarking techniques can be divided into various categories in various ways. The watermarks can be applied in spatial domain. An alternative to spatial domain watermarking is frequency domain watermarking. It has been pointed out that the frequency domain methods are more robust than the spatial domain techniques. (Chandra, et al, 2010).

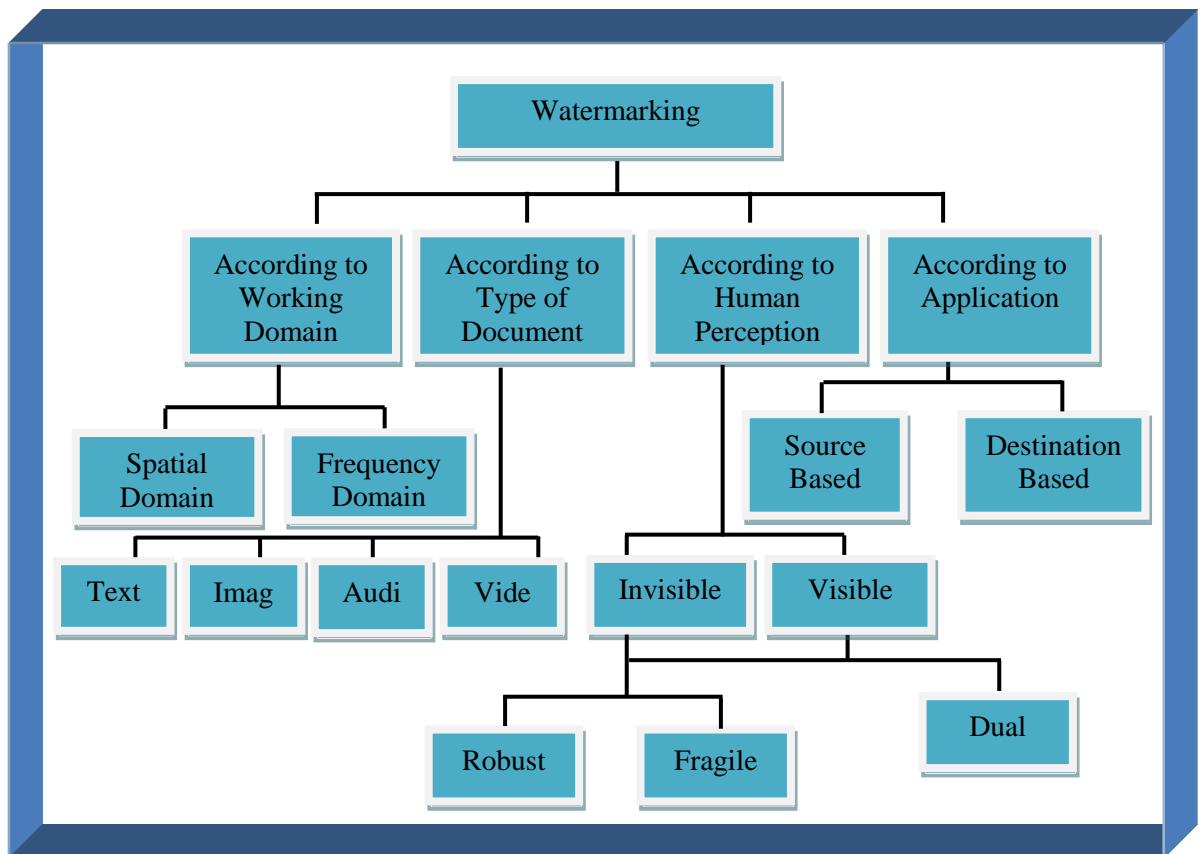


Figure 2.3: Type of watermarking (Chandra, et al, 2010).

Watermarking techniques can be divided into four categories according to the type of document to be watermarked as follows.

- **Image Watermarking**

Image watermarking deals with creating a metadata (a watermark) about the image content and hiding it within the image. Image watermarking deals with creating a metadata (a watermark) about the image content and hiding it within the image (Furht, 2008).

- **Video Watermarking**

Digital watermarks can be used to enable copy control of video in a DVD and VCD. In this combination, the recording device scans the digital data stream for an existing watermark and enables or disables the recording action for a specific movie or stream. Such technology could extend the pay-per-view concept and close the gap between the applied cryptographic approach and its usability. By limiting available DVDs to CSS-compliant DVD players, manufacturers had to integrate new encoders that are secured by patent law regulations in their devices to maintain market position (Seitz, J, 2005).

- **Audio Watermarking**

Audio watermarking system based on Gammatone filterbank and coded-image has an expectative imperceptibility and robustness against typical attacks, especially random samples cropping and zeros inserting (Yiqing & Abdulla, 2010).

- **Text Watermarking**

Text watermarking is an emerging technique in the intersection of natural language processing and the technologies of security. Text watermarking aims at embedding additional information in the text itself with the goals of subliminal

communication and hidden information transport, of content and authorship authentication, and finally of enriching the text with metadata (Meral, et al, 2009).

According to the human perception, the digital watermarks can be divided into three different types as follows:

- Visible watermark

Visible watermark is a translucent overlaid into an image and is visible to the viewer. Visible watermarking is used to indicate ownership and for copyright protection (Singh, 2011). Visible watermarking is one of the ways to prevent illegal use from the unauthorized users by observing the visible logo by human eyes. Visible watermarking should be perceptually visible and robustness. Lossless property is more important in visible watermarking because greater distortion is may occur (Shu-Kei, et al, 2006).

- Invisible watermark

Invisible watermark is embedded into the data in such a way that the changes made to the pixel values are perceptually not noticed. Invisible watermark is used as evidence of ownership and to detect misappropriated images (Singh, 2011). Invisible watermarking is the digital data that cannot be perceived because of its different applications. Invisible watermarking, which is destroyed when the image is manipulated digitally in any way may be useful in proving authenticity of an image. If the watermark is still intact, then the image has not been "doctored." If the watermark has been destroyed, then the image has been tampered with. Invisible watermarking is very resistant to destruction under any image manipulation might be useful in verifying ownership of an image suspected of misappropriation. Digital detection of the watermark would indicate

the source of the image. Invisible watermarking, the watermark should be perceptually transparent and robustness. (Shu-Kei, et al. 2006)

- Dual watermark

Dual watermark is the combination of visible and invisible watermark. An invisible watermark is used as a backup for the visible watermark (Singh, 2011). First, insert the visible watermark in the original image and then an invisible watermark is added to the already visible watermarked image. The final watermarked image is the dual watermark image (Mohanty, 1999).

In order to give more strength to the security of digital image content, the previous researchers have made the evolution in the processing of the watermark. In general, the researchers divided the watermarking into two categories according to the needs of specific applications. They are robust watermark and fragile watermarks.

- Robust watermarking is mainly used to sign copyright information of the digital works, the embedded watermark can resist the common edit processing, image processing and lossy compression, the watermark is not destroyed after some attack and can still be detected to provide certification. (Jiang, X, 2010). In order to meet the robustness requirement of digital watermarking and to weather the various processing of the watermarked image, such as low pass filtering, compressing, adding noise, cropping and so on, the watermarking image should be placed on the important sense part of human visual system in the original image, namely corresponding to the low frequency part of the wavelet transform domain of the original image. (Zhang, et al, 2007).

- A fragile watermark is simply a mark likely to become undetectable after a work is modified in any way. Until now, we have consider fragility undesirable, seeking instead to design robust watermark that can survive many forms of distortion. However, fragility can be an advantage for authentication purposes. If a very fragile mark is detected in a work, it means the work has probably not been altered since the watermark was embedded (Cox, et al, 2008). Fragile watermarking is mainly used for integrity protection, which must be very sensitive to the changes of signal. We can determine whether the data has been tampered according to the state of fragile watermarking. (Jiang, X, 2010).

Digital watermarking can be divided into copyright protection watermarking, tampering tip watermarking, note anti-counterfeiting watermarking, and anonymous mark watermarking based on its purpose. Copyright protection watermarking means if the owners want others to see the mark of the image watermark, then the watermark can be seen after adding the watermark to the image, and the watermark still exists even if it is attacked. Tampering tip watermarking protects the integrity of the image content, labels the modified content and resists the usual lossy compression formats. Note anticounterfeiting watermarking is added to the building process of the paper notes and can be detected after printing, scanning, and other processes. Anonymous mark watermarking can hide important annotation of confidential data and restrict the illegal users to get confidential data (Jiang, X, 2010).

Digital watermarking can be divided into visual watermarking and blind watermarking according to the detection process. Visual watermarking needs the original data in the testing course, it has stronger robustness, but its application is

limited. Blind watermarking does not need original data, which has wide application field, but requires a higher watermark technology (Jiang, X, 2010).

2.2.3. Watermarking Requirement

Obviously, an implicit requirement for a blind watermark is that it is invisible to the naked eye and should look indistinguishable from the original. There are also other requirements for successful watermarking techniques. Literature lists the following common requirements: robustness, imperceptible to statistical methods, recovery with or without the original data, extraction or verification of a given watermark, security issues and use of keys, speed, and capacity (Brannock, et al, 2008).

Digital watermarks can be classified and measured on the basis of certain characteristics and properties that depend on the type of application (Seitz, J, 2005). These characteristics and properties include the difficulties of notice, the survival of common distortions and resistance to malicious attacks, the capacity of bit information, the coexistence with other watermarks, and the complexity of the watermarking method (Heileman, Pizano, & Abdallah, 1999). In general, they are described as fidelity, robustness, fragility, tamper resistance, data payload, complexity, and other restrictions. Digital watermarks must fulfill the following, often contradictory, requirements (Kutter & Hartung, 2000):

- **Robustness:** It may not be possible without knowledge of the procedure and the secret key to remove the watermark or to make it illegible. Robustness also means the resistance ability of the watermark information changes and

modifications made to the original file. As modifications, resizing, file compression, rotation, and common operations will be particularly considered. Above all, commonly used operations such as lossy compression (JPEG, MPEG) should not destroy the digital watermark (Hanjalic et al, 2000). Further examples are linear and nonlinear filters, lossy compression, contrast adjustment, gamma correction, re-coloring, re-sampling, scaling, rotation, small nonlinear deformations, noise adding, pixel permutations, and so forth. Robustness does not include attacks on the embedding scheme based on the knowledge of the algorithm or on the availability of the detector function. Robustness means resistance to common operations applied in the imaging, motion picture, or audio field (Fridrich, 1998).

- Nonperceptibility: It is important to recognize whether they brought bit sample of the watermark produces perceptible changes acoustically or optically. A perfect nonperceptible bit sample is present if data material marked with watermark and the original cannot be distinguished from each other. This classifier is based on the idea and properties of the human visual system (HVS) and human audio system (HAS). The watermark is nonperceptible or invisible if a normal human being is unable to distinguish between the original and the carrier.
- Nondetectable: The data material with the brought watermark information is not detectable if it is consistent with the origin data. In this case, an embedding algorithm could use for example, steganographically the noise components of the data source of a picture to hide the watermark information. Nondetectability cannot directly be linked to nonperceptibility that is based on the concepts of

human perceptions. Nondetectability is related to the data source and its components. It describes the consistency with the original data (Fridrich, 1998).

- **Security:** It is assumed that the attackers have full knowledge about the applied watermark procedure; however, no secret key would be known. Therefore, an attacker will try to manipulate the data to destroy the watermark, or again print and scan to win the original material without a copyright-protection note. The complexity is also connected with the security, that is, the algorithm for bringing in and reading watermark information should work with enough long keys to discourage the search for the appropriate secret key. However, for certain applications and persons, the watermark must be also detectable. The problem of secure key exchange emerges.
- **Complexity:** Complexity describes the expenditure to detect and encode the watermark information. A measurement technique could be the amount of time (Dittmann, 2000). It is recommended to design the watermarking procedure and algorithm as complex as possible so that different watermarks can be integrated. Thus, “trial-and-error” attacks can be avoided (Voyatzis, Nikolaidis, & Pitas, 1998).
- **Capacity:** Capacity refers to the amount of information that can be stored in a data source. In using digital watermarking for simple copy control applications, a capacity of one bit (one = allow/zero = deny) seems to be sufficient. On the other hand, intellectual property applications require about 60 to 70 bit information capacity to store data about copyright, author, limitations, or International Standard Book Number (ISBN), International Standard Recording Code (ISRC), or OEM numbers.

2.2.4. Embedding / Encoding

Selecting a 512x512 image as the original image, a 64x64 image as watermarking; we do researches on the digital image watermarking technique by means of two-dimensional discrete wavelet transform. The intention to uniformly scramble the original watermarking image is to enhance the difficulty which unauthorized persons detect and extract the image watermarking, or which vicious attackers attack the watermarked image.

In order to combine a watermark with a digital document, for example, images, you need an image (I), a signature ($S = s_1, s_2, s_n$) that contains the watermarking information, and an encoding algorithm (E) to create a watermarked image (I'). The encoder takes the signature and the cover document, and generates the watermarked image, that is described as a function: $E(I, S) = I'$. In this case, secret or public keys and other parameters can be used to extend the watermarking encoder. (Seitz, J, 2005)

The input text is subjected to a number of level decomposition using the RC4 Algorithm. The perceptually important coefficients of each subband are detected and classified into two groups with respect to a threshold value. Also, another group of coefficients is formulated containing the region around the edges. This is accomplished using a morphological dilation operation with a structuring element of 9×9 .

2.2.5. Extracting

The watermarked image is also respectively decomposed at three levels of the watermarked image with the corresponding ones of the original image respectively at different positions and different levels, extracting the corresponding wavelet transform coefficients of the embedded watermarking image.

The watermark is extracted using a decoder function. In this case, the decoder D loads the watermarked, normal or corrupted image $I(w)$, and extracts the hidden signature S . Using non-blind watermarking techniques, the decoder D loads an additional image I , which is often the original image, to extract the watermarking information by correlation. Such methods are more robust against counterfeit attacks. The process can be described as $D(I, I(w)) = S$ (Seitz, J, 2005). The extracted watermarking image is the scrambled watermarking image and is not the original watermarking image. Therefore, the extracted watermarking image must be reverted to the original watermarking image by inverse Arnold transform.

2.3. Watermarking Technique

2.3.1. Least Significant Bit (LSB) Method

LSB algorithm is the first to propose the international digital watermarking algorithm, is a typical spatial domain information hiding algorithm. LSB algorithm is the core of selected pixels of the most significant bits in turn replaced by secret information. Information were Embedded in the image when you first select a subset of vector pixel, then the LSB carrier exchange with secret information. When extracting the first to find out the selected image pixel vector sequence, LSB will be

lined up and secret information then reconstructed. The main advantage of LSB algorithm can achieve high capacity and good invisibility. But the algorithm is easy to find and get a third party, destroyed the image of various operations, such as compression, cropping and so on, will be affected the reliability of the algorithm. In order to enhance the performance of the algorithm, a variety of improved methods, such as the use of pseudo-random sequence of "random" in order to modify the image of the LSB; in the case of the key must be used in order to get the correct sequence of embedding, and so on (Zhang & Li, 2009).

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image. Unfortunately, it is vulnerable to even a slight image manipulation. Converting an image from a format like GIF or BMP, which reconstructs the original message exactly (lossless compression) to a JPEG, which does not (lossy compression), and then back could destroy the information hidden in the LSBs (Meena, et al, 2011).

Least Significant Bit (LSB) watermarking describes a straightforward and basic way to integrate watermark information in digital documents. Considering a basic grayscale image, the pixel and its values can be sliced up into significant and irrelevant levels. Because the significant levels merely represent a digital noise pattern, it could be easily used for digital watermarking. In changing selected pixel values of the noise pattern using a special or key-based algorithm, the watermarking information can be easily integrated (Hanjalic et al, 2000). LSB is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. The LSB is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digit further to the right.

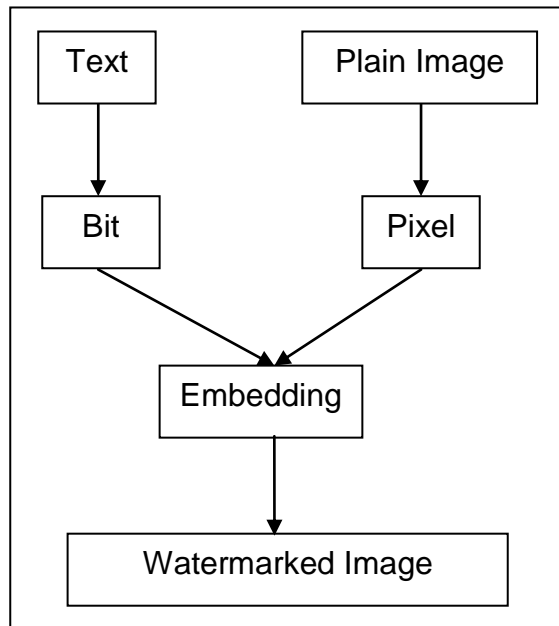


Figure 2.4: LSB watermark system (Embedding) (Hanjalic et al, 2000)

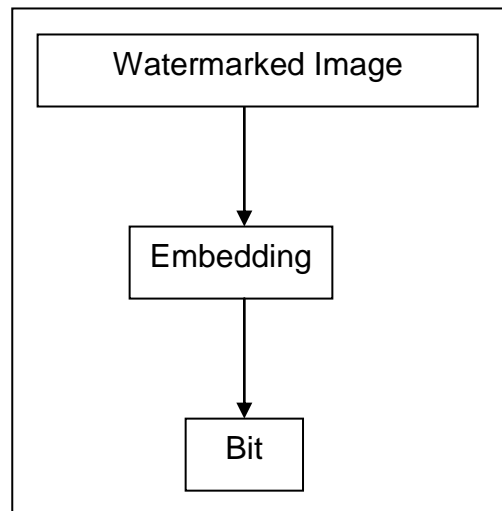


Figure 2.5: LSB watermark system (Extracting) (Hanjalic et al, 2000)

Figure 2.4 and 2.5 represents the overall system of suggested model. In figure 2.4, the embedding scheme explains the changing process of each pixel from a plain image into a raster data. In the first phase, LSB method will read the pixel of image in RGB into a raster data that represent using binary. The second phase, LSB

read the text message that represent by characters into ASCII in decimal. Then the ASCII number will be converted to bits message in binary. Change every lowest bit into bit message. If lowest bit equal to bit message, then the raster data will not be changed. If lowest bit lower than bit message, then raster data will be add by 1. But if lowest bit greater than bit message, then raster data will be subtract by 1. To read the new pixel, LSB will change the raster data to RGB then convert it to pixel. (In figure 2.5)

Least Significant Bit (LSB) method is the common and simplest approach of watermarking techniques that embed the bits of the message directly into the least-significant bit plane of the cover image in a deterministic sequence. A digital image consists of a matrix of color and intensity values. In a typical gray scale image, 8 bits/pixel are used. In a typical full color image, there are 24 bits/pixel, 8 bits assigned to each color components.

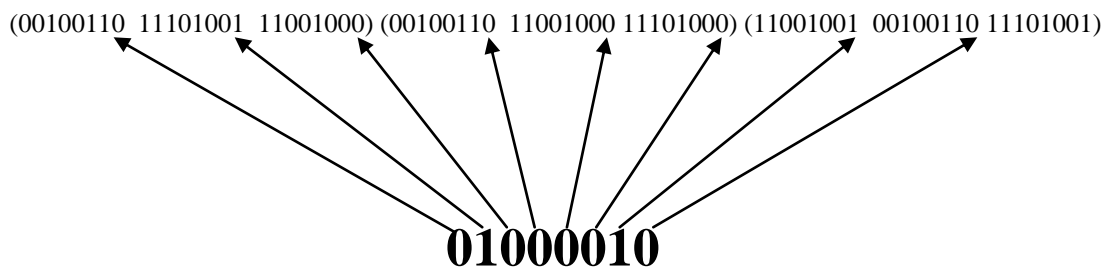
To hide an image in the LSBs of each byte of 24-bit image, 3 bits of hidden messages can be stored in each pixel. A larger amount of information can be stored if the message is being compressed first before embedding it into the image. For example, the letter B can be hidden in three pixels (assuming no compression has been made) Neil F.J & Sushil J. 1998). The original raster data for 3 pixels (9 bytes) may be:

```

00100111    11101001    11001000
00100111    11001000    11101001
11001000    00100111    11101001

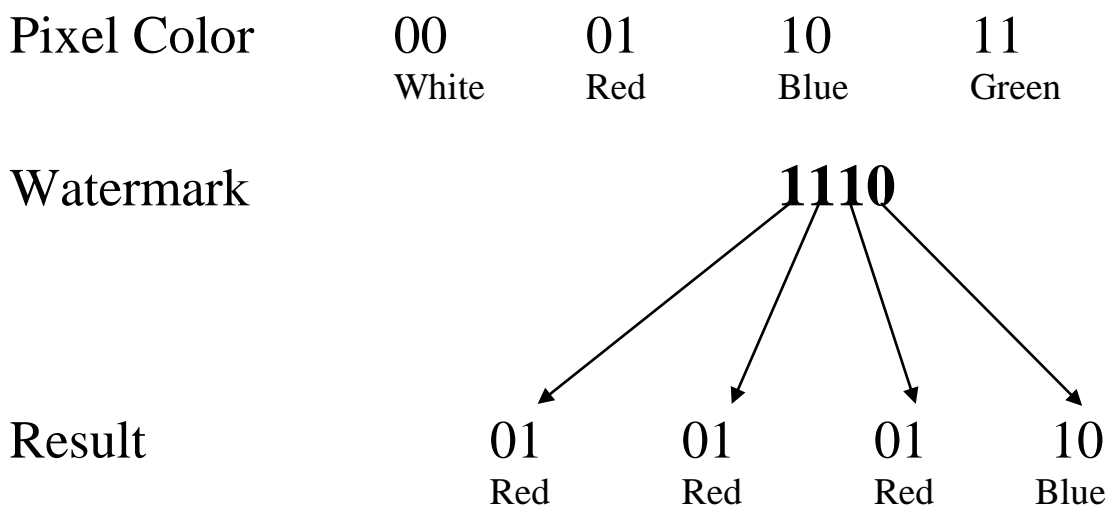
```


The binary value for B is 01000010. Inserting the binary value for B in the three pixels would result in :



The arrowed bits are the bits that have been changed in the 8 bytes used. On average, the LSB only requires half of the bits in an image to be changed. Data can be hidden in the least and second least significant bits and still the human eye cannot detect the data.

The 8-bit images are not quite good for LSB manipulation because of color limitations. To implement LSB in 8-bit images, first the cover must be carefully selected so that watermark image will not broadcast the existence of an embedded message. When message is inserted into LSBs of the raster data, the pointers to the color entries in the pixel color are changed. The example of LSB method for 8-bit image is as follows:



The entries of simple four-color pixel of white, red, blue, and green has the corresponding pixel position entries of 0 (**00**), 1 (**01**), and 3 (**11**), respectively. Hiding number 14 valued 1110 changed the raster data to **01 01 01 10**, which is red, red, red, blue. These gross changes in the image are visible and it clearly shows the weaknesses of the 8-bit images. On the other hand, there is little visible difference noticed between adjacent gray values (Neil F.J & Sushil J, 1998).

The advantage of the LSB method is its simplicity and many techniques use this method. LSB is also allows high perceptual transparency. Modulating the least significant bit also does not result in a human perceptible difference because the amplitude of the change is small. However, there are many weaknesses when robustness, tamper resistance, and other security issues are considered. LSB encoding is very sensitive to any kind of filtering or manipulation of the watermark image.

2.3.2. Spatial Domain Watermarking

Digital watermarking techniques in the spatial domain use the values of the color channels, luminance or brightness signals of a digital image (Dittmann, 2000). One straightforward and rapid technique is based on the principle of generating a pseudo-generated noise pattern and integrates it into specific chrominance or luminance pixel values (Darmstaedter, et al, 1998).

Such pseudo-random noise patterns consist of black (1), white (-1), and neutral values (0). The pseudo noise is generated with a “secret” key and algorithm. Additionally, the process could be adjusted to the image components or feature vectors to achieve a higher level of invisibility. In general, the watermark $W(x, y)$ is integrated into the image components $I(x, y)$ by a factor that allows amplification of the watermarking values in order to obtain the best results.

$$I W(x, y) = I(x, y) + k * W(x, y)$$

The detection of the watermark is based on the principles of correlation. In this case, a specific detector compares the watermarked image $IW(x, y)$ with the original image and automatically decides, based on a specific correlation level, whether a watermark exists (Hanjalic et al., 2000). Such techniques particularly enable the integration of one-bit watermark information. In integrating more information, various techniques have been invented. Such methods have the possibility to save up to 500 bits in one 512x512 image (Hanjalic et al., 2000). In order to clarify this main procedure, the original image will be subdivided into small blocks. Now, the selected blocks are watermarked or not and produce a bit sequence in the detection process. In this case, the watermarking detector scans the image and generates the bit sequence according to a specified correlation level. Using CRC method could improve the error tendency.

2.3.3. Frequency Domain Watermarking

The basic principles of adding or changing components of digital images and other digital documents can be transferred to other value domains. In order to integrate watermark information into frequency components, the document has to be

transformed into its frequency components using discrete cosine, discrete Fourier, or Hadamard transformations (Dittmann, 2000; Hanjalic et al., 2000). As such transformations are used in lossy compression techniques, for example, MPEG and JPEG, the watermark appears to be very resistant to the usual attacks. Furthermore, in integrating watermarks in the most important frequency components improve security and resistance, because every change significantly reduces the quality of the image (Hanjalic et al., 2000). Therefore, it is important to identify the coefficients of the transformation that are less infected by the attack method. In most cases digital watermarks are integrated into the mid-band frequencies. Research has determined a specific sensibility of high-band frequencies against filter operations, lossy compression, and noise insertion whereas manipulating low frequencies seems to produce visible artifacts anytime (Hanjalic et al., 2000).

2.3.4. Spread Spectrum Watermarking

Spread spectrum techniques used in digital watermarking is borrowed from the communication field. The basic idea of spread spectrum is to spread the data across a large frequency band. In the case of audio, it is the entire audible spectrum; in the case of images, it is the whole visible spectrum. Spread spectrum is a military technology designed to handle interferences and disturbances. In most cases, signals that represent the information are modulated at low intensity across the source bandwidth. Spread spectrum communication is used in radar, navigation, and communication applications. The information is weaved into the source material using a secret key or an embedding procedure (Hanjalic et al., 2000).

2.4. Summary

This chapter has discussed concept and definition of the information hiding and other issues regarding the watermark concept. An overview of history watermarking, watermarking development, and watermarking requirement has been discussed also. Finally this chapter outlined the perception and previous related work, which eventually helped in determining the watermarking technique based on certain approaches. The next chapter will discuss on research methodology.

CHAPTER III

RESEARCH METHODOLOGY

The research methodology for this study will be described in this chapter. A research methodology defines what the activity of research is, how to proceed, how to measure progress, and what constitutes success. Research methodology consists of the combination of the process, methods and tools that are used in conducting a research. What compose research in general are its objectives and the methods. This project has been conducted by applying the design research methodology. Further of this chapter will elaborate the definition of desing research methodology that has been chosen as the guide to conduct this study.

3.1. Research Definition

Research can be generally defined as an activity which contributes to the understanding of any phenomenon (Kuhn, 1996, Lakatos, 1978). For the design research aspect, most part of the phenomenon may be invented as opposed to the naturally occurring. The phenomenon can be defined as set of behavior that will attract the interest of researcher or maybe the entire research community. A set of activities that the research community considered appropriate to the production of understanding to the knowledge are its methods or techniques.

3.2. Design Definition

Design means to plan or protocol for carrying out or accomplishing something (Merriam-Webster, 2003). Design usually deals with creating something that is new and does not exist in nature. In the early years design is only distinguished in the professions of sciences such as architecture, medicine, and law domain, which are centrally concerned about the process of design (Simon, 1996). However, in these recent days, natural sciences almost drove out the design into all professions including management science and computer science. Design is a form of techniques and methods for performing a mapping that will show how something is done that will satisfy a certain set of functional requirements.

3.3. Research Design Technique

In this research, the techniques used for Digital Image Watermarking using LSB method is the Waterfall Model. The steps enumerated of the waterfall model:

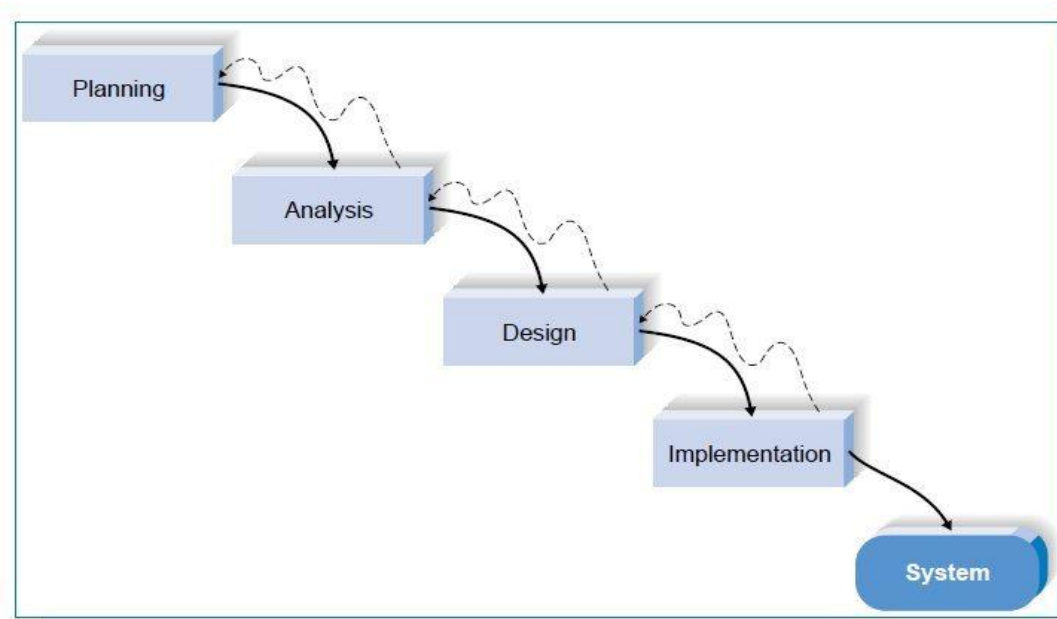


Figure 3.1: Steps in Waterfall Model (Dennis, et al, 2005)

3.3.1. Planning Phase

The first step here is the planning phase. The basic objective in this step of digital watermarking project is to protect a digital image by designing an application for hiding information in the form of text messages. Before this can be done, it is necessary that a kind of feasibility study is conducted in such that such a study will help in prolonging the study, making it a reality and sustain the future impression on it. This feasibility study will be useful in ensuring that the task involved in the work would have been ascertained before it is started so that it will not be abandoned along the line.

During this phase, I have done planning which aimed to define the requirements from many different stakeholders and drive them into a series of software releases. This process will need to account for a variety of problem and constraints. In this process, researcher can make a Gantt chart to illustrate a project schedule. A Gantt chart is a simple tool which represents time as a bar or a line on a chart. It gives an at-a-glance perspective on the names and timing of tasks, as well as their current status. A Gantt chart consists of a list of project activities to reflect activity duration (Basu, R. 2004). From a Gantt chart, it can show us the due date of each activities in digital watermarking project phase because it explain the start and finish dates of the terminal elements and summary elements of a project.

Here is the Gantt chart of this project:

Table 3.1: Gantt chart

Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Literature Review	1/3/2011 – 9/6/2011														
Proposal Submission			20/3												
Data Collection															
Analysis															
Design															
Implementation															
Draft Report Submission															
Final Presentation															
Correction and Final Report Submission															

3.3.2. Analysis Phase

The second and another vital stage here is the analysis phase. This is a stage where there is fusion of ideas and practical implementation. In order to ascertain which methods will be bet applicable, there is use of multiple methods which will be tested, observed in investigating and getting feedback from the data collected.

In this phase, I try to do the list of requirement in order to capture the intended behavior of the system. The list of requirement behavior may be expressed as services, tasks or functions the system is required to perform. List of requirement

can explain the system behaviors and functions of the digital image watermarking system using LSB method.

Each function or behavior in list of requirement has their own priority. Those priorities are mandatory, desirable, and optional. Mandatory is the most important priority that the system must do. Based on Oxford Dictionary, mandatory is required by law or rules, mandate, compulsory to be provided. it synonyms with obligatory, compulsory, binding, required, requisite. If the system does not have the mandatory, means the system is useless or cannot be work. Based on Oxford Dictionary, desirable means wished for being attractive, useful or desire to be developed. When a developer wants to develop any software, they can make any desirable function of item. It means the developer follow what customer wants to have and what customer needs. Desirable priority will complete the functions provided by the system. Based on Oxford Dictionary Optional means possible but not compulsory, based on personal choice, or we can conclude as what system may have and may not have. Optional priority will not affect the function of the system (Dictionaries, O, 2009).

List of requirement can be specified as functional list of requirement and non functional requirement. In functional requirement, it explains software features of the system. Here, the software features that need to be developed are the encryption and decryption of the system. In order to encrypt and decrypt the hidden message into a picture, then the system need to have the cryptic code of the message. Cryptic code later will be used as its key to get the message from an image. In a non functional requirement, it explains the non features function of the system. This non functional requirement explains about what the system should do in order to deal with the

system background. For example: if there is a systems breakdown, it should behave perfectly normal when reloaded again.

Here is list of requirement table of LSBWatermark System. Listed below are the functional requirements and non-functional requirement of the system. In the priority column, the following short hands are used:

- M – mandatory requirements (something the system must do)
- D – desirable requirements (something the system preferably should do)
- O – optional requirements (something the system may do)

Table 3.2: Functional Requirements

No.	Requirement ID	Requirement Description	Priority
	Watermark_01	Software Features	
1.	Watermark_01_01	User can enter cryptic code	M
2.	Watermark_01_02	User can open image	D
3.	Watermark_01_03	User can insert text message	D
4.	Watermark_01_04	User can generate image result	M
5	Watermark_01_05	User can save image result	D

Table 3.3: Non-Functional Requirements

No.	Requirement ID	Requirement Description	Priority
	Watermark_02	Reliability issues	
1.	Watermark_02_01	For a single user, the system should crash no more than once per 10 hours.	M
2.	Watermark_02_02	If the systems crash, it should behave perfectly normal when reloaded again.	M

3.3.3. Design Phase

This third stage is design phase; it is referred to software and system design activity. In this activity, researcher transform all of the informal ideas into detailed implementation descriptions. The outcome of this activity is technical drawings in product development. According to Blessing and Chakrabarti (2009) software and system design stage has six, iteratively related, steps:

- a. Architectural design: The sub-systems making up the system and their relationships are identified and documented.
- b. Abstract specification: For each sub-system, an abstract software specification is produced of the services it provides and the operation constraints.
- c. Interface design: The interface of each sub-system with other sub-systems is designed and documented.
- d. Component design: Services are allocated to different components and the interfaces of these components are designed.
- e. Data structure design: The data structures used in the system implementation are designed in detail and specified.
- f. Algorithm design: The algorithms used to provide services are designed in detail and specified.

In this design phase researcher did use case diagrams, flow charts, activity diagrams, sequence diagrams, collaboration diagrams, class diagrams, and software interface. In detail each diagram are explained below:

- **Use Case Diagram**

The basic objective of Use Case Diagram is to explain a graphical representation of all functionalities provided by the system in terms of actor. In this phase user is required to enter cryptic code. This cryptic code is needed later when the information will look back on text messages that have been inserted earlier. User also needs to insert an image as a raw data, and insert text message as its modifiers. After that, the system will encode the text message to the image. Finally, user can save the result which is a watermarked image.

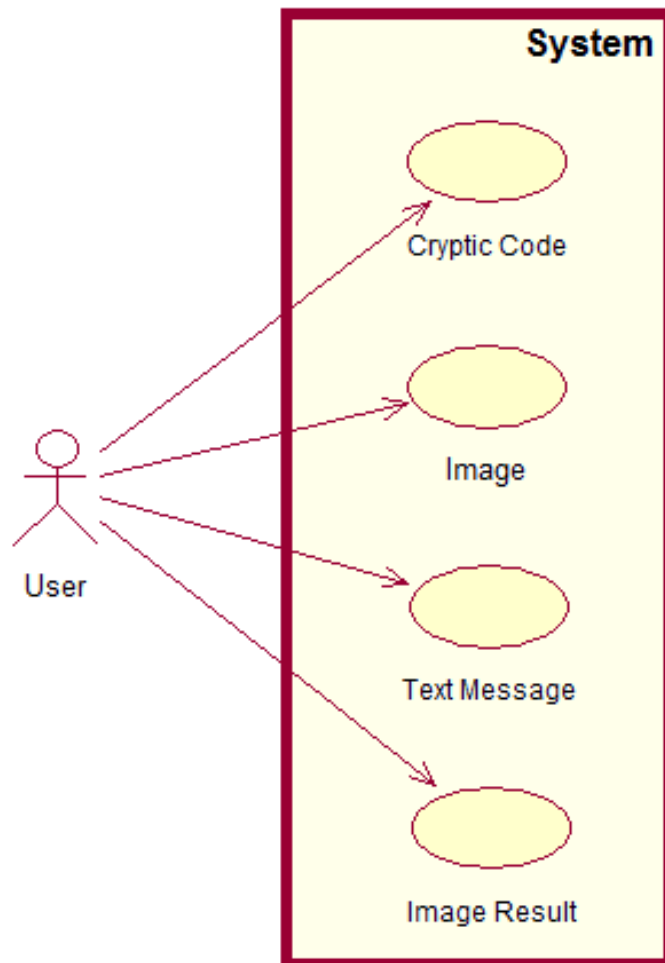


Figure 3.2: Use case diagram for encryption

In this use case diagram explains the basic step of system which is including encryption and decryption. In encryption, user need to enter cryptic code and plain image before they hide the text message, then create watermarked image. For decryption, they just need to enter the cryptic code and watermarked image, system will generate text message automatically.

- **Activity Diagram & Flow chart**

This part explains the whole process of software in general. This process start by key in the cryptic code then user inserts a plain image (original). System will verify the type of image, if the image type is wrong, user need to insert another image again, but if the image type is correct, user continue insert text message as its modifiers. Here, the system will verify the size of text inserted to the image. If the text size larger than the image size, the software will ask us to re-enter text. The text size should smaller than image size. After that, the system will encode the text message to the image. Finally, user can save the result which is a watermarked image

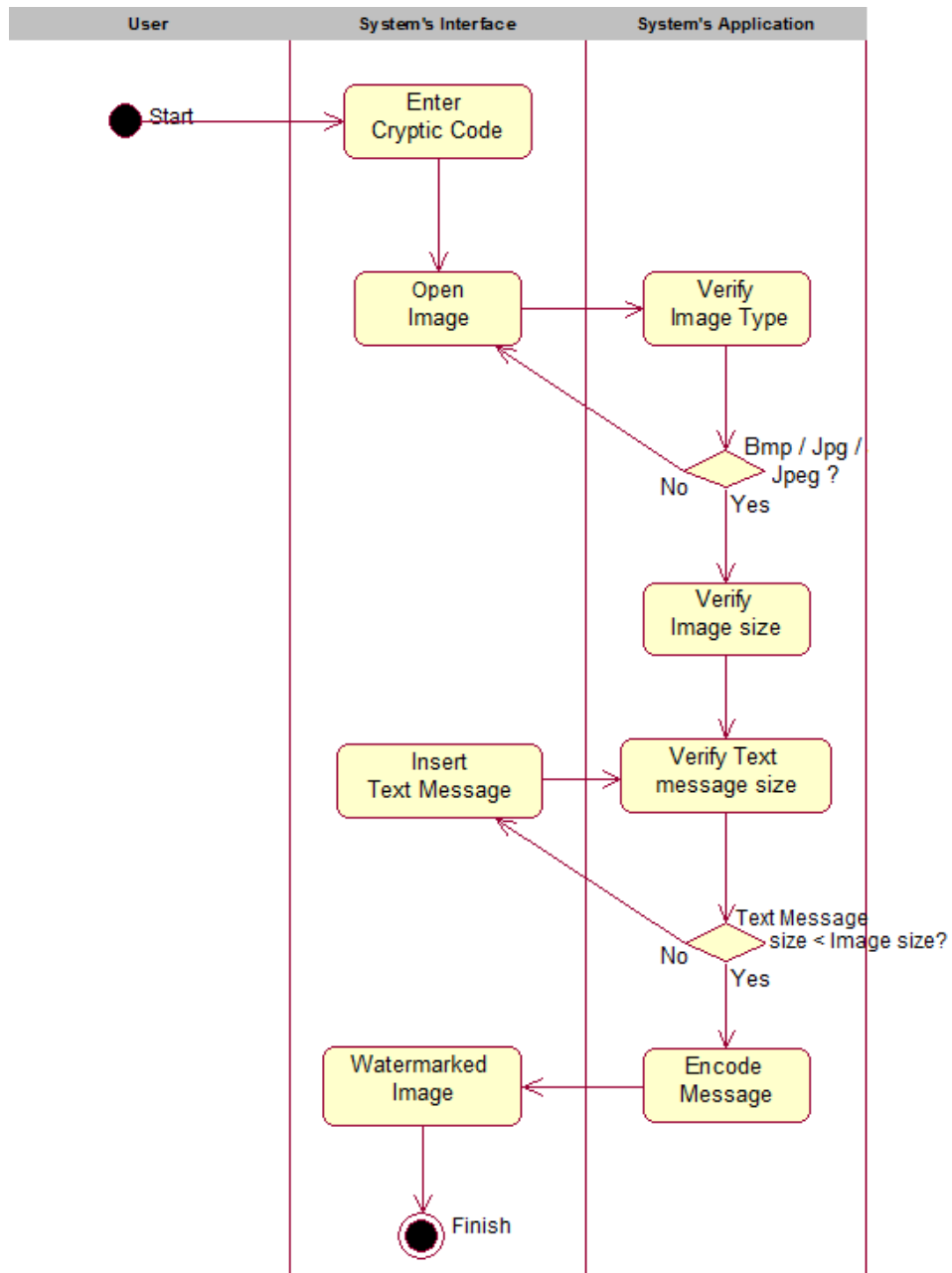


Figure 3.3: Activity diagram for encryption

This activity diagram explains about the encryption step in the system. To start the process, user need to enter a cryptic code, then they will allowed to open any plain image which are .bmp, .jpg, or .jpeg; then the system will verify it. If the image type is correct, then it will continue to image size verification. After image size is known, the user can insert the text message. They can type it or open any documents which are .txt or .rtf; then the system will verify the size of it. If the text size is

greater than image size, the program will not execute and give an error message. Otherwise if the text message is smaller than image size, the system can process it and generate a watermarked image.

Here is the activity diagram of decryption digital image watermark:

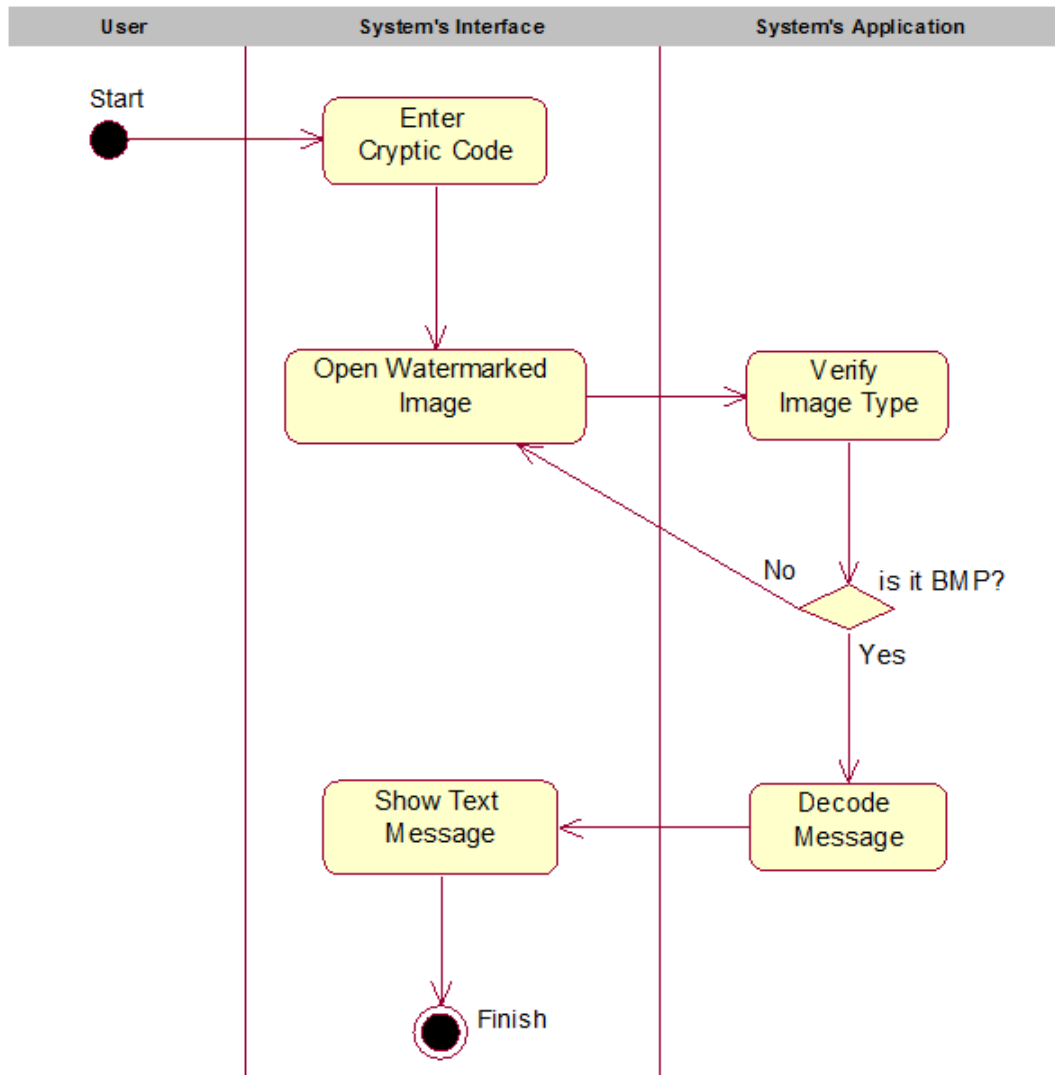


Figure 3.4: Activity diagram for decryption

This activity diagram explains about the decryption step in the system. It begin by enter a cryptic code, then they will allowed to open any watermarked image which is .bmp; then the system will verify it. If the image type is correct, then it will

continue to text and image size verification. If the text size is greater than image size, the program will not execute and give an error message. Otherwise if the text message is smaller than image size, the system can process it and display the hidden message inside it.

An activity diagram resembles a horizontal flowchart that shows the actions and events as they occur. Activity diagrams show the order in which the actions take place and identify the outcomes. Figure 3.4 shows an activity diagram for a decryption process in the system. Notice that the user initiates the activity by inserting cryptic code and a watermarked image. Activity diagrams also can display multiple use cases in the form of a grid, where classes are shown as vertical bars and actions appear as horizontal arrows.

- **Flow Chart**

Flowchart is a diagram with graphical symbols which express a different type of program. Flowchart or algorithm can provide a tool to facilitate the design of a program logic flow sequence, making it easy to track the source of program errors, and a tool to explain the logic of the program (Ambler, S, 2004; Marrer, G, 2004). The following images are an encryption and a decryption flowchart which represents a digital watermark program:

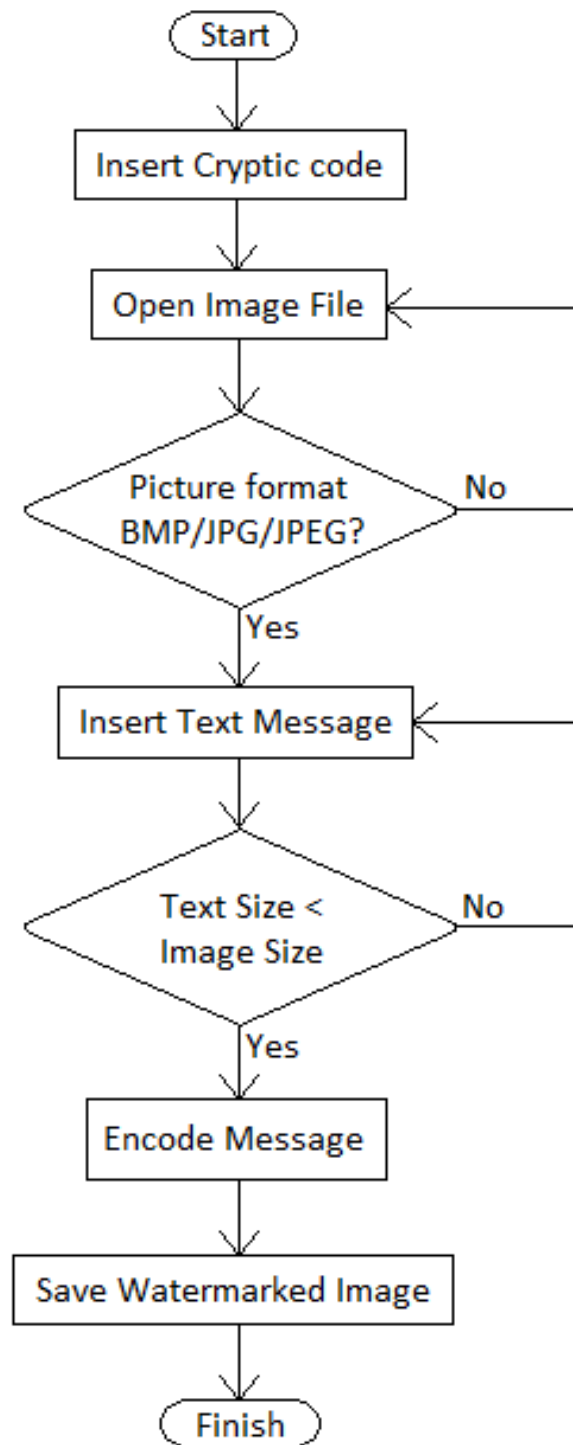


Figure 3.5: Flow Chart for encryption

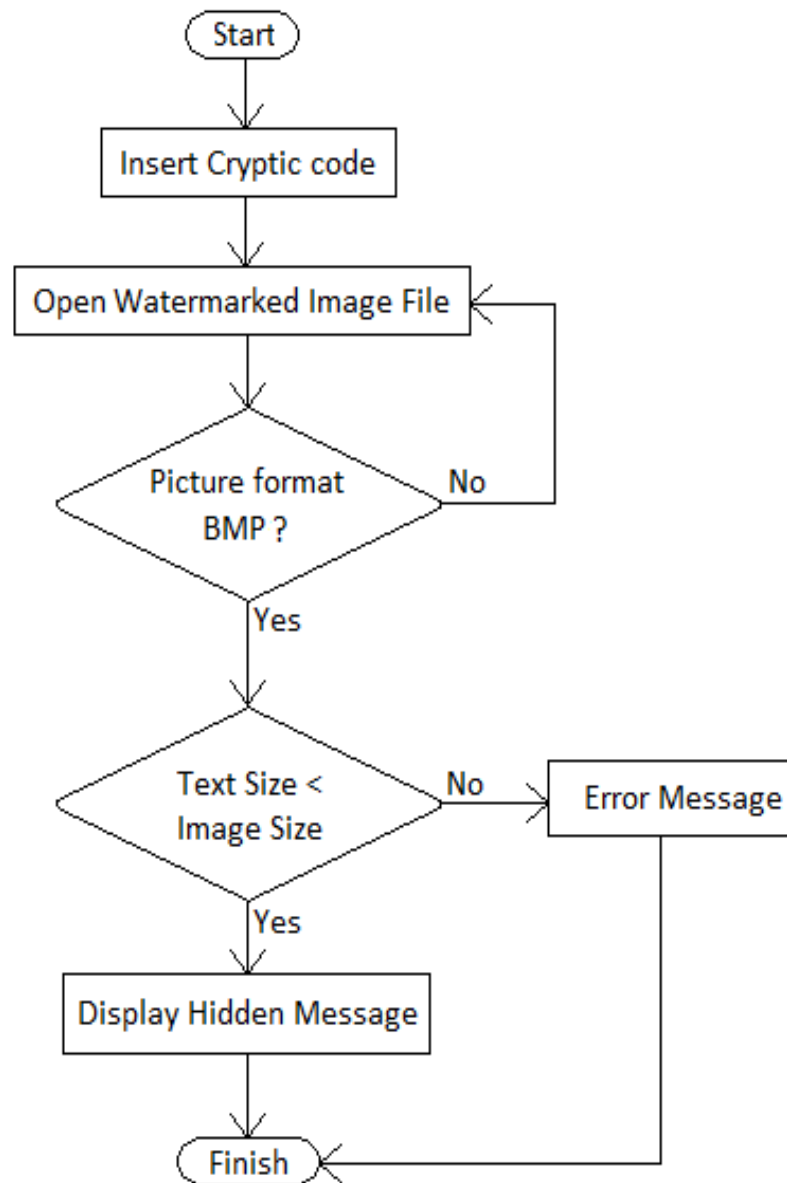


Figure 3.6: Flow Chart for decryption

- **Sequence Diagram & Colaboration Diagram**

A sequence diagram is a dynamic model of a use case, showing the interaction among classes during a specified time period. A sequence diagram graphically documents the use case by showing the classes, the messages, and the timing of the messages. Sequence diagrams include symbols that represent classes, lifelines, messages, and focuses (Shelly, et al, 2009).

A collaboration diagram is one of two kinds of interaction diagrams used in UML. These interaction diagrams provide a transition between the requirements analysis work performed in association with use case, state, and activity diagrams and the detailed design of the software system embodied in the state and class, component, and node diagrams. Thus, the interaction diagrams offer what could be called a "preliminary design environment." The collaboration diagram illustrates the structural relationships among objects and the messages that must interconnect them to accomplish some activity (Grady, 2006).

Encryption Sequence Diagram

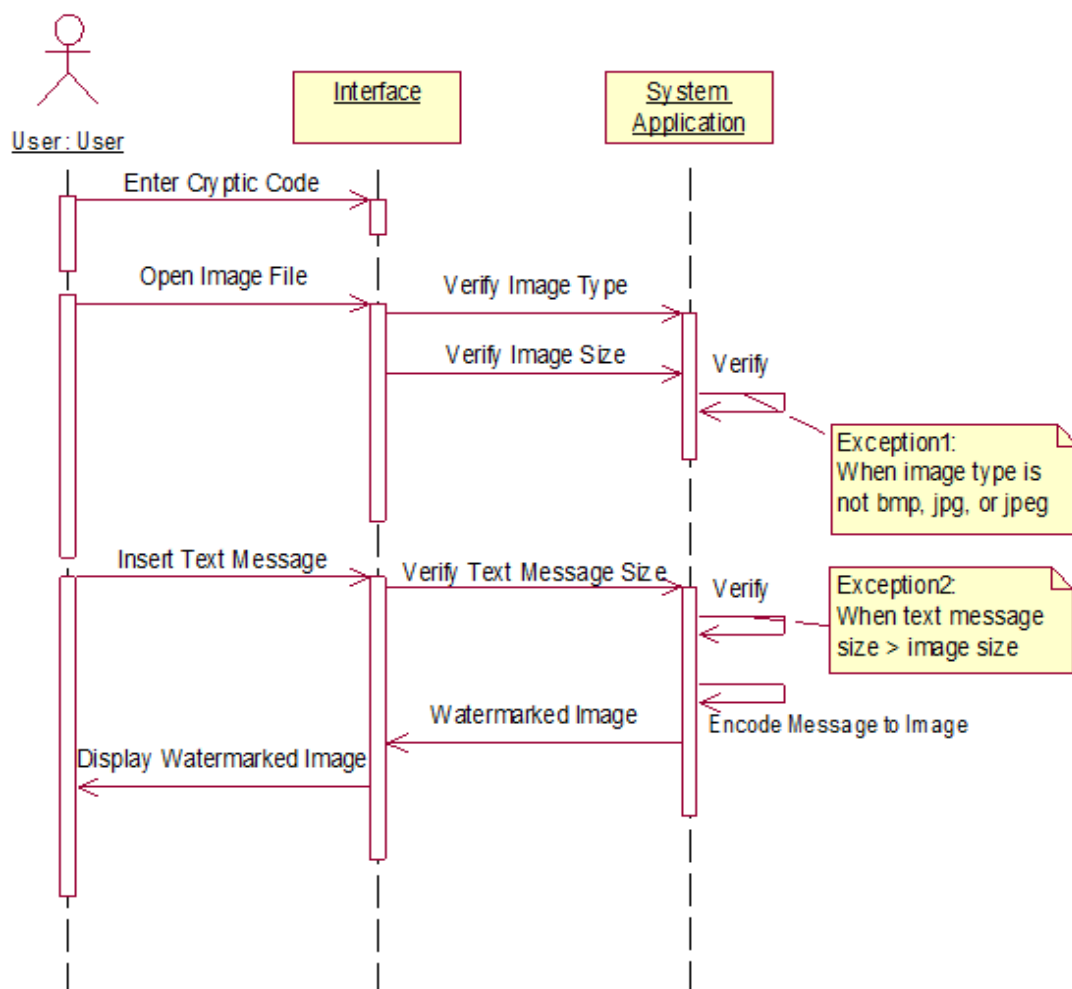


Figure 3.7: Basic Flow that show overall encryption step

Encryption Collaboration Diagram

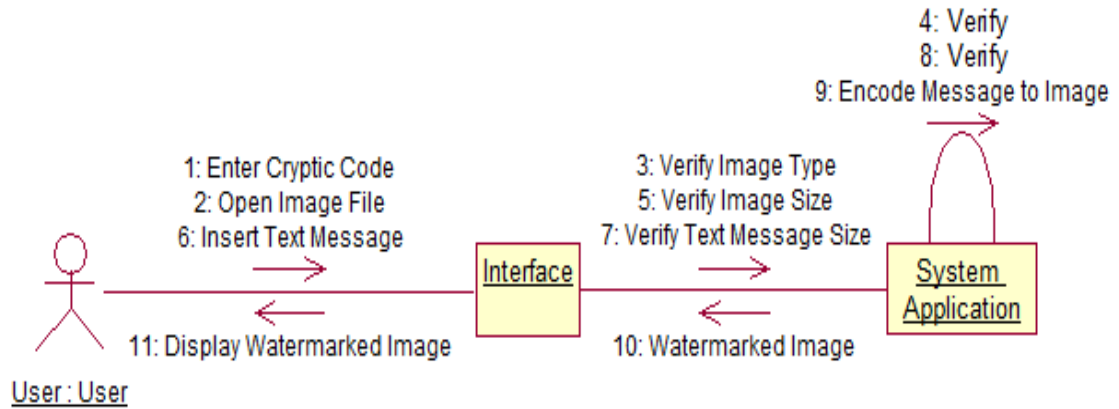


Figure 3.8: Collaboration Diagram show overall encryption

These sequence and collaboration diagrams are explaining encryption steps which start by user entering the cryptic code and open the plain image. The system will verify image type and size, if it is wrong, it will go to exception 1, but, when it's correct, then user can enter text message that want to be hidden, and then the system will verify the text message size. If the text message is too large it will go to exception 2, otherwise if text message smaller than image size, system can continue to create a watermarked image.

Decryption Sequence Diagram

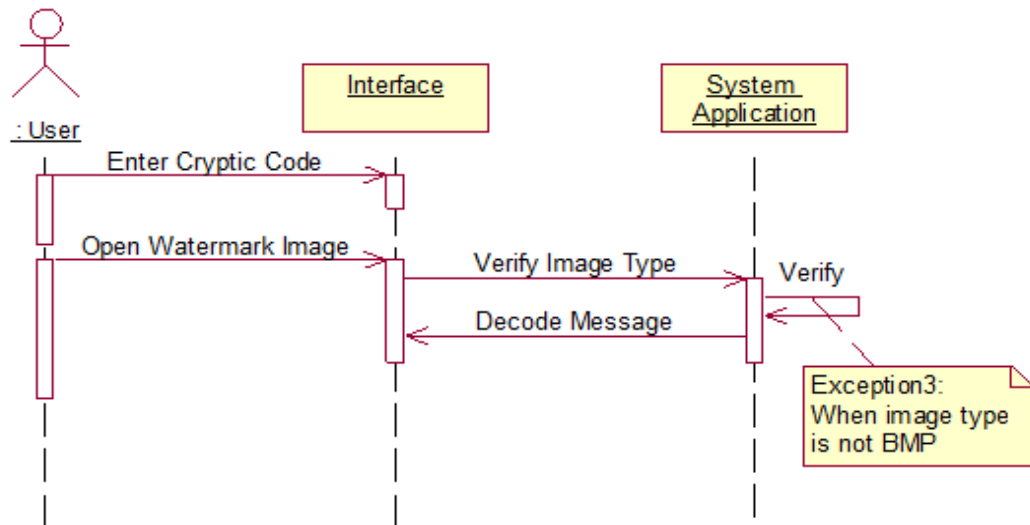


Figure 3.9: Sequence diagram show overall decryption

Decryption Collaboration Diagram

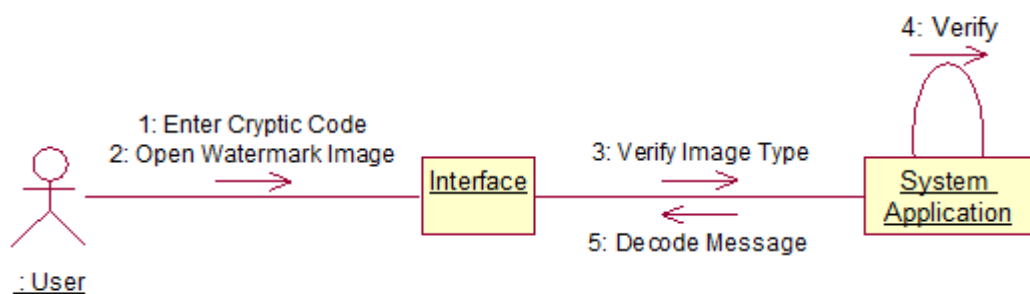


Figure 3.10: Collaboration Diagram show overall decryption

These sequence and collaboration diagrams are explaining decryption steps which start by user entering the cryptic code and open the watermarked image. The system will verify image type, if it is wrong, it will go to exception 3, but, when it's correct, then system will verify image size and hidden text message size inside it. If text message smaller than image size, system can continue to read the text message which hidden in watermarked image.

Exceptions

Exception 1 : Invalid Plain Image Type

Sequence Diagram

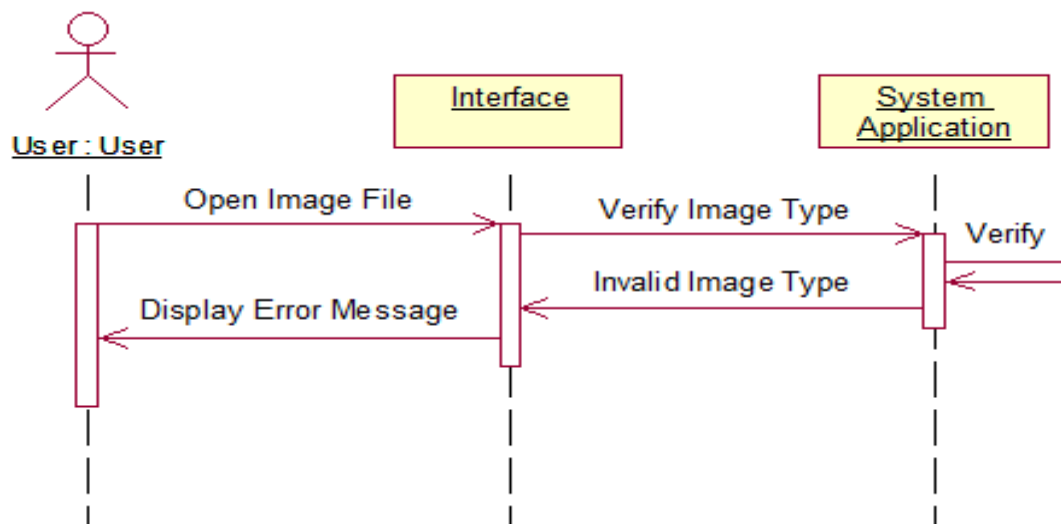


Figure 3.11: Sequence Diagram of Exception Flow

Collaboration Diagram

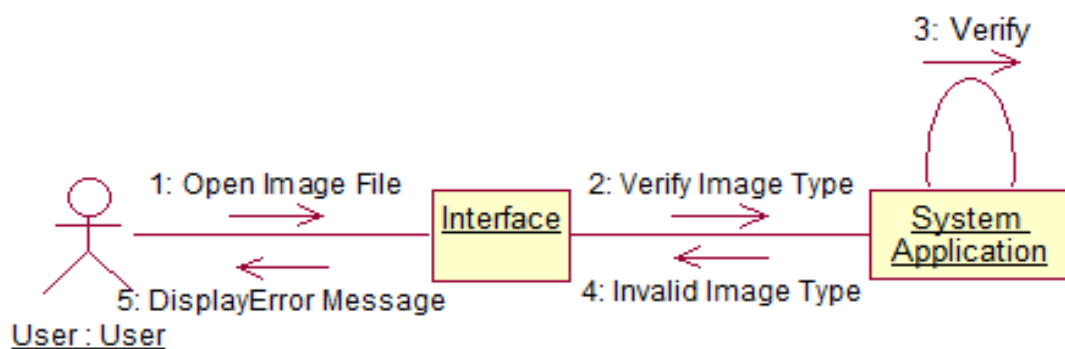


Figure 3.12: Collaboration Diagram of Invalid Image Exception

This is an exception of invalid plain image size. If the system receives a wrong plain image type, the system will not execute and display error message.

Exception 2: Too Large Text Message Size

Sequence Diagram

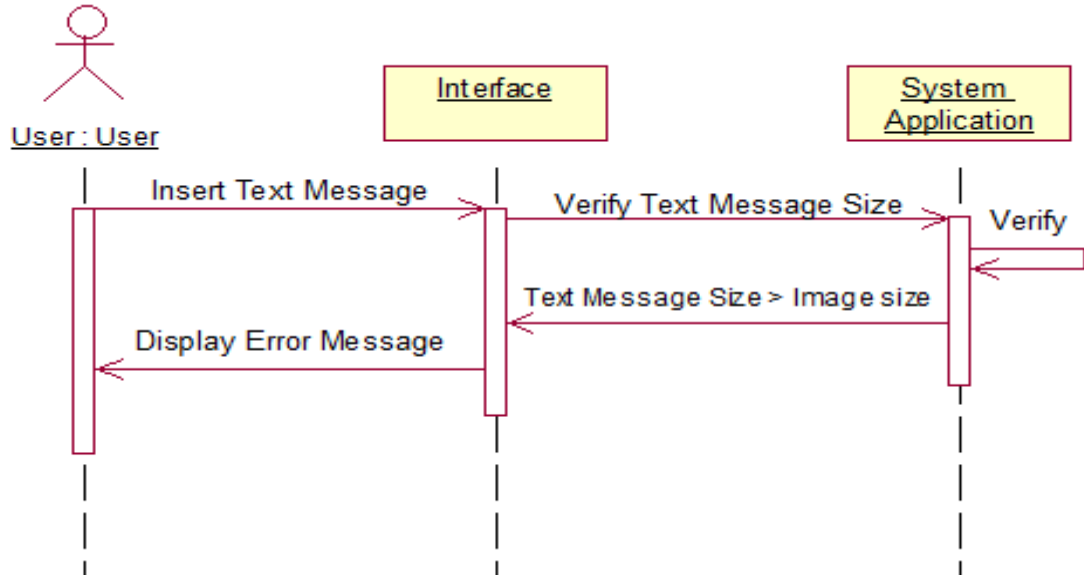


Figure 3.13: Sequence Diagram of Large Text Message Size

Collaboration Diagram

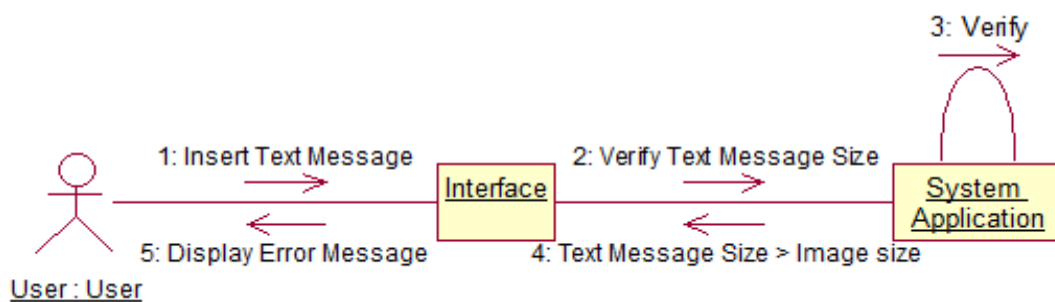


Figure 3.14: Collaboration Diagram of Large Text Message Size

This is an exception of too larger text message size. If the system receives a too larger text message size or message size greater than plain image size, the system will not execute and display error message.

Exception 3: Watermarked Image Type not BMP

Sequence Diagram

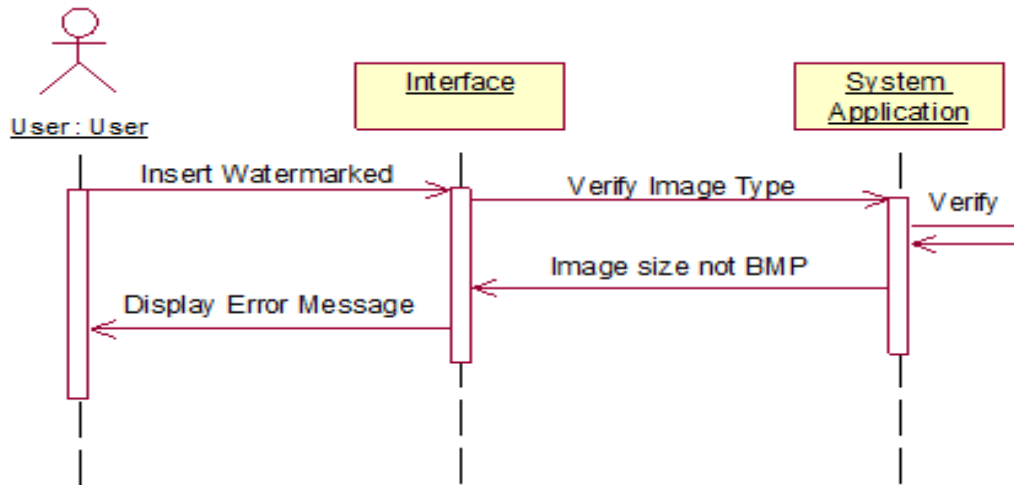


Figure 3.15: Sequence Diagram of watermarked image type not BMP

Collaboration Diagram

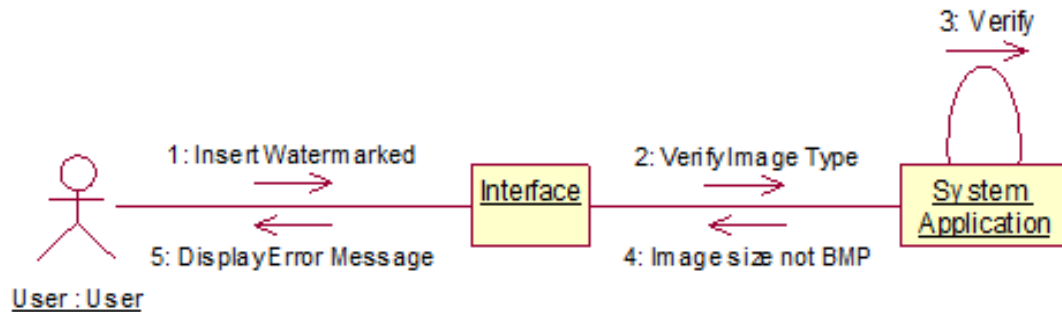


Figure 3.16: Collaboration Diagram of watermarked image type not BMP

This is an exception of invalid watermarked image size. If the system receives a wrong watermarked type, the system will not execute and display error message.

- **UML Class Diagram**

In addition of sequence and collaboration objects above, it is useful to create a static view of the class definitions with a design class diagram. This illustrates the attributes and methods of the classes (Larman, C, 2002).

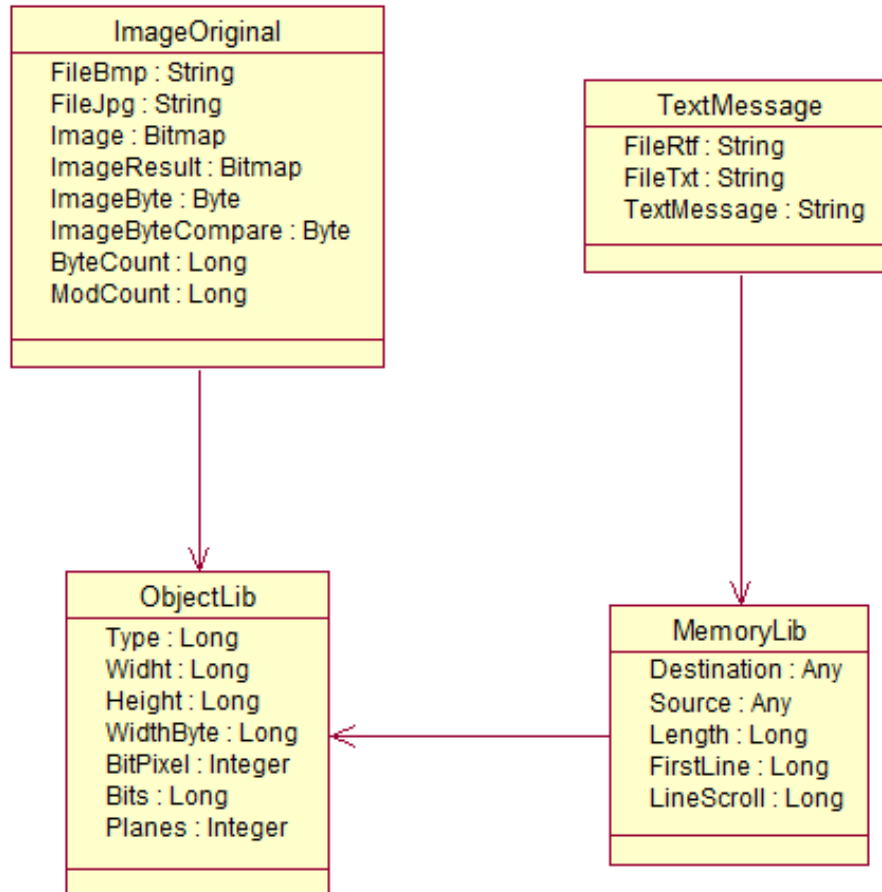


Figure 3.17: Class Diagram

This class diagram explains what system has inside the coding of digital watermarking system. Based on this class diagram, author can start write the program in a structured manner in accordance with object oriented programming.

- **Software Interface Design**

After all the design processes created, the interface design is needed to give a clear illustration about the software interface and its interactions. User interface design is the design of computers, appliances, machines, mobile communication devices, software applications, and websites with the focus on the user's experience and interaction.



Figure 3.18: Software interface design

3.3.4. Implementation Phase

When all the previous testing and formation stages have been passed and result achieved, this stage take care of the implementation of the result achieved there from. This is described as the mother of all the phases. At this stage, the system is formed, policies are made, theories are generated and conclusions are drawn. All irregularities and inadequacies would have been removed at this stage.

Finally a proposed framework had been created and which will apply digital watermarking technique into LSB method. A fragile image watermarking technique has been proposed to be implemented into the current general model of the digital watermarking using LSB method in order to measure robustness of the watermarked image. Further explanations of this process will be elaborated in chapter 4.

3.4. Conclusion

Software prototype, an activity during a specific software development, is the creation of prototypes, namely the full version of the software programs developed. A prototype is usually only simulate some aspects of the features of the program eventually.

The aim is to enable users of conventional prototype software to evaluate developers' proposals to design products that finally by actually trying them out, rather than having to interpret and evaluate the design based on the description.

3.5. Summary

In this day, computer technology growth increased rapidly. There are so many digital data falsification has occurred. By applying the watermarking technique using LSB that insert inside to the throw-away prototyping methodology, the falsification problem of digital data will be resolved. Prototyping has several advantages such as obtaining feedback from users at the beginning of the project. It also allows the researcher some insight into the accuracy of initial estimates and whether the project deadlines and milestones proposed can be successfully met.

CHAPTER IV

FINDING AND RESULT

The session in this chapter researcher analyze data requirements needed to determine what inputs will be processed on the software programs to be designed. This chapter also discusses about the implementation of LSB insertion methods in watermarking applications from programs that will be built.

In general, the software programs that will be made in this project has the function to hide the information in the form of digital data behind other digital data, in this case the media used is a digital image and should be a concern that in the process of modification changes that occur between the media container with a container not modified media be too flashy or in other words by naked eyes, changes in the image of a container that has been modified not too visible (invisible watermark).

In order for a confidentiality of information contained in the object image of digital reservoir remain intact (integrity), so not just anyone will be able to retrieve information from the image of the reservoir, it takes a key that is used to retrieve a secret message contained in the image of a container object called by the term key. Without this key lay people who do not know the key word, will not be able to get the information contained in the image placeholders.

4.1. Digital Image File

Image files consist of either (geometric) data or vector rasterized pixel to pixel when displayed (with some exceptions) in the display vector graphics. Pixel is an image that was ordered as a grid (columns and rows); each pixel consists of figures representing the amount of brightness and color.

In this system, user has to open a plain image as a raw data. The file format of the image should be BMP, JPG and JPEG. The encoding process of text files will be done later after the image file opened.

4.2. Text File

The text file is a file type of characters, symbols and numbers are structured as a sequence of lines. End of a text file is often denoted by placing one or more special characters, known as a point (.), after the last line in a text file. It refers to the type of container, while the plain text refers to the type of content.

In this digital image watermark, the process of embedding a text message consists of two parts. First, user can process to insert the text message typed by themselves. Second, user also can process to insert the text message that stored in the text file. The text files are stored in WordPad (rtf format) or Notepad (txt format).

4.3. Image File with Hidden Text Message

Image watermarking deals with creating a metadata about the content and hiding it within the image. Information hiding in the watermarked image is as a nearly invisible embedding of information within various host data sets. After the message is encoded to the plain image file, the system will produce an output image as a watermarked image. The type of image file you've pasted the text messages will be stored as BMP (bitmap).

4.4. LSB Analysis in Application

Least Significant Bit (LSB) is the bit position in a binary integer giving the units value, that is, determining whether the number is even or odd. The LSB is sometimes referred to as the right-most bit, due to the convention in positional notation of writing less significant digit further to the right. The bit that carries is the lowest value or weight in binary notation for a numeral. It is analogous to the least significant digit of a decimal integer, which is the digit in the ones (right-most) position.

LSB method will read the pixel of image in RGB into a raster data that represent using binary. Then LSB read the text message that represent by characters into ASCII in decimal and converted to bits message in binary. LSB change every lowest bit into bit message. If lowest bit equal to bit message, then the raster data will not be changed. If lowest bit lower than bit message, then raster data will be add by 1. But if lowest bit greater than bit message, then raster data will be subtract by 1. To read the new pixel, LSB will change the raster data to RGB then convert it to pixel.

The process of hiding messages in digital images is done by changing the Least Significant Bit digital image placeholders in the message information that would be hidden. Before the process of hiding the message carried will be determined first in hiding location. The determination of the location of hiding the message is as follows:

- i. To the original RGB image $c(x, y) = [R_c, G_c, B_c]$ size $m \times n$, specify a random seed and generate pseudo-random number and then collated into a pseudo-image RGB $p(x, y) = [R_p, G_p, b_p]$ $m \times n$ size.
- ii. Calculate the distance between $c(x, y)$ and $p(x, y)$ using the distance formula of two vectors, namely

$$d(x, y) = \sqrt{(R_c - R_p)^2 + (G_c - G_p)^2 + (B_c - B_p)^2}$$

- iii. Hiding starting from the location with the smallest distance to the largest distance.

4.5. Decryption and Encryption Process Analysis of Text Messages

To strengthen the data hiding technique, every bit of confidential data is not used to replace the sequential bytes, but the byte array is selected at random. For example, if there are 50 bytes and 6-bit data bytes will be hidden and replaced its LSB bits chosen at random, say byte number 36, 5, 21, 10, 18, 49. Random numbers can be generated with the program pseudo-random-number-generator (PRNG). PRNG using a secret key used to generate the position of pixels that will be used to hide bits. PRNG in built a number of ways, one using block-based cryptographic algorithm (block cipher). The purpose of this encryption is to produce a set of

random numbers for each of the same encryption key by selecting bits from a block of data encryption basis. The cryptographic algorithm used in this research program is the RC4 algorithm.

RC4 algorithm was developed in 1987 by Ron Rivest, developers of the RSA algorithm. RC4 uses a byte operation is more easily implemented in a program, especially in the machine an 8-bit or 16 bit machine. The Rivest Cipher version 4 (RC4) Algorithm, designed by Ron Rivest in 1987, is a symmetric cryptographic method for which a variable key-size stream cipher is used, which is generated by a pseudo-number generator. RC4 initial key (from 1 to 256 bytes) stored in a temporary vector T creates a variable 256-byte key in a state vector S; thus, the RC4 performs byte-wise operations and random permutations. (Kartalopoulos, S. V. 2009).

4.6. Size Analysis Data Hidden



Size of data which will be hidden is depending on the size of the data container. In 8-bit image of size 256x256 pixels there are 65,536 pixels and each pixel size of 1 byte. Once converted into 24-bit image, the data size to be 196,608 bytes (65,536 multiplied by 3 bytes). Since each byte can only hide 1 (one) bit in LSB, so the size of data to be concealed inside a maximum image becomes 24,576 bytes (196 608 divided by 8 bytes). The size of this data must be reduced with long file names, due to confidential data hiding not only hide the contents of the data but also conceals the name of the file. The greater hidden data in the image can be damaged by manipulate image container.

4.7. Digital Image Watermark Testing

4.7.1. Image Selection

In this experiment, researcher use image as raw data and text as its manipulation, thus as a result, it will produce an image with BMP extension. There are 2 types of plain images which are BMP and JPEG the first plain image is Computer.jpeg and the second plain image is UUMLogo.bmp.

Table 4.1: Raw data images

Image 1	Image 2
 <p data-bbox="475 1413 671 1447">Computer.jpeg</p>	 <p data-bbox="1034 1424 1241 1458">UUMLogo.bmp</p>

4.7.1.1 JPEG Image



Figure 4.1: Plain image in Jpeg file format

This is the first plain image which is Jpeg format. This image consists of pixels which extracted and converted to binary using WinHex:

	00	01	02	03	04	05	06	07	08	09	0a
00000000	11111111	11011000	11111111	11100000	00000000	00010000	01001010	01000110	01001001	01000110	00000000
00000010	00000000	01100000	00000000	00000000	11111111	11100001	00010000	11100000	01000101	01111000	01101001
00000024	00000000	00101010	00000000	00000000	00000000	00001000	00000000	00000100	00000001	00111011	00000000
00000030	00000000	00000000	00001000	01001010	10000111	01101001	00000000	00000100	00000000	00000000	00000000
00000040	10011100	10011101	00000000	00000001	00000000	00000000	00000000	00001110	00000000	00000000	00010000
00000050	00000000	00000000	00001000	00001100	00000000	00000000	00000000	00111110	00000000	00000000	00000000
00000060	00000000	00001000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000070	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000080	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
00000090	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
000000a0	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000

Figure 4.2: Plain image which is already converted to binary as a raw data

Here, researcher takes the first 9 pixels of the image which converted to the binary as the image binary that will be encoded: “11111111 11011000 11100000 00000000 00010000 01001010 01000110 01001001 01000110”

4.7.1.2. BMP Image



Figure 4.3: Plain image in BMP file format


This is the second plain image which is BMP format. This .bmp image consists of pixels which extracted and converted to binary using WinHex:

	00	01	02	03	04	05	06	07	08	09	0a
00000000	01000010	01001101	00100110	01101010	00000110	00000000	00000000	00000000	00000000	00000000	00110110
00000010	00000000	00000000	01010000	00000001	00000000	00000000	10100001	00000001	00000000	00000000	00000001
00000020	00000000	00000000	00000000	00000000	00000000	00000000	00100011	00101110	00000000	00000000	00100011
00000030	00000000	00000000	00000000	00000000	00000000	00000000	11111101	11111101	11111101	11111101	11111101
00000040	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101
00000050	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101
00000060	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101
00000070	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101
00000080	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101
00000090	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101
000000a0	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101	11111101

Figure 4.4: Plain image which is already converted to binary as a raw data

For image binary, we just take the first 9 pixels of the image. And the first 9 pixels are converted to the binary as the image binary that will be encoded: “01000010 01001101 00100110 01101010 00000110 00000000 00000000 00000000 00000000”

4.7.2. Hidden Texts

As manipulation, researcher use plain text by directly type to the system, and for .txt/.rtf file. User can upload .rtf or .txt file using  “Load Text” button. The manipulation texts that will be used in this project are “Copyright” for typing text, “hideText1.txt”, and “hideText2.rtf”.

4.7.2.1. Typing Text

Text message that will be encoded is “Copyright” which has binary: “01000011 01101111 01110000 01101001 01110010 01101001 01100111 01101000 01110100”

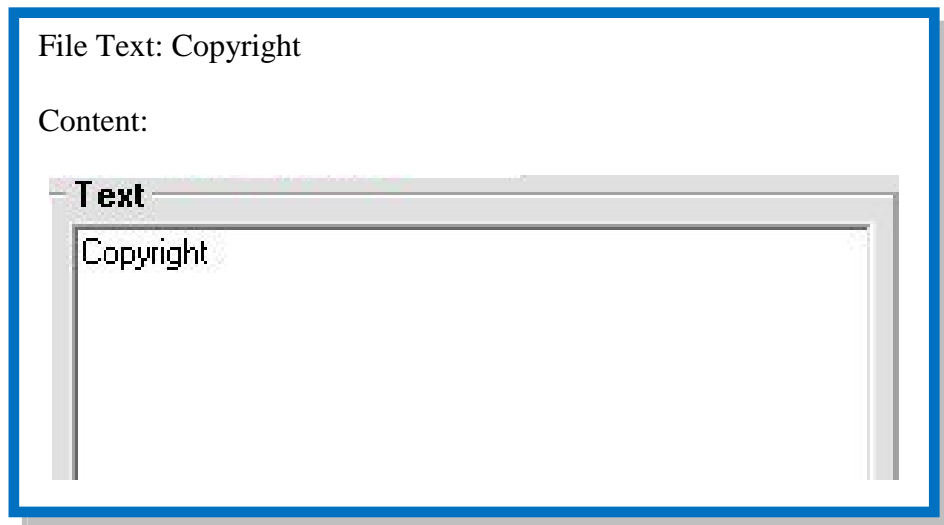


Figure 4.5: Typing text

4.7.2.2. TXT File

The second process to insert the text message that stored in the text file. In this step, researcher uses a .txt file named “hideText1.txt”. Here is the txt file’s content.

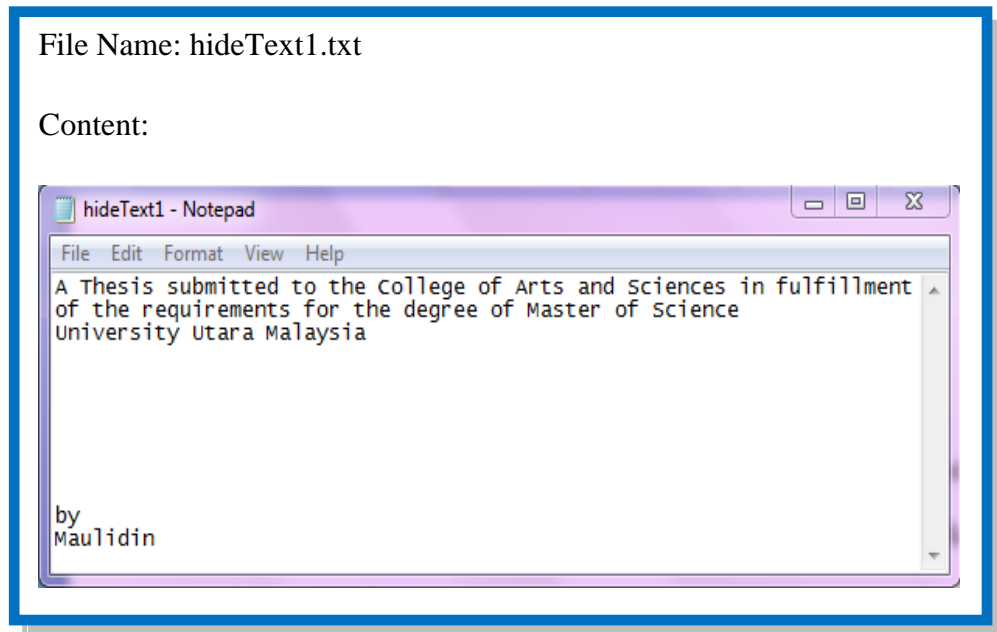


Figure 4.6: Notepad text file

This is the txt format which converted to binary using WinHex

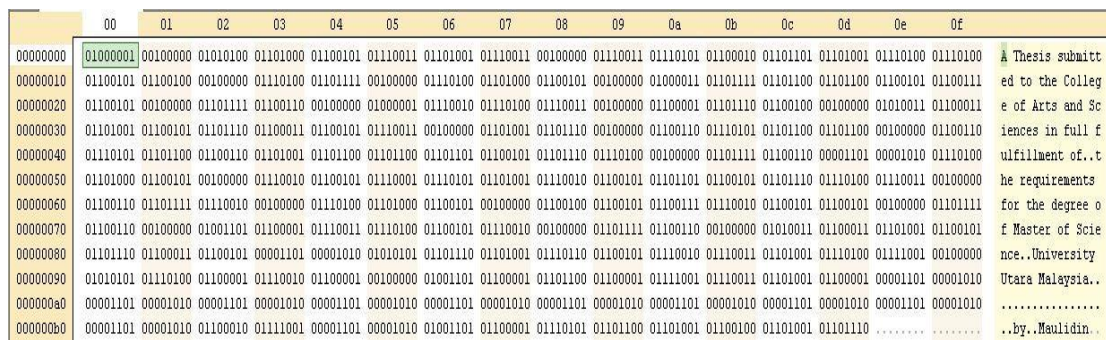


Figure 4.7: Notepad text file using WinHex

4.7.2.3. RTF File

In this step, researcher uses a .txt file named “hideText2.rtf”. Here is the txt file’s content.

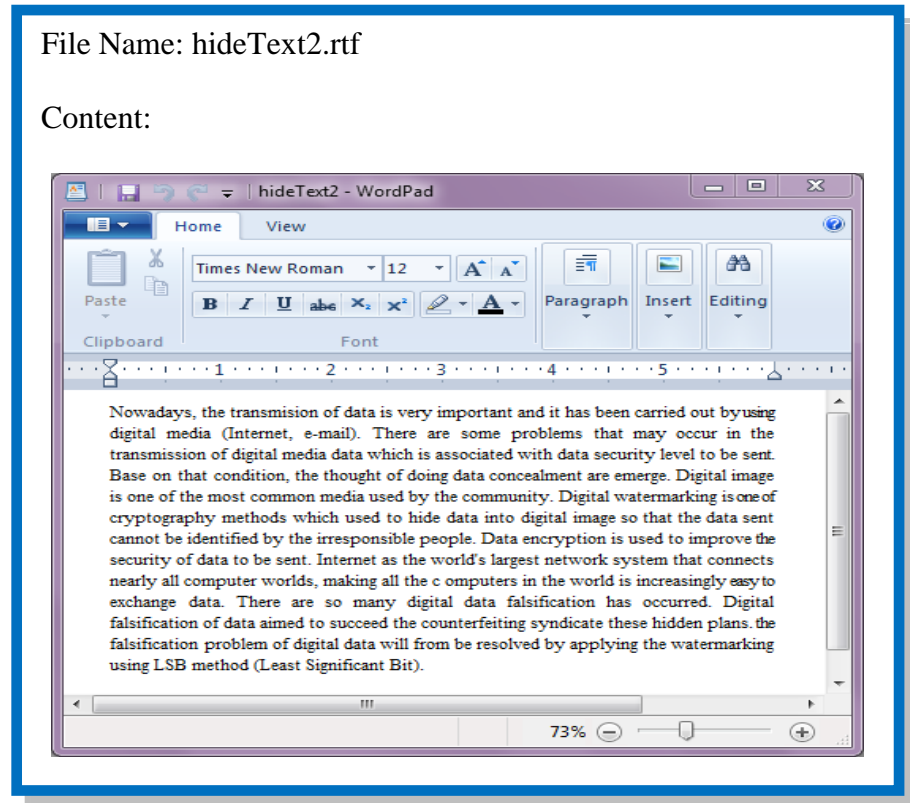


Figure 4.8: WordPad text file

This is the rtf format which converted to binary using WinHex

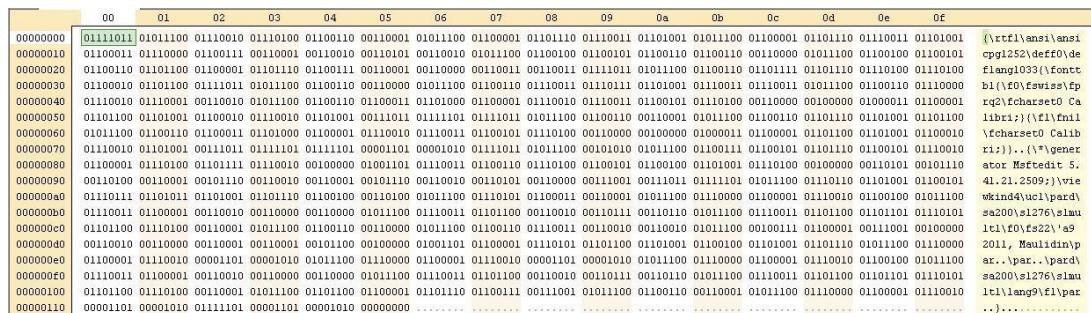


Figure 4.9: WordPad text file using WinHex

4.7.3. LSB Analysis in Image

As a sample researcher will analyze manipulation of image using “Copyright” text. Below is calculation analysis of LSB insertion method for Jpeg and BMP image. Table 4.2 show the LSB encryption image for Jpeg and text message:

Table 4.2: LSB raster process computer.jpeg

Image (Raw Data)	Letters	Type	Text Binary (Raster Data)	Result
11111111	C	Capital letter	01000011	11111111
11011000	o	Small letter	01101111	11011001
11100000	p	Small letter	01110000	11100000
00000000	y	Small letter	01101001	00000001
00010000	r	Small letter	01110010	00010000
01001010	i	Small letter	01101001	01001011
01000110	g	Small letter	01100111	01000111
01001001	h	Small letter	01101000	01001000
01000110	t	Small letter	01110100	01000110

The table above explains about the process of LSB. Here, if the plain image bits are lower than the text bit, so, the LSB will add 1 bit into the least binary. But if the plain image bits are greater than the text bit, means the LSB will deduct 1 bit. Otherwise when the bits are same, LSB will not change it. From that process, we get the result: “**11111111 11011001 11100000 00000001 00010000 01001011 01000111 01001000 01000110**”.

Table 4.3: LSB raster process for image uumLogo.bmp

Image (Raw Data)	Letters	Type	Text Binary (Raster Data)	Result
01000010	C	Capital letter	01000011	01000011
01001101	o	Small letter	01101111	01001101
00100110	p	Small letter	01110000	00100110
01101010	y	Small letter	01101001	01101011
00000110	r	Small letter	01110010	00000110
00000000	i	Small letter	01101001	00000001
00000000	g	Small letter	01100111	00000001
00000000	h	Small letter	01101000	00000000
00000000	t	Small letter	01110100	00000000

Based on LSB process, we get result for watermarked uumLogo.bmp as: “**01000011 01001101 00100110 01101011 00000110 00000001 00000001 00000000 00000000**”. After we get this binary result, it will be converted again as a new pixel then change into a new image which called as watermarked image. This watermarked image will indicate the copyright owner, and provide additional information of digital content.

4.7.4. Software Implementation

The implementation of this program consists of two parts, which are: insertion process of text message and extraction process of image to see the hidden message which inserted previously.

4.7.4.1. Insertion Process of Text Message

For the first time, user should insert the cryptic code to begin the process of the software. This cryptic code will be needed later when decryption process running to get the text messages that have been inserted earlier. Before the cryptic code inserted, the button "Take Picture", "Insert Message", and "Save Messages" become inactive and will be active after it typed. Once the cryptic code entered, the image "load" button will become active. The next step is to select the image file you want then enter the text message. The display in the current program will select the image file is as follows:

- **JPEG Image**



Figure 4.10: Load plain Jpeg image

After insert the cryptic code, user can select an image file, user can type a text message on the in the Text Field. For the first encryption, researcher use “Copyright” as hidden text message. Then when the image opened, "load" button will be active. Thus user can press the "Load" to insert the text message into image files that have been opened before.

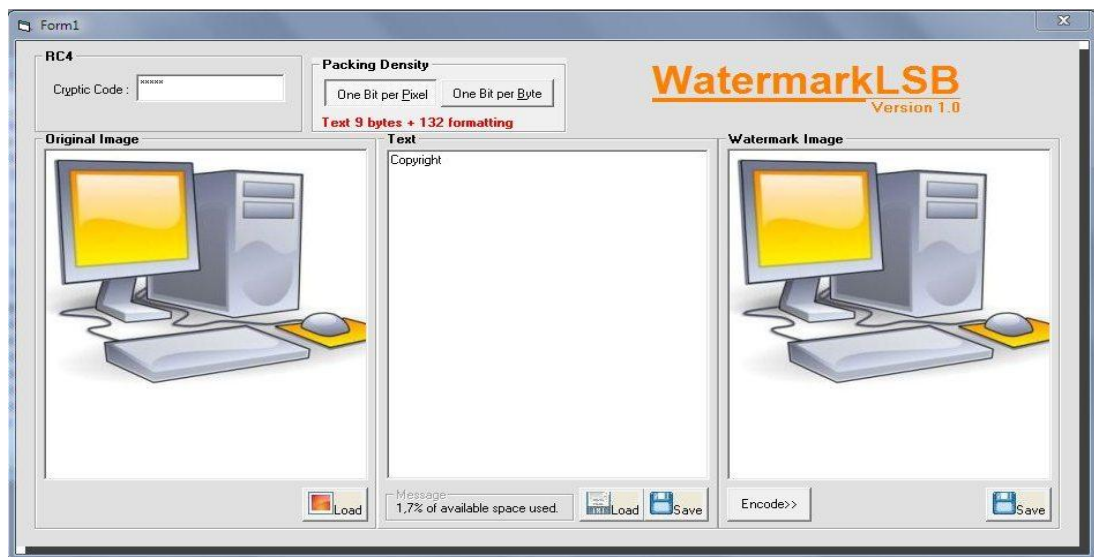


Figure 4.11: Jpeg watermark result with typing text message

After the encryption of text messages, the watermark image will be appearing. Researcher can press "Save" to save image. The specify storage location of image file that is the result of the merger earlier. The result of the encryption of a text on the image file will be saved in bitmap format (BMP). Image files that are the result of the encryption did not change the text message width and height of image file. After pressing the "Save", the text column will be empty and the original image and image insertion results will be displayed in the program.

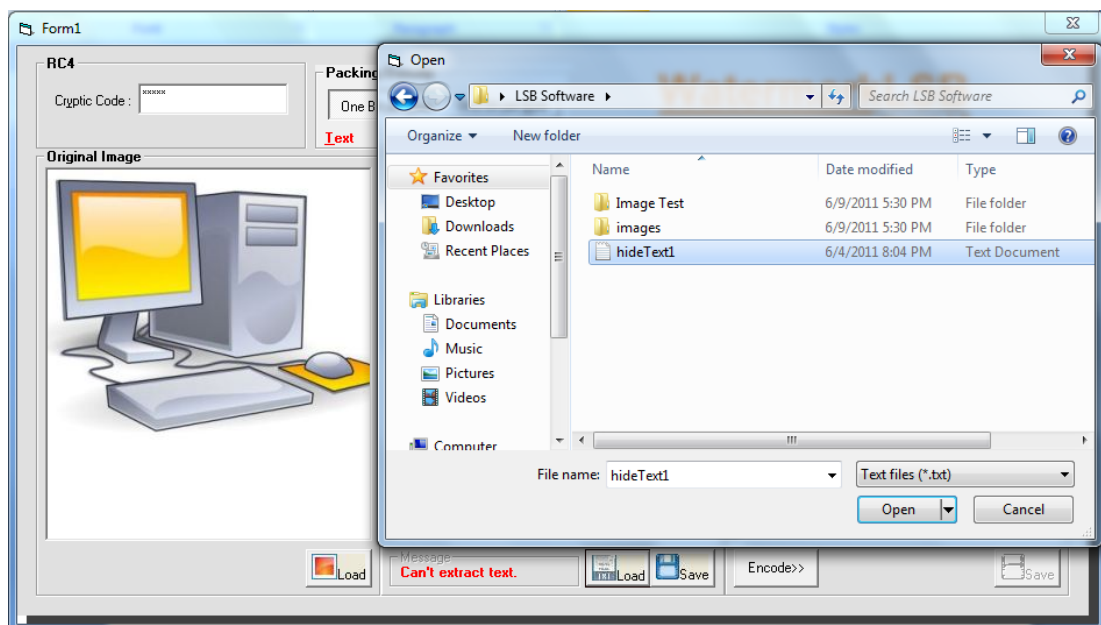


Figure 4.12: Jpeg image with txt text file

The same step also performed on the figure 4.12 above. In figure 4.12 above describes the selection of a text file that will be included. Here the researcher try to insert the file named "hideText1.txt" which made through a notepad.

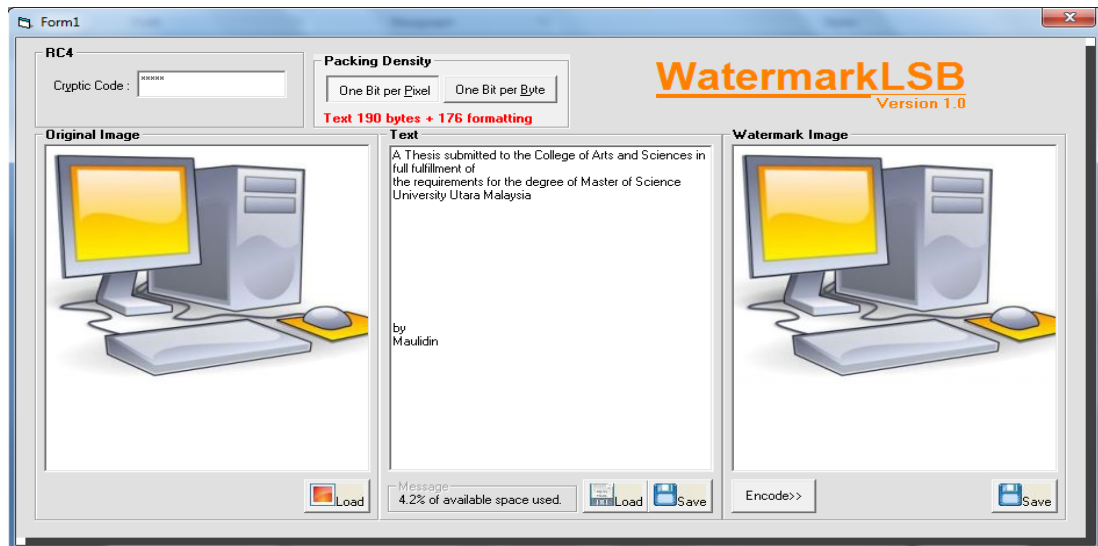


Figure 4.13: Jpeg watermark result with txt text file

In figure 4.13 above describes the insertion process of “hideText1.txt” text file that will be included. Here the researcher try to insert the file named "hideText1.txt" which made through a notepad.

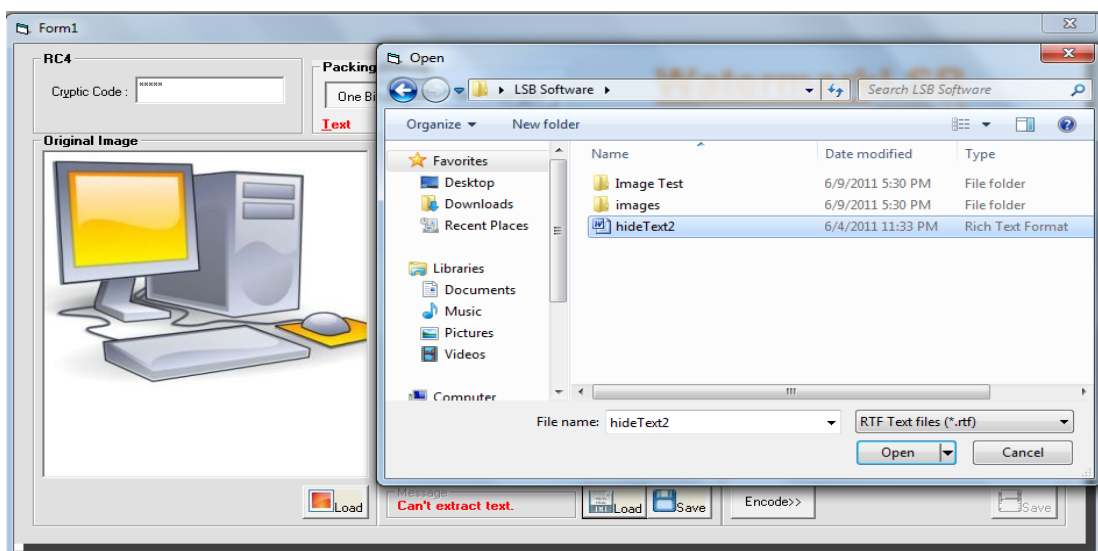


Figure 4.14: Jpeg image with rtf text file

On the figure 4.14 above the same step also performed. It describes the selection of rtf text file that will be included. Here the researcher try to insert the file named "hideText2.rtf" which made through a WordPad.

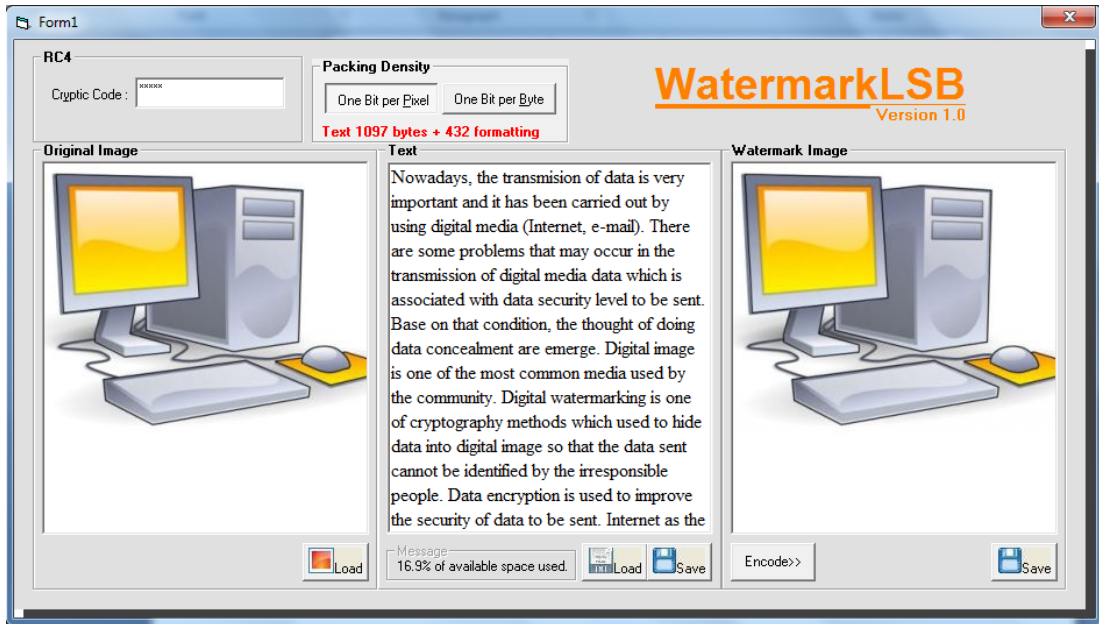


Figure 4.15: Jpeg watermark result with rtf text file

In figure 4.15 above describes the insertion process of “hideText2.rtf”. Here the researcher try to insert the contents in rtf file named "hideText1.rtf" which made through a WordPad.

- **BMP**



Figure 4.16: Load plain BMP image

The same process also applied to BMP image encryption. Here, researcher loads uumLogo.bmp as original image. In process of BMP image encryption, researcher also used “Copyright” as its hidden text message.

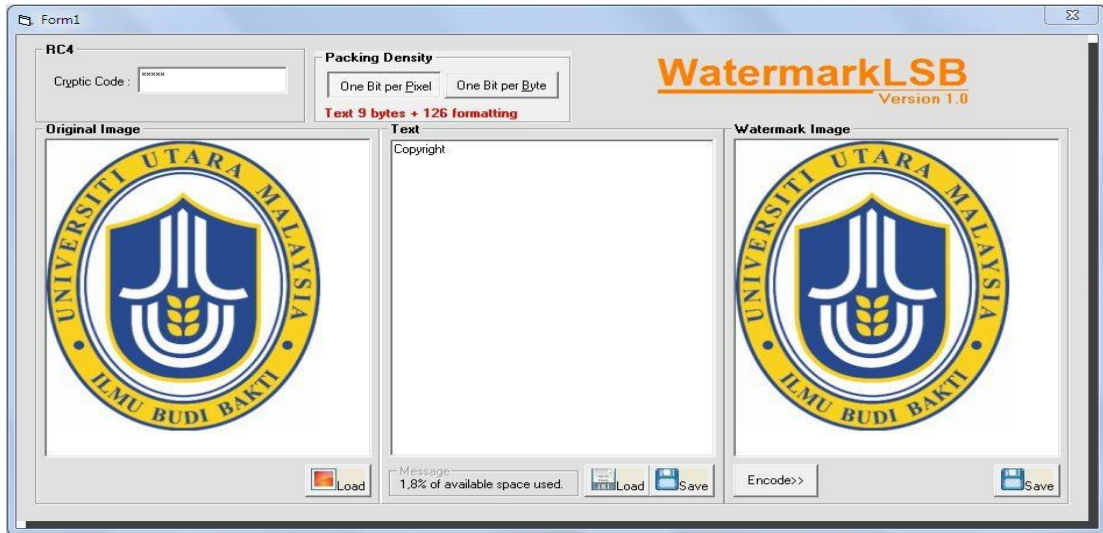


Figure 4.17: BMP watermark result with typing text message

Here, researcher press “Encode” button to encrypt the text message into the image. After the encryption of text messages, the watermark image will be appear and researcher can press "Save" to save image in bitmap format (BMP).

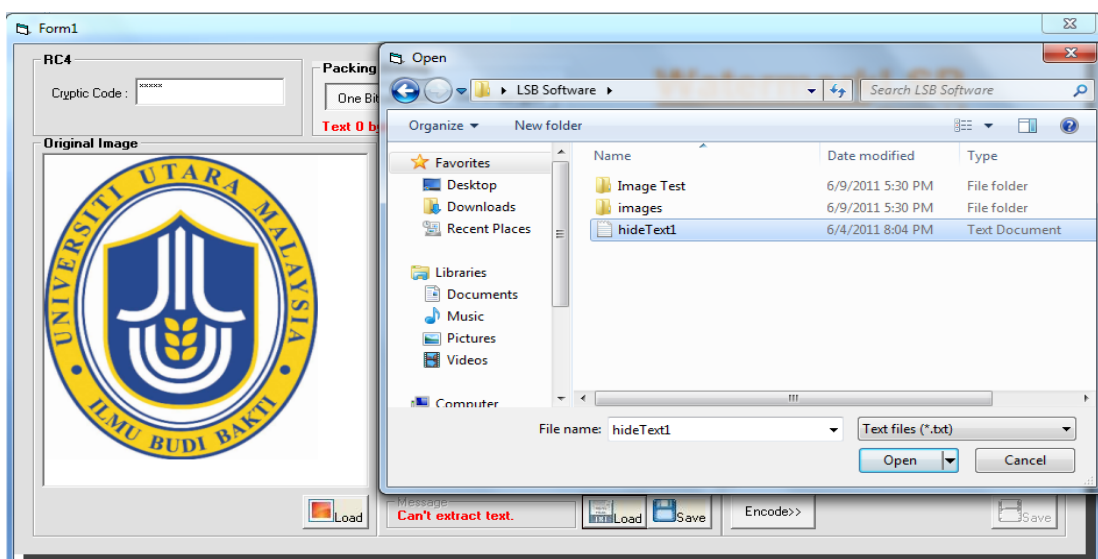


Figure 4.18: BMP image with txt text file

The same step also performed on the figure 4.18 above. In figure 4.18 above describes the selection of a text file that will be included. Here the researcher try to insert the file named "hideText1.txt" which made through a notepad.

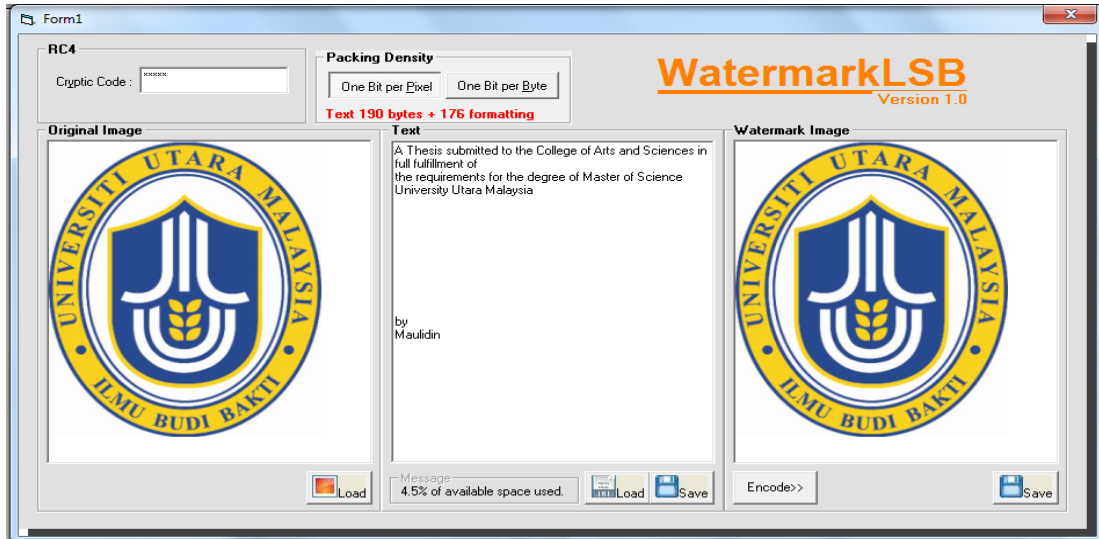


Figure 4.19: BMP watermark result with txt text file

In figure 4.19 above describes the insertion process of "hideText1.txt" text file that will be included. Here the researcher try to insert the file named "hideText1.txt" which made through a notepad

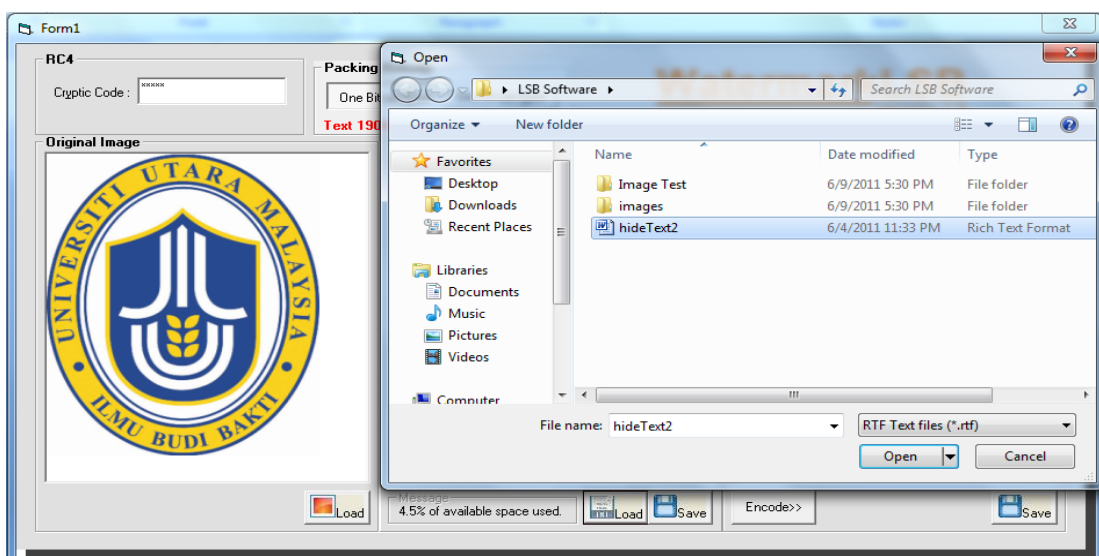


Figure 4.20: BMP image with rtf text file

On the figure 4.20 above the same step also performed. It describes the selection of a rtf text file that will be included. Here the researcher try to insert the file named "hideText2.rtf" which made through a WordPad.

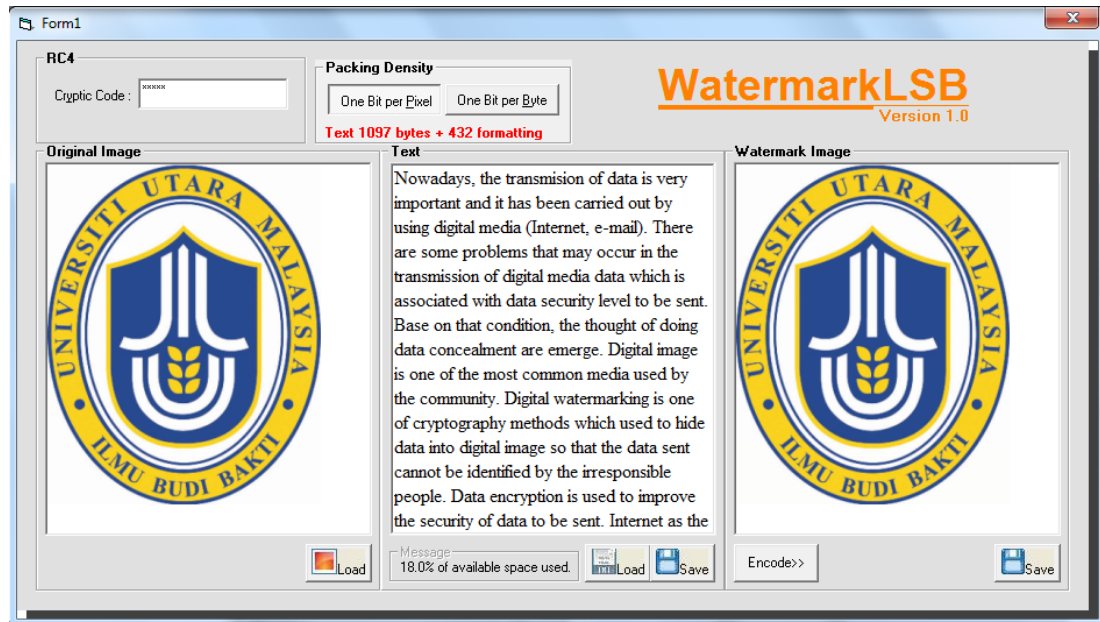


Figure 4.21: BMP watermark result with rtf text file

In figure 4.21 above describes the insertion process of "hideText2.rtf". Here the researcher try to insert the contents in rtf file named "hideText1.rtf" which made through a WordPad.

4.7.4.2. Extraction Process of Text Message



Figure 4.22: Text message decryption in Jpeg

The first decryption that used here is “result1.bmp”, as we know before this picture is a result from “computer.jpeg” encryption. To decrypt an image, we need to remember the cryptic code that we used before. If the cryptic code is wrong, this software will not execute the decryption.

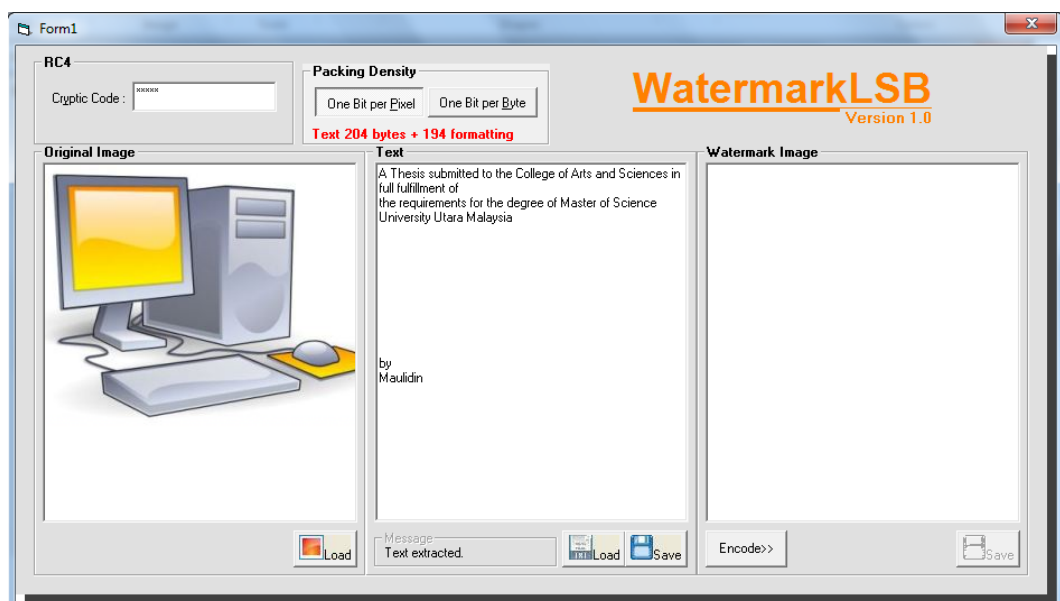


Figure 4.23: Txt file text message decryption in Jpeg

The second extraction that used here is “resultTxt.bmp”, this extraction used 204 bytes. This software gives a content of Txt file text as its outcome.

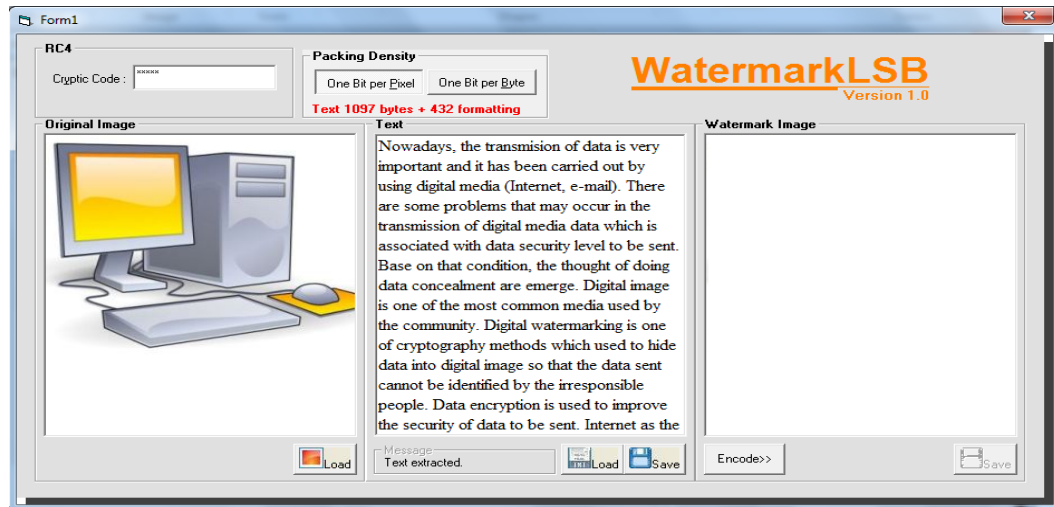


Figure 4.24: Rtf file text message decryption in Jpeg

The third extraction that used here is “resultRtf.bmp”, this extraction used 1097 bytes. This software gives a content of rtf file text as its outcome.

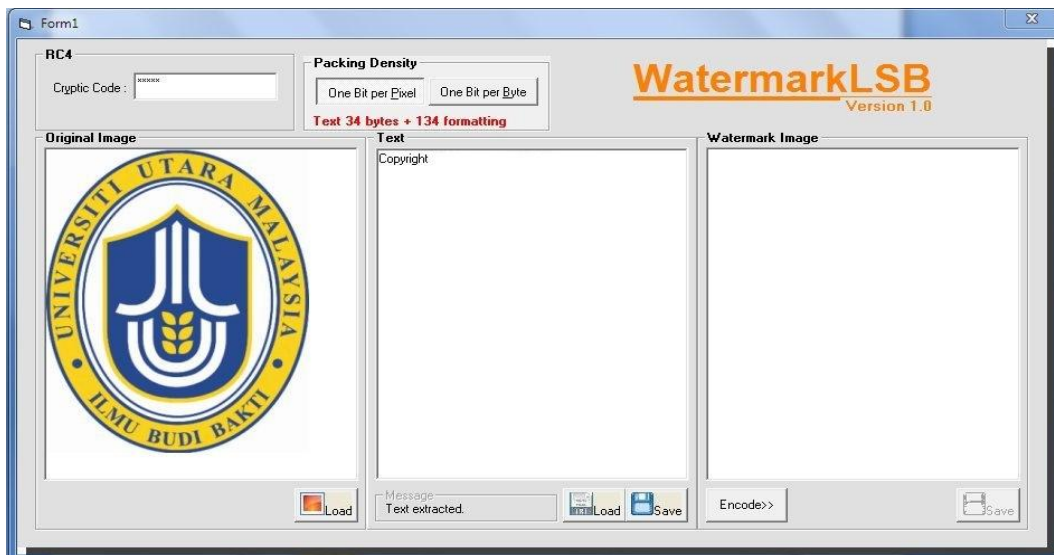


Figure 4.25: Text message decryption in BMP

The fourth decryption that used here is “result2.bmp”, as we know before this picture is a result from “uumLogo.bmp” encryption.

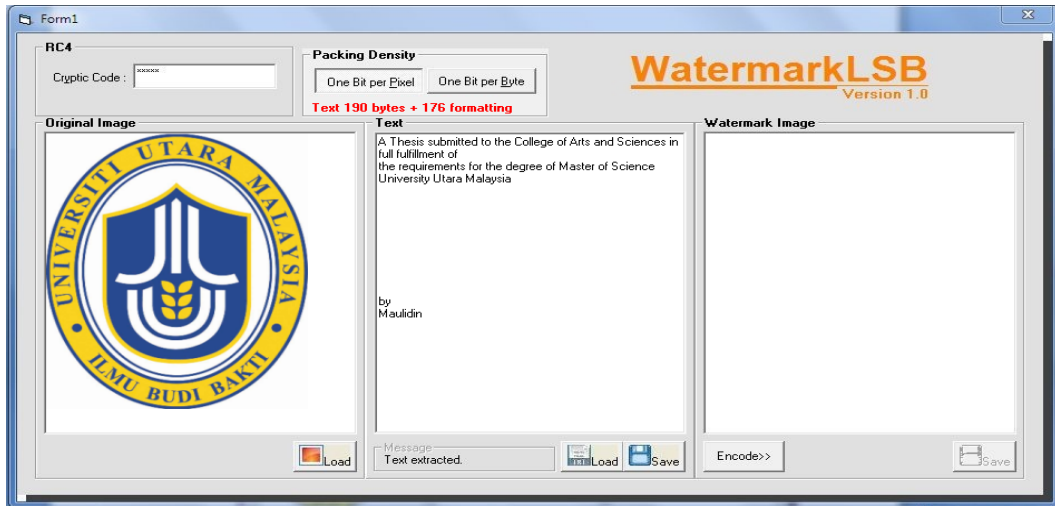


Figure 4.26: Txt file text message decryption in Jpeg

The fifth extraction that used here is “resultTxt2.bmp”; this extraction used 204 bytes. This software gives a content of Txt file text as its outcome.

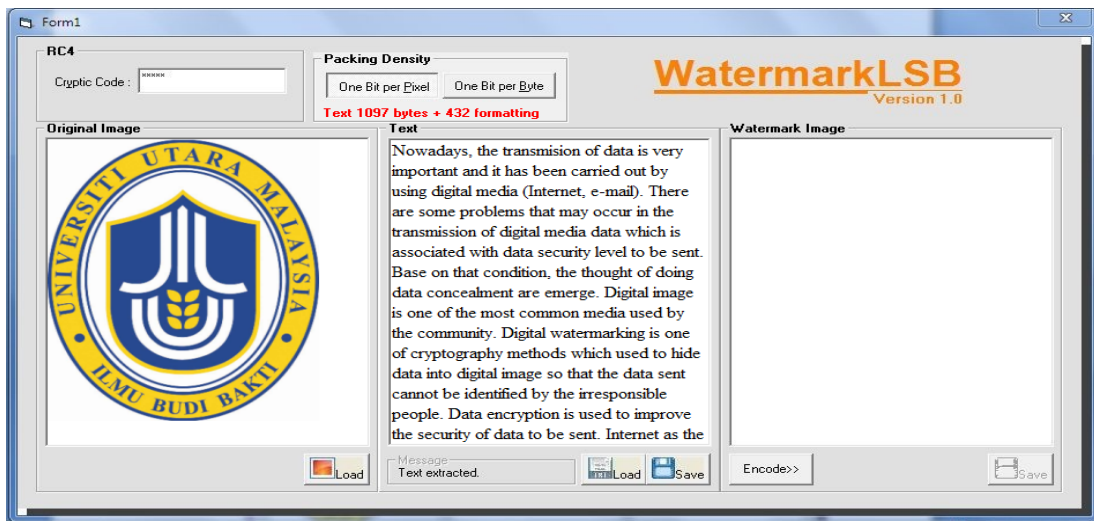


Figure 4.27: Txt file text message decryption in Jpeg

The sixth extraction that used here is “resultTxt2.bmp”; this extraction used 1097 bytes. This software gives a content of TXT file text as its outcome.

4.7.5. Robustness Testing

In order to test the security of confidential messages inside image is by scaling, rotating and cropping. Figures below show the image that has been enlarged, narrowed, rotated, and cropped.

- **Enlarge**

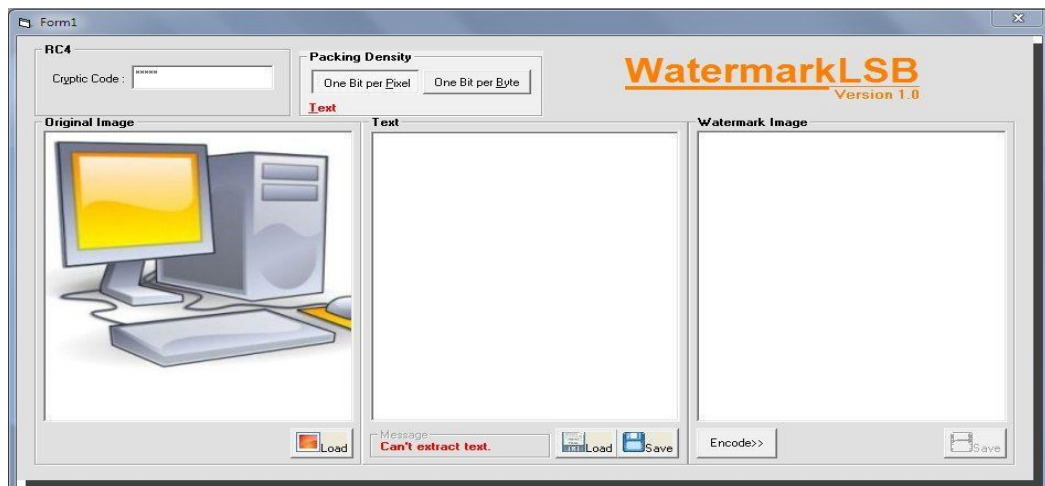


Figure 4.28: Enlarging manipulation step of computer.jpeg



Figure 4.29: Enlarging manipulation step of uumLogo.bmp

In figure 4.28 and 4.29 show extracting process of the image that has been enlarged. The results show that, the pictures have been enlarged cannot be read the hidden text messages contained.

- **Narrow**

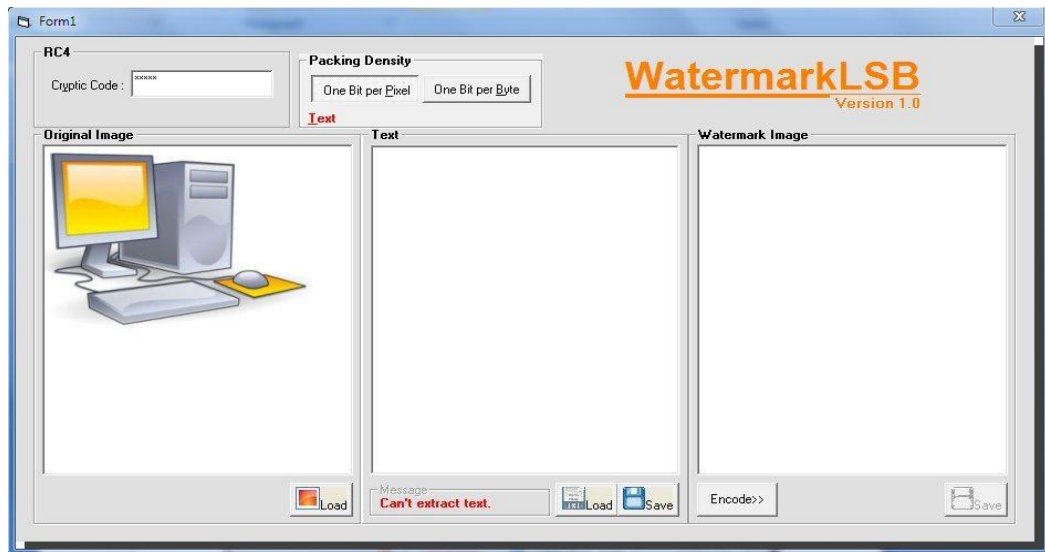


Figure 4.30: Narrow manipulation step of computer.jpeg



Figure 4.31: Narrow manipulation step of uumLogo.bmp

Figure 4.30 and figure 4.31 show extracting process of the image that has been narrowed. The results show that, the pictures have been narrowed cannot be read the hidden text messages contained.

- **Rotate**

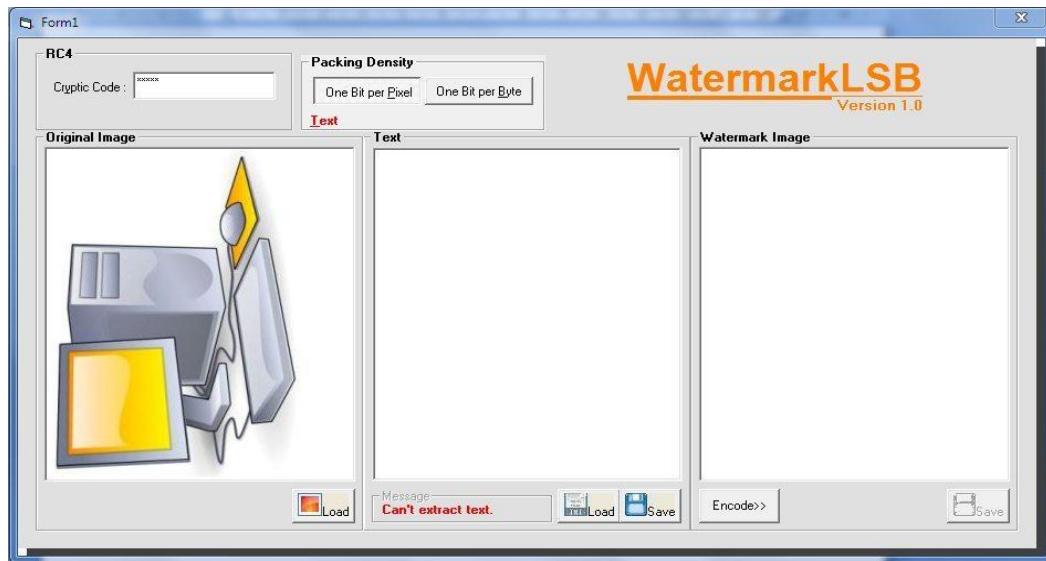


Figure 4.32: Rotation manipulation step of computer.jpeg



Figure 4.33: Rotation manipulation step of uumLogo.bmp

Figure 4.32 and figure 4.33 show extracting process of the image that has been rotated. The results showed that, the pictures have been rotated cannot be read the hidden text messages contained.

- Crop



Figure 4.34: Cropping manipulation step of computer.jpeg

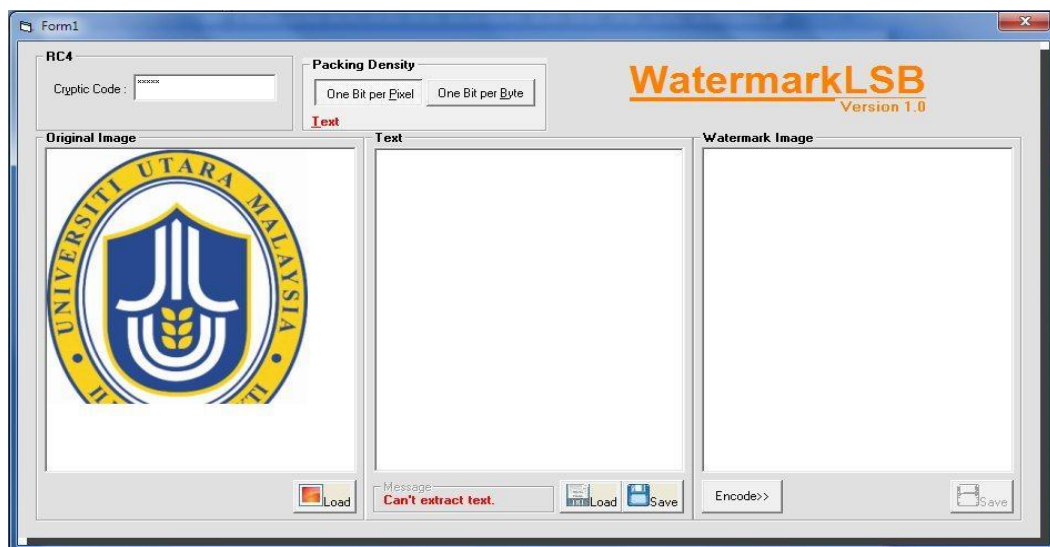


Figure 4.35: Cropping manipulation step of uumLogo.bmp

Figure 4.34 and figure 4.35 show extracting process of the image that has been cropped. The results showed that, the pictures have been cropped cannot be read the hidden text messages contained.

Based on the software testing that has done above, the result of manipulation test on watermarked image can be shown in table 4.4:

Table 4.4. Result of manipulation test

Image	Text	Manipulation	Robustness	
			Strong	Weak
Computer.jpeg	"Copyright" typing text	Enlarge	√	-
		Narrow	√	-
		Rotate	√	-
		Crop	√	-
	hideText1.txt	Enlarge	√	-
		Narrow	√	-
		Rotate	√	-
		Crop	√	-
	hideText2.rtf	Enlarge	√	-
		Narrow	√	-
		Rotate	√	-
		Crop	√	-
UUMLogo.bmp	"Copyright" typing text	Enlarge	√	-
		Narrow	√	-
		Rotate	√	-
		Crop	√	-
	hideText1.txt	Enlarge	√	-
		Narrow	√	-
		Rotate	√	-
		Crop	√	-
	hideText2.rtf	Enlarge	√	-
		Narrow	√	-
		Rotate	√	-
		Crop	√	-

From the results above of the software testing was obtained security level of image's secret message sent to the recipient in safe, this is viewed from several trials or experiments to make changes or add disturbances or noise in the image but cannot reveal the secret messages contained in that image .

4.8. Summary

From the analysis and results above, the implementation of the protection of confidential messages on digital image using Least Significant Bit method can be concluded that the method has the nature of the Least Significant Bit Fragile (easily broken) by interference from outside, such as magnification, rotation, cropping, and noise disturbance that would be a secret message secured from unauthorized person who tried to forcibly open the secret message.

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

This chapter will provide the summary of the whole research. This chapter will be divided into research summary, research contribution, and recommendation for future work. In the research summary, the steps of this research are summarized again. Next, the discussion of research contribution which point out to the benefits of the research is presented. Finally the limitation and future work will be discussed.

5.1. Conclusions

Previously, all of the requirements and basic information about the proposed digital image watermarking using LSB method has been explained though chapter one, the objective of this study was:

- i. To hide the information in the form of text messages on the digital image data using LSB.
- ii. To understand the contents of text messages hidden in the digital image data using LSB (authentication).

The two objective stated above have been achieved at the end of the system development phase. Based on the on the finding of the research protection confidential messages on digital image Using Least Significant Bit Method can be concluded that digital image that has been inserted in the text message does not

change the original digital image by naked eye. In addition, this method has the nature of the Least Significant Bit Fragile (easily broken) by interference from outside such as: enlarge, narrow, rotation, and cropping disturbance, so that the secret message will be more secured from unauthorized person who tried to open the secret message.

5.2. Research Contributions

Contributions from this research carried out is for it to improve copyright protection in digital image watermarking method that can be extracted without using the original image and is invisible. In addition, this research also contributes to education, especially in the field of concealment of information for those who want to develop a watermarking technique in digital data, especially digital image. This research can also be used as a reference for research on digital data security system further.

5.3. Limitations and Recommendations

In this research, there are some limitations of digital image watermark which only can be used for JPEG and BMP image files. For the future, researchers expect the developments in watermarking research that will further increase along with the development of civilization and technological development. In addition, researchers expect any development in other watermarking method that can be applied also to get maximum results. This indicates that the method of embedding watermarking by means LSB has a fragile nature so as to make secret messages that will be attacked be lost. The next research development expected to create a strong message that insertion method will not be amended at the time of the attack and still be read by the recipient.

REFERENCES

- Afolabi, M. (1992). The Review of Related Literature in Research. *International Journal of Information and Library Research*. 4(1), 59-66.
- Ambler, S. W. (2004). *The Object Primer: Agile Modeling-Driven Development With UML 2.0*. Cambridge: Cambridge University Press.
- Anan, T., Kuraki, K., & Nakagata, S. (2007). Watermarking Technologies for Security - Enhanced Printed Document. *Fujitsu Sci. Tech. J.* 43(2), 197-203. Retrieved from <http://www.fujitsu.com/downloads/MAG/vol43-2/paper06.pdf>
- Barreto, P. S. L. M., Kim, H. Y., & Rijmen, V. (2002). Toward Secure Public-Key Blockwise Fragile Authentication Watermarking. *Vision, Image and Signal Processing, IEE Proceedings*, 149(2), 57-62. doi: 10.1049/ip-vis:20020168.
- Basu, R. (2004). *Implementing Quality: A Practical Guide to Tools and Techniques: Enabling The Power of Operational Excellence*. London: Thomson Learning.
- Blessing, L. T. M., & Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. London: Springer.
- Bourner, T. (1996). The Research Process: Four Steps to Success in T. Greenfield (Ed.). *Research Methods: Guidance for Postgraduates*. London: Arnold.
- Brannock, E., Weeks, M., & Harrison, R. (2008). Watermarking with Wavelets: Simplicity Leads to Robustness. *Southeastcon, IEEE*, 3-6 April 2008, 587-592. doi: 10.1109/SECON.2008.4494361.
- Chandra, M., Pandey, S., Chaudhary, R. (2010). Digital Watermarking Technique for Protecting Digital Images. *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, 9-11 July 2010. 7, 226-233. doi: 10.1109/ICCSIT.2010.5565177.
- Celik, M.U., Sharma, G., Saber, E., and Tekalp, A.M. (2001). A Hierarchical Image Authentication Watermark With Improved Localization And Security. *Image Processing, 2001. Proceedings. 2001 International Conference on*, 7-10 Oct 2001, 2, 502-505. doi: 10.1109/ICIP.2001.958538.
- Chen, P., Zhao, Y., & Pan, Jeng-Shyang. (2006). Image Watermarking Robust to Print and Generation Copy. *Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on*, 30 Aug 2006- 1 Sept 2006, 1, 496-500. doi: 10.1109/ICICIC.2006.98.
- Cox, Ingomar J., Miller, Mathew L., Bloom, Jeffrey A., Fridrich, J., & Kalker, T. (2008). *Digital Watermarking and Steganography*. USA: Morgan Kaufman Publishers.

- Darmstaedter, V., Delaigle, J.F., Quisquater, J.J., & Macq, B. (1998). Low Cost Spatial Watermarking. *Computer & Graphics*, 22(4), 417–424.
- Dennis, A., Wixom, B., & Tegarden, D. (2005). *System Analysis Design with UML 3.0*. USA: Wiley.
- Dictionaries, O., Waite, M., & Hawker, S. (2009). *Oxford paperback dictionary and thesaurus / edited by Maurice Waite, Sara Hawker*. Great Britain: Oxford University Press.
- Dittmann, J. (2000). *Digitale Wasserzeichen: Grundlagen, Verfahren, Anwendungsgebiete*. Berlin: Springer.
- Fridrich, J. (1998). Applications of Data Hiding in Digital Images. *Tutorial of the ISPACS '98 Conference, Melbourne*. Australia.
- Furht, B., Muharemagic, E., & Socek, D. (2005). *Multimedia Encryption and Watermarking*. USA: Springer.
- Furht, B. (2008). *Encyclopedia of Multimedia* (2nd ed). USA: Springer.
- Grady, J. O. (2006). *System Requirements Analysis*. USA: Elsevier Academic Press.
- Hanjalic, A. (2000). *Image and Video Databases: Restoration, Watermarking, and Retrieval*. Netherlands: Elsevier Science B. V.
- Hartung, F., & Kutter, M. (1999). Multimedia Watermarking Techniques. *Proceedings of the IEEE*, Jul 1999, 87(7), 1079–1107. doi: 10.1109/5.771066.
- Heileman, G.L., Pizano, C.E., & Abdallah, C.T. (1999). Performance Measures for Image Watermarking Schemes. *Proceedings of the Fifth Baiona Workshop on Emerging Technologies in Telecommunications, Baiona, Spain*. Retrieved from http://www.ece.unm.edu/controls/papers/Hei_Piz_CTA.pdf
- Jiang, X. (2010). Digital Watermarking and Its Application in Image Copyright Protection. *Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on*, 11-12 May 2010, 2, 114-117. DOI 10.1109/ICICTA.2010.625.
- Kartalopoulos, S, V. (2009). *Security of Information and Communication Networks*. Canada: Wiley.
- Katzenbeisser, S., & Petitcolas, F.A.P. (2000). *Information Hiding: Techniques for Steganography and Digital Watermarking*. Norwood, MA: Artech House Books.
- Kazakeviciute, G., & Rosenbaum, R. (2001). *Information Hiding on Wavelet Based Schemes under Consideration of Jpeg2000*. Preprint CS-10-0. University of Rostock, Department of Computer Sciences, Insitute of Computer Graphics. Retrieved from <http://www.idav.ucdavis.edu/~rene/publications/Rosenbaum-RIB01.pdf>
- Kuhn, T. (1996). *The Structure of Scientific Revolutions*. Chicago, University of Chicago Press.
- Kutter, M., & Hartung, F. (2000). Introduction to Watermarking Techniques. In: Katzenbeisser, S.; Petitcolas, F. A. P. (Ed): *Information Hiding Techniques for Steganography and Digital Watermarking*. Boston: Artech House Publishers.

- Lakatos, I. (1978). *The Methodology of Scientific Research Programmes*. (John Worrall and Gregory Curie, Eds). Cambridge, Cambridge University Press.
- Larman, C. (2002). *Applying UML and Patterns: an Introduction to Object-Oriented Analysis and Design and the Unified Process*. USA: Prentice Hall PTR.
- Li, Y., Hao, Y., & Wang, C. (2010). A Research on the Robust Digital Watermark of Color Radar Images. *Information and Automation (ICIA), 2010 IEEE International Conference on*, 20-23 June 2010, 1091-1096. doi: 10.1109/ICINFA.2010.5512166.
- Marrer, G. (2004). *Fundamentals of Programming: With Object Orientated Programming*. USA: Gary Marrer.
- Maximus, U. *Steganomaximus*. Retrieved 20 march, 2011, from <http://www.planet-source-code.com/vb/scripts/ShowCode.asp?txtCodeId=38372&lngWid=1>
- Meena, M, K., Kumar, S., & Gupta, N. (2011). Image Steganography tool using Adaptive Encoding Approach to Maximize Image Hiding Capacity. *International Journal of Soft Computing and Engineering (IJSCE)*, 2 May 2011, 1(2) 7-11. Retrieved from http://www.ijscce.org/attachments/File/Vol-1_Issue-2/A020031111.pdf
- Meggs, Philip B. (1998). *A History of Graphic Design* (3rd ed). John Wiley & Sons, Inc. ISBN 978-0471291985.
- Meral, H. M., Sankur, B., Ozsoy, A. S., Gungor, T., & Sevinc, E. (2009). Natural Language Watermarking via Morphosyntactic Alterations. *Computer Speech and Language.*, 23(1), 107-125. doi: 10.1016/j.csl.2008.04.001.
- Merriam-Webster, I. (2003). *Merriam-Webster's Collegiate Dictionary* (11th ed). USA: Merriam-Webster, Inc. Retrieved from <http://books.google.com.my/books?id=TAAnheeIPcAEC>
- Mohanty, S. P. (1999). *Digital Watermarking: A Tutorial Review* (Masters Project Report), Dept. Of Electrical Engineering. Indian Institute of Science, Bangalore, India. Retrieved from <http://www.cse.unt.edu/.../MohantyWatermarkingSurvey1999.pdf>
- Moulin, P., & O'Sullivan, J, A. (2003). Information-Theoretic Analysis of Information Hiding. *Information Theory, IEEE Transactions on*, 10 Mar 2003, 49(3), 563-593. doi: 10.1109/TIT.2002.808134.
- Neil, F. J., & Sushil, J. (1998). Exploring Steganography: Seeing the Unseen. *Computer*, Feb 1998, 31(2), 26-34. doi: 10.1109/MC.1998.4655281.
- Petitcolas, F.A.P., Anderson, R.J., & Kuhn, M.G. (1999). Information Hiding-A Survey. *Proceedings of the IEEE*, 87(7), 1062–1078.
- matis, P., Levicky, D. (2010). Using DCT Coefficients Flipping for Information Hiding in Still Images. *Radioelektronika (RADIOELEKTRONIKA), 2010 20th International Conference*, 19-21 April 2010, 1-4. doi: 10.1109/RADIOELEK.2010.5478578.
- Rojas, R & Hashagen, U. (2000). *The First Computers - History and Architectures*. Cambridge: MIT Press.
- Seitz, J. (2005). *Digital Watermarking for Digital Media*. Hershey, USA: Information Science Publishing.

- Shelly, G. B., Cashman, T. J., & Rosenblatt, H. J. (2009). *Systems Analysis and Design*. USA: Thomson Course Technology.
- Shi, H., & Lv, F. (2010). A Blind Digital Watermark Technique for Color Image Based on Integer Wavelet Transform. *Biomedical Engineering and Computer Science (ICBECS), 2010 International Conference on*, 23-25 April 2010, 1-4. doi: 10.1109/ICBECS.2010.5462499.
- Shu-Kei, Y., Au, O. C., Chi-Wang, H., & Hoi-Ming, W. (2006). Lossless Visible Watermarking. *Multimedia and Expo, 2006 IEEE International Conference on*, 9-12 July 2006, 853-856. doi: 10.1109/ICME.2006.262635.
- Simon, H. (1996). *The Sciences of the Artificial* (3rd Ed). USA: MIT Press.
- Singh, V. (2011). Digital Watermarking: A Tutorial. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*, 10-21. Received from <http://www.cyberjournals.com/Papers/Jan2011/02.pdf>
- Tran, Nicholas. (2002). Hiding Functions and Computational Security of Image Watermarking Systems. *Computer Security Foundations Workshop (CSFW'02). Proceedings of the 15th IEEE*, 295-303. doi: 10.1109/CSFW.2002.1021823.
- Voyatzis, G., Nikolaidis, N., & Pitas, I. (1998). Digital Watermarking-An Overview. *Proceedings of IX European Signal Processing Conference (EUSIPCO)*, 8-11 Sept 2011, 1, 9-12.
- Wang, X., & Ye, J. (2010). Information Hiding Technology in Electronic Notes System. *E-Business and E-Government (ICEE), 2010 International Conference on*, 7-9 May 2010, 1627-1630. doi: 10.1109/ICEE.2010.412.
- Yiqing, L., & Abdulla, W. H. (2010). Robust Audio Watermarking Technique Based On Gammatone Filterbank and Coded-Image. *Signal Processing and Its Applications, 2007. ISSPA 2007. 9th International Symposium on*, 12-15 Feb 2007, 1-4. doi: 10.1109/ISSPA.2007.4555328.
- Yu, Y. H., Chang, C. C., & Lin, I. C. (2007). A New Steganographic Method for Color and Grayscale Image Hiding. *Computer Vision and Image Understanding*, Sep 2007, 107(3), 183-194. doi:10.1016/j.cviu.2006.11.002.
- Zhang, D., Dong, H., & Zhou, C. (2007). Researches on Digital Image Watermarking. *Electronic Measurement and Instruments, 2007. ICEMI '07. 8th International Conference on*, Aug. 16 2007-July 18 2007. 818-821. doi: 10.1109/ICEMI.2007.4350805.
- Zhang, J., & Li, X. G. 2009. The Application Research of Information Hiding Technology in Network Security. *Second International Symposium on Information Science and Engineering*, 26-28 Dec 2009. 208-212. doi: 10.1109/ISISE.2009.70.