Steganography-Based Secret and Reliable Communications: Improving Steganographic Capacity and Imperceptibility

A thesis submitted for the degree of Doctor of Philosophy

by

Adel Almohammad

Department of Information Systems and Computing

Brunel University

August, 2010

ABSTRACT

Unlike encryption, steganography hides the very existence of secret information rather than hiding its meaning only. Image based steganography is the most common system used since digital images are widely used over the Internet and Web. However, the capacity is mostly limited and restricted by the size of cover images. In addition, there is a tradeoff between both steganographic capacity and stego image quality. Therefore, increasing steganographic capacity and enhancing stego image quality are still challenges, and this is exactly our research main aim. Related to this, we also investigate hiding secret information in communication protocols, namely Simple Object Access Protocol (SOAP) message, rather than in conventional digital files.

To get a high steganographic capacity, two novel steganography methods were proposed. The first method was based on using 16x16 non-overlapping blocks and quantisation table for Joint Photographic Experts Group (JPEG) compression instead of 8x8. Then, the quality of JPEG stego images was enhanced by using optimised quantisation tables instead of the default tables. The second method, the hybrid method, was based on using optimised quantisation tables and two hiding techniques: JSteg along with our first proposed method. To increase the steganographic capacity, the impact of hiding data within image chrominance was investigated and explained. Since peak signal-to-noise ratio (PSNR) is extensively used as a quality measure of stego images, the reliability of PSNR for stego images was also evaluated in the work described in this thesis. Finally, to eliminate any detectable traces that traditional steganography may leave in stego files, a novel and undetectable steganography method based on SOAP messages was proposed.

All methods proposed have been empirically validated as to indicate their utility and value. The results revealed that our methods and suggestions improved the main aspects of image steganography. Nevertheless, PSNR was found not to be a reliable quality evaluation measure to be used with stego image. On the other hand, information hiding in SOAP messages represented a distinctive way for undetectable and secret communication.

DEDICATION

This thesis is dedicated, with deepest love and everlasting respect, to numerous precious persons.

To my parents for their continuous love, support and encouragement which helped me to achieve my dream.

To my wife and daughter who I am indebted to, without their continuous sacrifices and patience the completion of this project could not be possible.

To my brothers and all my friends, for their support and patience throughout these stressful years.

To the principles of each and every ambitious person who knows that patience and hard work make the dreams.

With my love…
Adel Almohammad
10 Ramadan, 1431
20 August, 2010

ACKNOWLEDGEMENTS

Firstly, all praise is due to Allah (Glorified and Exalted is He), without his immeasurable blessings and favours none of this could have been possible.

During the course of this research, there have been numerous people who have provided me with guidance, inspiration and support for completing this thesis. I am indebted to my supervisor Dr. George Ghinea for his continuous encouragement and support at all stages of this research. His involvement proved vital to the completion of this thesis. I would like to thank him for his useful and valuable advice. I also owe thanks to Professor Robert M. Hierons, my second supervisor, for his insightful and encouraging comments and suggestions throughout this research.

On the personal front, I would like to express my gratitude to my family, for their continuous love, support and sincere prayers during the course of this research. I must also express my thanks to all my colleagues in the Department of Information Systems and Computing at Brunel University. I will also like to thank my friends and other participants who helped me a lot by participating and motivating me in the course of my research.

Lastly but not the least, I will like to thank the Ministry of Higher Education in Syria and the University of Aleppo for providing me a chance and funding to keep this research going.

DECLARATION

The following papers have been published (or have been submitted for publication) as a direct result of the research discussed in this thesis:

Journals:

1. Almohammad, A., and Ghinea, G., (2010). Stego Image Quality: Objective and Subjective Perspectives. Submitted to *IEEE Transactions on Image Processing,* IEEE.
2. Almohammad, A., and Ghinea, G., (2010). Information Hiding in Grayscale and Colour Images. Submitted to *Signal Processing: Image Communication,* Elsevier.

Reviewed Conferences:

1. Alrouh, B., Almohammad, A., and Ghinea, G., (2010). Information Hiding and SOAP Messages. *The 5th International Conference for Internet Technology and Secured Transactions (ICITST 2010), London, UK, 8-11 November, 2010,* (Accepted), IEEE.
2. Almohammad, A., and Ghinea, G., (2010). Stego Image Quality and the Reliability of PSNR. *International Conference on Image Processing Theory, Tools and Applications (IPTA 2010), Paris, France, 7-10 July, 2010*, 215-220, IEEE.
3. Almohammad, A., and Ghinea, G., (2010). Image Steganography and Chrominance Components. *10th IEEE International Conference on Computer and Information Technology (CIT 2010), Bradford, UK, 29 June -1 July, 2010,* 996-1001, IEEE.
4. Almohammad, A., Ghinea, G., and Hierons, R. M., (2009). JPEG Steganography: a Performance Evaluation of Quantisation Tables. *International Conference on Advanced Information Networking and Applications (AINA 2009), Bradford, UK, 26-29 May, 2009,* 471-478, IEEE.
5. Almohammad, A., Hierons, R. M., and Ghinea, G., (2008). High Capacity Steganographic Method Based Upon JPEG. *The Third International Conference on Availability, Reliability and Security (ARES 2008), Barcelona, Spain, 4-7 March, 2008,* 544-549, IEEE.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1: Information Security and Steganography

1.1 Overview

Digital steganography is the art and science of hiding communications; a steganographic system thus embeds secret data in public cover media so as not to arouse an eavesdropper's suspicion. A steganographic system has two main aspects: steganographic capacity and imperceptibility. However, these two characteristics are at odds with each other. Furthermore, it is quite difficult to increase the steganographic capacity and simultaneously maintain the imperceptibility of a steganographic system. Additionally, there are still very limited methods of steganography to be used with communication protocols, which represent unconventional but promising steganography mediums.

Digital image steganography, as a method of secret communication, aims to convey a large amount of secret data, relatively to the size of cover image, between communicating parties. Additionally, it aims to avoid the suspicion of non-communicating parties to this kind of communication. Thus, this research addresses and proposes some methods to improve these fundamental aspects of digital image steganography. Hence, some characteristics and properties of digital images have been employed to increase the steganographic capacity and enhance the stego image quality (imperceptibility).

This chapter provides a general introduction to the research by first explaining the research background. Then, the main motivations of this study and the research problem are defined and discussed. Next, the research aim is identified based on the established definition of the research problem and motivations. This chapter ends by presenting the structure of the rest of the thesis.

1.2 Research Background

Cole (2003) stated that: *"Security through obscurity says that if you hide the inner workings of your system you will be secure. This philosophy doesn't work when it comes to security, and it doesn't work when it comes to cryptography"*. Thus, information security based on concealed methods or algorithms is known as security through obscurity. Additionally, systems that rely on such kind of security are no longer considered as secure systems since they need to be tested and validated. Hence, the algorithm must be released to the public and have really smart cryptanalysts beat on it for years and if they cannot find a way to break it, then it is considered secure (Cole, 2003).

Communication security should not be based on the secrecy of the communication method used. Various cryptosystems have been used to assure the security of transmitted data. In such systems, data is encrypted before transmission and they have been considered as secure systems. Cryptographic techniques could be vulnerable due to computers' capabilities increasing fastly during last years. Also, the availability of cipher texts enhances this vulnerability since the attackers have the chance to implement cryptanalysis techniques on this system in order to break it. However, this vulnerability can be significantly reduced using steganography, which is a kind of covert communication. Additionally, this technique of information security can be defined as the process of hiding secret data within other public information, such as digital images, in such a way that the existence of secret data is imperceptible and undetectable.

Encryption represents a well-known information security method. It encodes secret information in such a way that only the intended recipient can successfully decode it. Since encryption scrambles the secret message and makes it unreadable, the encrypted message would be suspicious enough to attract attention of eavesdroppers. Thus, the security problem can be solved by hiding the secret data in another cover or object so that it draws no special attention.

Essentially, it is important for any information security algorithm to satisfy three main objectives or requirements: confidentiality, data integrity and authentication (Venkatraman et al., 2004). Neither mere steganography nor simple encryption ensures information security. Like any security technology, steganography is not perfect and it does not address all security requirements. However, it satisfies most of the requirements of secret communication, sometimes in combination with other techniques such as cryptography. As cryptography and steganography complement each other, it is recommended to use these two techniques together for a higher level of security.

1.2.1 Steganography and Cryptography

Cryptography and steganography achieve separate goals. Cryptography conceals only the meaning or contents of a secret message from an eavesdropper. However, steganography conceals even the existence of this message (Lou and Liu, 2002). Furthermore, steganography provides more confidentiality and information security than cryptography since it conceals the mere existence of secret message rather than only protecting the message contents. Therefore, one of the major weaknesses of cryptosystems is that even though the message has been encrypted, it still exists.

Even though both cryptographic and steganographic systems provide secret communications, they have different definitions in terms of system breaking. A cryptographic system is considered broken if an attacker can read the secret message. However, a steganographic system is considered broken if an attacker can detect the existence or read the contents of the hidden message. Moreover, a steganographic system will be considered to have failed if an attacker suspects a specific file or steganography method even without decoding the message. As a result, this consideration makes steganographic systems more fragile than cryptography systems in terms of system failure. Additionally, steganographic systems must avoid all kinds of suspicion in order to achieve security and not be considered failed systems.

Since steganography adds an extra layer of protection to cryptography, combining steganography and encryption gives the ultimate in private communication. Therefore, the purpose of steganography is to complement cryptography and to avoid raising the suspicion of system attackers but not to replace cryptography.

1.2.2 Steganography and Watermarking

Steganography aims to hide the very existence of communication by embedding messages within other cover objects. However, watermarking aims to protect the rights of the owners of digital media such as images, music, video and software. Even if people copy or make minor modification to the watermarked file, the owner can still prove it is his or her file. Thus, both of steganography and watermarking are forms of data hiding and share some common characteristics. Nevertheless, the goal of steganography is the embedded message while the goal of watermarking is the cover object itself.

Watermarking is a data hiding technique that protects digital documents, files, or images against removal of copyright information. Even if someone knows that a watermark is exist (i.e. visible watermarking) in a given object, it should be impossible to remove the watermark from the watermarked object without causing a distortion or destroying the original (watermarked) object. This aspect or feature of watermarking is known as *"robustness"*. According to the kind of embedded information, two techniques of document marking can be distinguished: watermarking and fingerprinting. Watermarking is the process of embedding a specific copyright mark into digital documents in the same way. On the other hand, in order to detect any break of licensing agreement, a serial number is embedded in every copy of this digital document. This process is known as *"fingerprinting"*. Even if these markings are detected, it should be practically impossible to remove them.

1.2.3 Information Security and Steganography

Essentially, computer and network security have some requirements that should be addressed in order to get secure systems. Thus, in order to determine the performance of a security technology, three key concepts should be analysed: confidentiality, integrity, and availability. Cole (2003) identifies these concepts as follows:

1. "*Confidentiality* deals with protecting, detecting, and deterring the unauthorized disclosure of information". The main goal of cryptography is to garble a plaintext message in such a way that only the intended recipient can read it. This is precisely the goal of confidentiality.
2. "*Integrity* deals with preventing, detecting, and deterring the unauthorized modification of information". An integrity attack is potentially more dangerous than a confidentiality attack. Cryptography addresses integrity by performing a digital signature check across information.
3. "*Availability* relates to preventing, detecting, or deterring the denial of access to critical information". Cryptography can prevent confidentiality and integrity attacks, but it can not prevent availability attacks. Cryptography, like any other network security technology, is not a silver bullet. Therefore, it must be combined with other techniques to achieve a robust security solution.

In addition to the three key concepts of security, two other security goals are critical relative to cryptography: authentication and non-repudiation (Cole, 2003).

1. *Authentication*: "In most transactions you need to be able to authenticate or validate that the people you're dealing with are who they say they are".
2. "*Non-repudiation* deals with the ability to prove in a court of law that someone sent something or signed something digitally". Without non-repudiation, digital signatures and contracts would be useless.

Steganography, as a secret communication method, achieves most of these requirements since there is no method that can address all security concepts (Cole, 2003). Therefore, the key concepts of security that apply for steganography and equally take into consideration the main principles of information security requirements (discussed above) are as follows (Cole, 2003):

1. *Confidentiality:* Cryptography achieves the confidentiality by preventing unauthorized persons, who can see the information, from gaining access to this information. With steganography, unauthorized people do not even know there is secret data there.
2. *Survivability* means that all data processing takes place between sender and receiver does not destroy the hidden information. Additionally, this received information must be extractable and readable.
3. *No Detection*: Steganography fails if someone can easily detect where you hid your information and find your message. Therefore, even if someone knows how the steganography method embeds the secret information, he or she cannot easily find out that you have embedded data in a given file.
4. *Visibility:* The stego file must be undetectable and there must be no visible changes to the stego file.

The main goal of steganography is exactly the confidentiality of embedded data. Unlike cryptography which hides the content or meaning of the secret data, steganography hides the very existence of this data. Therefore, unauthorized people do not even know there is secret data there. From a confidentiality standpoint, steganography provides a higher level of information protection than cryptography.

To some extent, the survivability of data represents the integrity of this data since both of them (survivability and integrity) are aiming to prevent the manipulation of the transmitted data. In our study and proposed methods, like the most of other steganography techniques, consider the passive warden scenario (See section 2.5). The passive warden is restricted from modifying the contents of stego files during the communication process and he/she has the right to prevent or permit the message delivery (Cox et al., 2008). Therefore, if the stego file is received, then it will be exactly the file which is sent without any modification or changes added during the transmission process. Most steganography research is concerned with such kind of scenarios which assumes that the integrity of secret data is preserved between the sender and the receiver. Thus, maintaining the integrity of secret message means that the embedded message by the sender is exactly the same message extracted by the receiver (intact secret message). However, the integrity of the stego image means that the stego image sent by the sender is exactly the same stego image received by the receiver (identical and have similar statistical properties).

The goal of steganography is secret communication. Therefore, steganography aims to prevent others from thinking that such communication is taking place. Essentially, steganographic systems should identify the redundant (insignificant) bits of cover files or medium. Therefore, any modifications to these redundant bits should not destroy the integrity of these mediums. As a result, preserving the integrity of cover files enhances the undetectability of steganography (Anderson and Petitcolas, 1998). Usually, hiding secret data using steganography adds a slight change to the stego file properties. This makes the detection of steganography presence difficult or almost impossible. Additionally, even if the hiding method used is publically known, nobody should be able to prove the existence of hidden data. However, undetectability could be mainly achieved by adding no visible changes to the cover file. After the data hiding process, people have to see no visible traces in the stego file. Hence, if someone can tell or prove that a given file (i.e. stego file) has been modified in some way then the steganography is unsuccessful. For image-based steganography, the fidelity (i.e. PSNR) of the stego image is usually used to measure and evaluate the undetectability of steganography method used. However, *Fidelity* refers to our ability to detect differences between cover image and stego image. Thus, if we can not detect any difference between these two images then this steganography method is imperceptible. However, the integrity of the cover image is not preserved with steganography since some parts of the cover file should be changed or modified in order to hide the secret message and get the stego file.

1.3 Motivations and Research Problem

Free speech, taken for granted in many democratic countries, is not possible in many other countries. Since some governments restrict the use of encryption, this has motivated people to learn other secret communication methods. Steganography can be considered as a solution to exchange information and news between people or civil rights organisations around the world over the Internet without any fear of the message being detected.

On the other hand, there has been a great concern about preserving the intellectual property rights of digital media such as text, image, audio, and video. Another concern regarded the ban of using encryption techniques on the Internet. This has significantly motivated the interest in information hiding techniques over the recent years. Additionally, the growing concern about the ease of copying, reproducing, and theft of digital works has motivated and increased the interest of publishing and broadcasting industries in watermarking and authentication techniques.

Cryptography converts the secret information into a scrambled code in such a way that only the intended recipient, who has the decoding key, can read this secret message. Furthermore, a third party can tell that a secret message has been sent from one party to another but he/she can not read this message. However, steganography hides the very existence of this secret message. Thus, a third party can not even know that a secret message has been embedded within a stego file or sent over a network.

Digital images, particularly those using the JPEG format, are the most commonly used files for steganography since they are the most widely files available over the Internet and Web. Using encryption, the size of the secret message is not an issue but it represents a significant challenge for steganography since the size of cover files (i.e. image) mostly restricts the steganographic capacity. Furthermore, embedding a secret message in a cover image may change or modify some characteristics of this cover image and therefore attract the eavesdroppers' attention. Thus, the steganographic capacity and stego image imperceptibility are the most important aspects of image-based steganographic systems. Nevertheless, the amount of data that we can hide within a cover image is limited. Additionally, hiding more data within a given cover image makes the stego image more suspicious and therefore more detectable. Therefore, there is a kind of tradeoff between the steganographic capacity and the stego image imperceptibility.

1.4 Research Aim

The research motivations section has highlighted that the steganographic capacity and stego image imperceptibility are the most important aspects of image steganographic systems. Essentially, either increasing the steganographic capacity while maintaining the imperceptibility (stego image quality) or enhancing the imperceptibility while maintaining the steganographic capacity represents a contribution. This is exactly our research main aim, increasing the steganographic capacity and/or enhancing the stego image quality.

Traditional steganography uses digital files as cover of secret data. However, this carries the threat of detecting the stego file as these files are usually saved. Thus, available stego files may increase the opportunity of steganalysts to detect these stego images by applying various steganalysis techniques. Alternatively, communication protocols messages such as the ones of the Simple Object Access Protocol (SOAP) leave almost no trail as they are normally deleted after they have been received and the actual data de-serialized. Unlike the steganography methods used with conventional covers such as digital images, there are very few steganography methods that can feasibly be used with communication protocols such as SOAP. Thus, our research also aims to find out an undetectable steganography method based on SOAP messages.

1.5 Thesis Outline

This thesis is structured around eight chapters as follows.

1. **Chapter One** provides an introduction and background of this research. It defines the main motivations for researching digital steganography and explains the main research aim. Additionally, this chapter discusses the research context by providing a brief background of the research related domains.

2. **Chapter Two** provides a general review of techniques and methods of digital steganography used for both traditional covers and networks. It also discusses the fundamental properties of steganographic systems that are mainly used to evaluate and assess these systems. Additionally, this chapter explains the general approaches and common metrics used to measure and quantify the main aspects of steganographic systems.

3. **Chapter Three** Analyses and explains in-depth the nuts and bolts of JPEG steganography. Also, it discusses and describes the main aspects and attributes of SOAP steganography to provide the background necessary for the work discussed in this thesis. In addition, it presents a literature review on relevant steganographic systems; explains and critiques their approaches to determine the main issues that need to be addressed to improve digital steganography. This chapter also states the thesis objectives and briefly describes the methodology used to achieve these objectives.

4. **Chapter Four** develops two novel JPEG-based steganography methods to get high steganographic capacity and good stego image quality. These two proposed methods are based on modified quantisation tables and they are empirically evaluated and validated. Additionally, this chapter examines the main steganography aspects of these developed methods. Then, it evaluates our developed steganography methods against the JSteg, F5 and JMQT methods.

5. **Chapter Five** explores the usability of different colour image components for steganography to increase the steganographic capacity. This chapter provides in-depth analysis of the impact of data hiding in different image components on the main steganography aspects. For a given steganography method, this chapter also examines and evaluates the performance of both grayscale and colour versions of cover images in terms of the main steganography aspects.

6. **Chapter Six** evaluates the reliability of the peak signal-to-noise ratio (PSNR) metric as a measure of stego image quality. This chapter identifies and develops a subjective evaluation framework to examine the quality of stego images. Thus, it examines the relationship between the PSNR and the subjective quality of stego images.

7. **Chapter Seven** develops a novel SOAP-based steganography method. This method is based on rearranging the attributes or sub-elements of some SOAP message elements and is empirically evaluated and validated in this chapter.

8. **Chapter Eight** summarizes research findings and conclusions. It also presents an overview of the main research contributions to knowledge. The limitations of the research are also discussed and directions for further research explored in this chapter.

Chapter 2: Steganography and the Art of Covert Communication

2.1 Introduction

This chapter presents an overview of digital steganography basics as a method of covert communication. In this chapter, the steganography is defined, the different file types that can be used as cover files are described, and the main components of steganographic systems are identified. Therefore, the procedures of sending a secret message using steganography from one party to another is illustrated and explained. Assuming that there is an attacker monitoring all messages transmitted between these two communicating parties, we explain the different kinds of attacks that digital steganography can be subject to. Then, the main methods of steganography classification and the key techniques of steganography are presented.

Steganography techniques explain how we embed a secret message within a cover file. Therefore, network steganography techniques are presented to explain how we can hide a secret message or a stego file over a network. Also, our study aims to investigate the possibility of using steganography in SOAP protocols employed in Web services. Thus, the basics of Web services and SOAP messages are also presented in this chapter.

In order to measure the efficiency of a steganography technique, we have to measure some fundamental properties of this steganography technique and then compare the values of these properties with that of other techniques. Hence, these properties and the methods used to measure them are explained in detail. Finally, the steganalysis techniques that can be used to defeat steganography are also classified and explained in this chapter.

2.2 Steganography Defined

Linguistically, steganography means secret writing since the word *“Steganography”* is originally made up of two Greek words steganos (*secret*) and graphy (*writing*). Practically, it means the art and science of hiding or camouflaging secret data in an innocent looking dummy container in such a way that the existence of the embedded data is imperceptible and undetectable (Bailey et al., 2004; Cachin, 1998; Kahn, 1996). Therefore, steganography is the process of hiding secret data within public information.

Secret data can be a plaintext or ciphertext, or any kind of data that can be hidden in digital media. Since all kinds of secret data must be translated into binary, we always hide binary data whatever this secret data or file is. However, types of cover files that can be used for steganography will be presented in the next section. Basically, digital steganography can be considered as a multidisciplinary field since it combines digital signal and data compression methods, information theory, signal coding theory, digital communication theory, digital signal processing, cryptography and the theory of human visual perception, all employed to satisfy the needs of information security (Cole, 2003; Rabah, 2004).

2.3 Cover Files Used for Steganography

Basically, cover files represent the container of hidden data or secret messages. Additionally, some parts or characteristics of cover files will be modified, changed, or manipulated in order to hide these secret messages. However, these manipulations, which occur during the hiding procedure, should remain imperceptible to anyone not involved in the communication process. Therefore, the appearance or format of cover files must remain intact after hiding the secret data. As a result, it is not possible to use all types of files or data as cover files of steganography since every cover file must have a sufficient redundant area to be replaced by the secret message (Katzenbeisser and Petitcolas, 2000).

There is a variety of files that can be used as cover files of steganography such as executable files (i.e. exe files), HTML files, XML files and TCP headers. Essentially, many kinds of digital media such as image, audio, text, and video files can be used as cover files of steganography. However, the ability of such files to embed secret data depends on the availability of redundant or insignificant areas within these files. Thus, the cover files represent the container of hidden data and their size may determine the secret data size that can be embedded. To this end, cover files are the fundamental component of steganographic systems. However, the relationship between cover files and the other main components of steganographic systems will be discussed in the next section.

2.4 Main Components of Steganographic Systems

Steganographic systems have one general principle, described by Katzenbeisser and Petitcolas (2000) as follows. The sender (Alice), who wants to send a secret message (m) to the recipient (Bob), randomly chooses a harmless cover file (c). Afterwards, Alice embeds the secret message (m) in the cover (c) and probably uses a stego key (k). As a result, Alice gets a stego file (s) which must be undistinguishable from the cover file (c) neither by a human nor by a computer system. Therefore, the stego file (s) represents the original (cover) file (c) along with the secret message (m) embedded inside this cover file. Then, Alice transmits the stego file (s) to Bob over a communication channel. The purpose of the system is to prevent Wendy (a third party) from observing or noticing the hidden message (m). On the other side, Bob extracts the embedded message (m) since he knows the embedding method and the stego key (k) used in the embedding process (Katzenbeisser and Petitcolas, 2000).

Only the transmitter and the intended recipient should have the stego key. Therefore, most of steganographic systems prompt users to provide a stego key or password when they try to embed information in a cover file. Figure 2.1 shows the general principle of image-based steganographic systems.

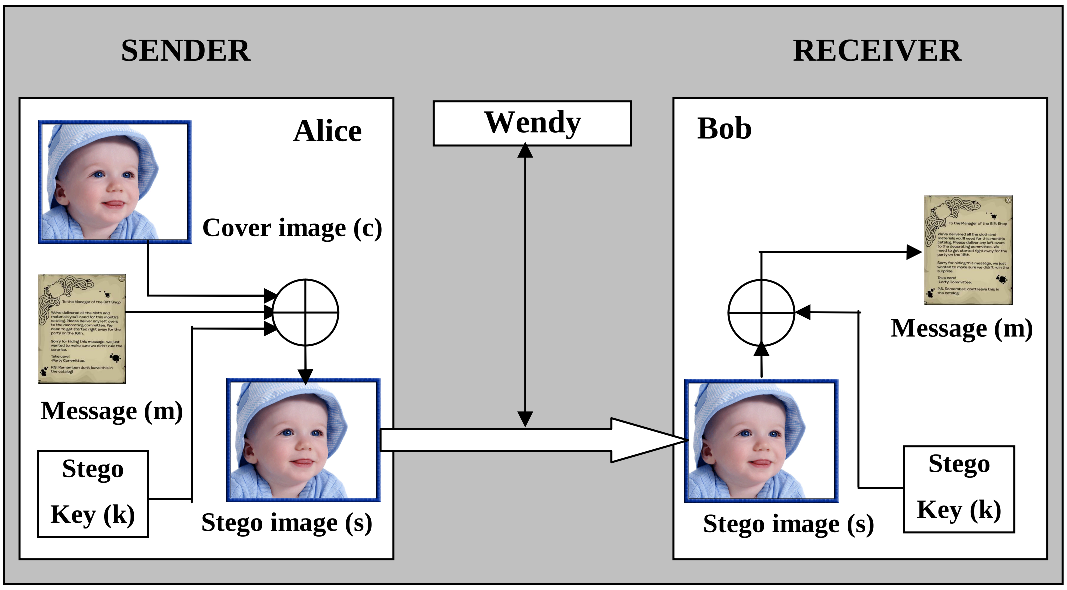


Figure 2.1: General Principle of Image Steganographic System

Sometimes, attackers (like Wendy) can detect a hidden message in a stego file and determine how the message was embedded, but they are unable to extract the hidden message. This system is known as a secure steganographic system because the secret message is unreadable unless one has the related stego key. Therefore, these stego keys must be chosen as strong as possible in order to prevent attackers from breaking the steganographic systems using all possible stego keys (Cox et al., 2008).

Thus, the security of steganographic systems must satisfy Kerckhoff's principle. Accordingly, the security of steganographic systems must be based on the assumption that attackers have full knowledge of the steganographic system design (the embedding and extracting algorithm). However, attackers only miss the stego key to suspect that a secret communication is taking place. Therefore, most of steganographic systems available nowadays meet this principle (Rabah, 2004). If the stego key used for embedding and the one used for extraction are the same, the steganographic system is symmetric. However, if these two stego keys are different then the steganographic system is asymmetric (Liu and Liao, 2008). The next section now describes three general types of steganography attacks.

2.5 Passive, Active, and Malicious Wardens

Steganography attackers are the interceptors of stego files in the communication channels in order to detect hidden messages in these stego files. In the *"Prisoners' Problem"*, Simmons (1983) links these attackers to wardens mediating the communication between two prisoners (the communicating parties, Alice and Bob). Three general types of steganography attacks can be distinguished and therefore three scenarios of digital steganography can be recognized in order to meet attacks. The first technique deals only with protecting the steganographic system against message detection in a passive attack, while the second protects the message against detection and modification by an active attack. However, the third technique protects the steganographic system against the forgery of a malicious attack. The next three subsections explain these three attacks in detail.

2.5.1 Passive Attack

Passive wardens just observe the communication without any interference. Therefore, if the warden is restricted from modifying the contents of stego files during the communication process, it is called a passive warden. The passive warden only has the right to prevent or permit the message delivery. Therefore, the communication between two parties will be blocked if the warden suspects that a secret communication is taking place. Otherwise the communication will be relayed (Cox et al., 2008).

Simmons (1983), in the *"Prisoners' Problem"*, clearly illustrated steganography with a passive attack. Alice and Bob are arrested and thrown in two different cells. They want to develop an escape plan, but all their communication is arbitrated by a warden (Wendy). Alice and Bob must communicate invisibly in order not to arouse Wendy's suspicion since she will transfer them to a high-security prison if she notices any suspicious communication. Alice and Bob will succeed only if Alice can hide and send information to Bob in such a way that Wendy does not become suspicious (Simmons, 1983).

Currently, most steganography techniques consider the passive warden scenario in which the warden does not interfere with the stego file in any way. Therefore, most steganography research is concerned with such kind of scenarios (Cox et al., 2008).

2.5.2 Active Attack

If the warden can intentionally modify the contents of stego files during the communication process, we are dealing with an active warden. An active attack is thus the process of altering stego files and introducing distortion during the communication process in order to prevent secret communication (Cox et al., 2008). In such kind of attacks, the attacker can capture and modify a stego file sent from Alice to Bob and then forward this modified file to Bob. Even though there is no suspicion that secret communication is taking place, the warden may modify the stego file or add random noise to the transmitted stego file in order to destroy any secret message that might be present (Katzenbeisser and Petitcolas, 2000).

On the other hand, many steganography applications such as watermarking and fingerprinting publicly reveal the existence of hidden information in some files. Therefore, protecting this hidden information against active attacks is a different problem (Cachin, 1998). However, steganographic systems that resist such kind of attacks and preserve the readability of secret data at the receiver endpoint are known as robust systems. These kinds of systems are most likely to be used for watermarking and authentication applications rather than for secret communication methods.

2.5.3 Malicious Attack

If the warden fakes messages or acts as one of the communication partners during the communication process, it is called a malicious warden. In the malicious attack, the warden may intentionally try to remove the hidden message, impersonate one of the communicating parties, or trick them (Cox et al., 2008). Therefore, in this kind of attack, the warden can pass his own message to a specific communication partner as if it is sent by the other communication partner. However, this attack is the most difficult and rare among all these three main attacks since the attacker here needs to know the stego key shared between the communicating parties. Additionally, he/she may have to know the personal encryption key of the sender person. Such kind of attacks are considered infrequently in both steganography and watermarking applications since it is difficult to apply and easy to be detected by the actual receiver.

2.6 Steganography: Methods of Classification

There are two general approaches to classify steganographic systems. The first approach is based on the type of cover file while the second approach is based on the hiding method or the layout of modification used in the embedding process (Cole, 2003; Katzenbeisser and Petitcolas, 2000). These two general classification approaches of steganography are explained in the next subsections.

2.6.1 Cover-Type Based Classification

Since many kinds of digital media can be used as cover files of steganography, the first approach of classification breaks down steganography according to the type of the cover file used. However, the properties of these cover files vary from one type to another and these properties control how the secret data can be hidden in these cover files. To this end, knowing the type of cover file can give us an indication or idea where the secret data might be hidden (Cole, 2003).

Mostly, steganographic systems are classified according to the cover file used. Accordingly, different steganography types can be distinguished such as: image, audio, video, text, and HTML steganography. For example, the steganographic system that uses digital images as cover files is an image-based steganographic system.

2.6.2 Hiding Method-Based Classification

Regardless of the cover type used for data hiding, steganography can be classified according to the method used to hide secret data. Furthermore, this approach of steganography classification is the most preferred approach in the steganography research community. Accordingly, there are three ways to hide secret data in cover files: insertion-based, substitution-based, and generation-based method (Cole, 2003; Kipper, 2004).

2.6.2.1 Insertion-Based Method

This method depends on finding some areas in cover files which are usually ignored by applications that read this cover file and then embedding the secret data in these areas. Since this method inserts the secret data inside the cover file, the size of the stego file would be larger than the size of the cover file. As a result, the main advantage of this method is that the contents of the cover file would not be changed after the embedding process since this method relies on accumulating or adding the secret data to the cover file.

An example of such a method is using a Word document to write a secret message in the areas between the end-text and begin-text markers. Because of the configuration of Word documents, which depends on ignoring anything written in such areas, the hidden message will not appear when this document is viewed in Word (Cole, 2003; Kipper, 2004).

2.6.2.2 Substitution-Based Method

Unlike the insertion-based method, this method does not add the secret data to the cover file data. However, substitution-based method depends on finding some insignificant regions or information in cover files and replacing this information with the secret data. Therefore, the sizes of both the stego file and the cover file are similar since some of the cover data is just modified or replaced without any additional data. However, the quality of the cover file can be degraded after the embedding process. Additionally, the limited amount of insignificant information in cover files restricts the size of secret data that can be hidden (Cole, 2003; Kipper, 2004).

2.6.2.3 Generation-Based Method

Unlike both methods explained above, this method does not need a cover file since it uses secret data to generate appropriate stego files. One of the steganography detection techniques depends on comparing cover files with their stego files. Therefore, one advantage of the generation-based steganography is preventing such kind of detection since only stego files are available and there is no cover files used. The major limitation of this method is the limited stego files which can be generated. Moreover, the generated stego files might be unrealistic files for end users (e.g. an image contains different shapes and colours without any sense or a text without any meaning). Therefore, the main media for such techniques are random-looking images and English text files (Cole, 2003).

2.7 Steganography Techniques

In addition to the general methods of information hiding presented above, many steganography techniques have been proposed during the last few years. These techniques differ in the mechanism or principle being used to hide a secret message or the changes that are taking place during the entire process of embedding. Therefore, there are six categories of steganography techniques: substitution systems, transform domain techniques, spread spectrum techniques, statistical methods, distortion techniques, and cover generation techniques (Katzenbeisser and Petitcolas, 2000; Kipper, 2004).

2.7.1 Substitution Systems

For a given cover file, it is important to find out some areas or data that can be modified without having any significant effects on this cover file (Cole, 2003). Therefore, a secret message can be embedded by replacing the redundant or insignificant parts of a cover file with secret message bits, without adding any significant noise to this cover file (Kipper, 2004).

Generally, digital covers have a large number of redundant bits (e.g. least significant bits (LSB)). In the substitution technique of steganography, the bits of the secret message substitute the LSB of the bytes of the cover file without causing a drastic change to this cover file. Moreover, the LSB technique is a spatial domain technique since it embeds the secret bits directly in the cover file. Since LSB substitution technique is relatively quick and easy to use, it is the most common technique used for digital steganography and especially with digital images. However, the embedded information using the LSB technique is highly vulnerable and could be destroyed entirely by applying a slight modification to the stego image such as JPEG compression (Rabah, 2004).

2.7.2 Transform Domain Techniques

Unlike spatial domain techniques (e.g. LSB technique), transform (frequency) domain techniques hide secret data in significant parts of the cover file. Therefore, frequency domain techniques are considered more robust to attacks than spatial domain techniques. Hence, most of robust steganographic systems known today rely on frequency domain techniques. There are many transforms used to map a signal into the frequency domain. Discrete cosine transform (DCT), discrete wavelet transform (DWT), and discrete Fourier transform (DFT) are methods used as mediums to embed secret data in digital images. However, when we add a slight noise or secret data to some frequency domain components, it changes the whole image rather than changing only this part of the image. Thus, secret and embedded data will be spread across the entire image and will not be concentrated on one certain area or region.

2.7.3 Spread Spectrum Techniques

Marvel et al. (1998) define spread spectrum communication as *"the process of spreading the bandwidth of a narrowband signal across a wide band of frequencies"*. In spread spectrum steganography, the frequency domain of the cover file is considered to be a communication channel and the secret message as a signal that is transmitted through it. Since the secret message is spread through a wide frequency band, this technique is relatively robust against stego file modification or message removal (Marvel et al., 1998).

2.7.4 Statistical Techniques

These techniques embed only one bit of secret data in a cover file. Therefore, it is known as "1-bit" steganography scheme. If "1" is hidden in a cover file, some statistical characteristics (e.g. entropy and probability distribution) of this cover file must be changed significantly to clearly indicate the existence of a message. However, if the hidden bit is "0", the cover file is left unmodified. Therefore, this technique entirely depends on the ability of the receiver to differentiate between changed and intact cover files (Katzenbeisser and Petitcolas, 2000; Kipper, 2004).

2.7.5 Distortion Techniques

Most of the steganography techniques are blind, which means that a receiver does not need the original cover file to extract the hidden message from the corresponding stego file. However, if a distortion technique is used, the receiver requires the original cover file in order to recover the secret message. For a receiver, the embedded message is the difference between the modified cover file received (the stego file) and the original cover file (Katzenbeisser and Petitcolas, 2000).

2.7.6 Cover Generation Techniques

All steganography techniques presented above need cover files to be used as containers for secret data. However, cover generation techniques do not require cover files but instead they create stego files just for the purpose of hiding information (Katzenbeisser and Petitcolas, 2000).

2.8 Network Steganography Techniques

The main goal of information hiding and steganography is secret communications. Therefore, steganography aims to provide a secure, secret and easy way of communication among people. Information hiding in a file is useful practice, but the main purpose of steganography is being able to transmit or send that secret message to others over a network or the Internet. Therefore, four types of network-based information hiding techniques can be distinguished.

2.8.1 Information Hiding in an Attachment

Basically, this technique represents the most frequent form of information hiding techniques used. Usually, different steganography methods can be used to hide a secret message within a cover file. Then, the stego file would be attached to some other form of network traffic. However, there are three common methods to do this, by email, by file transfer such as file transfer protocol (FTP), or by posting a stego file on a web site (Cole, 2003).

2.8.2 Information Hiding in a Transmission

In the techniques covered so far, we needed one tool, method, or program to hide a secret message within a cover file. Then we needed another program to transmit the stego file to the intended recipient. Therefore, a steganography program and email program are needed in order to send our secret information. However, the technique that only uses a single program to hide secret information in a cover file and then sends the stego file is known as information hiding in a transmission. Therefore, this kind of steganography technique has a built-in transfer feature that enables stego files to be transmitted to the other communication parties (Cole, 2003).

2.8.3 Information Hiding in Network Headers

In networks, the main information required to route data packets properly is included in the Internet Protocol (IP) header. Furthermore, using steganography requires some redundant or insignificant parts or fields in the cover file that can be changed or replaced without affecting the communication. Network headers contain many fields that are either optional or unused for normal transmission.

The IP header contains one field, the IP identification number, which can be modified without having any effect on its operation. Therefore, we can put any number as an IP identification number and the protocol will still function properly. So, IP headers represent a good possible candidate for steganography.

Furthermore, transmission control protocol (TCP) headers have sequence and acknowledgment numbers used for reliable communication. The first number indicates how much data was sent while the second number indicates how much data was received. However, the values of these two numbers are randomly generated during the initial handshake. Therefore, the secret data can be hidden in these fields for the first packet only, since we can not use them after establishing a communication (Cole, 2003).

2.8.4 Information Hiding in an Overt Protocol

In this kind of steganography, the appearance, shape or format of secret data is modified or adjusted in order to make this data looks like an overt protocol. Therefore, we can put this data in normal network traffic without attracting any suspicion. Usually, Web traffic contains HTML. Thus, we can add symbols such as < > </> to the secret data in order to make this data looks like normal Web traffic. Essentially, the technique of making some kind of data look like another kind of data is called data camouflaging (Cole, 2003).

Simple Object Access Protocol (SOAP) is a data transfer protocol over the Web. Additionally, SOAP represents a communication protocol between clients and web services. Since this research investigates the techniques of information hiding in communication protocols, we explain and discuss the basics of SOAP message in the next section.

2.9 Web Services and SOAP Protocol

It is well known that the Web represents the world's premier network and that the Extensible Mark-up Language (XML) represents the world's premier data representation format (Newcomer, 2002). Additionally, XML is playing an increasingly important role in the exchange of a wide variety of data on the Internet. As such, XML documents are considered to be a language of Web pages and digital contents. Moreover, they are used for the data exchange between organizations. Whilst, Web services require a data exchange in the form of XML documents, the Simple Object Access Protocol (SOAP) exactly provides this kind of data transport. Therefore, SOAP supports a common data transfer protocol for effective communication over the Web (Newcomer, 2002).

Web services provide a platform neutral and programming language independent technology that supports interoperable machine-to-machine interaction over a network. Moreover, clients and other systems interact with the web service using a standardized XML messaging system, such as the SOAP protocol (Zhang et al., 2007). Therefore, structured and typed information can be exchanged between peers of distributed environment using SOAP messages.

A SOAP protocol is the most important of all Web services technologies since it achieves the most important of Web services: transporting the data from one place to another over the network. Thus, SOAP transfers an XML document from one place to another across the Web and other types of networks to reach a Web service implementation (Newcomer, 2002).

In summary, the SOAP supports a common data transfer protocol for effective communication over the Web. The interaction between service providers and requesters in Web Services occurs via XML-based SOAP messages. Thus, secret data can be embedded in SOAP messages and sent over the network to an intended destination. Hence, SOAP messages can be used as steganography cover files and may provide an imperceptible and reliable way of covert communication. In the next chapter, we will investigate the methods of data hiding which may fit SOAP messages. However, digital images represent the most common cover files of steganography. In the next section, we explain the reasons behind the wide-use of digital images for steganography.

2.10 Digital Images and Steganography

Steganography in digital media is the most developed field of information hiding nowadays. Wayner (2002) stated that: *"There are millions of images floating about the Net used as window dressing for Web sites and who knows what. Any one could hijack the bits to carry their own messages"*. Additionally, information hiding in the noise of digital images represents one of the most popular methods of steganography and hence digital image steganography is growing in use and application. As a result, digital images are the most widespread cover files used for steganography due to the insensitivity of the human visual system. Furthermore, digital images can easily be used as cover files without any suspicion because of their omnipresence on the Internet (Artz, 2001; Liu and Liao, 2008).

The purpose of steganography is to keep others from thinking that a secret message even exists within stego files. However, Steganalysts or steganography attackers may suspect some images and may detect the existence of hidden messages within such images. In order to avoid this, the design of almost all steganographic systems takes into account the characteristics of the human vision system (HVS) (Chang et al., 2002). Thus, the success of a steganographic system greatly relies on the limitations of the HVS in order to embed data in cover images (Artz, 2001; Chang and Tseng, 2004). For example, information is embedded in noisy regions and edges of images rather than in smoother regions since the HVS is more sensitive to the degradations in smoother regions (Zeng et al., 2006).

For quantised DCT coefficients in JPEG coding, the human vision system (HVS) is much more sensitive to the noises in low frequency components than those in higher frequencies. This is because the energy of natural images is concentrated on the lower frequency components. Therefore, introducing a distortion to high frequency components is visually acceptable, while this is not so for lower frequencies. Thus, the low frequency coefficients should be avoided when hiding secret data in order to achieve better imperceptibility and security (Chang et al., 2002). This is an example of how we can exploit some HVS characteristics to find a redundant and undetectable area to be used for steganography.

2.11 Fundamental Properties of Steganographic Systems

In order to examine the pros and cons of a steganographic system, many features of this system should be considered. Basically, steganographic systems have two fundamental characteristics which must be investigated in order to evaluate the system. The security or undetectability and the hiding capacity are the most important requirements that must be addressed in every steganographic system (Wang and Wang, 2004). Thus, the effectiveness of a steganography technique can be measured using two key principles: the amount of data that can be embedded and the difficulty of detection of this data (Cole, 2003). Thus, designing information hiding algorithms that are statistically undetectable and can hide a large amount of data is the main goal of steganography (Cox et al., 2008).

2.11.1 Undetectability or Perceptual Transparency (Security)

Generally, a steganographic system fails if an attacker is able to prove the existence of a secret message or if the embedding technique arouses suspicions of attackers. Therefore, steganographic systems can be considered secure if it is impossible for attackers to detect the presence of hidden data in the stego files by using any accessible means. Additionally, the hidden message must be invisible both perceptually and statistically in order to avoid any suspicions of attackers. Thus, a steganographic system is perfectly secure if the statistics of the cover file and that of the stego file are identical. Therefore, the characteristics and attributes of cover files should not be changed and no distortions should be produced during the embedding process. (Venkatraman et al., 2004). However, the presence of statistical anomalies (i.e. histograms and a variety of higher-order statistics) may be used by an adversary to prove that a secret communication is taking place (Cox et al., 2008).

In most steganography publications, the term of security is usually equivalent to undetectability. Therefore, secure steganographic systems refer to imperceptible steganographic systems (Cox et al., 2008). Chang et al. (2002) stated that *"The better quality the stego image has, the more secure the steganography system will be"*. Thus, imperceptible steganographic system means that the hidden information cannot be perceived by the human visual system or other statistical means. Nevertheless, hiding secret information in a cover image may introduce some noise or modulate this cover image (Venkatraman et al., 2004). Yet, the introduced noise must not degrade the perceived quality of stego image in order to get a secure steganographic system.

2.11.2 Capacity or Data Payload

Steganographic capacity is the maximum number of bits that can be embedded in a given cover file with a negligible probability of detection by an adversary. However, the embedding capacity is the maximum number of bits that can be embedded in a given cover file. Therefore, the embedding capacity is likely to be larger than the steganographic capacity (Cox et al., 2008). Moreover, the size of the hidden information relative to the size of the cover image is known as embedding rate or capacity (Venkatraman et al., 2004). Steganographic systems, mainly used for secret communication, aim to maximise the steganographic capacity and minimise the perception of hidden messages in stego images (Wang and Wang, 2004). Cole (2003) stated that *"the more data you can hide, the better the technique"*. However, the steganographic capacity tends to be restricted by the size of cover files (Artz, 2001; Rabah, 2004). Therefore, developing a steganography technique should take into consideration how to increase the amount of secret data that can be hidden without affecting the properties of stego files.

2.11.3 Robustness

A watermarking system is a robust system if the watermark is still detectable or recoverable after some signal processing operations. Therefore, watermarking systems that can endure and survive against all kinds of attacks are called robust systems. Thus, robustness represents the ability to detect the watermark after such kind of attacks. Moreover, it entails the survivability of this watermark against all kinds of signal processing and transformations (e.g. scaling, rotating, sharpening, filtering, adding noise, and blurring) (Cox et al., 2008; Venkatraman et al., 2004).

Since the majority of steganographic systems use computer networks and the Internet as communication channels that cause no degradation, the robustness is rarely considered. Thus, the recipient receives exactly what the sender transmitted. Therefore, robustness is not an issue or a top priority for steganographic systems. Moreover, the design of most steganographic systems does not consider robustness as a fundamental requirement, since the majority of these systems assume the passive warden scenario (Cox et al., 2008). Hence, steganographic systems are either not robust against modifications or have limited robustness against technical modifications such as compression, format conversion, or digital-to-analogue conversion. However, watermarking systems must be robust and resist any kind of transformations or manipulations that may attempt to remove the watermark. Thus, this is because the watermark must remain uncorrupted and recoverable in order to consider the watermarking system effective and successful system (Wang and Wang, 2004).

2.11.4 Tradeoff between Requirements

The main aim of steganography is to increase the steganographic capacity and enhance the imperceptibility or undetectability (Chang et al., 2006). However, steganographic capacity and imperceptibility are at odds with each other. For example, hiding more data in cover images (higher capacity) introduces more artefacts into cover images and then increases the perceptibility of hidden data (Wang and Wang, 2004; Wang and Chen, 2006). Furthermore, it is not possible to simultaneously maximize the security and capacity of a steganographic system. This is because of the tradeoff between the amount of embedded information, the amount of artefacts introduced to the cover file, and the system immunity against stego file modification (Marvel et al., 1998; Venkatraman et al., 2004). Consequently, steganographic systems must achieve a balance among these requirements. Steganographic systems do not need to be robust; but they should satisfy high steganographic capacity and secret data imperceptibility. However, watermarking schemes do not require a large embedding capacity or watermark imperceptibility. Instead of this, they need a large robustness against malicious and unintentional attacks (Marvel et al., 1998).

The capacity of spatial domain schemes is better than that of frequency domain schemes. However, the frequency domain schemes have better robustness than those of the spatial domain (Yu et al., 2005). Since watermarking schemes need to be robust, most watermarking schemes used are frequency domain schemes. Moreover, many novel embedding techniques have been suggested in order to enhance the security and increase the capacity of steganography methods (Chu et al., 2004; Lee and Chen, 2003; Lee and Chen, 2000; Li et al., 2006; Tseng and Chang, 2004). The next section explains the main techniques used to detect the steganography. Moreover, the relationship between the main aspects of steganographic systems and these detection methods will be discussed.

2.12 Steganalysis Techniques

Steganalysis can be defined as the science and the art of detecting and often decoding secret messages hidden within stego files (Artz, 2001). Nonetheless, steganography is considered broken if merely the presence of secret data within a stego file is detected, with it not being necessary to decode the secret message. The increasing number of steganography techniques available have stimulated steganalysis research. Thus, the significance of reliable detection techniques is increasing. Furthermore, it is suggested that such steganalysis techniques should be included in every virus-detection program in the future (Fridrich et al., 2001).

Basically, most steganographic systems leave behind (in the stego files) some traces, so these traces make these files detectable even though these traces are indiscernible by humans. Generally, modifying some parts of a cover file changes the properties of this file in some way. Therefore, this can be a sign that there is a hidden message within this stego file (Provos and Honeyman, 2003). Therefore, a simple comparison between a stego file and its corresponding cover file may reveal the existence of a hidden message within this stego file. In order to avoid such a comparison, cover files used should not be publicly available or should be destroyed after usage, since the absence of cover files represents the weakest form of steganalysis (stego only attack) (Artz, 2001).

Many forms of steganalysis can be distinguished. Some forms aim to detect the presence of secret messages in stego files while others involve the extraction of hidden messages from stego files. However, some forms of steganalysis aim to destroy all possible stego files (Johnson and Jajodia, 1998). On the other hand, steganalysis techniques can be classified into two main categories according to the detection means used. Thus, visual attacks rely on humans' inspection while statistical attacks rely on examining the statistical properties of stego files. Accordingly, visual attacks shall now be looked at in more detail.

2.12.1 Visual Attacks

Visual analysis is defined as the process of detecting hidden messages in stego files through inspection by naked eye or by assistance of a computer. Therefore, the visual attack represents one of the easiest steganalysis methods (Wang and Wang, 2004). Usually, such attacks may detect LSB steganography techniques but this is not true for JPEG steganography methods (Provos and Honeyman, 2003).

Visual attacks examine the entire stego file (i.e. image) or only the LSB of this file in order to detect any alteration or irregularity. Thus, steganography methods that leave some kind of trail or signature would be vulnerable to such attacks. Such signatures could be: adjacent pixels in an image have very different colours, the number of an image colours has drastically been increased or decreased, the image size has been changed, and the image quality has been modified (Bailey et al., 2004).

2.12.2 Statistical Attacks

Statistical analysis relies on examining the contents of files. Moreover, this kind of attacks is more powerful than the visual attack since it reveals even tiny modifications which have occurred in the statistical properties of files (Artz, 2001). The statistics of a file may reveal that it has been modified in some way but this doesn't specify which technique was used for modification. This represents one of the difficulties of the statistical analysis (Watters et al., 2005). For example, the variation of an image from the original can be determined by checking their colour histogram.

There are many image processing programs and tools to expose the statistical properties of these images. Some of these statistical properties which can be analysed and investigated are: standard deviation, differential values, median, skew and kurtosis (Cole, 2003). However, it is difficult to find out an effective tool of steganalysis that can detect all steganography methods and techniques. Nonetheless, steganalysis tools that can detect a specific type of embedding are more common and effective than the general tools (Wang and Wang, 2004).

2.13 Steganographic Systems Evaluation

In order to make a decision of which steganographic system or technique is better than another, an evaluation scheme for steganographic systems is needed. Currently, no standard test or measure is available in order to evaluate the performance or the effectiveness of steganographic systems. However, there are some guidelines and general procedures that can be considered when evaluating or designing steganographic systems (Cox et al., 2008).

The amount of hidden information and the difficulty of detection of stego files are the two most important aspects of any steganographic system. Therefore, measuring these two characteristics will determine the superiority of a steganography technique over another. Consequently, the measures of steganographic capacity and undetectability are needed in order to evaluate the efficiency of steganographic systems.

2.13.1 Evaluation of Steganographic Capacity (Payload)

Since the main application of information hiding and steganography is the secret communication, it is important to determine how many bits a steganographic system can embed imperceptibly in comparison to the other methods. Therefore, evaluating the capacity of a steganography technique means to find out the maximum number of bits that can undetectably be hidden.

It is mentioned before, that there is a tradeoff between the steganographic capacity and imperceptibility. Nevertheless, steganography techniques that embed larger-size messages in cover files and introduce more distortion to stego files are considered as worthless systems. On the other hand, increasing the steganographic capacity and maintaining an acceptable level of stego image quality is considered a good contribution. Additionally, improving the stego image quality while maintaining the steganographic capacity is also considered a significant contribution (Wu and Hwang, 2007).

2.13.2 Evaluation of Imperceptibility

Methods or techniques that can be used to evaluate the undetectability or imperceptibility of steganographic systems are different from one system to another depending on the type of cover file used for information hiding. For example, image quality represents an indication for the undetectability of image based steganography, while file size may reveal the presence of hidden data within a text file and therefore lead to its detection.

Two types of perceptibility can be distinguished and evaluated in signal processing systems, namely fidelity and quality. Fidelity means the perceptual similarity between signals before and after processing. However, quality is an absolute measure of the goodness of a signal. For example, we can use a grayscale, distorted and low resolution image (considered to be of low quality) for data hiding. The stego image looks identical to the cover image but it is also has low quality. However, because it is indistinguishable from the cover image, it has high fidelity. For image based steganography, the fidelity is defined as the perceptual similarity between the original cover image and the stego image. Therefore, the fidelity evaluation requires both versions of the image before and after embedding. However, attackers, and most likely recipients, do not have access to the unmodified original cover image. Additionally, steganographic systems must avoid attracting the attention of anyone not involved in the secret communication process and therefore stego images must have very good quality. Therefore, quality is the major perceptual concern for most steganography techniques in order to avoid any suspension and therefore detection (Cox et al., 2008). Even though the PSNR and the mean square error (MSE) are by definition fidelity metrics, they are pervasively known as quality measures, since they also represent perceptual distance metrics used to measure the distortion amount added to an image. Thus, in this thesis we are going to go with this direction also. Accordingly, a high quality image entails a large PSNR value and therefore both cover image and stego image are very similar and quite undistinguishable. Significantly, *"Fidelity"* is defined as the perceptual quality of stego files and therefore PSNR and MSE describe how imperceptible the secret message is (Cox et al., 2008).

Thus, it is very important that there is no visual difference between the cover image and the stego image. Moreover, the difference between these two images (cover and stego image) must be perfectly imperceptible for the human visual system. Accordingly, the higher the quality of stego images, the larger the imperceptibility of the steganographic system. Therefore, evaluating the quality of stego images is a significant measure to be used for evaluating the performance of image steganography techniques (Wu and Hwang, 2007).

2.13.2.1 Evaluating the Quality of Digital Images

The usage of image compression, coding, or processing technologies has increased significantly during last few decades. Furthermore, evaluating and measuring the quality of compressed images still represent a significant issue in many image processing applications, such as image coding algorithms and digital image steganography. Thus, image quality represents a key factor in most applications and assessing the perceived quality of digital images is very important (Tan et al., 1998). Nevertheless, evaluating the image quality of many image compression algorithms (i.e. lossy compression and image based steganography) has many difficulties, such as the amount of degradation induced in the reconstructed image.

Generally, there are two primary ways to measure image quality: objective quality methods (automated) and subjective quality methods (human based) (Stoica et al., 2003). The objective methods measure the physical aspects of images and psychological issues while the subjective methods are psychologically based methods. Additionally, subjective methods use human observers in order to evaluate the quality of images. For example, subjects can be asked to compare a modified (or processed) image with its original (unprocessed) version in order to know how much this modified image is degraded (Wu and Rao, 2006).

2.13.2.2 Objective Quality Evaluation (Automated Evaluation)

In order to get a faster and cheaper measure of image quality, responses of observers can be predicted and modelled (Cox et al., 2008). Therefore, designing image quality evaluation metrics that can automatically predict the perceived image quality is the main goal of objective image quality assessment research (Wang et al., 2003). Thus, the assessment algorithms designed for objective image quality evaluation should be in close agreement with subjective human evaluation regardless of the image content, the distortion amount, or the distortion type (Sheikh et al., 2006).

Objective image quality evaluation metrics are classified into three generic categories according to the availability of the unmodified or original image (reference). These categories are: full-reference (FR), no-reference (NR), and reduced-reference (RR) image quality assessment (Wang et al., 2003). The full-reference means that the original image and the test (impaired) image are available. However, the no-reference means that only the test image is available. On the other hand, the reduced-reference means that the test image and some information about the original image are available (Ponomarenko et al., 2008).

In the literature, the peak signal-to-noise ratio metric (PSNR) has shown the best advantage almost over all objective image quality metrics under different image distortion environments and strict testing conditions (Wang et al., 2002a). Indeed, PSNR and the MSE metrics are the most common measures used to evaluate the quality of image coding and compression (Costa and Veiga, 2005). However, these two objective quality evaluation metrics do not offer good results in terms of human perception when used for colour image. As a result, they are not reliable predictors of perceived quality. Accordingly, subjective evaluation methods have also been utilized in the literature (Stoica et al., 2003; Wu and Rao, 2006).

2.13.2.2.1 PSNR and MSE

PSNR and MSE are the most common and widely-used full-reference (FR) metrics for objective image quality evaluation. Furthermore, PSNR is used in many image processing applications and considered as a reference model to evaluate the efficiency of other objective image quality evaluation methods (Wang et al., 2002b).

The PSNR measures the similarity between two images (how two images are close to each other), while the MSE measures the difference between these two images. Since the computing of these two metrics is very easy and fast, they are widely-used and very popular (Wang et al., 2003). The MSE is the statistical difference in the pixel values between the original and the reconstructed image. Moreover, PSNR and MSE are defined as follows (Stoica et al., 2003; Wang et al., 2003):

$$MSE = \left(\frac{1}{MN}\right) \sum_{i=1}^{M} \sum_{j=1}^{N} \left(X_{ij} - \overline{X_{ij}} \right)^2 \tag{2.1}$$

$$PSNR = 10.\log_{10} \frac{I^2}{MSE} db \tag{2.2}$$

where:

X_{ij} is the i^{th} row and the j^{th} column pixel in the original (cover) image,

$\bar{X}_{ij}$ is the i^{th} row and the j^{th} column pixel in the reconstructed (stego) image,

M and N are the height and the width of the image,

I is the dynamic range of pixel values, or the maximum value that a pixel can take, for 8-bit images: *I*=255.

However, the MSE for colour images is defined as follows (Yu et al., 2007):

$$MSE_{AVG} = \frac{MSE_R + MSE_G + MSE_B}{3} \tag{2.3}$$

where: MSE_R, MSE_G, and MSE_B are the MSE of red, green, and blue components respectively. Thus, the best image quality can be found when the MSE value is very small or going to be zero since the difference between the original and reconstructed image is negligible. However, PSNR values between 20 and 40 can be considered as typical values (Cole, 2003). Moreover, the higher the PSNR value of a stego image, the better the degree of hidden message imperceptibility. For example, it is difficult for the human visual system to recognize any difference between a greyscale cover image and its stego image if the PSNR value exceeds 36 dB (Wu and Hwang, 2007).

2.13.2.3 Subjective Quality Evaluation (Human Evaluation)

In this kind of evaluation, humans are asked to observe some images and then to evaluate or assess their visual quality. However, the visual sensitivity varies from person to another and it changes over time in anyone. Therefore, different viewers will behave differently. Nevertheless, almost all objective image quality measures do not perfectly reflect the impression of humans. Hence, the subjective quality measure represents a true performance benchmark for image processing tools (Stoica et al., 2003). Unlike objective quality measures, subjective measures represent the most reliable method to determine the actual image quality since human beings are the ultimate proposed receivers in most applications. Furthermore, it has been stated that the subjective test is the best method to evaluate the quality of images surely and reliably (Marini et al., 2007).

Accordingly, subjective measures use structured experimental designs and real end users or human subjects to assess the quality of images (Tan et al., 1998; Wu and Rao, 2006). Furthermore, they are the most widely recognized methods for image quality evaluation since they quantify the actual perceived quality. However, subjective experiments of image quality evaluation are complex, difficult to repeat, very expensive, and time consuming (Grgic et al., 2004; Wu and Rao, 2006). Generally, observers are asked to rate the quality of images, sometimes with reference to other images, according either to a quality scale or an impairment scale. Table 2.1 summarises these scales to be used in evaluating image quality (ITU-R-BT.500-11, 2002).

The average scores for these rated images are called the mean opinion scores (MOS) (Simone et al., 2009). The MOS, which is a subjective quality measure representing the average score of number of subjects' scores, is considered as a reliable measurement of image quality. Moreover, the mean opinion score is calculated for each test condition k (i.e. steganography method) as follows:

$$MOS_k = \frac{\sum_{n=1}^{N} m_{nk}}{N} \tag{2.4}$$

where m_{nk} is the score of subject n for the test condition k and N is the number of subjects (Simone et al., 2009).

Five-Grade Quality and Impairment Scale			
Quality Scale		Impairment Scale	
5	Excellent	5	Imperceptible
4	Good	4	Perceptible, but not annoying
3	Fair	3	Slightly annoying
2	Poor	2	Annoying
1	Bad	1	Very annoying

Table 2.1: Quality and Impairment Scales

The Recommendation ITU-R Rec.500-11 describes the subjective evaluation of visual quality. Furthermore, it suggests criteria for selection of observers, test materials, viewing conditions, evaluation procedures, and analysis methods. This international standard has adopted many methods of subjective evaluation for image quality as they have stable and repeatable results. The double stimulus continuous quality scale (DSCQS) method, the double stimulus impairment scale (DSIS) method, and the single stimulus continuous quality evaluation (SSCQE) method represent some of these methods adopted. The single stimulus continuous quality evaluation (SSCQE) and double stimulus continuous quality scale (DSCQS) are the most adopted methods for subjective image quality evaluation in both research community and the industry (Wang et al., 2003). Moreover, the double stimulus continuous quality scale method (DSCQS) is the best available test in the existing standardisation literature (Baroncini, 2006).

2.13.2.3.1 Double Stimulus Continuous Quality Scale (DSCQS)

In the DSCQS method, the subject is asked to observe a pair of images from the same source and then to assess the quality of both images. One of these two images is the original image (directly from the source) and the other is the image under test. The series of presentations are internally random and randomly presented. Therefore, each presentation or pair of images consists of one unimpaired (reference) image while the other one might or might not contain impairments. However, observers are not told which image is the reference one since the position of the reference image is changed randomly (ITU-R-BT.500-11, 2002). Basically, the reference image represents the clean unmodified image while the test image represents the reconstructed, modified, or manipulated image (Wu and Rao, 2006).

In this method, there are two presentations for each trial and the subject has to provide two responses for each trial. The scale used in this method is double, 10 cm in length and divided into five equal intervals. These intervals are from top to bottom: Excellent (100-80), Good (79-60), Fair (59-40), Poor (39-20) and Bad (19-0). Therefore, this method is considered as a continuous rating system. In this method, the quality of a test image can be rated as better, equal or worse than that of its reference being compared but this is not true for other subjective methods. Moreover, the subjects are asked to assess both test and reference images since they have no idea from trial to another which image (test or reference) is presented first (Wu and Rao, 2006). Therefore, the subjective impairment judgment is the difference between the reference image score and the test image score (Wang et al., 2003). Generally, most participants will tend to avoid the end of the scales (100 and 0). Therefore, it might be a good idea to instruct them on what reference quality and worst case are look like (Wu and Rao, 2006).

The continuous rating scales used in the DSCQS method avoid quantising errors and are divided into five equal lengths. Ratings of subjects for each presentation are then converted into scores in the range from 0 to 100. Afterwards, the difference between these two scores (for the reference image and the test image) is calculated. Thus, these scores obtained should not be treated as absolute scores (Simone et al., 2009).

Some methods of subjective evaluation are designed in such a way that they will be sensitive to context effects. This kind of effects occurs when the impairment ordering or severity within test sessions has an effect on the subjective ratings. However, one of the advantages of the DSCQS method is its lower sensitivity to these context effects. Therefore, the DSCQS method is widely accepted and used as an accurate test method (Pinson and Wolf, 2003).

2.14 Summary

In this chapter we have introduced the reader to the main issues concerning digital steganography. We have identified the core components of a steganographic system and the key techniques of steganography. Additionally, using steganography over networks and Internet has been considered and explained. We have discussed file types that can be used for steganography, particularly digital images and SOAP messages. Furthermore, the main aspects of steganography techniques needed to evaluate their efficiency have been identified. Then, we have investigated issues concerning attacks and steganalysis. The details of information hiding within JPEG files and text files will be investigated in the next chapter. Also, the previous work related to improving the steganographic capacity and imperceptibility of steganography (JPEG and text) will be critically analysed there.

Chapter 3: JPEG and SOAP Steganography: Capacity and Imperceptibility

3.1 Introduction

The steganographic capacity and imperceptibility represent the most important aspects of any steganography technique. Therefore, designing a steganographic system must essentially consider these two properties. In this chapter we introduce issues relating to DCT-based JPEG compression and JPEG based steganography since our study focuses on improving JPEG steganography in the first place. Thus, the main procedures of JPEG encoding and decoding are illustrated. Also, the common methods of JPEG based steganography are identified. Then, the impact of the JPEG quantisation tables on the efficiency of steganography techniques is presented. Additionally, we investigate and analyse related studies from four different perspectives: improving the JPEG steganographic capacity and stego image quality, enhancing the quality of JPEG images, using chrominance components for steganography, and evaluating the quality of stego images.

Additionally, our research aims to investigate the capability of SOAP messages, which are communication protocols, to be used for steganography. Thus, the structure of SOAP messages and their ability to hide secret data are illustrated and explained in this chapter. Since SOAP is a formatted text, the previous studies related to information hiding in text and SOAP messages are presented and analysed. According to the related studies analysed in this chapter, we state our research aims and objectives in section 3.12. Then, the research methodology, used in Chapters 4, 5, 6 and 7, is described and justified.

3.2 DCT-based JPEG Compression

Generally, JPEG-compressed images have both small size and reasonable quality. Furthermore, this kind of images represents one of the most popular images widely used over the Internet as well as in local usage (Tseng and Chang, 2004).

JPEG is an international standard for continuous-tone still image compression (ISO-DIS, 1992). Moreover, JPEG compression using the discrete cosine transform (DCT) is the most common compression standard for still images. Additionally, it provides a large compression ratio and maintains high image quality (Munirajan et al., 2004; Noda et al., 2006; Rongrong et al., 2006; Tseng and Chang, 2004; Wong and Wong, 2001).

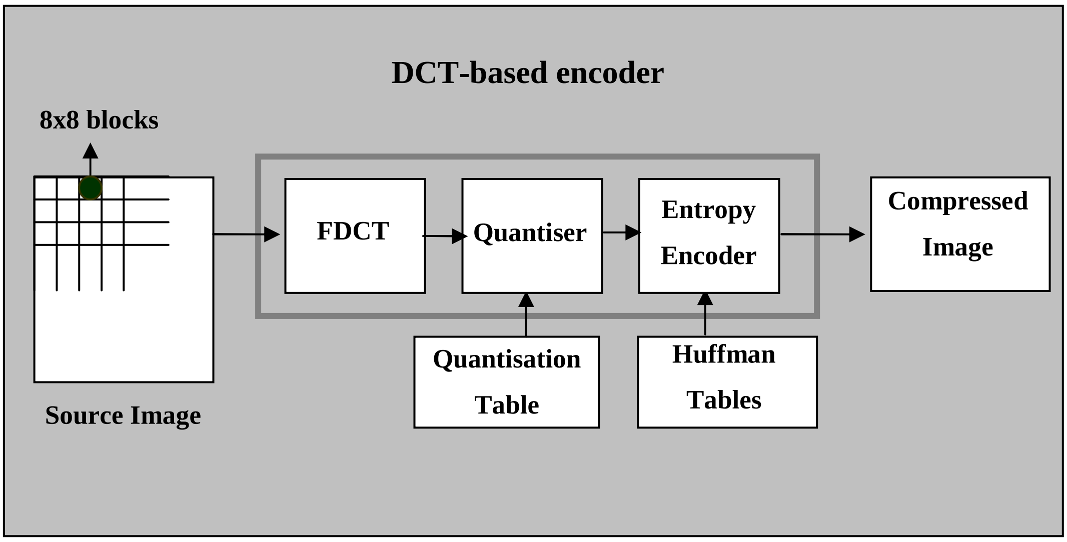


Figure 3.1: The Block Diagram of the JPEG Encoding

The JPEG encoding process consists of three main steps: forward DCT (FDCT), quantisation, and entropy encoding, as shown in Figure 3.1 (ISO-DIS, 1992). The input image is first converted into the YCbCr colour space (Y represents the image luminance while Cb and Cr represent the image blueness and redness respectively) and then divided into disjoint blocks of 8x8 pixels. Then, each block is transformed by the FDCT into a set of 64 DCT coefficients. For MxN block, the mathematical definitions of the FDCT and inverse DCT (IDCT) are as following:

$$F(u,v)=\frac{2}{\sqrt{M.N}}C(u)C(v)[\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}f(x,y)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}] \quad (3.1)$$

$$f(x,y)=\frac{2}{\sqrt{M.N}}[\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}C(u)C(v)F(u,v)\cos\frac{(2x+1)u\pi}{2M}\cos\frac{(2y+1)v\pi}{2N}] \quad (3.2)$$

for *u=0, 1, 2,…,N-1* and *v=0, 1, 2,…,M-1* where

$$\left.\begin{array}{l} C(u),C(v)=\frac{1}{\sqrt{2}} \text{ for } u,v=0 \\ C(u),C(v)=1 \text{ otherwise.} \end{array}\right\} \quad (3.3)$$

For 8x8 blocks:

$$F(u,v)=\frac{1}{4}C(u)C(v)[\sum_{x=0}^{7}\sum_{y=0}^{7}f(x,y)\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16}] \quad (3.4)$$

$$f(x,y)=\frac{1}{4}[\sum_{u=0}^{7}\sum_{v=0}^{7}C(u)C(v)F(u,v)\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16}] \quad (3.5)$$

Here, $F(u,v)$ presents a DCT coefficient at the coordinate (u,v) while $f(x,y)$ presents a pixel value at the coordinate (x,y). The values of DCT coefficients are the relative amount of the two dimensional spatial frequencies contained in the 64-point input signal. Thus, the DCT maps the 64-point vectors between the image (spatial domain) and the frequency domain in a one-to-one mode. Therefore, the DCT just transforms the source image samples in order to be encoded more efficiently without introducing any loss.

$F(0,0)$ is called the (DC) component which represents the average value of intensity for each block in the spatial domain. Therefore, the DC coefficient is the coefficient with zero frequency in both dimensions. However, in the case of $u \neq 0$ and $v \neq 0$, $F(u,v)$ is called the (AC) component. Therefore, the remaining 63 coefficients in each 8x8 DCT block are called the AC coefficients. Moreover, the upper left coefficients of each block are called the low frequency components while the lower right coefficients are called the high frequency components (Wallace, 1991).

In the quantisation step, all DCT coefficients of each block are divided by predefined quantisation values (contained in a quantisation table). These values can be any integer from 1 to 255. Then, each quantised DCT coefficient is rounded to the nearest integer (equation (3.6)). The quantisation step is a many-to-one mapping and therefore it is essentially a lossy process because of the rounding error.

$$F^{Q}(u,v) = Round(\frac{F(u,v)}{Q(u,v)}) \tag{3.6}$$

Finally, these quantised DCT coefficients are encoded using an entropy encoder (Huffman coding or arithmetic coding), which is a lossless process (ISO-DIS, 1992). The quantised DC coefficient is encoded as the difference from that of the previous block. This is because most of the image energy is concentrated in the DC coefficients of the DCT blocks and the DC coefficients of adjacent blocks have a strong correlation.

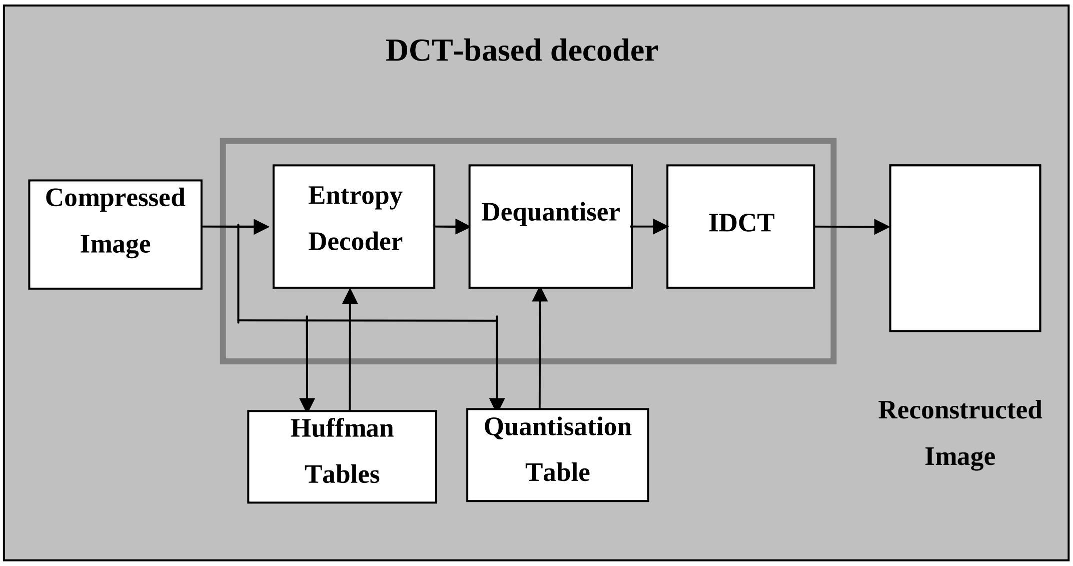


Figure 3.2: The Block Diagram of the JPEG Decoding

Figure 3.2 shows the JPEG decoding process which also consists of three main steps: entropy decoding, dequantisation, and inverse DCT (IDCT) (ISO-DIS, 1992). The compressed code is entropy decoded and the quantised DCT coefficients are obtained. In the dequantisation step, each block of quantised DCT coefficients is multiplied with the quantisation table to convert these coefficients to their approximate value. Afterwards, the IDCT is used to convert the dequantised DCT coefficients to their spatial values.

3.3 JPEG Based Steganography

The JPEG is the most suitable format to be used as cover image for steganography since JPEG is the most common compression standard used for still images (Tseng and Chang, 2004). Furthermore, the majority of steganography techniques used for JPEG images, such as JSteg and Outguess, adopt the standard JPEG compression. As shown in the previous section, JPEG is based on breaking the image into non-overlapping blocks of 8x8 pixels and fitting discrete cosine transformations to these pixels. Additionally, the compression ratio can be increased or decreased by setting more or fewer quantised DCT coefficients to zero (Chang et al., 2002; Lee and Chen, 2003; Li et al., 2006; Munirajan et al., 2004; Rongrong et al., 2006; Tseng and Chang, 2004; Wong and Wong, 2001). Mostly, the least significant bits (LSBs) of quantised DCT coefficients are used as redundant bits to be replaced by the bits of the secret message. Moreover, modifying a single DCT coefficient affects all 64 block pixels. Therefore, hiding secret information in JPEG images mainly occurs during the JPEG encoding process.

Recently, many JPEG based steganography methods have been proposed and developed: JSteg, Outguess, and F5 (Fridrich et al., 2002). Moreover, all these methods manipulate the quantised DCT coefficients in order to embed the secret data (Fard et al., 2006). Figure 3.3 shows a generalized steganography framework for compressed JPEG images (Cherukuri and Agaian, 2007). Obviously, secret information is mainly embedded after the quantisation step and before the entropy coding step since the entropy coding procedure is lossless. This means that all data will remain intact after entropy coding. Therefore, the secret bits embedded in the quantised DCT coefficients will not be destroyed.

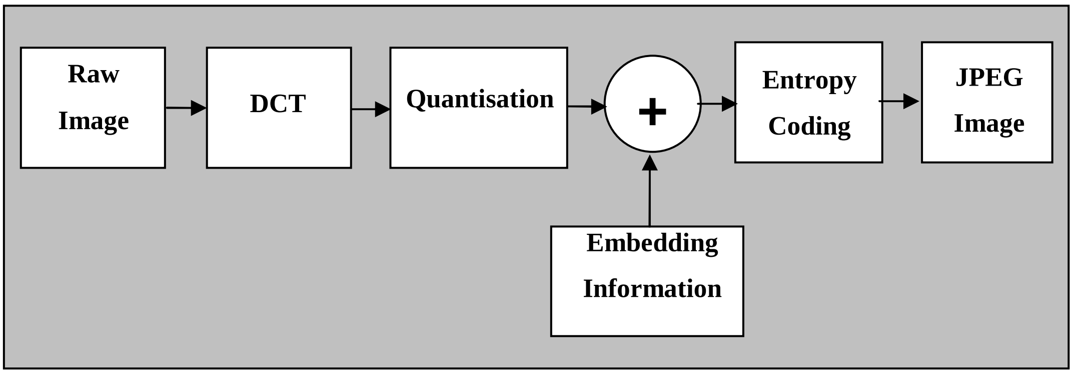


Figure 3.3: The Generalized Steganography Framework for JPEG Images

3.3.1 JSteg Approach

This widely known steganography approach builds upon the standard JPEG compression and represents one of the first programs designed for JPEG based steganography (Provos and Honeyman, 2003). It was found by Derek Upham and allows us to hide a secret message or file as we compress an image (Upham). In the cover image, all disjoint blocks of 8x8 pixels are transformed using the DCT. Afterward, the DCT coefficients are scaled according to the default JPEG quantisation table (Table 3.1). This approach sequentially replaces the LSB of the quantised DCT coefficients with secret bits and skips all coefficients whose magnitudes are 0 or 1.

JSteg embeds the secret message over the entire cover image in a zigzag scan order until either embedding the whole message or exhausting the capacity of the cover image. Thus, its capacity has been considered relatively high (around 13% on an average). In fact, the steganographic capacity of this approach is very limited since the number of zero-values of the quantised DCT coefficients is large.

Additionally, the number of quantized DCT coefficients whose magnitude are 1 (not used by JSteg for embedding) is large also and a bit smaller than that of zero-value coefficients (Cherukuri and Agaian, 2007).

Even though manipulating the LSB of quantised DCT coefficients (frequency domain) can harm the stego image quality, the effects of this tweaking is hardly distinguished (Wayner, 2002). Therefore, JSteg is resistant against visual attacks. Nevertheless, the existence of hidden messages in stego images of JSteg method can be easily detected by simple statistical attacks (e.g. X^2 test or chi-square attack) (Cherukuri and Agaian, 2007; Westfeld, 2001a).

3.3.2 Outguess Software

Outguess is a steganography software written by Niels Provos (Provos, 2001). This steganography method preserves the first order statistics (histogram) of DCT coefficients in stego images in order to counter the statistical chi-square attack (Fridrich et al., 2002). Moreover, it carefully uses the LSB technique to avoid causing statistical distortions which may attract the attention of attackers to steganography existence. Outguess and JSteg are almost similar techniques. However, Outguess scatters the locations of embedding by using a pseudo random number generation (PRNG) to shuffle the ordering of the coefficients.

Outguess uses half the number of all quantised DCT coefficients whose magnitudes are not equal to 0 or 1 for secret message embedding. However, it uses the second half of the quantised DCT coefficients to correct the alteration made to the histogram of DCT coefficients and adjust this histogram to its original value. For example, changing “0” to “1” to embed a single bit requires changing “1” to “0” at the same time to correct the histogram of quantised DCT coefficients. Thus, the capacity of channel is reduced to half while the security is increased significantly (Cox et al., 2008; Wayner, 2002). Moreover, the Outguess algorithm can withstand against simple statistical attacks since it preserves the first-order statistical properties. Additionally, the maximum steganographic capacity of cover images is 6.5% while its embedding efficiency is around 1 (Sallee, 2003).

3.3.3 F5 Algorithm

F5 algorithm was proposed and created by Westfeld in 2001 to increase the steganographic capacity of JPEG images without sacrificing security (Westfeld, 2001a). Moreover, the goal of designing and developing the F5 algorithm was to preserve the histogram shape of DCT coefficients. The F5 algorithm skips the DC coefficients and the AC coefficients whose magnitudes are zeros, so it does not use them for embedding. Moreover, it decreases the absolute value of a quantised DCT coefficient by one to embed a secret bit instead of flipping the LSB of this coefficient. Therefore, if a given secret bit does not match the LSB of a predefined coefficient, the algorithm decrements the absolute value of this coefficient. Otherwise, the value of this coefficient will not be changed (Cox et al., 2008).

Basically, F5 does not embed the secret message sequentially within a cover file, but the DCT coefficients are randomly chosen to be used for secret data embedding (like Outguess). Furthermore, it uses a matrix encoding technique which spreads the secret information out among more bits. Thus, this technique significantly decreases the necessary number of changes required for embedding the secret data and therefore, F5 improves the embedding efficiency (Westfeld, 2001a).

It has been suggested that the steganographic capacity of the F5 algorithm should not be more than 14% of the cover image size in order to resist the visual attack. Moreover, the steganographic capacity should be smaller than 1% of the cover image size in order to resist any kind of attacks (Westfeld, 2001a). However, the maximum steganographic capacity of the F5 algorithm is roughly 13% while the embedding efficiency is 1.5 (Sallee, 2003).

3.4 JPEG and Quantisation Tables

In image compression with a JPEG baseline system, the quantisation and dequantisation processes are very important since they represent the main cause of lossy compression (Cherukuri and Agaian, 2007). Therefore, the quantisation process maintains the DCT coefficients needed to achieve the desired image quality whilst it zeroes out most of high frequency DCT coefficients and discards information that is visually irrelevant (Yildiz et al., 2007).

Usually, it is possible to control the image quality and compression ratio by controlling the values of the quantisation table (Hamamoto, 1999). Furthermore, the quantiser step size used before the entropy encoding determines the amount of compression. Thus, the quantisation table plays a fundamental role in compression as regards the fidelity performance of JPEG coding (Chang et al., 1999; Costa and Veiga, 2005; Hamamoto, 1999; Monro and Sherlock, 1996; Shohdohji et al., 1999). As a result, using an optimised quantisation table for JPEG compression may provide a reconstructed image with better quality than using the default JPEG quantisation table.

Although the JPEG standard uses 8x8 quantisation tables, it does not specify default or standard values for quantisation tables. Hence, specifying the quantisation values is left up to the application. Nevertheless, the JPEG standard provides a pair of sample quantisation tables. They were tested empirically and found to generate good results (ISO-DIS, 1992). One of these quantisation tables is for the luminance component (Table 3.1) while the other one is for the chrominance components (Table 3.2). Since these quantisation tables are widely used, they will be referred to as JPEG default quantisation tables.

Additionally, there are no samples for quantisation tables larger than 8x8 in the JPEG standard. This is because the DCT calculation for block-sizes larger than 8x8 pixels may require much more running time and may increase the computational operations and complexity (Bracamonte et al., 1997). Therefore, this might be one of the reasons why the JPEG standard uses blocks of 8x8 pixels. On the other hand, using block sizes larger than 8x8, and therefore larger quantisation tables, may lead to better results in terms of image quality and compression ratio. Additionally, the fast developments in computers technology, capability and competency may overcome the computation complexity which was existing in computers many decades before.

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

Table 3.1: The Default JPEG Quantisation Tables for Luminance

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Table 3.2: The Default JPEG Quantisation Tables for Chrominance

Quantisation tables can be arbitrarily generated and, if the quantisation values are set properly, there should be no perceptible difference for a human observer between the input image and the compressed image (Miano, 1999). Thus, several methods have been proposed to find out the optimum quantisation table for JPEG compression since the loss of fidelity in JPEG coding occurs entirely in the quantisation step (Yildiz et al., 2007). However, some methods for optimising the JPEG quantisation tables can be found in (Chang et al., 1999; Costa and Veiga, 2005; Hamamoto, 1999; Monro and Sherlock, 1996; Shohdohji et al., 1999).

As a result, everyone can come up with his/her own quantisation table which satisfy his/her requirements and applications. Hence, some applications require high quality images regardless of the compression ratio or the image size, while other applications need small-sized images of acceptable quality. At the end, users decide what they need and what quantisation table is to utilize. Furthermore, in order to balance the quality of a reconstructed image with its size, a quality factor can be applied to the quantisation table (equation (3.7)), so that the quantisation values can be scaled by a constant factor (Tseng and Chang, 2004).

$$Q(u,v)' = round[\frac{Q(u,v)}{Q_F}] \tag{3.7}$$

3.5 Text Files and Steganography

Text steganography refers to the process of hiding secret information in text files. For security and imperceptibility reasons, it is very important for stego texts not to show any detectable artefacts. Thus, readers should not notice or discover the modifications made in the stego text files. Generally, the redundant information in text files is very limited in comparison to image and audio files. Therefore, using text as cover files of steganography represents the most difficult way of information hiding (Bender et al., 1996).

Basically, there are three major methods to hide secret data in text files. The first method, open space method, manipulates white spaces in the text. Therefore, it exploits inter-sentence spacing, end-of-line spaces, and inter-word spacing. The second method, syntactic method utilizes punctuation. However, the third method, the semantic method, manipulates the words of the text themselves (Bender et al., 1996).

Unlike digital images, text files have less redundant information which could be used for steganography. Furthermore, information hiding in text needs to be done manually by the user since it can not be automated (Katzenbeisser and Petitcolas, 2000). Thus, information hiding in text files is the most difficult kind of steganography.

Secret information can be embedded either directly in the text or in the text format. Thus, introducing spelling errors, replacing words by their synonyms, and omitting commas are some examples of direct hiding within text files. Furthermore, DOC, LATEX, XML, and HTML represent a formatted form of text files. Therefore, secret information can be hidden in the format rather than in the text itself. Changing the spaces size between text lines and adding white space characters are some examples of this kind of steganography. However, reformatting the text, retyping it, or converting it from one format to another may destroy all hidden information in the text format (Katzenbeisser and Petitcolas, 2000).

As a result, there is a quite small amount of redundant data within the text of formatted text files available to be used for steganography. This represents the main challenge of text steganography and explains the limited number of studies related to this kind of steganography. As a possible solution for this limited steganographic capacity, a communication protocol, that mainly uses a formatted text, could be utilized as a cover for secret data since we can communicate a huge number of messages without attracting suspicion. This shall now be explained in more detail in the next section.

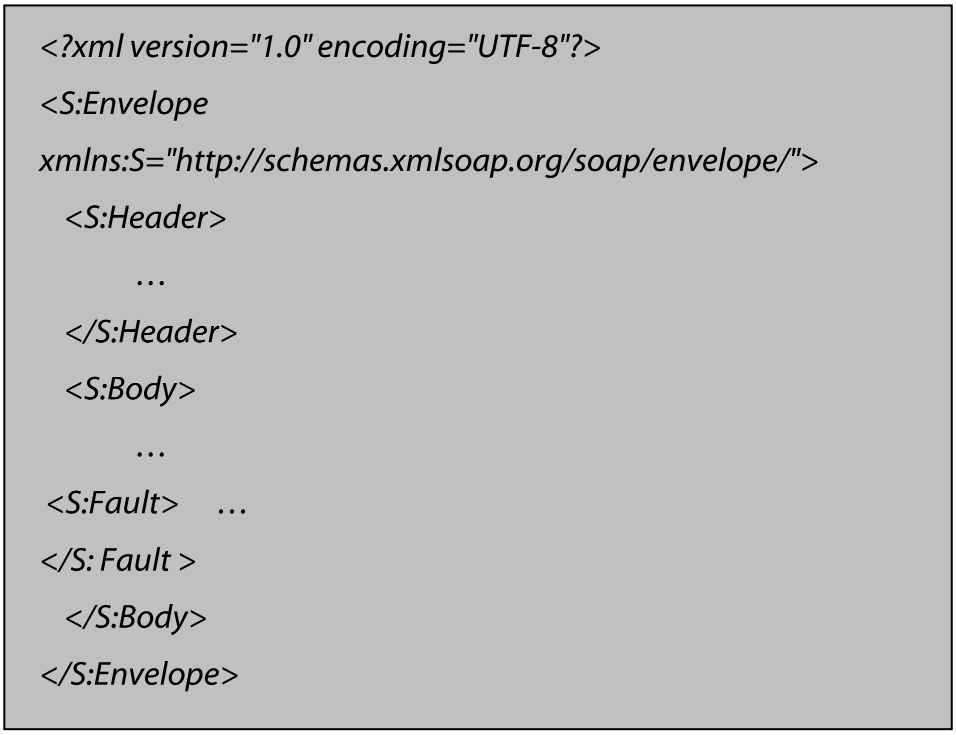
3.6 SOAP Message and Steganography

Communications and interactions between clients and the web service are achieved using SOAP which supports a common data transfer protocol for effective communication over the Web. Thus, structured and typed information can be exchanged between peers of distributed environment using SOAP messages. A SOAP message is an XML document created in a specific format and it mainly consists of envelope, header, body and fault elements, as shown in Figure 3.4.

The SOAP *Envelope* is the root element that defines the XML document as a SOAP message. Furthermore, it indicates the start and the end of the message.

Therefore, it lets the receiver know when an entire message has been received and it is ready for processing.

SOAP can be extended to include additional features and functionality such as security, reliability, and other quality-of-service attributes. Thus, SOAP *Headers* can be used to carry such application-specific information. *Headers* are optional but we can use many headers within a SOAP message. Also, *Headers* can be used for adding some features to a SOAP message in decentralized way without prior agreement between the communicating parties. Additionally, *Headers* may contain commands to SOAP processors either to understand these *Headers* or to reject the SOAP message.



```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
        …
  </S:Header>
  <S:Body>
        …
 <S:Fault>    …
</S: Fault >
  </S:Body>
</S:Envelope>
```

Figure 3.4: SOAP Message Structure

The SOAP *Body* contains the actual data (application-defined XML data) being exchanged in the SOAP message. Therefore, mandatory information that must be delivered to the intended recipient should be included within the *Body* part of SOAP message.

The optional *Fault* element is used to identify error messages. If an error occurs during SOAP processing, a SOAP *Fault* element will emerge in the body of the message. Then, the sender of the SOAP message will get the *Fault* response returned to him/her. The error information that might be returned by the SOAP fault mechanism include: a predefined code, a description, the address of the SOAP processor that generated the *Fault*, and application-generated details about the error. Furthermore, if a detail element is present within a *Fault* response message then the error occurred during *Body* processing. However, a single *Fault* block in the body of the message can be carried by a SOAP response message.

When two parties communicate through SOAP messages, the actual data (i.e. fields and properties of objects or parameters and return values of methods) in the sender endpoint are converted (serialized) into an XML stream that conforms to SOAP specifications. This serialized XML document is the SOAP message that needs to be de-serialized at the receiver endpoint to reconstruct the actual data.

In Web Services, the interaction between service providers and requesters occurs via XML-based SOAP messages. Therefore, such messages offer a kind of steganography cover files. Hence, secret information can be embedded in SOAP messages and sent over the network to an intended destination. Essentially, a SOAP message is a kind of XML document, yet XML documents essentially contain text since they represent a formatted form of text files. Therefore, steganography methods used for text files and XML documents can theoretically be used for SOAP messages. Practically, some or all of these methods might be infeasible since the SOAP protocol has its own structure and application.

In conclusion, SOAP messages are extensively used over the Web and so represent a valuable kind of steganography cover. Thus, it would be a good idea and practice to find out a steganography method that can be used for SOAP messages which would not attract the attention of attackers.

The next five sections discuss and analyse the previous work related to five directions: improving JPEG steganographic capacity and stego image quality, enhancing the quality of JPEG images, image steganography and chrominance, evaluation the quality of stego images and SOAP based steganography.

3.7 Improving JPEG Steganographic Capacity and Stego image Quality

The main goal of steganography is secret communications. Mostly, the data payload of a steganography method should be at least equal to the size of the secret message transmitted between two parties. Therefore, developers of steganography frequently attempt to increase the steganographic capacity in order to hide much more information in cover files. However, many attempts have been made by researchers in order to balance the two conflicting requirements: imperceptibility and capacity. So, various solutions have been suggested and presented for JPEG steganography (Yu et al., 2008).

To the best of our knowledge, this is the first work that attempts to use larger and optimised JPEG quantisation tables to increase the JPEG steganographic capacity and/or stego image imperceptibility. This section describes the previous studies related to these two points of view which represent the first and second objectives of our study: improving the capacity of JPEG steganography and enhancing the quality of stego images.

Basically, the majority of proposed steganography methods for JPEG images consider the steganographic capacity and imperceptibility aspects. Thus, various methods and techniques have been suggested in this regard. In this part, we discuss and analyse some key studies in the area of steganographic capacity and stego image quality improvement for JPEG-based steganography.

Alturki and Mersereau (2001) proposed a novel method to increase the capacity and security of transform domain steganography in still images. They suggested increasing the number of transform coefficients that can be used for data hiding by introducing some middle and high frequency components into the image. They have done this by decorrelating the samples of the image in the spatial domain using a key. Thus, this key scrambles the image pixels in such a way that the resulting image will be seen as a white noise. Therefore, this image whitening process spreads the image energy uniformly over all frequency bands, so it increases the data transmission bandwidth and then increases the embedding capacity (2 bits per transform coefficient). Hence, this method hides secret bits by rounding the quantised DCT coefficient to the nearest odd or even integer according to the secret bit. Since the statistical properties of both cover and stego images are similar and the format of extracted data (the difference between the cover image and the stego image) looks like a Gaussian noise, this method is considered secure (Alturki and Mersereau, 2001). Thus, this method uses small quantisation step sizes and therefore almost all DCT coefficients may be kept and used for steganography. As a result, this may significantly decrease the compression ratio and therefore increase the stego image size, which may arouse someone's suspicion.

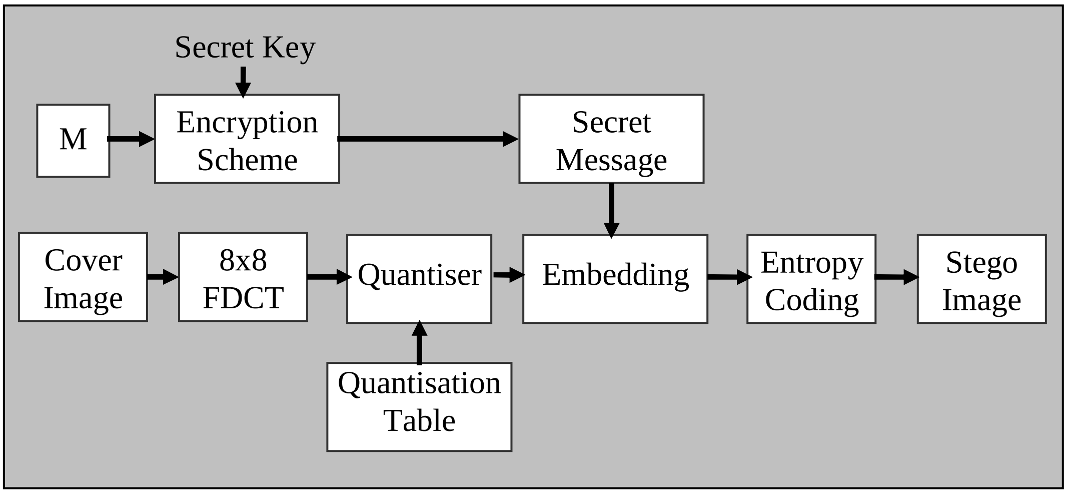


Figure 3.5: The Block Diagram of the JMQT Embedding Method

Chang et al. (2002) developed a novel steganography method in order to improve the steganographic capacity of JSteg method. The proposed method is based on JPEG and a modified quantisation table, so it will be called as JMQT method (Figure 3.5). The middle range of quantised DCT coefficients was used for secret information embedding. Firstly, they used a modified quantisation table to quantise the DCT coefficients. Secondly, they utilized only the middle frequency range of quantised DCT coefficient for data hiding. Thus, they embedded the secret data in the least two-significant bits (2-LSBs) of these coefficients. Accordingly, each quantised DCT coefficient in the middle frequency part will hide two secret bits. In conclusion, the security level of the proposed method and JSteg method was similar (hidden quantisation table and LSB embedding technique). Moreover, the stego images of JSteg method have better quality than those of this method. However, the proposed method has a larger steganographic capacity (around 24%) than the JSteg method (Chang et al., 2002).

Sallee (2003) proposed a general steganography method to answer this question: *what is the maximum embedding capacity that is statistically undetectable and how can this be achieved?* Thus, information theory and statistical modelling are used to generate a general framework for steganography. Furthermore, an example model-based JPEG steganography method is proposed in order to resist statistical attacks and achieve higher embedding capacity than F5 and Outguess. This method models only non-zero AC coefficients using a parametric density function which is based on Generalized Cauchy distribution. Thus, it can hide a secret message twice as large as that Outguess algorithm, so its capacity is around (13%). However, it is resistant to first order statistical attacks and maintains individual coefficients histogram (Sallee, 2003).

In related work Tseng and Chang (2004) proposed a steganography method in order to balance the steganographic capacity, bit rate, and stego image quality (Figure 3.6). Firstly, they used a JPEG compressor with a low scaling factor, then another JPEG with a high scaling factor. Then, the quantisation error, the difference between these two images, is calculated. Furthermore, the difference between the dequantised DCT blocks and the DCT blocks are called quantisation error tables (QET). Thus, DCT coefficients that intend to become zero after quantisation are used for embedding and the secret bits will be embedded in the quantised DCT coefficients according to the QET values. However, the quantisation values corresponding to the selected DCT coefficients are modified to 1 in order to keep these coefficients intact. As a result, the capacity of this method for a bit rate of 2.54 (bit per pixel) was around 22% (Tseng and Chang, 2004).

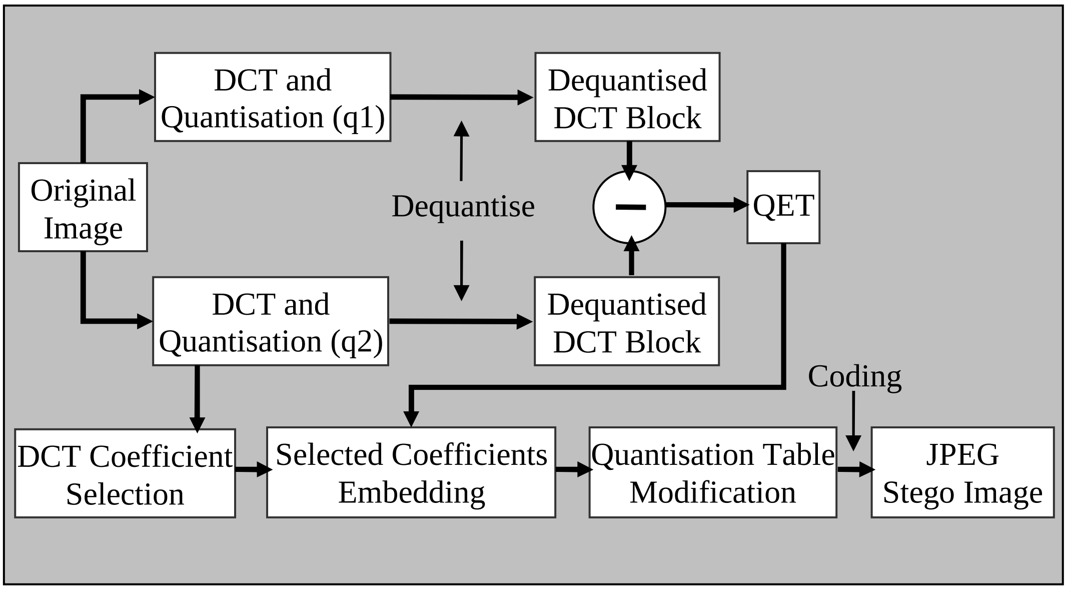

DCT and Quantisation (q1)
Dequantised DCT Block
Original Image
Dequantise
−
QET
DCT and Quantisation (q2)
Dequantised DCT Block
Coding
DCT Coefficient Selection
Selected Coefficients Embedding
Quantisation Table Modification
JPEG Stego Image

Figure 3.6: The Steganography Method Proposed by Tseng & Chang (2004)

Noda et al. (2006) presented two JPEG steganography methods mainly to preserve the histogram of quantised DCT coefficients. Hence, these methods hide data just during the quantisation of DCT coefficients, not after the quantisation step as is usually the case. Additionally, the least 21 frequency components including the DC component in the zigzag order were used for embedding. As a result, these two methods are secure against histogram based attacks, have higher embedding capacity than the F5 algorithm (around 15.2%) and their stego images have a larger PSNR than those of the F5 algorithm (Noda et al., 2006).

Agaian et al. (2006) proposed a JPEG based steganography algorithm which is applicable to mobile platforms and non-mobile computing platforms. This algorithm uses a switching embedding technique. Moreover, this technique determines the DCT coefficients that can be used for embedding using two different thresholds: an image-dependent threshold and a block-dependent threshold. The first threshold determines the embeddable and non-embeddable blocks of quantised DCT coefficients while the second threshold identifies the coefficients within the determined blocks to be used for embedding. This algorithm preserves the first order statistics of the cover image since it maintains the histogram after embedding. Therefore, it resists visual and statistical attacks.

As a result, the steganographic capacity of this algorithm was around 9%. Furthermore, this algorithm has a lower impact on the mean square error (MSE), and therefore its stego images have larger PSNR than that of the F5 and Outguess algorithms (Agaian et al., 2006).

More recently, Cherukuri and Agaian (2007) presented a novel JPEG steganographic system based on switching theory (Figure 3.7). This theory depends on the fact that the energy distribution of the quantised DCT coefficients varies from block to block and from one AC coefficient to another within a given block. Nevertheless, the switching mechanism consists of a threshold value which is a parameter defined according to the user requirements and input data characteristics. Thus, the coefficients' energy is used to calculate this threshold value. Additionally, it contains two different embedding methods. Thus, one specific method from these two methods is selected according to certain constraints and the calculated threshold value. The suggested steganographic system displayed a minimum of coefficients' distortion, preserved the cover image histogram before and after embedding, and increased the steganographic capacity (around 20%) (Cherukuri and Agaian, 2007).

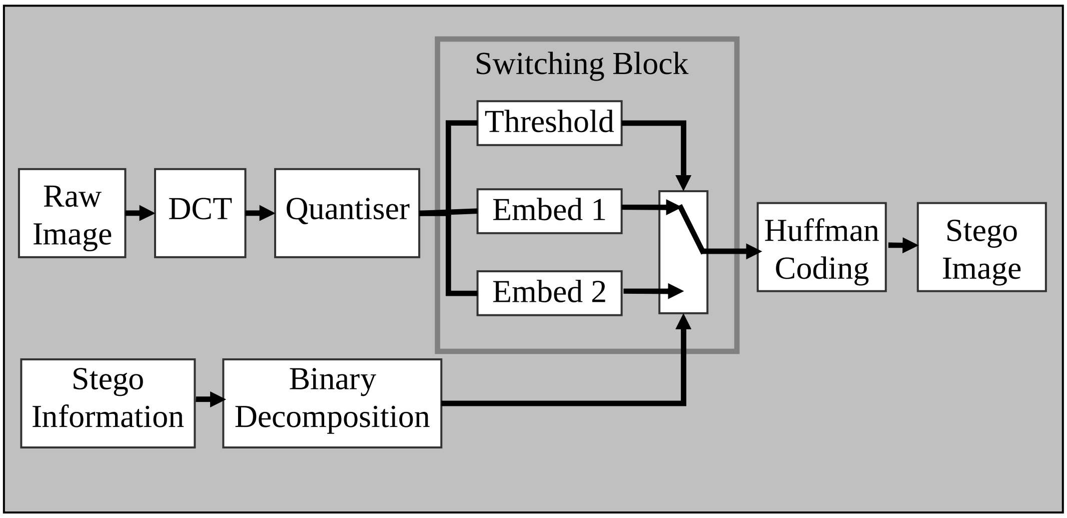


Figure 3.7: The Switching Theory-based Steganographic System

Li and Wang (2007) proposed a novel steganography method based on JPEG and Particle Swarm Optimization (PSO). This method has been developed to increase the steganographic capacity of the JMQT method. The PSO algorithm has been used to improve the stego images' quality. Thus, the optimal substitution matrix is produced in order to transform and then embed the secret message into the cover image. The JMQT method embeds the secret message in the middle frequency quantised DCT coefficients only. However, the proposed method embeds the secret message in the DC-to-middle frequency range of the quantised DCT coefficients. As a result, the steganographic capacity of this method was larger than that of the JMQT method. Additionally, the stego images of this method have better quality (larger PSNR) than those of JMQT method. However, the sizes of the stego images and the computational time required for this method were larger than those of the JMQT method. Thus, the computational time of this method was 12 times larger than that of the JMQT method (around 35 seconds), which is a very long time and represents one of this method's drawbacks (Li and Wang, 2007).

Liu and Liao (2008) proposed a high capacity and secure JPEG steganography method. Mostly, JPEG-based steganographic systems flip the LSB of the quantised DCT coefficients to embed the secret message. However, this method embeds the secret bits in the cover image by adding one or subtracting one from the quantised non-zero DCT coefficients. In order to minimise the detection probability using statistical attacks, the proposed method employs a complementary embedding strategy. According to a predefined divider, the secret bits and the quantised DCT coefficients have been divided into two parts. Moreover, two different embedding algorithms have been used to embed these two parts of secret bits in the corresponding DCT coefficients. As a result, the steganographic capacity of this method was larger (around 17.5%) than that of the JSteg, Outguess, and F5 algorithms. However, the quality of this method's stego images was slightly worse than the quality of the other methods' stego images (lower PSNR) (Liu and Liao, 2008).

3.8 Enhancing the Quality of JPEG Images

Many methods have been suggested to improve the compression rate and enhance the quality of JPEG encoded images. In the previous section, some steganography methods that enhance the quality of stego images have been discussed. Additionally, it has been argued, in section 3.4, that the quantisation table of JPEG compression has a significant role in controlling the encoded image quality and the compression ratio. Therefore, almost all methods used to improve the JPEG image quality rely on quantisation table optimisation.

To this end, Wu and Gersho (1993) presented an algorithm to generate a JPEG quantisation table. Using a large-step quantisation table (low bit rate), this algorithm starts decreasing the step size in one entry of the quantisation table every time and then repeats the process until a desired bit rate is obtained. As a result, the quantisation tables designed according to this algorithm produced higher PSNR and better perceptual quality the default JPEG quantisation tables (Wu and Gersho, 1993).

In another study, Sherlock et al. (1994) presented a model to generate optimal JPEG quantisation tables and therefore get a desirable compression ratio. This strategy uses simulated annealing to find out the optimal quantisation table. Quantisation coefficients were expressed as functions of their position in the quantisation table and the target compression ratio. Therefore, quantisation tables were represented by a three-parameter model and to get a general model, each parameter was expressed as a function of the compression (Sherlock et al., 1994).

In related work, Crouse and Ramchandran (1995) proposed a novel image-adaptive JPEG encoding scheme. In order to enhance the quality of JPEG still images, they jointly performed coefficients thresholding, quantisation table optimisation, and adaptive Huffman entropy-coding. Furthermore, they formulated an algorithm that jointly optimises coefficient thresholding and quantisation table. As a result, this algorithm improved the PSNR of coded images and therefore the quality of these images was better than that of custom JPEG images (Crouse and Ramchandran, 1995).

Based on their earlier work (Sherlock et al., 1994), Monro and Sherlock (1996) presented a general strategy, three-parameter model, to generate optimal quantisation tables using simulated annealing. Moreover, the designed quantisation tables could be used both for standard JPEG compression and its extension. Unlike standard JPEG compression, the extended JPEG compression uses blocks other than 8x8 pixels (from 8 to 20). As a result, the optimised quantisation tables of this model have significantly improved the compression performance (fidelity) in comparison with the default JPEG quantisation tables (Monro and Sherlock, 1996).

Bethel et al. (1997) proposed a method to generate an optimal JPEG quantisation table in terms of rate-distortion performance. Thus, this method has been developed to get the best possible tradeoff between high compression and low distortion. This method uses the “equal gradient” technique for quantisation table optimisation and utilizes blocks of 16x16 pixels rather than blocks of 8x8 pixels. As a result, this method yielded a better rate-distortion performance than that of default JPEG compression (Bethel et al., 1997).

The quality of JPEG images and the compression ratio can be controlled by adjusting the values of the quantisation tables. Thus, optimising JPEG quantisation tables may produce a higher compression ratio and better image quality. Hence, Hirata et al. (1999) proposed to apply the Hamiltonian algorithm for quantisation table optimisation in order to improve the quality of JPEG images and reduce the bit rate. In this method, the high frequency DCT components were finely quantised to improve the image quality while the low frequency DCT components were roughly quantised to decrease the rate (Hirata et al., 1999).

Chang et al. (1999) designed a novel HVS-based quantisation table for JPEG encoders in order to improve the PSNR and then the quality of reconstructed images. The proposed method incorporated the human visual system with the quantiser to determine the perceptual importance of different DCT coefficients and therefore to derive a perceptual quantisation table. As a result, the performance of the proposed quantisation table was better than that of the default JPEG table. Additionally, JPEG images produced by using the proposed quantisation table had a higher PSNR and better perceptual quality than those produced by using the default JPEG table (Chang et al., 1999).

More recently, Costa and Veiga (2005) proposed a method to generate a JPEG quantisation table using Genetic Algorithm techniques. The final goal of this model was to improve the quality (enlarge the PSNR) of reconstructed JPEG images. As a result, JPEG images reconstructed using the generated quantisation table had better quality than those reconstructed using the default quantisation table (Costa and Veiga, 2005).

Chang et al. (2007) proposed a reversible JPEG steganographic system. They modified the components of the default quantisation table in the medium and high frequency range (values are diminished) to increase the image quality. Therefore, a modified quantisation table has been used instead of the default quantisation table. As a result, the quality of stego images has been improved when the same steganographic capacity has been applied (Chang et al., 2007).

3.9 Image Chrominance and Steganography

A grayscale image can be defined as a continuous-tone image that has only one component (e.g. Y). However, a colour image is a continuous-tone image that has more than one component (e.g. Y, Cb and Cr) (ISO-DIS, 1992). Almost always the colour space Y, Cb, Cr is used to store JPEG images. The component Y (luminance) represents the intensity of the image while the components Cb and Cr (chrominance) specify the blueness and redness of the image respectively. Using only the Y component in such a colour model (Y, Cb, Cr) produces a grayscale representation of the colour image (Miano, 1999). Thus, grayscale images represent special cases of colour images. As a result, colour images can be used as cover images but we have to take all of these components (Y, Cb, and Cr) into consideration. Nevertheless, most studies focus on data hiding in the luminance component and therefore grayscale images rather than hiding in the chrominance components of colour images. Even though some works have used colour images for data hiding, few studies investigate the usage of the chrominance components for steganography.

One of these is Mastronardi et al (2001), who studied the effects of steganography in JPEG images. They proposed modifying only the quantised DCT coefficients whose values are greater than a given threshold in order to maintain the compression ratio. Thus, higher thresholds reduce the capacity and therefore reduce the noise injected to the stego image. Additionally, the downsampling process applied to the chrominance components (Cb and Cr) reduces the size of each component to be one-quarter of the image size. Besides, the compression algorithm works much better with the chrominance coefficients since most quantised DCT coefficients become zeros. As a result, the capacity of the chrominance coefficients (Cb and Cr) is much less than that of the luminance coefficient (Mastronardi et al., 2001). However, this study used only one image as a cover image (Lena 512x512) and did not address the main steganography requirements such as the stego image quality and steganographic capacity. However, using a threshold of T=2, 5000 bytes were embedded in the Lena image (in all components) with 50% of the coefficients representing a noise.

Seitz (2005) states *"It is important for the designer of data compression techniques to understand that the human eye is less perceptive to colour than to luminance"*. Therefore, data compression techniques should consider that the chrominance components can tolerate more noise than the luminance component without affecting the perceived image quality in a significant way. Thus, when the size of chrominance data is chosen to be one half to one quarter of luminance size, it will not influence human perception (Seitz, 2005).

In the YUV image model, Y represents the luminance or the brightness component while U and V represent the colour components. The HVS is less sensitive to image colour than to image brightness. Therefore, the impact of changes in the image brightness is more discernable by the human eyes than the changes in the image colour components (Stanescu et al., 2007). Depending on this limitation of human eye, Stanescu et al. (2007) proposed a steganography method based on hiding secret information in the (V) channel. This method transforms (RGB) image into (YUV) model, embeds data in the (V) channel, and then transforms the stego image back to (RGB). The average difference (error) between the cover image and the stego image was around 0.76%. Since this error is very small, it cannot be perceived by the human eye. Also, the capacity of different image components has not been addressed in this study.

Generally, in JPEG compression, both chrominance components (Cb and Cr) are downsampled simultaneously in order to reduce the amount of information they offer. However, Garg (2008) downsampled only one chrominance component every time in order to analyse the impact of each chrominance component on the JPEG compression. And found that downsampling the chrominance blue component (Cb) provided a better compression ratio than downsampling the chrominance red component (Cr). Additionally, downsampling chrominance red component provided a larger PSNR than downsampling the chrominance blue component (Garg, 2008).

Baba et al. (2009) proposed a new watermarking scheme for digital images. Thus, they embedded a watermark in different colour components in a semi-random way. Instead of hiding the secret bits directly in the LSB of quantised DCT coefficients, a pseudorandom bit sequence is generated in order to indicate which component (Y, Cb, or Cr) to be used for secret data hiding. Furthermore, a block number sequence generator is used to indicate the blocks that will be used for hiding. Therefore, the LSB of a selected AC coefficient is used to hide a secret bit (watermark bit) (Baba et al., 2009). So, the embedding capacity of this method is one bit per block which is quite enough for watermarking applications, but is very limited for covert communication purposes.

This section described the previous studies related to information hiding in image chrominance. To this end, the chrominance components of JPEG images have not been considered as a genuine or valid cover medium of steganography apart from some attempts to investigate their applicability for steganography. To the best of our knowledge, this is the first work that attempts to analyse and evaluate the impact of using JPEG chrominance components for steganography. Thus, these components may be used to increase the steganographic capacity and/or imperceptibility of JPEG steganography. This is exactly the third objective of our research; investigating the impact of using chrominance components for steganography.

3.10 Evaluating the Quality of Stego Images

Both objective and subjective image quality evaluation methods have been investigated and used in the literature for various purposes. Hence, some applications use the subjective methods in order to evaluate some compression algorithms. Additionally, some applications evaluate many objective methods by using the subjective methods as reference methods. Other applications investigate the relationship between the subjective and objective methods.

Basically, the objective methods, and particularly the PSNR metric, are used to evaluate the quality of stego images. Thus, PSNR measures the efficiency of a particular stego method over another in terms of imperceptibility or stego image quality. This measure has been tested and validated to be used with many image processing applications. However, what if the PSNR is not a suitable or reliable measure for stego image quality? Thus, this section describes the previous studies related to both objective image quality evaluation methods (PSNR) and subjective image quality evaluation methods.

Ogihara et al. (1996) proposed a new DCT-based steganography method. Hence, they evaluated the efficiency and effectiveness of their method by measuring the quality of stego images using a subjective evaluation method. Therefore, they asked 20 persons to arrange ten printed and shuffled images (one reference and nine stego images) according to their quality (Ogihara et al., 1996).

Stoica et al. (2003) performed both objective and subjective tests on two high resolution images compressed using two JPEG 2000 algorithms. The objective quality was evaluated by three methods (PSNR, normalized MSE, and normalized colour difference methods). However, they performed the subjective tests with a panel of ten observers and three images were presented at the same time; the reference image was placed in the middle, while both compressed images were randomly positioned on the left or the right of the reference image. As a result, objective image quality measures did not offer the same results as the subjective quality evaluation tests. Additionally, the subjective evaluation was in contradiction with the objective measures (Stoica et al., 2003).

Kong et al. (2003) stated that: *"Most researchers still use Peak-Signal-to-Noise Rate (PSNR) for their perception evaluation, while ignoring the specialty of steganography"*. Therefore, they developed an objective image quality evaluation method based on both characteristics of steganography and the HVS. As a result, this method was superior to the PSNR (Kong et al., 2003).

Grgic et al. (2004) investigated the reliability of nine objective picture quality measures for application in still image compression systems. Thus, the correlation of these measures with subjective image quality measures has been evaluated. So, they examined the correlation between the MOS and each objective measure. Additionally, they evaluated the effects of different image compression algorithms, compression ratios, and image contents on the image quality. As a result, they found that some of these objective image quality measures correlate well with the subjective image quality (perceived image quality) for a given compression algorithm but they are not reliable for other different algorithms. This study used the double stimulus impairment scale (DSIS) as a testing method and 20 non-expert viewers (Grgic et al., 2004).

In related work, Bailey and Curran (2006) evaluated the strengths and weaknesses of seven steganography methods using visual inspection and automated detection techniques. The methods chosen were implemented and analysed using only GIF (Graphics Interchange Format) images. Furthermore, all steganography methods tested use the least significant bits of pixel values technique or the rearrangement of image colours technique to hide the secret information in GIF images. The stego images were evaluated by comparing them with original images using passive observation. Moreover, 13 independent observers have been asked about the vulnerability of detection for these seven steganography methods through visual inspection. Basically, viewers were provided with 18 folders and each folder contained 6 images (one was stego image while the other 5 were original images). Therefore, the viewers were asked to identify the stego image within each folder (Bailey and Curran, 2006).

Marini et al. (2007) claimed that insufficient attention has been devoted to assess the quality of watermarked images. The PSNR and MSE, which are commonly used to assess the fidelity of information hiding algorithms, are neither suitable nor reliable to assess the visual impairment present in an image. Since these statistical differences do not represent visual impairment, they used subjective experiments to evaluate the invisibility of many watermarking algorithms. They proposed to use the double stimulus impairment scale (DSIS) method since there is not a standard method for watermarking purposes. Moreover, this subjective method was compared with the PSNR metric and other five objective methods (not specifically designed to evaluate the perceptual quality of watermarking). As a result, the correlation between the PSNR and subjective evaluation was 0.68 and all other objective metrics tested outperformed the PSNR in this regard (Marini et al., 2007).

Since the quality evaluation of images is different from the quality evaluation of moving images described in the ITU standard (ITU-R-BT.500-11, 2002), Simone et al. (2009) developed a subjective evaluation method for still images. The proposed method was a modified version of double stimulus continuous quality scale (DSCQS) method. Here, images are shown simultaneously in each presentation and the subject has no time limit to assess the quality of each image. Moreover, the subject is asked to detect the impaired image in each pair and assess only its quality instead of assessing the quality of both images (Simone et al., 2009).

The suitability and efficiency of the PSNR as a fidelity measure for watermarked images has been investigated in (Marini et al., 2007). However, in the subjective assessment, observers are told which image is watermarked and are asked to judge its quality compared to the quality of the original image. Furthermore, this study uses the DSIS method, so if the quality of a watermarked image is better than the reference one for instance, viewers are unable to reveal such a decision or vote. Thus, this represents one of the limitations of DSIS. The work described in this thesis attempts to evaluate the validity and reliability of PSNR measure to be used with JPEG stego images by using the DSCQS method. Thus, this represents the fourth objective of our research.

3.11 Text and SOAP Based Steganography

There is a relatively small number of text steganography studies in comparison to that of image, video and audio based steganography. As alluded to earlier, this might be due to the lack of redundancy in text files (Inoue et al., 2001).

In this context, Por and Delina (2008) improved the open space method proposed by Bender et al. (1996). Hence, they proposed a hybrid steganography method for text by combining both inter-word spacing and inter-paragraph spacing methods. To increase the steganographic capacity, whitespaces between words and right-justified paragraphs are used for data hiding. Additionally, the cover text is dynamically generated according to the size of the secret message (Por and Delina, 2008).

Shirali-Shahreza (2008) proposed a new steganography method for texts. This method is based on the different spelling of some words in English between UK and US. For example, "centre" has different spellings in the UK (centre) and the US (center). Table 3.3 shows a list of some words which have different spelling in UK and US (Shirali-Shahreza, 2008).

American Spelling	**British Spelling**
Favorite	Favourite
Criticize	Criticise
Fulfill	Fulfil
Center	Centre
Dialog	Dialogue
Medieval	Mediaeval
Check	Cheque
Defense	Defence
Tire	Tyre

Table 3.3: List of Some Words which have Different Spelling in UK and US

In another study, the model proposed by Shirali-Shahreza and Shirali-Shahreza (2008) defines a text steganography method based on substituting the words which have different terms in UK and US. For example, (Gas) has different terms in the UK (Petrol) and the US (Gas) (Shirali-Shahreza and Shirali-Shahreza, 2008). Table 3.4 shows a list of some words which have different terms in UK and US.

American English	British English
Account	Bill
Candy	Sweets
Closet	Cupboard
Fall	Autumn
Gas	Petrol
Mail	Post
Movie	Film
Package	Parcel
Soccer	Football

Table 3.4: List of Some Words which have Different Terms in UK and US

Liu et al. (2009) proposed a text steganography method to be used in online chats. This method is based on an Internet meme named typoglecymia, which means that changing the order of word's middle letters has a slight to no effect on the ability of skilled readers to understand the text (e.g. Guitar and Guiatr). Thus, this steganography method uses the redundancy found in the interior letters order. Since this letter randomisation looks like a common error made by chatters due high speed typewriting, it is likely to be used in online chats. Moreover, online chat texts usually contain mistakes (Liu et al., 2009).

Inoue et al. (2001) proposed five techniques of information hiding in XML documents. The first method uses the representation of the empty XML element which has two forms: either a start-tag immediately followed by an end-tag, or an empty-element tag. The second method uses white spaces in tags to embed the secret data. In the third method, the order in which elements appear is exchanged to embed the secret data; yet, the appearing order of attributes in the element is exchanged in the fourth method. The fifth method uses two or more elements that could contain each other and exchanges the inner-tags with outer-tags to embed the secret data (Inoue et al., 2001).

Memon et al. (2008) designed four steganography techniques to ensure that the confidentiality and integrity of data is maintained in XML documents. In the first technique, random characters are inserted between XML tags and their values while the second technique uses the procedure of shuffling the XML tags in a predetermined sequence. In the third technique, the original order of tags (before the shuffling process) is saved in attributes. However, the fourth technique reverses the sequence of characters (Memon et al., 2008).

Zhang et al. (2007) proposed a steganography method depending on the text characteristic of SOAP technology in order to hide information in SOAP messages. Therefore, the physical properties of SOAP keywords and namespaces (self-defined) are used as cover message. A character string is initialized by converting every letter in these keywords and namespaces into lowercase. Coordinating every secret bit with every letter of the character string, a specific letter is converted into a capital letter only when the secret bit is “1”. However, the number of SOAP keywords is limited for a short SOAP message (Zhang et al., 2007).

To the best of our knowledge, there is only one study regarding the steganography with SOAP messages (Zhang et al., 2007). In this method, the stego SOAP looks suspicious since some characters of this message are in lowercase while others are in uppercase. Therefore, the overall shape of the stego SOAP attracts the attention of possible observers. Additionally, this method does not comply with the case-sensitivity nature of XML documents. Nowadays, SOAP messages represent a very important way of communication between parties over Web and other networks. So, SOAP may represent a significant medium and non-conventional cover of steganography. Thus, the fifth objective of our study investigates the capability of hiding secret data undetectably within SOAP messages and we will propose a novel method in this regard.

3.12 Research Aims and Objectives

It has been shown that the steganographic capacity and stego image quality/ imperceptibility are the most important aspects of image-based steganographic systems. Furthermore, there is a tradeoff between the steganographic capacity and stego image quality. Therefore, both increasing the steganographic capacity while maintaining the stego image quality, and enhancing the stego image quality while maintaining the steganographic capacity represent good contributions. As such, the main aim of our research is to improve the steganographic capacity and stego image imperceptibility of image based steganography. Moreover, our research aims to examine and investigate the capability of using SOAP message as steganography cover instead of using conventional digital files. Thus, in order to achieve theses research aims, a series of five investigations will be carried out, each targeting a major research objective of our research.

- **Objective 1: Investigating the impact of using 16x16 blocks and quantisation tables on JPEG based steganography**. To this end, we intend to investigate the impact of using 16x16 non-overlapping blocks and quantisation table with JPEG instead of standard 8x8 on the main properties of reconstructed images. In addition, we are going to find out a novel high-capacity steganography method based upon 16x16 quantisation table. Our work in this respect will be detailed in Chapter 4.
- **Objective 2: Investigating the impact of using optimised JPEG quantisation tables on the steganographic capacity and the quality of JPEG stego images.** In order to improve the quality of JPEG stego images, we intend to examine the impact of JPEG quantisation values on the quality of coded images. Therefore, an optimisation method will be employed to get optimised quantisation tables and then to use these optimised quantisation tables with JPEG steganography. Additionally, we are going to find out a novel steganography method that enhances the quality of stego images and/or increases the steganographic capacity. Our work in this respect will be detailed in Chapter 4.
- **Objective 3: Investigating the impact of using chrominance components for steganography.** To further consider the aim of steganographic capacity increasing, we intend to investigate the impact of using image chrominance components for steganography on the main aspects of steganography. Therefore, we are going to hide secret data within chrominance and/or luminance component/s and then evaluate the

impact of this on the main aspects of steganography. Thus, in order to increase the steganographic capacity, the feasibility of using all image components for steganography will be examined. Our work in this respect will be detailed in Chapter 5.

- **Objective 4: Evaluating the reliability of PSNR as a quality measure of stego image.** Peak signal-to-noise ratio (PSNR) is almost the merely measure used to evaluate the quality of stego images. Additionally, superiority of a steganography method over another one (from stego image quality perspective) is mostly evaluated and measured by this metric. Therefore, we intend to evaluate the reliability and consistency of PSNR measure when used to measure the quality of stego images. Since subjective quality evaluation methods are the most reliable measures of image quality, a subjective measure will be adapted and used for this purpose. Our work in this respect will be detailed in Chapter 6.
- **Objective 5: Examining the capability of using SOAP message for steganography.** Unlike conventional steganography methods that use digital cover files, communication protocols may offer a feasible path for secret information to be transmitted over networks. Relying on the structure and characteristics of the SOAP message, we intend to examine the capability of hiding secret data within SOAP messages in undetectable way. However, the research area of SOAP steganography is emerging and needs more investigation. Therefore, we intend to find out a novel method of steganography to be used with SOAP messages. Our work in this respect will be detailed in Chapter 7.

3.13 Research Methodology

The only way to make causal inferences is to use experimental methods. In order to test a theory or explain a phenomenon, we can use an experimental method that enables us to control independent variables and then measure the values of dependent variables. Thus, experiments represent a straightforward and standard practice of manipulating independent variables and then statistically analysing the generated data in order to test research hypotheses. Furthermore, experimental design is used to determine significant differences between controlled conditions. One of the advantages of experimental research is that it enables other researchers to easily replicate the experiment and validate the results. Hence, it is considered as an accurate method of research (Shuttleworth, 2008). As a result, the researcher can establish a causal relationship between variables by manipulating independent variable(s) to assess the effect upon dependent variable(s).

Structured experiments follow a predefined order in order to measure pre-designed and specific experimental factors. However, non-structured experiments do not measure pre-designed experimental factors. They therefore provide a rich source of information which requires more complicated data-analysis methods. On the other hand, laboratory (true) experiments assist the researchers to undertake accurate and highly-focused studies and enable them to control all variables. Consequently, this kind of experiments is considered to represent an accurate and consistent research methodology. Nevertheless, field experiments usually focus on investigating a causal relationship between variables within a real world environment. Thus, field experiments do not give a full control of the environment and it is almost impossible to control confound variables. However, the ability to generalise the results of laboratory experiments is limited due to the artificial (unreal) environment (Coolican, 1994).

The choice between structured and non-structured experiments represents the choice between limited or rich data. However, we aim to evaluate the main steganography aspects which are clearly identified and structured (measuring specific variables). Thus, structured experimentation will be used throughout this study to measure pre-defined steganography aspects. Mainly, the choice between laboratory and field experiments represents the choice between controllable and non-controllable variables. In our study, we aim to measure the effect of our proposed scenarios and nothing else on the steganography aspects. Hence, confound variables must be controlled, which is facilitated by laboratory experiments rather than other kind of experiments.

Accordingly, experimental design based on structured laboratory experiments represent the best choice for our work aim and objectives since our concern is improving a predefined set of steganography key-requirements and we need a full control over the experiment variables. In the previous section, we have identified a clear set of structured aims and objectives which facilitates the effective measurement of critical experimental factors. In addition to the objective measurements, subjective measurements were used in our study in order to evaluate the reliability of objective metrics.

Experimental designs should clearly explain the form of the problem under investigation, the type of the experimental design, the implementation of the experiment, the analysis of the data, and the interpretation of the results (Hinkelmann and Kempthorne, 2008).

In our work, we wish to improve the main aspects and features of steganography, mainly the steganographic capacity and imperceptibility. Accordingly, we will propose novel steganography methods, utilising various scenarios to hide data in different image components, and using different measures to evaluate the quality of stego images. Thus, our hypothesis is that all these actions and manipulations (independent variables) have significant effects on the main steganography aspects (dependent variables). Since the experimental design is regarded as the most accurate and obvious standard for testing a hypothesis (Shuttleworth, 2008), we will use this methodology in order to achieve our research aim and objectives.

The main aspects and features of our proposed methods and scenarios represent the variables we are trying to understand and study. Thus, we can implement an experiment to test the relationship between a specific input scenario (i.e. a proposed technique) and the outcome (i.e. steganography aspects). Essentially, we will use the outcome of other methods or scenarios in order to evaluate the performance or effectiveness of our methods' outcome. Therefore, we will assess this relationship by comparing the outcome of proposed method with that of other methods since we do not use measurable independent variables. Such type of experiments is commonly called a comparative experiment (Hinkelmann and Kempthorne, 2008).

As a result, the experimental design based on structured laboratory experiments is used in Chapters 4, 5, 6 and 7. Nevertheless, the design and implementation of experiments is explained in detail within each chapter. The data for each experiment is then analysed and the results of each experiment is interpreted.

3.14 Summary

In this chapter we have introduced the reader to issues concerning DCT-based JPEG encoding and the role of quantisation tables. We have identified the main JPEG steganography methods. Additionally, key aspects of text steganography and SOAP messages have been explained. We have also reviewed the literature of JPEG and text steganography. Accordingly, we have identified a need for improving the main characteristics of JPEG steganography and explained the benefits of using SOAP messages for steganography. We then summarised the aims and objectives of our study and finally we described the research methodology used in Chapters 4, 5, 6 and 7. In the next chapter, we will consider the first and second objectives of our research.

Chapter 4: JPEG Steganography and Quantisation Tables

4.1 Introduction

The two most important aspects of any steganographic system are the capacity and the imperceptibility. Nevertheless, it is quite difficult to increase the steganographic capacity and simultaneously maintain the imperceptibility of stego images, so it is still a research challenge. In our study, we aim to increase the steganographic capacity and maintain/enhance the quality of JPEG stego images. This will be achieved through two approaches. Firstly, we intend to use extended, modified, and optimised 16x16 quantisation tables (quantisation tables based techniques). Secondly, we will use image chrominance components for steganography in addition to the luminance component (chrominance based techniques). Therefore, the impact of using grayscale or colour versions of a given cover image will be examined in terms of the main steganography requirements. Consequently, in this chapter, we intend to consider objectives 1 and 2, which concerns the usage of quantisation tables to improve the image steganographic capacity and stego image quality. For that reason, we will propose two novel steganography methods.

The structure of this chapter is as follows: section 4.2 provides a description for the role of quantisation tables in JPEG compression. Additionally, it provides a summary of the relevant studies relating to the impact of quantisation tables on steganography. In section 4.3, we present two steganography methods which we propose in order to improve both steganographic capacity and stego image quality. Information concerning the research methodology used and the experiments employed specifically to measure the performance of these steganography methods are presented in section 4.4. The relationship between the DCT calculation and computational complexity is explained in section 4.5. In section 4.6, we consider our results, and lastly, in section 4.7, the conclusion is presented.

4.2 The Role of JPEG Quantisation Tables

In the JPEG encoder, we can get rid of unimportant DCT coefficients by dividing these coefficients by quantisation values and then rounding the result to the nearest integer. Thus, selecting an appropriate quantization table is something of a black art and it is likely that future research will yield better tables in terms of compression ratio and perceived image quality. Additionally, implementation of improved quantization tables causes no compatibility problems since decoders don't care how the quantisation table was picked; they only read the tables from the compressed image. As a result, the values of quantisation tables can possibly be manipulated in order to control the image quality and compression ratio. Thus, the quantisation tables have a considerable impact on the performance of JPEG coding.

Since the quantisation table is not a part of the JPEG standard, users are allowed to design or redefine the quantisation table in order to control the quality of reconstructed image and the compression ratio (Chang et al., 1999). Therefore, specifying the quantisation values is left up to the application (ISO-DIS, 1992). Miano (1999) states that *"If you are implementing a JPEG encoder you can come up with your own scaling or use any other method you want for generating quantisation values"*. Therefore, it is important to find out or design a quantisation table which produces a better image quality than that obtained by the JPEG default tables (Kong et al., 2003). As a result, using optimised quantisation tables enable us to either get stego images with better quality or hide more secret data. In fact, there is a tradeoff between the stego image quality and steganographic capacity, so we can transform this quality enhancement into capacity improvement by hiding more data and getting a stego image with quality nearly equal to that before the enhancement process.

Also, many methods have been proposed to find out the optimum quantisation table for JPEG compression. Accordingly, the majority of methods used to improve the JPEG image quality relies on quantisation table optimisation. Thus, using optimised quantisation tables instead of the default tables for JPEG compression produces better reconstructed images in terms of quality.

In this chapter, we propose a novel steganography method that utilizes 16x16 non-overlapping blocks and then a 16x16 quantisation table in order to increase the steganographic capacity. Additionally, we propose another steganography method, based on optimised quantisation tables (8x8 and 16x16), in order to improve the quality of stego images and increase the steganographic capacity.

4.3 JPEG Steganography Methods Proposed

Quantisation tables of JPEG encoding have a fundamental impact on the compression ratio and the quality of the reconstructed or compressed image. Therefore, depending on the fact that there is no default or standard quantisation table for JPEG compression, we propose two steganography methods in this section. The first method depends on the fact that there is no 16x16 quantisation table in the JPEG standard but such quantisation tables may provide better results than 8x8 tables in terms of reconstructed image quality. Additionally, dividing the input image into non-overlapping blocks of 16x16 may provide a wider range of middle frequency components which can be used for steganography and therefore this may increase the hiding capacity. However, the second method depends on the user capability to generate application-based quantisation tables to be used instead of the JPEG default tables. Thus, using optimised quantisation tables along with steganography may enhance the stego image quality and/or steganographic capacity.

4.3.1 Steganography Method Based on 16x16 Quantisation Tables

As there are no samples for quantisation tables larger than 8x8 in the JPEG standard, Bracamonte et al. (1997) used disjoint blocks of 16x16 pixels in order to improve the JPEG compression. They found that just four coefficients had significant magnitudes (low frequency coefficients). Therefore, only these four coefficients have to be calculated and this reduced the computational overhead significantly. As a result, they got a better compression ratio by using larger block-sizes for JPEG compression.

In image blocks of 16x16-pixels, more middle frequency coefficients can be used for steganography since the significant DCT coefficients are limited in such blocks and the range of middle frequencies is wide. This might increase the embedding capacity and preserve the stego image quality. In this section, we propose a novel JPEG steganography method based upon blocks of 16x16 pixels and modified 16x16 quantisation table.

As mentioned before, almost all steganography research done in the JPEG transformation domain divides a given cover image into non-overlapping blocks of 8x8 pixels. However, since computational capabilities have improved significantly over the last decade, calculating the DCT for blocks of 16x16 pixels or larger may be much more feasible and faster than before. In our method, we will divide the cover image into non-overlapping blocks of 16x16 pixels and use a 16x16 quantisation table.

Basically, this method is inspired by the JMQT method, so it takes the general framework from this method. In fact, the most significant part of each block of an image (the energy of the image) is concentrated in the low frequency part (upper left) of these blocks. Thus, the coefficients of this part should be kept intact as possible to maintain the image quality since modifying these coefficients may degrade the reconstructed image. However, the insignificant part of blocks is concentrated in the high frequency part (bottom right) of these blocks. Also, most of these coefficients will be zeroed and discarded after the quantisation process in order to get rid of redundant data and compress the image. Therefore, we should avoid hiding data within these areas since hiding secret data within such coefficients may be lost. As a result, hiding secret data within the middle frequency may represent a good choice of steganography in order to balance both stego image quality and compression rate.

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

Table 4.1: The Default JPEG Quantisation Table for Luminance

8	6	5	8	12	20	26	31
6	6	7	10	13	29	30	28
7	7	8	12	20	29	35	28
7	9	11	15	26	44	40	31
9	11	19	28	34	55	52	39
12	18	28	32	41	52	57	46
25	32	39	44	52	61	60	51
36	46	48	49	56	50	52	50

Table 4.2: The Default JPEG Quantisation Table (Luminance)-Quality Factor=2 (JSteg and F5 Methods)

In order to verify the performance of our method, we will compare its aspects with that of JSteg, F5, and JMQT methods. The main characteristics of these methods have been explained in Chapter 3. These methods have been chosen to compare their aspects with that of our method for various reasons: JSteg (Upham), for instance, is a widely used and simple LSB-based tool in the DCT domain (Li and Wang, 2007; Upham). Moreover, it uses the default JPEG quantisation table, build upon the standard JPEG compression, and it has a limited steganographic capacity. Additionally, it hides secret data by modifying the LSB of quantized DC and low-frequency DCT coefficients and this degrades the stego image. On the other hand, increasing the steganographic capacity of JPEG images was one of the designing goals of F5 algorithm (Westfeld, 2001a). Furthermore, this tool is publicly available and it hides secret data by decrementing the absolute values of non-DC and low frequency coefficients rather than flipping their LSB. Finally, we build our method based on the JMQT method. Moreover, this method hides secret data by modifying the two LSBs of each pre-defined middle frequency quantised DCT coefficient of each image block. Thus, it has a very large steganographic capacity compared to other JPEG based steganography methods.

8	6	5	8	1	1	1	1
6	6	7	1	1	1	1	28
7	7	1	1	1	1	35	28
7	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	56	50	52	50

Table 4.3: The Modified Quantisation Table (JMQT Method)

16	8	7	6	6	1	1	1	1	1	1	1	1	1	1	1
7	7	6	6	1	1	1	1	1	1	1	1	1	1	1	30
7	6	6	1	1	1	1	1	1	1	1	1	1	1	30	28
6	8	1	1	1	1	1	1	1	1	1	1	1	32	35	29
8	1	1	1	1	1	1	1	1	1	1	1	32	35	32	28
1	1	1	1	1	1	1	1	1	1	1	35	40	42	40	35
1	1	1	1	1	1	1	1	1	1	35	44	42	40	35	31
1	1	1	1	1	1	1	1	1	35	44	44	50	53	52	45
1	1	1	1	1	1	1	1	31	34	44	55	53	52	45	39
1	1	1	1	1	1	1	31	34	40	41	47	52	45	52	50
1	1	1	1	1	1	30	32	36	41	47	52	54	57	50	46
1	1	1	1	1	36	32	36	44	47	52	57	60	60	55	50
1	1	1	1	36	39	42	44	48	52	57	61	60	60	55	51
1	1	1	39	42	47	48	46	49	57	56	55	52	51	54	51
1	1	41	46	47	48	48	49	53	56	53	50	51	52	51	50
1	43	47	47	48	48	49	57	57	56	50	52	52	51	50	50

Table 4.4: The Suggested 16x16 Quantisation Table

The default (luminance) JPEG quantisation table will be used for JSteg and F5 methods but with a quality factor equal to two (Table 4.1 and Table 4.2). However, a modified version of Table 4.2 will be used as a quantisation table for JMQT method (Table 4.3), similar to that used in (Chang et al., 2002). Since quantisation tables can arbitrarily be generated, we suggested and generated a 16x16 quantisation table. However, since our proposed method hides data in the middle frequency part of each quantised DCT block, the middle frequencies of this 16x16 table were set to be one (Table 4.4).

In this stage, we will investigate the effectiveness and efficiency of our method based on 16x16 non-overlapping blocks, 16x16 quantisation table, and middle frequency quantised DCT coefficients. Thus, the generation (creation) process of the proposed quantisation table is not considered as important issue since we just built it from scratch for research purposes and without any simulation or optimisation algorithms. However, optimisation of quantisation tables is considered and investigated further in our second method proposed.

4.3.1.1 Embedding and Extracting Procedures

The process of embedding a secret message (M) in a cover image (C) and then extracting the message (M) from the stego file using our first steganography method is illustrated in Figure 4.1. However, the embedding procedure consists of five steps as shown in Figure 4.2.

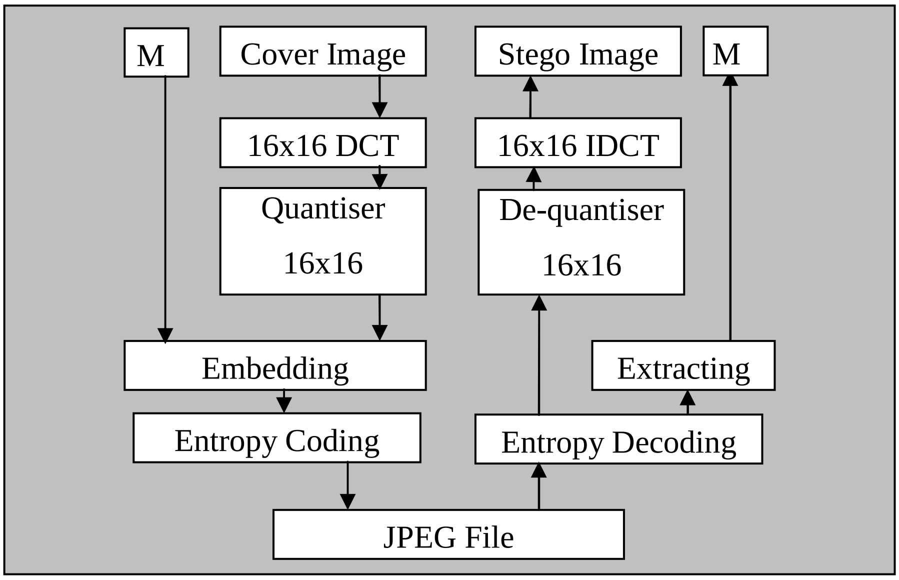


Figure 4.1: The Block Diagram of Embedding and Extracting Procedures

On the other hand, the procedure of extracting the embedded message from the stego JPEG file consists of the following steps; the JPEG file (stego image) is entropy decoded using the coding tables (Huffman tables) located in the image header. As a result, we get the blocks of quantised DCT coefficients modified according to the secret message. From each pre-defined middle frequency coefficient of each block we retrieve the least two-significant bits (secret bits). We put these retrieved bits in the same order in which they were embedded to get the secret message (M).

1. *The message (M) to be embedded in the cover image is randomly generated.*

2. *The cover image is divided into non-overlapping blocks of 16x16 pixels and then the DCT is used to transform each block into DCT coefficients.*

3. *The DCT coefficients are scaled by the modified 16x16 quantisation table (Table 4.4). In this quantisation table, the values of (1) represent the middle frequencies to be used for embedding (121 values). The quantised DCT coefficients of each block are rounded to the nearest integers and then set in zigzag scan order.*

4. *The least two-significant bits of each middle frequency coefficient in the quantised DCT blocks are modified to embed two secret bits.*

5. *The JPEG entropy coding (DPCM, Run-Length coding, and Huffman coding) is applied to compress these resultant blocks, and then the JPEG file is obtained.*

Figure 4.2: The Embedding Procedure of our First Steganography Method

4.3.2 Steganography Method Based on Optimised Quantisation Tables

In this section, we propose a new hybrid JPEG steganography method based on optimised quantisation tables. This method has a higher embedding capacity and better stego image quality compared to other methods that use default tables. Furthermore, we investigate and evaluate the impact of using optimised quantisation tables (Opt-QT) instead of the default quantisation tables (Def-QT) on the performance of JPEG steganography.

Since the quantisation table is not part of the JPEG standard, users are allowed to design or redefine the quantisation table to control the quality of the reconstructed image and the compression ratio. Hence, many methods have been proposed in order to optimise the quantisation tables of JPEG compression (refer to Chapter 3).

We need an optimisation method for the 8x8 and 16x16 quantisation tables so they can be used with JSteg, F5, JMQT, and our method proposed above (refer to section 4.3.1). Therefore, the general strategy for modelling optimal quantisation tables presented in (Monro and Sherlock, 1996) will be used in order to generate optimised quantisation tables. This method can be used for both the JPEG image compression standard and for an extension of the JPEG approach to image block sizes other than 8x8 pixels. This is the main reason behind our selection of this flexible optimisation method, since it is appropriate for both 8x8 and 16x16 quantisation tables. In the next subsection, we explain this optimisation objective and process and provide a background of relevant theory.

4.3.2.1 The Optimisation Method of JPEG Compression

Monro and Sherlock (1996) presented a general strategy for modelling and generating optimal quantisation tables for both JPEG compression and its extension (general block sizes). Their aim was minimising the Root Mean Square (RMS) error between original and recovered images while maintaining the compression ratio. Thus, the objective of this optimisation method was producing optimal rate-distortion performance.

Firstly, they selected a block of 128x128 pixels from three different images and the quantisation tables were obtained using simulated annealing on all 64 parameters in the quantisation table. At each step, the DCT coefficients were compressed (compression ratio *C*) and the Root Mean Square (RMS) total error (*E*) between the original and reconstructed blocks was measured. Thus, they minimised a composite cost function F (keeps *C* close to a desirable value C_0).

$$F = E^2 + (C - C_0)^2 \tag{4.1}$$

For a selected C_0, a quantisation table can be located by making *E* minimal. These optimised quantisation tables gave better results than the default tables when applied to other test images.

Then, directly optimised tables were produced for each image and compared with that of JPEG. Examining the relationship between the results of these quantisation tables (4 quantisation tables), it was found that, for a given compression ratio, the quantisation table can be modelled by an equation as follows.

$$Q_{xy} = A + By^C + D\,|\,z - E\,|^F \tag{4.2}$$

where x and y are the DCT coefficient indices,

$z = x + y$ is the Manhattan distance of a coefficient from $(0,0)$, and A, B, C, D, E and F are six model parameters.

They optimised for several compression ratios by using the simulated annealing on the six parameters, which is faster than direct optimisation on 64 parameters. However, it was found that there are tradeoffs between B, C and D parameters were complicating the results. To simplify the model, they experimentally chose the following 3-parameter model for JPEG quantisation tables:

$$Q_{xy} = A + Dz^F \tag{4.3}$$

As a result, the error of the modelled table was lower than that of the default JPEG table for the same compression ratio. This model was generalised in order to get good fidelity at higher compression ratios. Thus, larger blocks of pixels were used to achieve this. The three model parameters A, D and F for a specific compression ratio were determined using further optimisations and modelled as following:

For 8x8 blocks:

$$A = 5.43 + 2.15C_R \tag{4.4}$$

$$D = 0.0969 - 0.0565C_R + 0.00749{C_R}^2 \tag{4.5}$$

$$F = 1.83 \tag{4.6}$$

where C_R is the desired compression ratio.

For 16x16 blocks: x→7x/15 and y→ 7y/ 15

$$A = 10.7 + 1.34C_R \tag{4.7}$$

$$D = -0.129 + 0.0117C_R + 0.00188{C_R}^2 \tag{4.8}$$

$$F = 2.70 \tag{4.9}$$

8	8	8	9	10	10	11	12
8	8	9	10	10	11	12	14
8	9	10	10	11	12	14	15
9	10	10	11	12	14	15	16
10	10	11	12	14	15	16	18
10	11	12	14	15	16	18	20
11	12	14	15	16	18	20	22
12	14	15	16	18	20	22	23

Table 4.5: The Optimised Quantisation Table (JSteg and F5 Methods)

8	8	8	9	1	1	1	1
8	8	9	1	1	1	1	14
8	9	1	1	1	1	14	15
9	1	1	1	1	14	15	16
1	1	1	1	14	15	16	18
1	1	1	14	15	16	18	20
1	1	14	15	16	18	20	22
1	14	15	16	18	20	22	23

Table 4.6: The Optimised and Modified Quantisation Table (JMQT Method)

In order to get approximately the same compression ratio as with the default quantisation table, we selected the compression ratio to be $C_R = 12$ and the quality factor to be $Q_F = 4$. Therefore, Table 4.5 represents an optimised 8x8 quantisation table to be used with the JSteg and F5 methods. Moreover, the modified version of this table is used with the JMQT method (Table 4.6).

However, Table 4.7 represents an optimised and modified 16x16 quantisation table to be used with our first steganography method proposed.

7	7	7	7	7	1	1	1	1	1	1	1	1	1	1	1
7	7	7	7	1	1	1	1	1	1	1	1	1	1	1	17
7	7	7	1	1	1	1	1	1	1	1	1	1	1	17	18
7	7	1	1	1	1	1	1	1	1	1	1	1	17	18	20
7	1	1	1	1	1	1	1	1	1	1	1	17	18	20	22
1	1	1	1	1	1	1	1	1	1	1	17	18	20	22	24
1	1	1	1	1	1	1	1	1	1	17	18	20	22	24	26
1	1	1	1	1	1	1	1	1	17	18	20	22	24	26	28
1	1	1	1	1	1	1	1	17	18	20	22	24	26	28	30
1	1	1	1	1	1	1	17	18	20	22	24	26	28	30	33
1	1	1	1	1	1	17	18	20	22	24	26	28	30	33	36
1	1	1	1	1	17	18	20	22	24	26	28	30	33	36	39
1	1	1	1	17	18	20	22	24	26	28	30	33	36	39	42
1	1	1	17	18	20	22	24	26	28	30	33	36	39	42	45
1	1	17	18	20	22	24	26	28	30	33	36	39	42	45	49
1	17	18	20	22	24	26	28	30	33	36	39	42	45	49	52

Table 4.7: The Optimised and Modified 16x16 Quantisation Table

As a result, the quality of stego images when optimised quantisation tables are used is better than that when default tables are used (shown in the section 4.6.2). In the next section, we will take advantage of this quality improvement in order to propose a hybrid JPEG steganography method which has higher steganographic capacity than the method proposed earlier in this chapter. To increase the steganographic capacity, our hybrid steganography method uses an optimised and modified 16x16 quantisation table and two different hiding techniques, namely JSteg and our first method proposed.

4.3.2.2 Embedding and Extracting Procedures

Since using optimised quantisation tables improves the quality of compressed images, we can embed more data and still have good stego image quality. Since there is a tradeoff between stego image quality and steganographic capacity, the quality enhancement of stego image by using optimised quantisation tables can be transformed into capacity improvement. Hence, if we use an optimised 16x16 quantisation table instead of that suggested in our first method proposed, we get better stego image quality. Therefore, we can hide much more data within cover images in order to get similar stego image quality to that of our first method proposed.

1. *The message (M) to be embedded in the cover image is randomly generated.*

2. *The cover image is divided into non-overlapping blocks of 16x16 pixels and then the DCT is used to transform each block into DCT coefficients.*

3. *The DCT coefficients are scaled by the optimised and modified 16x16 quantisation table (Table 4.7). In this quantisation table, the values of (1) represent the middle frequencies to be used for embedding (242 bits). The quantised DCT coefficients of each block are rounded to the nearest integers and then set in zigzag scan order.*

4. *The JSteg method is applied to the low frequency coefficients (top left part, 14 coefficients, of each block). More likely, modifying the DC coefficients, which represent the mean luminance within a block, introduces perceptible blocking artefacts. Thus, DC coefficients are not used for embedding in our proposed method.*

5. *The least two-significant bits of each middle frequency coefficient in the quantised DCT blocks are modified to embed two secret bits.*

6. *JPEG entropy coding (DPCM, Run-Length coding, and Huffman coding) is applied to compress these resultant blocks, and then the JPEG file is obtained.*

Figure 4.3: The Embedding Procedure of our Hybrid Steganography Method

In this section, we propose a hybrid steganography method based on an optimised 16x16 quantisation table, our first method proposed for middle frequency coefficients, and the JSteg method for low frequency coefficients. The procedure of embedding a secret message (M) in a cover image for our hybrid steganography method is described in Figure 4.3. However, the procedure of extracting the embedded message from the stego JPEG is described in Figure 4.4.

> ***1.*** *The JPEG file (stego image) is entropy decoded using the coding tables (Huffman tables) located in the image header. As a result, we get the blocks of quantised DCT coefficients modified according to the secret message.*
>
> ***2.*** *From each pre-defined middle frequency coefficient of each block we retrieve the least two significant bits (secret bits).*
>
> ***3.*** *From each low frequency coefficient of each block (where its value does not equal to 0 or 1), we retrieve the LSB.*
>
> ***4.*** *We put these retrieved bits (from 2 and 3) in the same order of embedding to get the secret message (M).*

Figure 4.4: The Extracting Procedure of our Hybrid Steganography Method

4.4 The Impact of Using Larger JPEG Blocks

JPEG standard divides the input image into non-overlapping blocks of 8x8. Additionally, there are no samples for quantisation tables larger than 8x8 in this standard. This is because the DCT calculation for block-sizes larger than 8x8 pixels may require much more running time and may increase the computational operations and complexity (Bracamonte et al., 1997). Therefore, this might be one of the reasons why the JPEG standard uses blocks of 8x8 pixels. On the other hand, using block sizes larger than 8x8, and therefore larger quantisation tables, may lead to better results in terms of image quality and compression ratio.

Additionally, the fast developments in computers technology, capability and competency may overcome the computation complexity which was existing in computers many decades before.

Monro and Sherlock (1996) stated that the impact of block size on the quality of the reconstructed image can be evaluated by compressing an image to the same ratio using different block sizes. They found that using small block sizes (8x8 pixels) causes excessive blocking artefacts. However, ringing effects become disturbing when very large block sizes are used (20x20 pixels).

Bracamonte et al. (1997) examined the impact of using an adaptive block-size (8x8 and 16x16 pixels) for JPEG coding instead of fixed block-size (8x8 pixels). This adaptive scheme was based on block classification according to some threshold values. As a result, they got a better compression ratio by using adaptive block-sizes for JPEG compression instead of using only the standard 8x8 blocks.

Currently, the relationship between the size of non-overlapping blocks of input image from one side and the reconstructed image quality, compression ration and steganography aspects from another side is not clear enough and needs further investigations.

Calculating 2-D DCT for 8 x 8 block, approximately 64x64=4096 multiply-accumulate operations are required (equation 3.1). However, calculating 2-D DCT for 16x16 block, approximately 256x256=65536 multiply-accumulate operations are required. If we consider larger blocks than 16x16 (for instance 32x32 pixels), approximately 1024x1024=1048576 multiply-accumulate operations are required. Therefore, if we consider the block 2-D DCT implementation, the computational complexity of using larger blocks is extremely higher than that of using smaller blocks and that 2-D DCT of larger blocks may require much more time to be calculated.

Since the DCT has a strong energy compaction property, it concentrates the energy of each block (the most significant coefficients) in a few low frequency coefficients. Thus, using larger block sizes may provide better compression ratio since we can discard more insignificant coefficients from the high frequency band. However, using larger block sizes, the energy of each block will be represented by fewer low frequency coefficients than using smaller blocks.

Therefore, this may degrade the reconstructed image and increase the distortion added to this image.

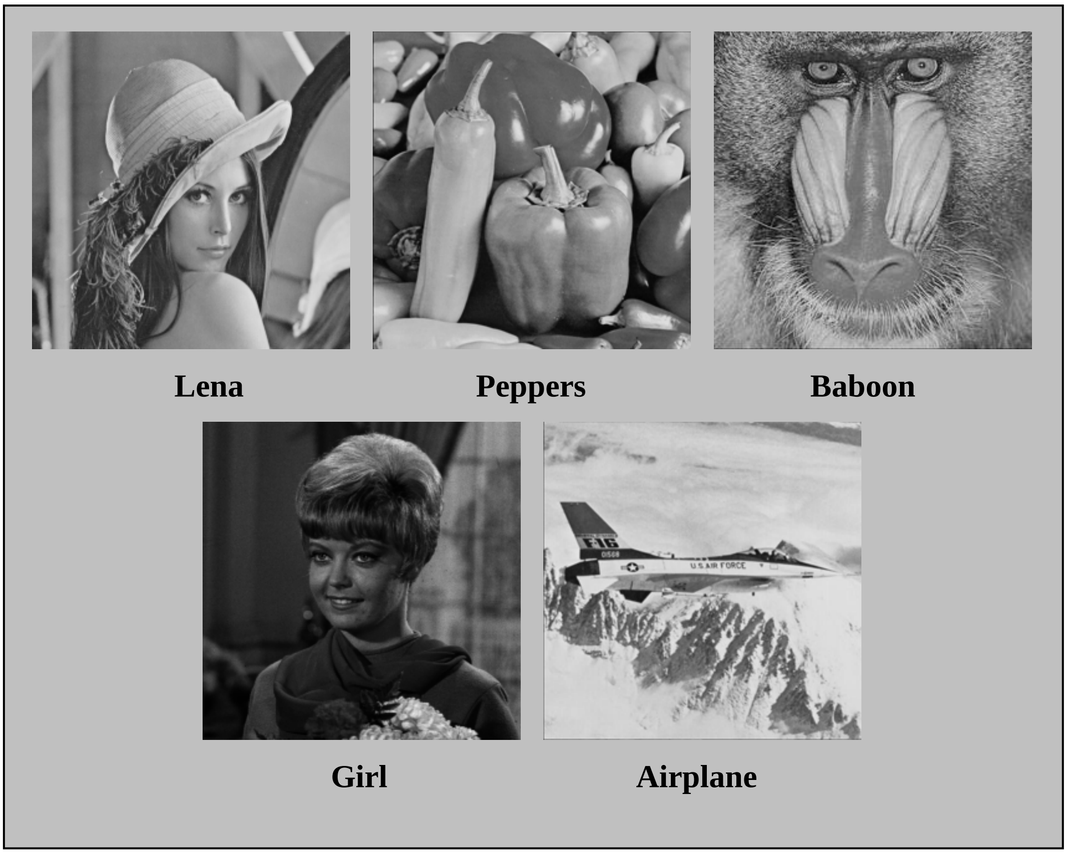
The middle frequency range of JPEG blocks can be expanded by using larger block sizes (such as 32x32 pixels) and appropriate quantisation table. Thus, this wide range of middle frequency coefficients can be used for data hiding, since it represents the most appropriate coefficients to be used for steganography. As a result, using larger blocks can increase the steganographic capacity of input images.

4.5 Experimental Design

In this section, we consider issues relating to the experimental approach, which is used in our study to measure the impact of quantisation tables on the main aspects of steganography. The experimental approach is used in our study to evaluate the performance of our proposed steganography methods. Thus, experiments have been conducted in order to evaluate the efficiency of these methods.

JPEG steganography methods; namely JSteg, JMQT and our proposed methods were coded in Matlab R2007a (V 7.4.0). However, F5 software was downloaded from (Westfeld, 2001b). Additionally, all the experiments were implemented and run on a PC Pentium 4 (2 CPUs, each of 2.13GHz) with 2GB of RAM under the Windows XP operating system.

Mainly, five grayscale images: Lena, Peppers, Baboon, Girl, and Airplane (Figure 4.5), each of 256x256 pixels were arbitrarily selected, downloaded and used as cover images. However, the only reason for choosing these images was being well-known and commonly used in the areas of digital image processing, compression and steganography (See, for instance, (Chang et al., 2007; Fard et al., 2006; Lee and Chen, 2003; Li and Wang, 2007; Toutant et al., 2006; Wang et al., 2002b; Yu et al., 2005; Yu et al., 2007)).



Lena **Peppers** **Baboon**

Girl **Airplane**

Figure 4.5: Test Image (256x256 pixels) Used in our Experiments

In the first stage of experiments, the steganography method used to embed secret data and then get a stego image was assumed the independent variable (JSteg, F5, JMQT and our proposed method). Thus, in order to evaluate the efficiency of our proposed steganography method (effect), four dependent variables were considered: the steganographic capacity, the quality of stego images, the size of stego images, and the computational time. The first two variables are the most commonly used aspects of image steganographic systems while the third variable is sometimes considered. Hence, the computational time has been considered to investigate the impact of using larger blocks and quantisation tables (16x16) on the processing time. Therefore, the computational time has been measured for both DCT transformation and quantisation process only.

Accordingly, for each steganography method (JSteg, F5, JMQT and our proposed method) and for each cover image (Lena, Peppers, Baboon, Girl and Airplane), we will measure the value of each dependent variable. However, the computational time was not considered for the F5 algorithm since we are using freeware software and the F5 is similar to the JSteg method in this part of system (DCT and quantisation). Having only the values of the dependent variables of our steganography method mean nothing since we can not decide how good these results are. As a result, we will compare the values of these dependent variables of the different steganography methods tested with that of our proposed method in order to assess its effectiveness.

In the second stage of experiments, we investigated the impact of optimised quantisation tables on steganography aspects. Thus, we used only one independent variable which is the quantisation table. As a result, we embedded secret data using four steganography methods (JSteg, JMQT, F5 and our first method) once with normal (default) quantisation tables (as in the first stage) and once again using optimised quantisation tables. Hence, steganographic capacity, stego image quality and stego image size were assumed the dependent variables.

In the third stage, the steganography method used to embed secret data and then get a stego image was assumed the independent variable (JSteg, F5, JMQT, our first method and our hybrid method). Additionally, we consider only three experimental variables to evaluate the performance of our proposed (hybrid) method; the steganographic capacity, the quality of stego images and the size of stego images. Therefore, all these variables are measured for the five test images and five steganography methods (JSteg, F5, JMQT, our first method, and our hybrid method).

4.6 DCT and Computational Complexity

The two most important aspects of image based steganography are the steganographic capacity and the stego image quality. The traditional JPEG based steganography methods divide images into non-overlapping blocks of 8x8 pixels. However, we proposed a novel steganography method based on JPEG and 16x16 quantisation tables. Thus, we used the computational time measure just to investigate whether using 16x16 non-overlapping blocks and quantisation table increases the computational overhead and complexity. Since image based steganography is not a real time application, it is not a crucial issue if the embedding process takes a bit longer time to be accomplished. We proposed a real time steganography method within a communication protocol (SOAP) and it will be discussed in detail later on in the Chapter 7.

The discrete cosine transform (DCT) has emerged as the most popular transform for many image/video compression applications, especially the still image compression standard JPEG (Acharya and Tsai, 2005). Since the DCT is a separable function, the two dimensional DCT (2-D DCT) can be computed using the one-dimensional DCT horizontally and then vertically across the image block.

For DCT-based JPEG coding, the images are usually divided into 8 x 8 blocks and the 2-D DCT is applied on this 8 x 8 block. A brute-force implementation of the 2-D DCT of an 8 x 8 block requires approximately 64x64=4096 multiply-accumulate operations (Acharya and Tsai, 2005). When the images are divided into 16x16 blocks and the 2-D DCT is applied on this 16x16 block, a brute-force implementation requires approximately 256x256=65536 multiply-accumulate operations. Therefore, if we consider the block 2-D DCT implementation, the computational complexity of using 16x16 blocks is higher than that of using 8x8 blocks.

On the other hand, by dividing the images of 256x256 pixels into 8x8 blocks we get 1024 non-overlapping 8x8 blocks. Moreover, by dividing these images into 16x16 blocks we get 256 non-overlapping 16x16 blocks. Thus, for a 256x256-pixels image, the number of the 8x8 non-overlapping blocks that require the implantation of the 2-D DCT is four times as more as that of the 16x16 non-overlapping blocks. Therefore, if we consider the number of non-overlapping blocks within an image, the computational time of using 16x16 blocks could be less than that of using 8x8 blocks.

Additionally, the fast developments in computers technology, capability and competency may overcome the computation complexity which was existing in computers many decades before. For example, the PC used for our experiments has two processors each of 2.13GHz. These developments in computers technology may lead to minimise the computational overhead of the 2-D DCT implementation for 16x16 blocks to be similar to that of 8x8 blocks. In our experiments, we measured the computational time of both DCT transform and the quantisation process together. Therefore, the results of our experiments related to the computational time (Table 4.11) do not represent the computational time of the DCT implementation only since the quantisation process may affect this time as well.

4.7 Experimental Results

4.7.1 Using 16x16 quantisation table for steganography

Figure 4.6 shows the capacity (bits) of our steganography method and the other three methods considered (Table 4.8). JSteg and F5 methods do not embed a fixed number of secret bits within a given cover image. However, the capacity varies from one block to another, so these methods include a code to count the number of secret bits embedded within each cover image. JMQT method uses the 2-LSBs of each predefined middle frequency coefficient (set to be 1 in Table 4.3) as redundant bits for data hiding. Since it embeds secret data in 26 quantised DCT coefficients of each block, then each block of 8x8 pixels can hide 2bits*26=52 secret bits. Therefore, a cover image of 256x256 pixels can hold:
52 x (256x256) / (8x8) =53248 secret bits.

Method / Image	Lena	Peppers	Baboon	Girl	Airplane
JSteg	10421	10854	19006	8593	11752
F5	15637	16112	24124	13572	17360
JMQT	53248	53248	53248	53248	53248
Our 16x16 Method	61952	61952	61952	61952	61952

Table 4.8: The Capacity(bits) of Our First Steganography Method

On the other hand, our proposed method can embed 242 secret bits in each block of 16x16 pixels using the same technique of JMQT and 16x16 quantisation table (Table 4.4). Therefore, the capacity of a cover image of 256x256 pixels is 242 x (256x256) / (16x16) = 61952 secret bits.

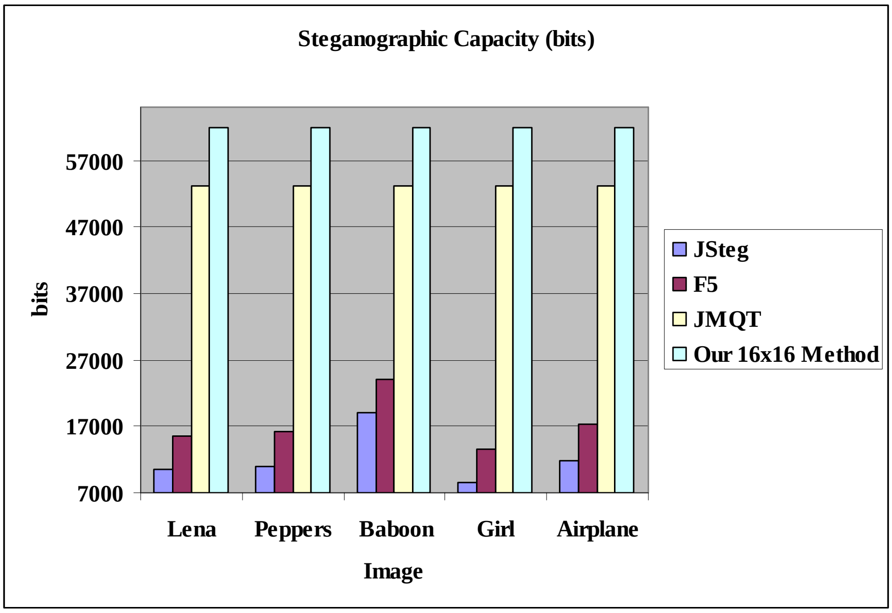


Figure 4.6: The Capacity(bits) of Our First Steganography Method

Figure 4.7 shows the PSNR values (quality) of obtained stego images (Table 4.9). The quality of our method's stego images is good and acceptable compared to the stego images' quality of other three steganography methods.

Method / Image	**Lena**	**Peppers**	**Baboon**	**Girl**	**Airplane**
JSteg	37.17	37.58	35.57	38.79	36.46
F5	37.45	38.33	36.08	38.81	37.06
JMQT	41.25	42.75	50.09	42.97	41.02
Our 16x16 Method	40.82	42.38	36.46	42.76	40.86

Table 4.9: The PSNR(db) of Stego Images-Our First Steganography Method

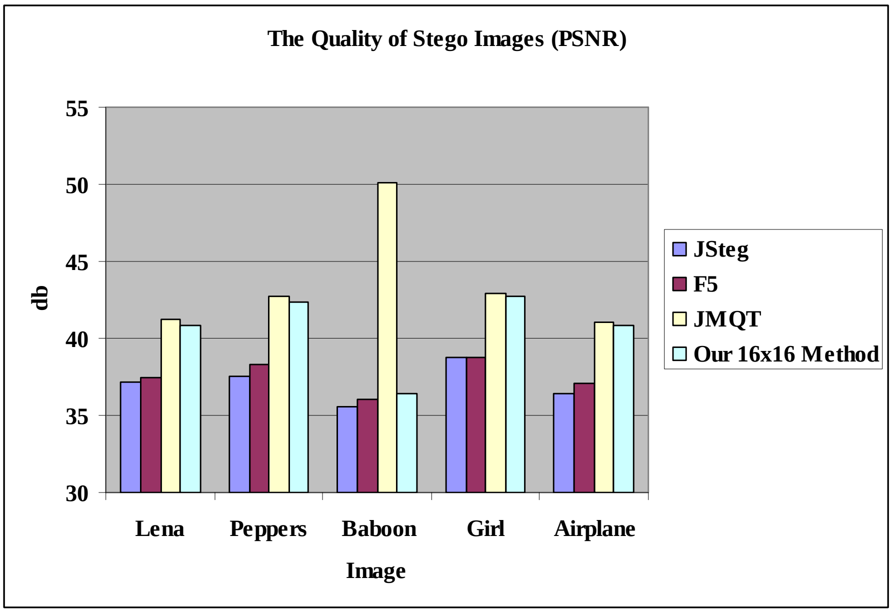


Figure 4.7: The PSNR(db) of Stego Images-Our First Steganography Method

In our method, we kept a considerable number of coefficients (set to be 1 in Table 4.4) to represent each block, hence stego images of our method have good quality. Even though we have slightly modified the middle frequency coefficients, it gives better results than discarding such coefficients. Moreover, in order to improve the quality of compressed images, smaller quantisation values can be used. Therefore, using more non-quantised coefficients provides better image quality. However, we embedded the secret bits by modifying these coefficients; therefore the stego images of our method have good quality but not as high as that of the JMQT method.

Figure 4.8 shows the sizes of the stego images (Kbytes) for the four steganography methods: JSteg, F5, JMQT, and our proposed method (Table 4.10). Comparing Table 4.3 with Table 4.4 we can notice the following aspects.

Firstly, in our method, 120 quantised high-frequency coefficients will mostly be discarded from each 16x16 block (the bottom right part of Table 4.4). However, in the JMQT method, 28 quantised high-frequency coefficients will mostly be discarded from each 8x8 block (the bottom right part of Table 4.3) and therefore 4x28=112 coefficients will mostly be discarded from each 16x16 block. Theoretically, and by considering this point of view, the size of the stego images of our method should be smaller than that of the JMQT method.

Method / Image	Lena	Peppers	Baboon	Girl	Airplane
JSteg	10.4	10.8	17.6	8.51	11.4
F5	15.4	15.9	24	13	16.7
JMQT	23.9	24.3	31.9	21.6	25.6
Our 16x16 Method	25.6	26.8	36.3	23	27.9

Table 4.10: The Sizes of Stego Images(KB)-Our First Steganography Method

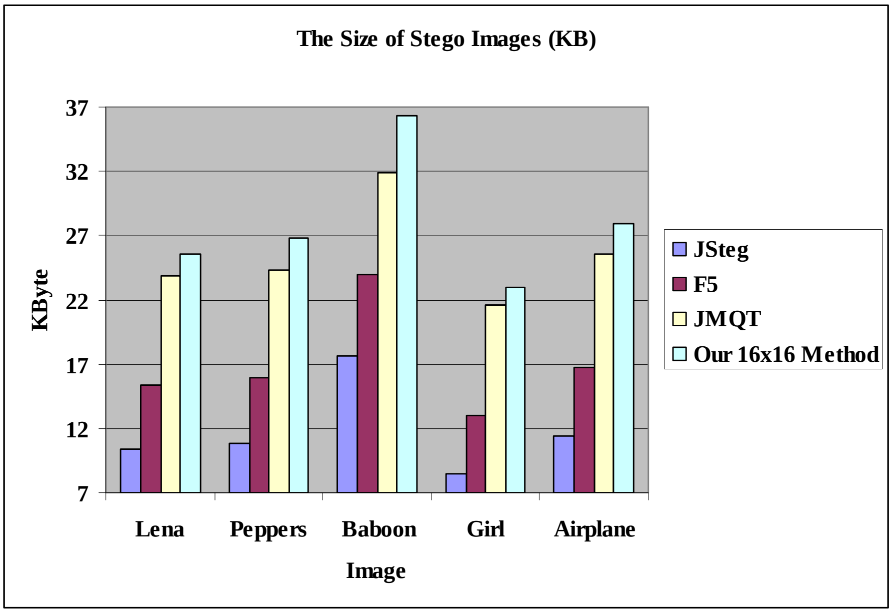


Figure 4.8: The Sizes of Stego Images(KB)-Our First Steganography Method

Secondly, our method uses more middle frequency coefficients for embedding than the JMQT method. This means that we keep more coefficients without quantisation (the value 1 in Table 4.4) which are usually large numbers and then maximise the average codeword length. Considering this point of view as well as the fact that the header of stego images for our method is larger than that of JMQT (larger quantisation table), the size of stego images of our method should be a bit larger than that of JMQT method.

Table 4.10 shows that the second point has more effect on the size of stego images than the first one, Therefore, the size of our method's stego images are a bit larger than that of JMQT method.

Method / Image	Lena	Peppers	Baboon	Girl	Airplane
JSteg	0.3531	0.3515	0.3528	0.3542	0.3546
JMQT	0.3533	0.3466	0.3447	0.3516	0.3504
Our 16x16 Method	0.1954	0.1957	0.1951	0.1955	0.1955

Table 4.11: The Computational Time(sec)-Our First Steganography Method

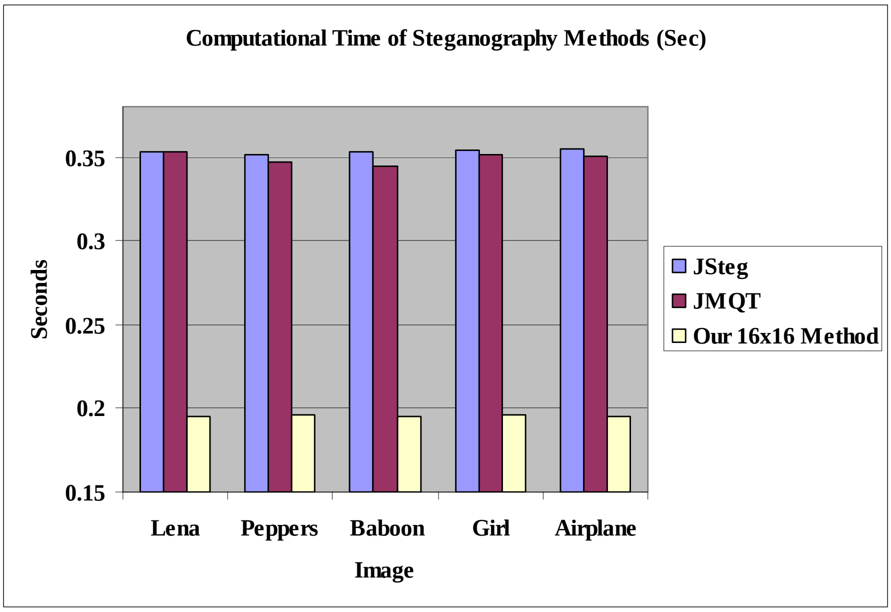


Figure 4.9: The Computational Time(sec)-Our First Steganography Method

Figure 4.9 shows the computational time of the DCT transformation and quantisation process for different test images and steganography methods (Table 4.11). The running time of our method (DCT transform and quantisation process) is almost half of that in the JSteg and JMQT methods. Thus, using 16x16 non-overlapping blocks and quantisation table for JPEG encoding requires less computational time than using 8x8 blocks and quantisation table. The reason behind this result may be related to the fact that fewer numbers of blocks have to be transformed and quantised.

The steganographic capacity and stego image imperceptibility represent the two main requirements of image steganography methods (Zhang and Wang, 2005). As a result, our steganography method provides larger steganographic capacity and acceptable stego image quality which makes it superior to the other steganography methods tested.

4.7.2 Using optimised quantisation tables for steganography

In order to evaluate the performance and efficiency of the optimised quantisation tables, we used them with JSteg, F5, JMQT, and our first method. Table 4.12 and Figure 4.10 show the capacity (bits) of these four methods once using the default tables (Table 4.2, Table 4.3, and Table 4.4 respectively) and once again using the optimised tables (Table 4.5, Table 4.6, and Table 4.7 respectively).

As mentioned before, the JSteg and F5 methods do not embed a fixed number of secret bits within a given cover image. However, the capacity varies from one block to another, so these methods include a code to count the number of secret bits embedded within each cover image. On the other hand, JMQT and our first method embed a fixed number of bits in each block and therefore in each cover image. As a result, the steganographic capacity of JSteg and F5 has been increased when the optimised quantisation table is used instead of the default table (Figure 4.10).

Method-QT / Image	Lena	Peppers	Baboon	Girl	Airplane
Jsteg-Def	10421	10854	19006	8593	11752
Jsteg-Opt	12363	12520	22207	9812	14304
F5-Def	15637	16112	24124	13572	17360
F5-Opt	21080	20971	24123	18541	22745
JMQT-Def	53248	53248	53248	53248	53248
JMQT-Opt	53248	53248	53248	53248	53248
Our 16x16 Method-Def	61952	61952	61952	61952	61952
Our 16x16 Method-Opt	61952	61952	61952	61952	61952
Our Hybrid Method	64651	64915	65062	64473	64576

Table 4.12: The Capacity(bits) of Our Hybrid Stego Method

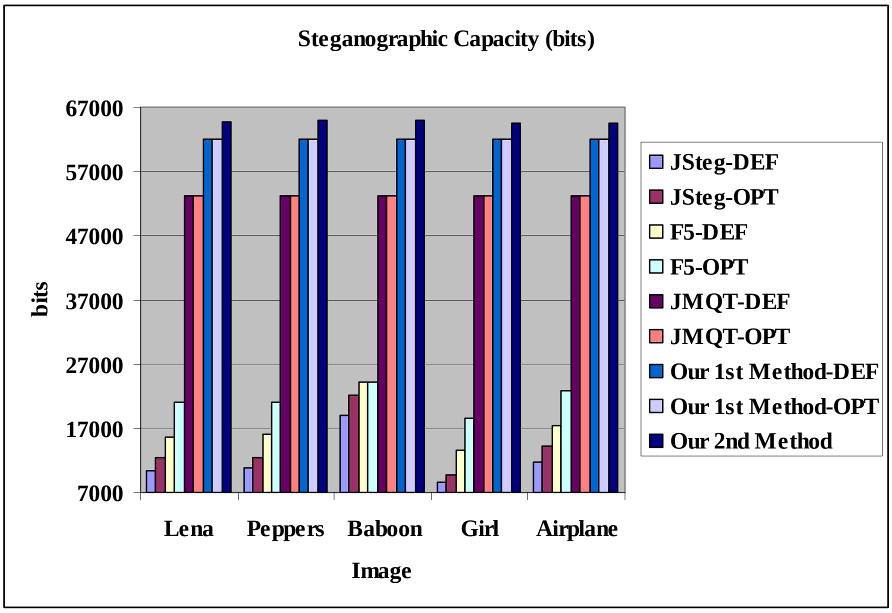


Figure 4.10: The Capacity(bits) of Our Hybrid Stego Method

Table 4.13 and Figure 4.11 show the quality of stego images of the four steganography methods before and after using optimised quantisation tables. Obviously, the quality of almost all stego images were considerably improved when the optimised quantisation tables are used instead of the default tables.

Method-QT / Image	Lena	Peppers	Baboon	Girl	Airplane
Jsteg-Def	37.17	37.58	35.57	38.79	36.46
Jsteg-Opt	38.66	38.79	37.68	39.48	38.19
F5-Def	37.45	38.33	36.08	38.81	37.06
F5-Opt	40.62	40.91	42.31	41.05	40.53
JMQT-Def	41.25	42.75	50.09	42.97	41.02
JMQT-Opt	44	44.66	46.73	44.47	44.05
Our 16x16 Method-Def	40.82	42.38	36.46	42.76	40.86
Our 16x16 Method-Opt	42.66	43.71	40.51	43.83	42.92
Our Hybrid Method	41.77	42.91	40.11	43.16	40.17

Table 4.13: The PSNR(db) of Stego Images-Our Hybrid Stego Method

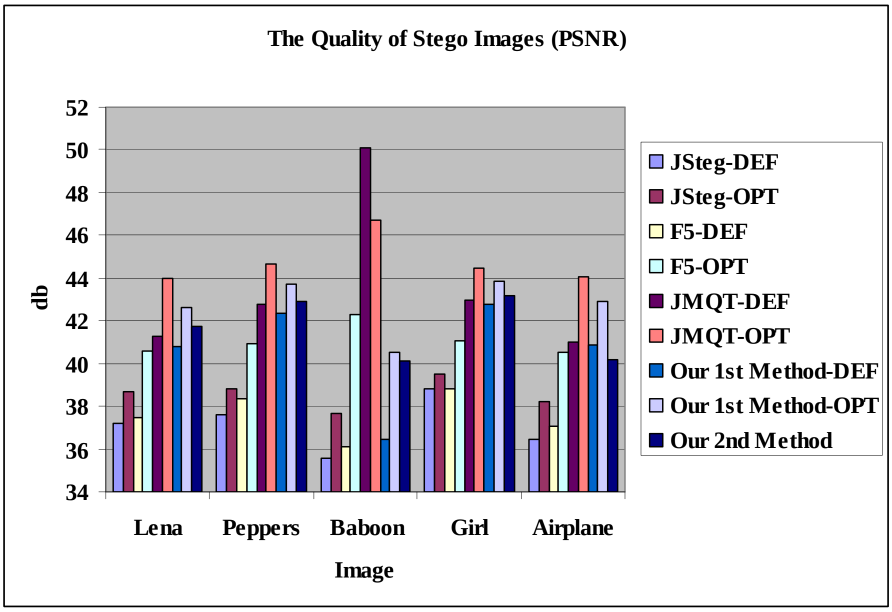


Figure 4.11: The PSNR(db) of Stego Images-Our Hybrid Stego Method

Table 4.14 and Figure 4.12 show the size of stego images of these four methods when default quantisation tables are used compared to that when optimised quantisation tables are used. Basically, the size of stego images are slightly increased when the optimised quantisation tables are used instead of the default tables.

Method-QT / Image	Lena	Peppers	Baboon	Girl	Airplane
Jsteg-Def	10.4	10.8	17.6	8.51	11.4
Jsteg-Opt	12.4	12.6	19.3	10.2	13.6
F5-Def	15.4	15.9	24	13	16.7
F5-Opt	18.7	18.9	27	16.5	19.9
JMQT-Def	23.9	24.3	31.9	21.6	25.6
JMQT-Opt	24.6	24.7	32.6	21.9	26.4
Our 16x16 Method-Def	25.6	26.8	36.3	23	27.9
Our 16x16 Method-Opt	26.4	27.5	37.8	23.5	28.8
Our Hybrid Method	26.4	27.4	37.8	23.5	28.8

Table 4.14: The Sizes of Stego Images(KB)-Our Hybrid Stego Method

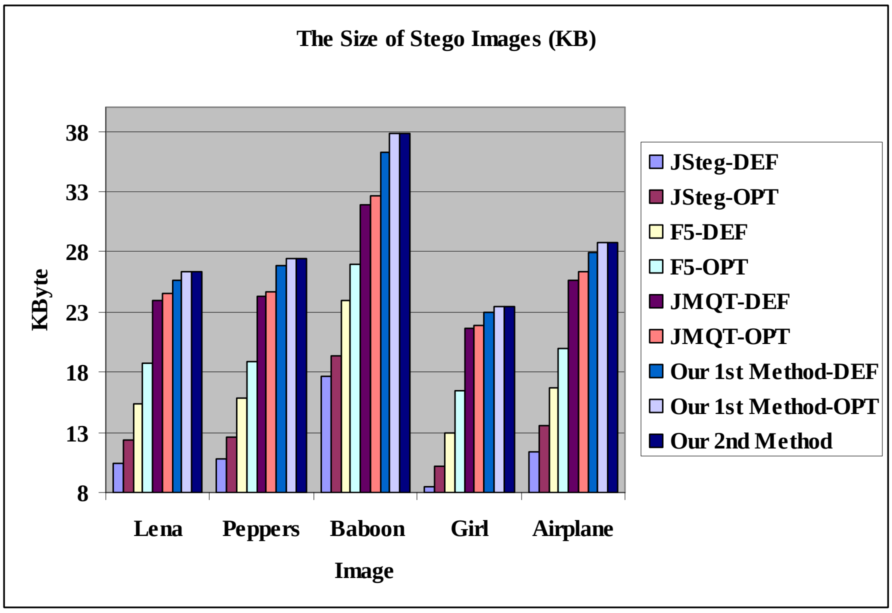


Figure 4.12: The Sizes of Stego Images(KB)-Our Hybrid Stego Method

We can, however, say that optimised quantisation tables have improved the performance of JPEG coding and steganography. This performance has been measured through three main aspects; steganographic capacity, stego image quality and stego image size. As shown in Figure 4.11, the stego image quality of our first method was improved by using an optimised 16x16 quantisation table instead of that arbitrary generated one. Then, we exploited this improvement in stego image quality to increase the steganographic capacity of our first proposed method. However, doing so will increase the steganographic capacity but will decrease the stego image quality due to the tradeoff between these two factors. Therefore, we proposed a hybrid steganography method based on our first method and the JSteg method.

Table 4.12 and Table 4.13 respectively show the capacity and stego image quality (PSNR) of our hybrid steganography method. Our hybrid steganography method which use an optimised quantisation table (Table 4.7) can embed much more data in the cover images than other methods tested; JSteg, F5, JMQT, and our first method. Using this hybrid method, the quality of the produced stego images is better than that of other methods (which use the default tables). Since the capacity and imperceptibility represent the two main requirements of any steganography technique, our hybrid method provides a larger steganographic capacity and improved stego images' quality.

4.8 Conclusion

The research aim of this thesis is to increase the steganographic capacity and improve the quality of JPEG stego images. Bearing in mind that, we can control the compressed image quality and compression ratio by manipulating the quantisation values, quantisation tables were used to improve the image steganographic capacity and stego image quality. Accordingly, in this chapter, we proposed two novel steganography methods.

Initially, our first steganography method is based upon JPEG compression and DCT transformation. Furthermore, non-overlapping blocks of 16x16 pixels and a modified 16x16 quantisation table are utilised in this method. The 2-LSBs of each quantised (middle frequency) coefficient are modified to embed two secret bits. As a result, our steganography method provided a larger steganographic capacity and acceptable stego image quality. Moreover, the computational time of our method is almost half of that of other methods which use 8x8 blocks and quantisation tables.

We then put forward the idea of using optimised quantisation tables instead of the default tables in order to improve the quality of stego images. Thus, the quality of stego images is enhanced significantly by using optimised quantisation tables. Additionally, we took the advantage of this quality improvement of stego images to embed more secret data in these images. Accordingly, we proposed a hybrid steganography method based on our first method, the JSteg method, and optimised quantisation tables. The steganographic capacity of this hybrid method is larger than that of other methods tested (including our first method). Moreover, the stego images of our hybrid method have better quality than that of other methods which use default quantisation tables.

Chapter 5: JPEG Steganography and Chrominance Components

5.1 Introduction

Some steganography methods use colour JPEG images as test images while others use grayscale images (Chang et al., 2009; Kharrazi et al., 2006; Lee and Chen, 2000; Stanescu et al., 2007). Therefore, both colour and grayscale images are used as cover images. To the best of our knowledge, the superiority of one kind of these images over the other kind is not examined when they are used with a given steganography method.

In this chapter, we intend to consider objective 3 of our research, which concerns the usage of chrominance components of images for steganography in addition to the luminance component in order to increase the steganographic capacity. Moreover, the pros and cons of using grayscale and colour versions of a given cover image will be examined in terms of steganography requirements. In meeting objective 3, we try to give an answer to the following question: *"which is better: the grayscale version or the colour version of a given cover image to be used with a given steganography method?"*. Additionally, the capability and impact of using the chrominance components for data hiding will be examined in order to increase the steganographic capacity.

The structure of this chapter is as follows: we start this chapter, in section 5.2, by summarizing the related work and introducing the work motivation. The experimental design used specifically to examine the performance of both grayscale and colour images when used for steganography is presented in section

5.3. In section 5.4, we consider our results, and lastly, in section 5.5, conclusions are discussed.

5.2 Image Chrominance and Steganography

The human eye is less sensitive to colour than to luminance. Therefore, data compression techniques should consider that the chrominance components of natural images can tolerate more noise than the luminance component without significantly affecting the perceived image quality. Usually, the size of chrominance data may be chosen to be one half to one quarter of the luminance data size without affecting human perception (Seitz, 2005). However, colour images provide more data for statistical analysis. Therefore, it is easier to detect steganography in colour images than in grayscale images (Cox et al., 2008).

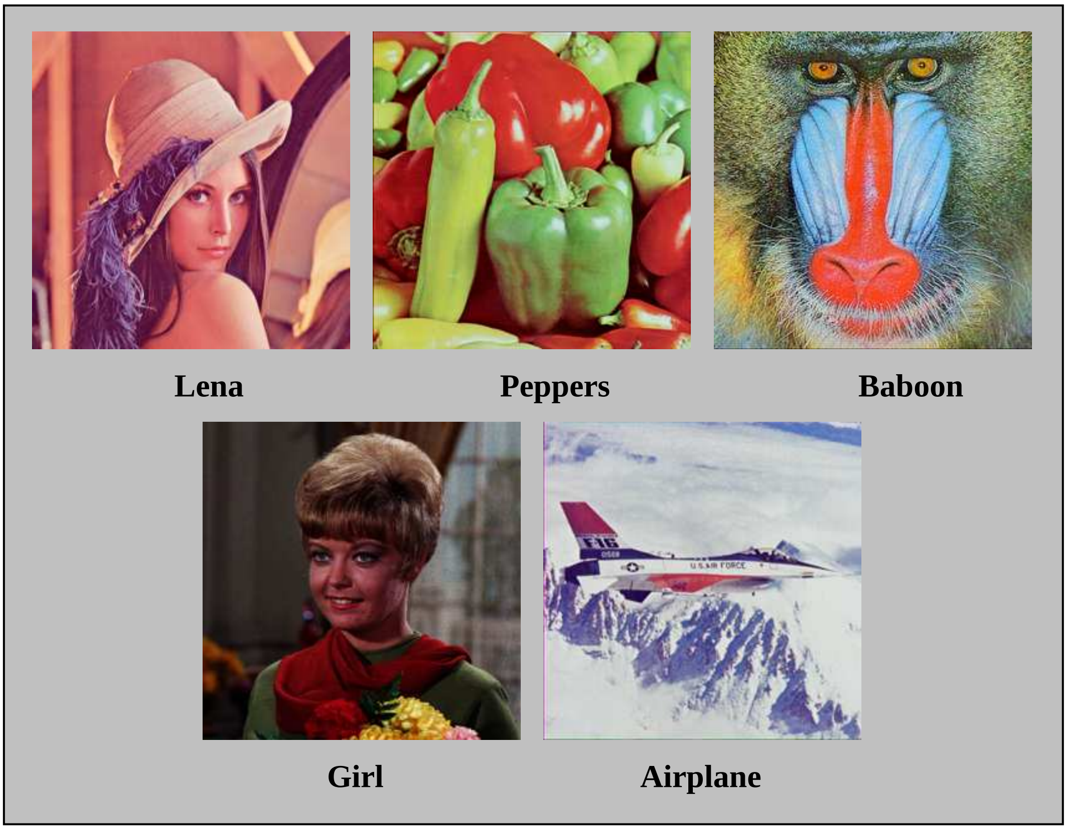
Some related work attempted to investigate the impact of chrominance components on the performance of data compression techniques. Therefore, the sensitivity of the human eye towards both of luminance and chrominance components is examined. Moreover, the capability of data hiding in the chrominance components is tested with RGB images. However, hiding data in Cb or/and Cr of DCT-based JPEG images is not, to the best of our knowledge, tested yet.

As stated before, not only grayscale images are used for steganography, but colour images as well. Since one of our study aims is to improve the steganographic capacity, we may hide extra data in the image chrominance components in addition to the luminance component. However, can the chrominance components be used as steganography cover medium or container? Moreover, using a given steganography method and having both grayscale and colour versions of a cover image, which image version is better than the other in terms of steganography? This chapter, therefore, examines the suitability of the chrominance components for data hiding and investigates the impact of using each single component for data hiding. Moreover, it evaluates the performance of both grayscale and colour versions of images when they are used with a given JPEG steganography method.

The steganographic capacity and imperceptibility, commonly used as steganography evaluation metrics, are used here in order to evaluate our suggested scenarios.

5.3 Experimental Design

In this section, we consider issues relating to the experimental approach, which is used in our study to evaluate the performance of both colour and grayscale images in terms of steganography. Therefore, experiments have been designed and conducted in order to investigate the capability and impact of using colour components for data hiding. The steganography methods used in these experiments were coded in Matlab R2007a (V 7.4.0) and run on a PC Pentium 4 with 2GB of RAM under the Windows XP operation system.



Lena

Peppers

Baboon

Girl

Airplane

Figure 5.1: Test Images (256x256 pixels) Used in our Experiments

Basically, five colour images and their grayscale versions, each of 256x256 pixels, are used as test images. These cover images are Lena, Peppers, Baboon, Girl, and Airplane (Figure 5.1). Both chrominance components (Cb and Cr) were downsampled by 2 in both horizontal and vertical directions. Therefore, we used a chrominance downsampling ratio equal to (4:2:0), so a macroblock consists of four 8x8 luminance blocks and two 8x8 chrominance blocks (one Cb and one Cr). Moreover, we have used two JPEG based steganography methods in our experiments, namely JSteg and JMQT.

Firstly, the impact of using both colour and grayscale versions of cover images for JPEG steganography was investigated. In this stage, the image version (colour or grayscale) of a test image was the independent variable. In order to evaluate the performance of these two image versions of each cover image on the main aspects of steganography (effect), three dependent variables were considered: the steganographic capacity, stego image quality, and the reconstructed stego image size.

Accordingly, for every steganography method, we have tested both colour versions and grayscale versions of cover images. For example, using “Lena” image as a cover image and “JSteg” as a steganography method, we used only the (Y) component of “Lena” colour image for data hiding without modifying the Cb and Cr components in the first phase. However, the (Y) component of “Lena” grayscale image, which is a sole component, is used for data hiding in the second phase.

Secondly, the capability and impact of using the chrominance components as a cover medium of steganography will be investigated. In this stage, we assume that the steganography scenario (Test Case) as the independent variable (cause) that will affect the dependent variables. Therefore, to evaluate the capability and impact of using chrominance components for data hiding on the main aspects of steganography (effect), three dependent variables were considered: the steganographic capacity, stego image quality, and the reconstructed stego image size.

Accordingly, we have embedded data in different image components according to various scenarios. Thus, for a given steganography method and specific cover image, hiding data in Cb, Cr, Y & Cb, Y & Cr, or Cb & Cr represent some scenarios to be investigated and tested. For example, using the “Lena” image and the “JSteg” method, we embedded secret data in the following components: (Cb), (Cr), both (Cb and Cr), and all (Y, Cb, and Cr).

5.4 Experimental Results

5.4.1 Colour and grayscale images as cover images

This set of experiments was conducted in order to investigate the difference between the grayscale and colour images when used as cover images. Firstly, we used the grayscale test images as cover images. Secondly, we used the colour test images as cover images even though we embedded the same secret data only in the (Y) component. We repeated these experiments using two different steganography methods, namely JSteg and JMQT. Moreover, the same amount of data is embedded in each pair of colour/grayscale images (Table 5.1).

Three measures are used as dependent variables, namely the quality, the capacity, and the size of stego images. The quality of stego images were measured using the PSNR metric (db). Hence, the capacity was considered as the size of data embedded within a cover image (KB). However, we are not going to consider the stego image size as an absolute value but we will consider the size of the stego image relatively to the size of its clean image (Stego_Size) (equation 5.1). This expression shows the increasing ratio of stego image size rather than the size of the stego image itself.

$$Stego\_Size = \frac{stego \;\; image \;\; size}{clean \;\; image \;\; size} \qquad (5.1)$$

The experimental results for both grayscale and colour test images used as steganography cover images are shown in Table 5.1. Since we used only the luminance component for data hiding in both grayscale and colour images, the capacity of each colour/grayscale pair of images is the same. Moreover, the quality of colour stego images (PSNR) is almost similar or slightly better than that of grayscale stego images (Figure 5.2). Furthermore, the Stego_Size of colour stego images is smaller, which means better, than that of grayscale stego images (Figure 5.3). Therefore, the following discussion will justify these results.

Method	Image	Gray/Colour	PSNR	Capacity	Stego_Size
JSteg	Lena	Grayscale	37.53	1.2721	1.0388
		Colour	37.55	1.2721	1.0327
	Peppers	Grayscale	38.09	1.325	1.036
		Colour	38.44	1.325	1.029
	Baboon	Grayscale	31.92	2.3201	1.045
		Colour	32.01	2.3201	1.039
	Girl	Grayscale	39.91	1.049	1.043
		Colour	40.55	1.049	1.037
	Airplane	Grayscale	36.41	1.4346	1.039
		Colour	36.43	1.4346	1.034
JMQT	Lena	Grayscale	39.88	6.5	2.38
		Colour	39.88	6.5	2.1634
	Peppers	Grayscale	39.37	6.5	2.324
		Colour	39.68	6.5	2.054
	Baboon	Grayscale	47.07	6.5	1.889
		Colour	47.12	6.5	1.779
	Girl	Grayscale	40.43	6.5	2.65
		Colour	41.05	6.5	2.396
	Airplane	Grayscale	39.06	6.5	2.321
		Colour	39.07	6.5	2.162

Table 5.1: Grayscale & Colour Covers Images: A Comparative Evaluation

The grayscale image has only one component (Y) used for data hiding. However, the colour image has one luminance component (Y) and two chrominance components (Cb and Cr). In order to evaluate the impact of using grayscale and colour images for steganography, only the (Y) component in colour images is used for data hiding. Thus, the other two components (Cb and Cr) are kept untouched. As a result, each pair of colour/grayscale images has the same capacity since the same amount of data is embedded in both images.

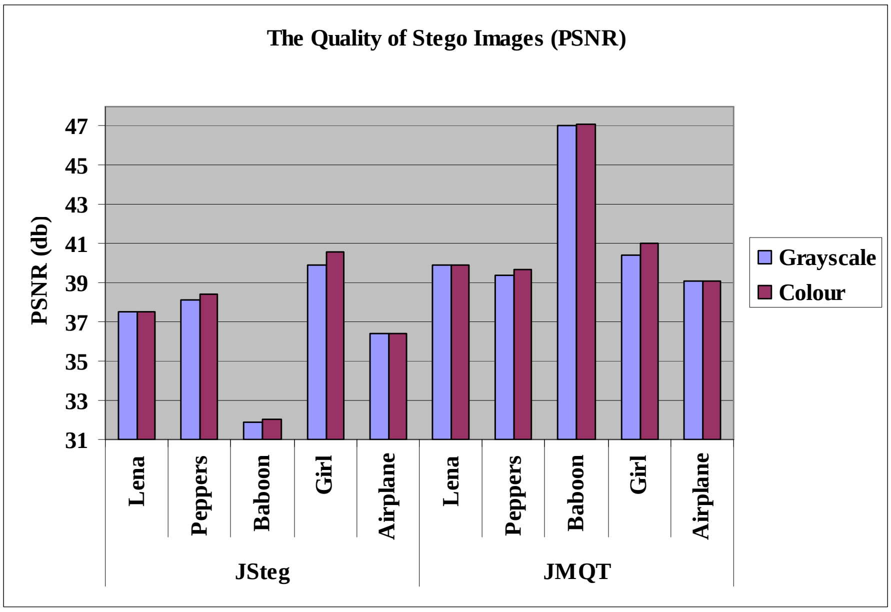


Figure 5.2: The PSNR(db) of Grayscale and Colour Stego Images

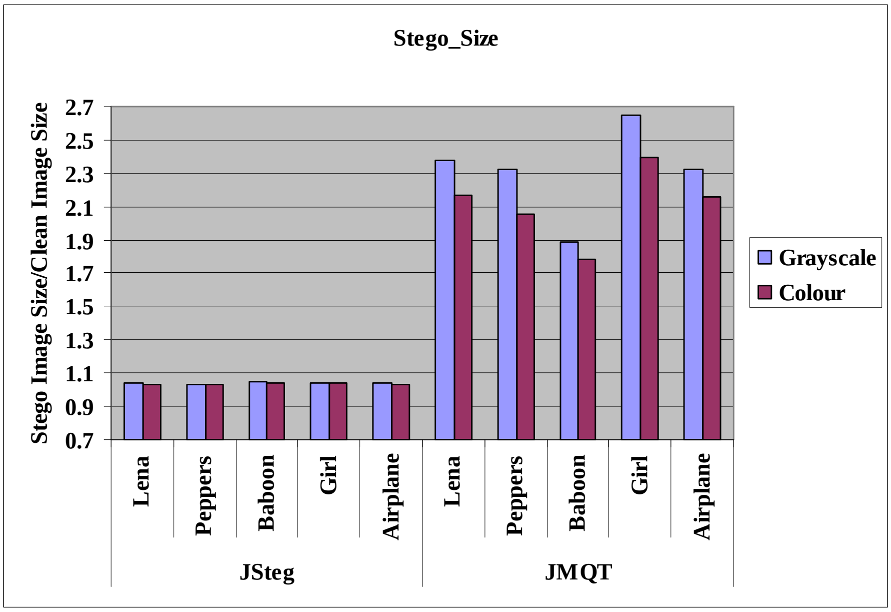


Figure 5.3: The Stego_Size of Grayscale and Colour Stego Images

Since the same amount of data is embedded in each pair of colour/grayscale images, we assume that the same amount of artefacts is added to both images. However, the MSE of colour images is divided by 3 (equation 2.3). Therefore, the PSNR of colour stego images is better than that of grayscale stego images.

It has been found that the size of all stego images was larger than that of clean images (Table 5.1). For a given grayscale image, assuming that the size of the clean image is “y” and the size of the stego image is “x”, then:

$$Stego\_Size_{grayscale} = \frac{x}{y} > 1 \tag{5.2}$$

Using the colour version of the same image, discussed above, we can add the size of both chrominance components “z” to both grayscale images “x” and “y”. Doing so, we find that the “Stego_Size” of colour stego images is smaller than that of grayscale stego images as follows:

Let $\mathbb{R}^{\oplus}$ denote the set of all positive real numbers. Then:

$$\forall x, y, z \in \mathbb{R}^{\oplus} \tag{5.3}$$

$$x > y \tag{5.4}$$

$$x.z > y.z \tag{5.5}$$

$$x.z + xy > y.z + xy \tag{5.6}$$

$$x.(y+z) > y.(x+z) \tag{5.7}$$

$$\frac{x}{y} > \frac{(x+z)}{(y+z)} \tag{5.8}$$

$$Stego\_Size_{greyscale} > Stego\_Size_{colour} \tag{5.9}$$

In conclusion, if we hide a similar amount of data in a pair of colour/ grayscale images we get better colour stego images than grayscale stego images in terms of quality and size. Therefore, the size increase of colour stego images was less (i.e. better) than that of grayscale stego images. In conclusion, using colour images is better than using grayscale images for data hiding.

5.4.2 Chrominance components and steganography

This set of experiments was conducted in order to investigate the capability and impact of using the chrominance components as a cover medium of steganography. For a given cover image, we used two different steganography methods, namely JSteg and JMQT. Moreover, we embedded secret data in one or more image components every time (Table 5.2, Table 5.3, and Table 5.4). For example, the Test Case (16) (Table 5.2) hides data in a colour test image using JSteg method for (Y) component and using JMQT method for both (Cb) and (Cr) components.

Firstly, we embedded secret data in the chrominance components only and then in both image components (luminance and chrominance). Then, we analysed and evaluated the impact of this on the main steganography aspects. Thus, the same three measures (dependent variables) defined and used before were utilised as evaluation metrics of the different steganography scenarios.

Test Case	**Hiding Method**			**Test Image**					**Avg**
	Y	**Cb**	**Cr**	**1**	**2**	**3**	**4**	**5**	
1	JMQT	N/A	N/A	39.8	39.3	47.0	40.4	39.0	41.1
2	JMQT	0	0	39.8	39.6	47.1	41.0	39.0	41.3
3	0	JMQT	0	42.3	41.0	47.0	46.6	41.0	43.6
4	0	0	JMQT	43.8	39.5	47.1	43.2	44.8	43.7
5	0	JMQT	JMQT	40.1	37.1	44.1	33.7	39.5	38.9
6	JMQT	JMQT	JMQT	37	35.3	42.4	38.4	36.3	37.9
7	JSteg	N/A	N/A	37.5	38.0	31.9	39.9	36.4	36.7
8	JSteg	0	0	37.5	38.4	32.0	40.5	36.4	37
9	0	JSteg	0	42.6	42.1	41.5	47.3	42.0	43.1
10	0	0	JSteg	43.9	39.7	42.0	43.9	45.4	43.0
11	0	JSteg	JSteg	40.2	37.7	38.7	42.3	40.4	39.9
12	JSteg	JSteg	JSteg	35.6	35.0	31.1	38.3	34.9	35.0
13	JMQT	JSteg	JSteg	37.0	35.6	38.1	38.6	36.6	37.2
14	JMQT	JMQT	JSteg	36.9	35.3	39.9	38.5	36.3	37.4
15	JMQT	JSteg	JMQT	37.0	35.5	39.6	38.4	36.6	37.4
16	JSteg	JMQT	JMQT	35.6	34.7	31.7	38.1	34.7	35

Table 5.2: The PSNR(db) of Stego Images-Different Test Cases

Test	Hiding Method			Test Image					Avg
Case	Y	Cb	Cr	1	2	3	4	5	
1	JMQT	N/A	N/A	6.5	6.5	6.5	6.5	6.5	6.5
2	JMQT	0	0	6.5	6.5	6.5	6.5	6.5	6.5
3	0	JMQT	0	1.625	1.625	1.625	1.625	1.625	1.62
4	0	0	JMQT	1.625	1.625	1.625	1.625	1.625	1.62
5	0	JMQT	JMQT	3.25	3.25	3.25	3.25	3.25	3.25
6	JMQT	JMQT	JMQT	9.75	9.75	9.75	9.75	9.75	9.75
7	JSteg	N/A	N/A	1.272	1.325	2.320	1.049	1.434	1.48
8	JSteg	0	0	1.272	1.325	2.320	1.049	1.434	1.48
9	0	JSteg	0	0.106	0.136	0.137	0.074	0.105	0.11
10	0	0	JSteg	0.106	0.180	0.130	0.096	0.072	0.11
11	0	JSteg	JSteg	0.213	0.317	0.267	0.171	0.178	0.22
12	JSteg	JSteg	JSteg	1.485	1.642	2.587	1.220	1.612	1.70
13	JMQT	JSteg	JSteg	6.713	6.817	6.767	6.671	6.678	6.72
14	JMQT	JMQT	JSteg	8.231	8.305	8.255	8.221	8.197	8.24
15	JMQT	JSteg	JMQT	8.231	8.261	8.262	8.199	8.230	8.23
16	JSteg	JMQT	JMQT	4.522	4.575	5.570	4.299	4.684	4.73

Table 5.3: The Capacity(KB) of Stego Images-Different Test Cases

Test	Hiding Method			Test Image					Avg
Case	Y	Cb	Cr	1	2	3	4	5	
1	JMQT	N/A	N/A	2.38	2.324	1.889	2.65	2.321	2.31
2	JMQT	0	0	2.163	2.054	1.779	2.396	2.162	2.11
3	0	JMQT	0	1.335	1.326	1.176	1.397	1.335	1.31
4	0	0	JMQT	1.330	1.348	1.177	1.416	1.306	1.31
5	0	JMQT	JMQT	1.654	1.664	1.346	1.799	1.629	1.61
6	JMQT	JMQT	JMQT	2.817	2.718	2.125	3.195	2.792	2.72
7	JSteg	N/A	N/A	1.038	1.036	1.045	1.043	1.039	1.04
8	JSteg	0	0	1.032	1.029	1.039	1.037	1.034	1.03
9	0	JSteg	0	1.003	1.003	1.002	1.004	1.003	1.00
10	0	0	JSteg	1.003	1.004	1.003	1.004	1.003	1.00
11	0	JSteg	JSteg	1.006	1.007	1.005	1.008	1.007	1.00
12	JSteg	JSteg	JSteg	1.039	1.036	1.044	1.045	1.041	1.04
13	JMQT	JSteg	JSteg	2.169	2.061	1.784	2.404	2.169	2.11
14	JMQT	JMQT	JSteg	2.501	2.385	1.957	2.798	2.5	2.42
15	JMQT	JSteg	JMQT	2.497	2.406	1.958	2.816	2.471	2.42
16	JSteg	JMQT	JMQT	1.686	1.693	1.385	1.836	1.664	1.65

Table 5.4: The Stego_Size of Stego Images-Different Test Cases

Statistical analysis of the experimental results allows us to test the objective 3 of our research. The research aim of this thesis is to increase the steganographic capacity and improve the quality of JPEG stego images. To achieve this, in objective 3, we investigated the impact of hiding secret data within different image components on the main aspects of steganography.

It is important to know the distribution of the data under analysis (normality of distribution) in order to perform the statistical analysis. Distribution of the data was analysed for each dependent variable measured. Moreover, we used Kolmogorov-Smirnov test to verify the normality of data distributions. However, the assumption of normality was not verified for these steganography aspects ($p<0.05$). Therefore, non-parametric statistics were used to analyse the data collected. Accordingly, a Kruskal-Wallis test revealed a statistically significant difference in these three measures (PSNR, capacity, and Stego_Size) across the sixteen different test cases (Test Case 1-16),

$$x^2_{PSNR}\,(15,\, n=80) = 51.626, \tag{5.10}$$

$$x^2_{capacity}\,(15,\, n=80) = 76.88, \tag{5.11}$$

$$x^2_{Stego\_Size}\,(15,\, n=80) = 75.244, \tag{5.12}$$

$$p_{PSNR} < 0.05, \tag{5.13}$$

$$p_{Capacity} < 0.05, \tag{5.14}$$

$$p_{Stego\_Size} < 0.05. \tag{5.15}$$

Therefore, the statistical analysis revealed a significant difference between the steganography methods (scenarios) used to hide secret data.

On the other hand, three schemes are used to evaluate the performance of chrominance components (Cb and Cr) when used for data hiding. In the first scheme, we compare the usage of the luminance (Y) for data hiding with the usage of chrominance (Cb or/and Cr) for data hiding (e.g. Test Case 2 with Test Cases 3, 4, and 5). Additionally, in the second scheme, we compare the usage of the luminance (Y) for data hiding with the usage of (Y), (Cb), and (Cr) for data hiding (e.g. Test Case 2 with Test Case 6). Finally, in the third scheme, we compare the usage of all image components (Y, Cb, and Cr) for data hiding using various steganography methods (e.g. Test Case 6 with Test Case 12). All of these will now be examined in more detail.

5.4.2.1 Hiding in (Y) vs. hiding in (Cb or/and Cr)

In Table 5.2, Test Cases (3, 4, and 5) and test cases (9, 10, and 11) hide data in the chrominance components (Cb, Cr, and both Cb and Cr respectively) while test cases (2) and (8) hide data in the luminance component (Y) only. It has been found that the chrominance components can be used for data hiding but we have to examine the impact of this hiding on the main aspects of steganography to verify their eligibility for data hiding.

The capacity of (Cb), (Cr), or both (Cb and Cr) is less than that of (Y) (Figure 5.4). This is because of the downsampling process applied to (Cb) and (Cr). Since the size of (Cb) or (Cr) is mostly equal to one quarter of the (Y) size, the small size of these two chrominance components restricts the hiding capacity.

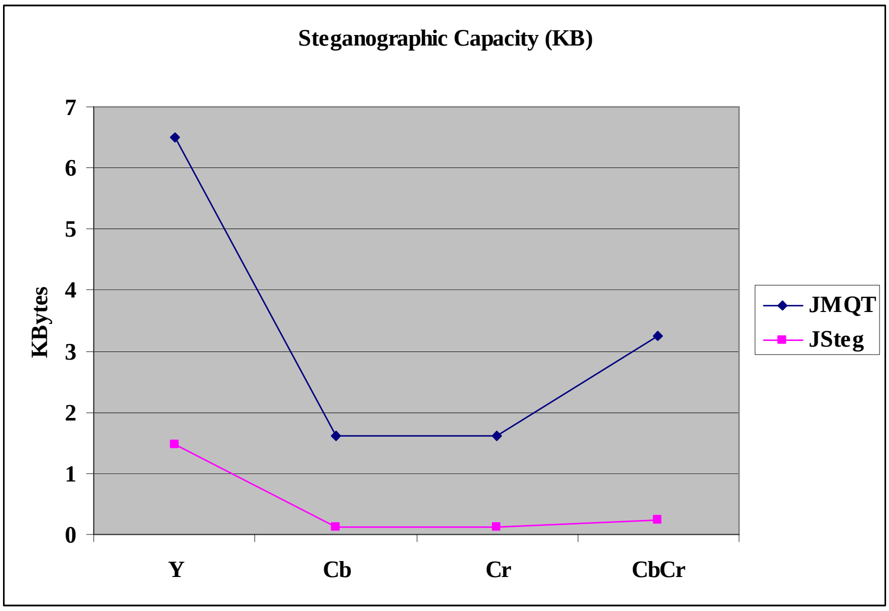


**Figure 5.4: The Average Steganographic Capacity(KB)-
Hiding in (Y) vs. Hiding in (Cb and/or Cr)**

Hiding data in (Cb) or (Cr) produces better stego image quality than hiding in the (Y) (Figure 5.5). Moreover, hiding data in both (Cb and Cr) using JSteg produces better stego image quality than hiding in the (Y). The size and the capacity of (Cb) and (Cr) are smaller than that of (Y). Thus, the number of DCT coefficients in (Cb) and (Cr) modified during data hiding is less than that of (Y). Therefore, hiding data in (Cb) and (Cr) adds fewer amounts of artefacts than hiding in (Y).

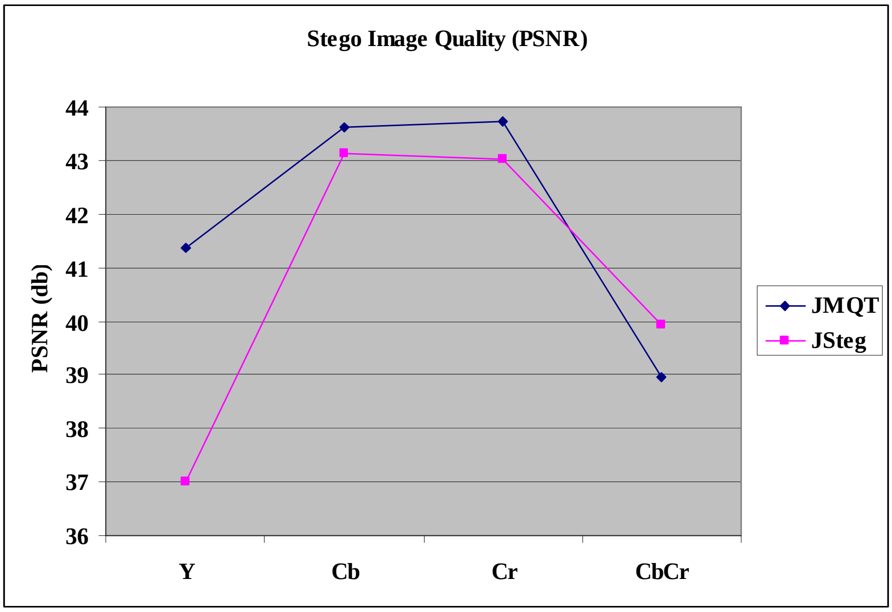


Figure 5.5: The Average PSNR(db) of Stego Images-Hiding in (Y) vs. Hiding in (Cb and/or Cr)

The size of stego images when (Y) is used for data hiding is larger than that when (Cb), (Cr), or both (Cb and Cr) is used for steganography (Figure 5.6). It has been mentioned that the size of stego images is larger than the size of clean images in general. Therefore, data hiding increases the size of stego images. Since the size of (Y) is larger than the size of both (Cb) and (Cr), hiding data in (Y) modifies more DCT coefficients (JMQT keeps more coefficients unquantised) than hiding in both (Cb) and (Cr) does. Therefore, data hiding in (Y) increases the stego image size more than hiding in (Cb), (Cr), or both (Cb and Cr). As future work, it could be a good idea to investigate the relationship between the size of the reconstructed stego image and the number of DCT coefficients modified in different image components.

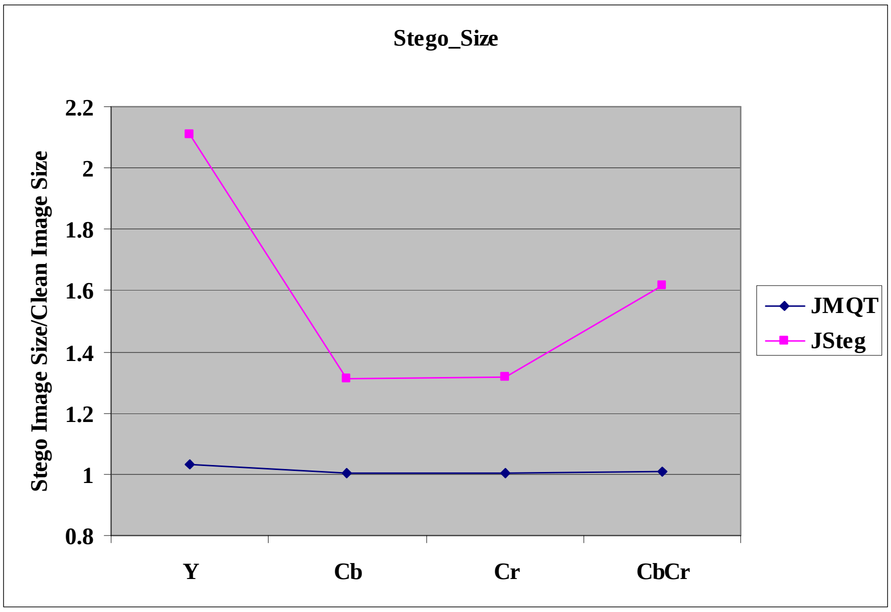


Figure 5.6: The Average Stego_Size of Stego Images-
Hiding in (Y) vs. Hiding in (Cb and/or Cr)

In conclusion, the chrominance components can be used for data hiding. Moreover, using these components for data hiding produces better stego image quality and more reasonable stego image size than using the luminance component. However, the chrominance components have smaller capacity than the luminance component. Therefore, these components are more efficient for data hiding when we want to hide a small amount of data such as a watermark or a fingerprint.

5.4.2.2 Hiding in (Y) vs. hiding in (Y, Cb, and Cr)

In Table 5.2, Test cases (2) and (8) hide data only in the luminance (Y) while Test Cases (6) and (12) hide data in both luminance (Y) and chrominance (Cb and Cr). As shown in Figure 5.7, capacity can be increased by hiding extra data in the chrominance components in addition to hiding data in the luminance component (Test Case 6) instead of hiding only in the luminance component (Test Case 2). Furthermore, hiding in all image components (Y, Cb, and Cr) increases the size of stego images more than hiding only in the luminance component (Figure 5.8).

This might be due to the larger number of modified DCT coefficients. Also, there is a considerable part (redundant data or middle frequency) of chrominance components kept to be used for data hiding in the JMQT method. However, this part of image is commonly discarded in the normal JPEG coding.

Since hiding in all components causes more artefacts than hiding in (Y) only, the quality of stego images is worse than the quality of stego images when (Y) only is used for embedding. Figure 5.9 shows that the PSNR of stego images of YCbCr embedding is less than that of stego images of only Y embedding.

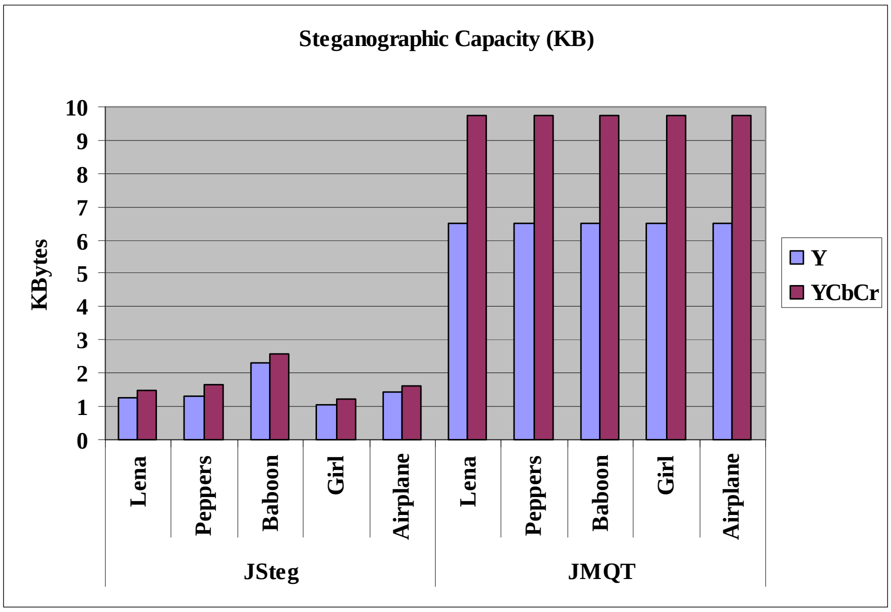


Figure 5.7: The Steganographic Capacity(KB)-
Hiding in (Y) vs. Hiding in (Y, Cb and Cr)

In conclusion, using the chrominance components in addition to the (Y) for data hiding increases the stego image size more than using only (Y). Moreover, it degrades the stego image more than using only the (Y). However, this increases the steganographic capacity significantly. Therefore, if (Y) is unable to hide a given amount of data, both luminance and chrominance can be used for data hiding while maintaining a good stego image quality (PSNR>35 db).

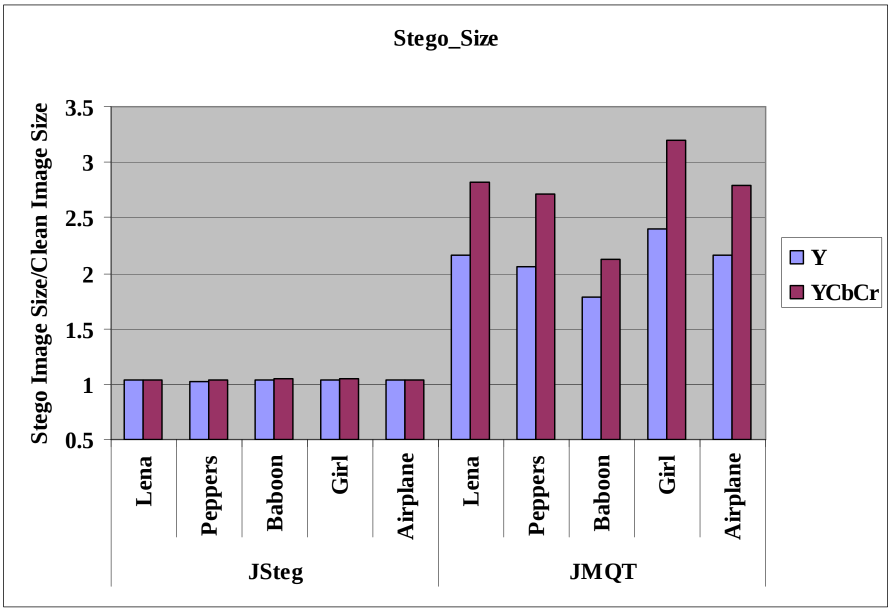


Figure 5.8: The Stego_Size of Stego Images-Hiding in (Y) vs. Hiding in (Y, Cb and Cr)

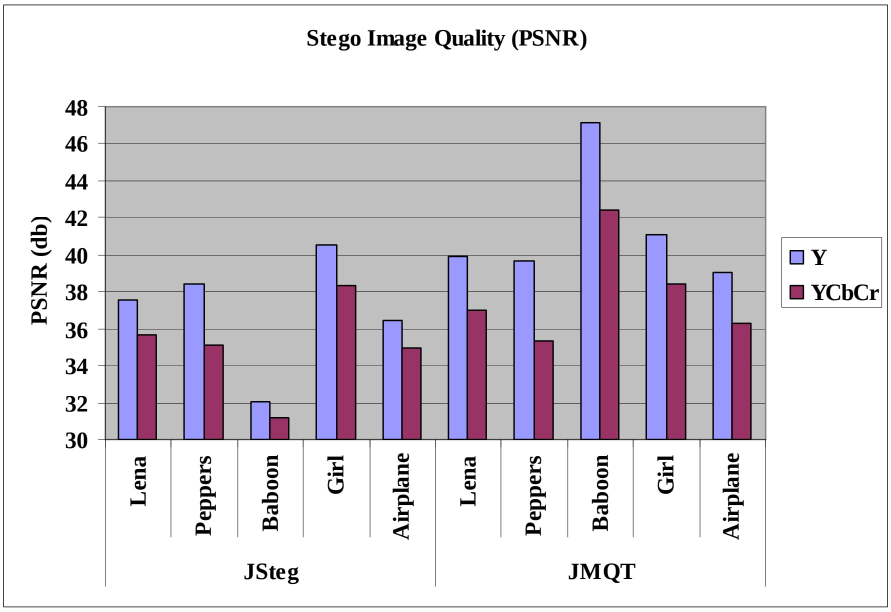


Figure 5.9: The PSNR(db) of Stego Images-Hiding in (Y) vs. Hiding in (Y, Cb and Cr)

5.4.2.3 Hiding in (Y, Cb, and Cr) using different steganography methods

In Table 5.2, Test Cases (6, 12, 13, 14, 15, and 16) hide data in both luminance (Y) and chrominance (Cb and Cr). However, the difference among these test cases is the steganography method used for each component. Tables 5.2-5.4 show that Test Case (6) uses the JMQT method for (Y, Cb, and Cr) while Test Case (16) uses the JSteg method for (Y) and the JMQT method for both (Cb and Cr).

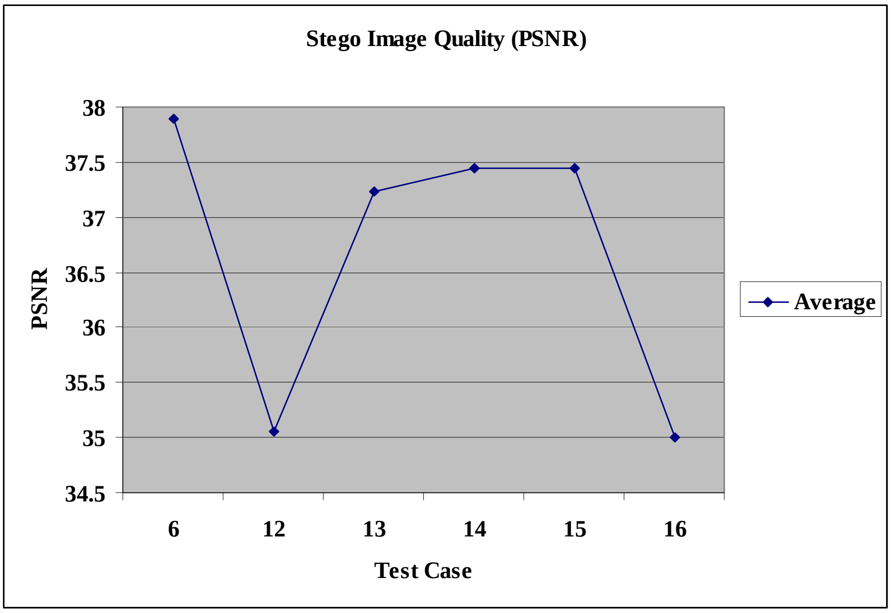


Figure 5.10: The Average PSNR(db) of Stego Images-Hiding in (Y, Cb and Cr) Using Different Stego Methods

Since the human eye is more sensitive to luminance than to chrominance and the chrominance part is downsampled in both directions, the (Y) component is likely to be the dominant part of the image. Therefore, the properties of (Y) as well as the manipulations applied on (Y) will mostly determine the characteristics of the reconstructed stego image. Figure 5.10 illustrates that all Test Cases that use the JMQT method for (Y) (6, 13, 14, and 15) have better stego image quality than Test Cases that use JSteg method for (Y) (12 and 16). This is because the JSteg method degraded the stego image more than JMQT method (discussed before).

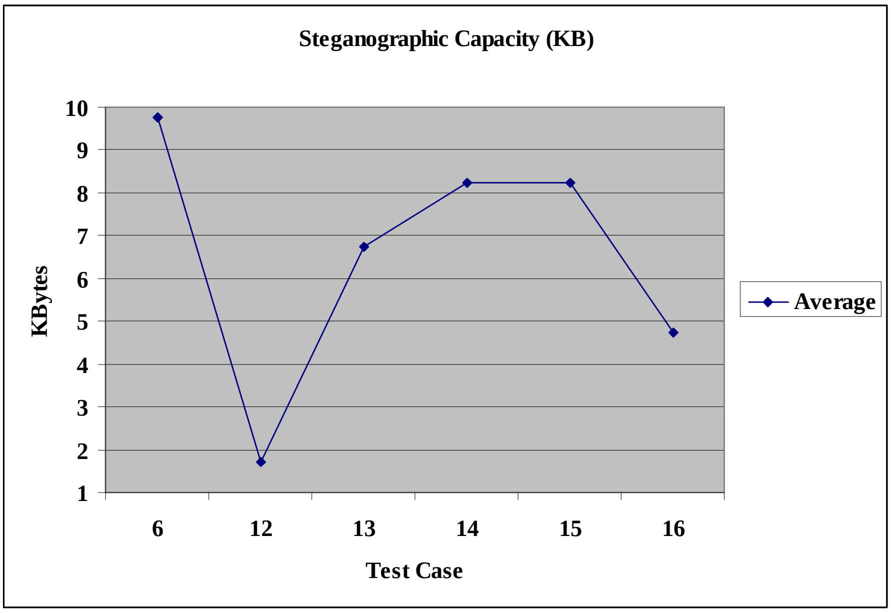


Figure 5.11: The Average Steganographic Capacity(KB)-Hiding in (Y, Cb and Cr) Using Different Stego Methods

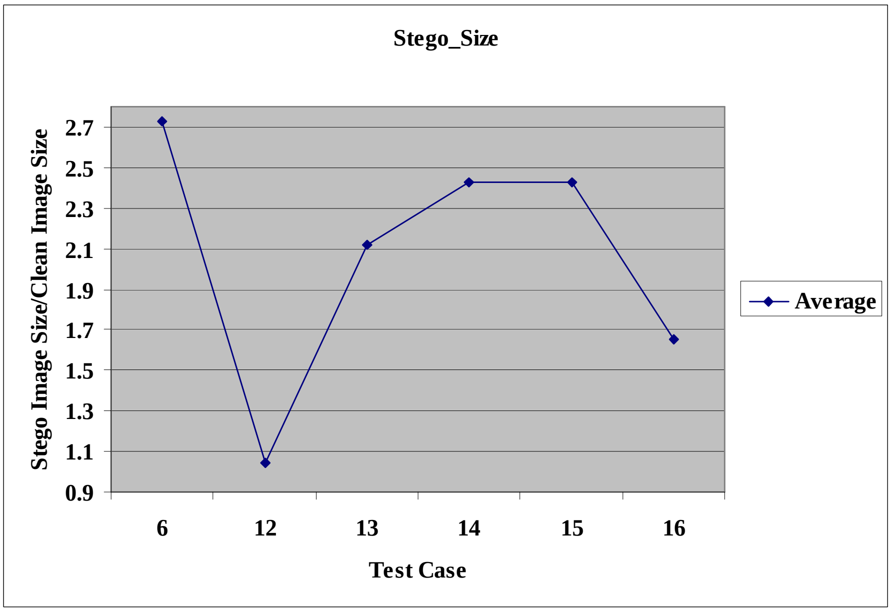


Figure 5.12: The Average Stego_Size of Stego Images-Hiding in (Y, Cb and Cr) Using Different Stego Methods

Additionally, the steganographic capacity of Test Cases which use the JMQT method for (Y) (6, 13, 14, and 15) is larger than that of Test Cases which use the JSteg method for (Y) (12 and 16) (Figure 5.11). However, Test Cases that use the

JMQT method for (Y) (6, 13, 14, and 15) produce larger-size stego images than Test Cases that use JSteg method for (Y) (12 and 16) (Figure 5.12).

This can be explained as follows. The JSteg method uses the low-frequency DCT coefficients for embedding and therefore it mostly discards other middle-to-high frequencies (smaller Stego_Size). However, JMQT uses the middle-frequency DCT coefficients for embedding so it discards less redundant data than the JSteg and has more data to be coded (larger Stego_Size). Comparing the Test Cases (1-6) from one side with the Test Cases (7-12) from the other side in Table 5.4, we can see that the JMQT method produces larger-size stego images than JSteg method. As a result, it is obvious that the steganography method used for luminance (Y) determines the features and properties of the stego image.

In conclusion, we can use various steganography methods to hide data in each component of colour image. Most likely, the steganography method used to hide secret data in the luminance component (Y) determines the properties of the stego image. Therefore, depending on our requirements, applications, or choice we can select one of the available scenarios in order to achieve our desirable outcomes.

5.5 Conclusion

In this chapter, we examined the suitability of the chrominance components for data hiding and investigated the impact of using each single component for data hiding. Additionally, we examined the performance of both grayscale and colour versions of five test images when used as steganography cover images. Therefore, for a given steganography method and cover image, we compared the usage of both grayscale and colour versions of test images for steganography. However, two steganography methods are used as test methods, JSteg and JMQT. Additionally, the steganographic capacity, stego image quality, and the stego image size are used as steganography evaluation metrics (dependent variables) in this chapter.

It has been concluded that using colour images is better than using grayscale images for data hiding. Moreover, Cb and Cr can be used for data hiding and they are more efficient for data hiding than using only (Y) when a small amount of data needs to be hidden. However, all image components can be used for data hiding in order to get a higher steganographic capacity while maintaining a good stego image quality (PSNR>35 db). Additionally, various steganography methods can be used for different image components. Nonetheless, the steganography method used for the luminance component determines the properties of the stego image.

Chapter 6: Stego Image Quality: Objective and Subjective Perspectives

6.1 Introduction

The usage of image compression, image coding, or image processing technologies has increased significantly during last few decades. Moreover, evaluating and measuring the quality of digital images still represent significant issue in many image processing applications, such as image coding algorithms and digital image steganography. Thus, image quality represents a key factor in most applications and therefore, assessing the perceived quality of digital images is a very important issue. However, evaluating the image quality of many image compression algorithms (i.e. lossy compression and image-based steganography) has many challenges such as the amount of degradation induced in the reconstructed image.

Basically, for security and imperceptibility reasons, it is very important for stego images not to show any detectable artefacts or distortions. Additionally, the original image (reference) and the stego image should look alike exactly under all possible comparison circumstances. Generally, there are two primary ways to measure image quality: objective quality methods and subjective quality methods (human-based). Objective and subjective image quality evaluation methods have been investigated for some applications (i.e. image compression) and for many types of artefacts (i.e. image blurring) (Grgic et al., 2004; Ogihara et al., 1996; Stoica et al., 2003). However, this still need further investigation for digital image steganography.

Indeed, JPEG steganography adds another type of distortion to stego images in addition to that added by image compression. Therefore, do objective methods (namely PSNR and MSE) represent reliable predictors of perceived stego image quality? In this chapter, we therefore intend to consider the objective 4 of our research, which concerns the quality evaluation of many JPEG stego images using a subjective image quality evaluation method. To this end, using JPEG stego images as test images, we will analyse and comparatively evaluate the relationship between this subjective method and the PSNR metric.

The structure of this chapter is as follows: in section 6.2, we describe the limitations of objective evaluation metrics. Information concerning the experimental design used to evaluate the quality of stego images is presented in section 6.3. In section 6.4, we consider our results, and in section 6.5, a brief discussion regarding these results is provided. Lastly, in section 6.6, conclusions are discussed.

6.2 Objective Evaluation Metrics

Generally, objective methods of image quality evaluation are faster and more cost-effective than subjective ones. Therefore, they are preferred over subjective quality evaluation methods. Mostly, digital image steganography uses the PSNR and the MSE metrics in order to evaluate the quality of stego images or the imperceptibility of hidden messages.

However, there is a poor correlation between objective quality estimation and the actual subjective evaluation in general (Nyman et al., 2006). Moreover, the PSNR and MSE are poor indicators of subjective image quality and they are extensively criticized for their poor correlation with actual measurement of perceived quality (Wang et al., 2002a; Wang et al., 2002b). Therefore, one of the main drawbacks of PSNR and MSE is the limited relationship with the quality perceived by human observers. For example, adding noise to an image can improve the subjective image quality in some cases, while it reduces the PSNR (Wu and Rao, 2006).

Furthermore, these two pixel-based metrics (PSNR and MSE) do not take into account the different effects of distortions on different image regions such as smooth areas and textured regions. Therefore, images with the same PSNR can actually have different perceived qualities. Thus, these pixel-based metrics provide inaccurate measures of quality for images that have different types of distortion (Wu and Rao, 2006).

Basically, the ability of existing objective image quality metrics to predict human judgment is still very limited. Additionally, the results obtained using these objective quality methods are not reliable due to a lack of standardization (Simone et al., 2009). Therefore, these criteria are not strongly related to the perceived image quality. Furthermore, they are widely criticized and consequently many methods of objective quality evaluation have been developed incorporating perceptual quality measures employing Human Visual System (HVS) characteristics (Baroncini, 2006; Nyman et al., 2006; Pinson and Wolf, 2003; Simone et al., 2009; Stoica et al., 2003; Wu and Rao, 2006).

PSNR and MSE are the traditional objective image quality measures. However, they are only applicable to luminance information. Therefore, it is not appropriate to use the computation of these measures for colour images (Wang et al., 2003). Moreover, they are used in the fields of image compression and image communication. However, steganography is a different field and stego images must not be suspected of concealing information. Therefore, the quality evaluation of stego images is more serious than the quality evaluation of compressed images (Kong et al., 2003).

6.3 Experimental Design

In this section, we consider issues relating to the experimental methodology used to study the reliability of PSNR and MSE metrics as quality evaluation measures of stego images. Thus, experiments were conducted in order to investigate the relationship between the PSNR and subjective evaluation.

Essentially, we are going to use some of the stego images obtained in the previous chapters (Chapters 4 and 5) as test images to be tested and evaluated subjectively in this chapter. As the quality of these stego images was objectively measured (PSNR) in the previous chapters, the performance of the steganography methods used to get these stego images was evaluated using the PSNR. Thus, we are not going to investigate only the relationship between the subjective quality of stego images and the PSNR. However, we also aim to examine the impact of this relationship on the performance evaluation of different steganography methods. For example, using PSNR as stego image quality measure may show that a steganography method (A) is better than (B) in terms of stego image quality. However, using the subjective evaluation as stego image quality measure may or may not lead to the same conclusion.

An adapted double stimulus continuous quality scale (DSCQS) method is used in our experiments as a subjective evaluation method of stego image quality. Thus, a Java application was developed and a 1280x1024 Dell monitor (E196FP, 19”) was used to display all images.

Five colour images and their grayscale versions, each of 256x256 pixels, are used as test images: Lena, Peppers, Baboon, Girl, and Airplane (Figure 4.5 and Figure 5.1). As a result, we will use some of the stego images obtained (Chapters 4 and 5) as input materials to be tested and evaluated. In this chapter, we are going to evaluate some steganography methods (including our proposed methods), in terms of the quality of their stego images, using subjective evaluation methods instead of the PSNR metric.

The steganography method or scenario used to embed secret data and then get a stego image was assumed the independent variable. Thus, in order to examine and evaluate the impact of these different steganography methods and scenarios on the stego images quality, one dependent variable is considered: the subjective quality (MOS) of these stego images. In Chapters 4 and 5, we have already measured the quality of these stego images using the PSNR.

In this chapter, we will investigate the relationship between the PSNR and subjective quality of stego images, and therefore evaluate the reliability of PSNR as a quality measure of stego images. Since the quality evaluation of stego image implicitly evaluates the performance of steganography methods used to get these stego images, the evaluation method used to measure the quality of stego images was assumed the independent variable.

6.3.1 Adapted DSQCS Method Proposed

The subjective evaluation method of image quality developed by Simone et al. (Simone et al., 2009) which was based on the (DSCQS) method, has inspired us to suggest another adapted method for subjective image quality evaluation. In our method, both images are shown simultaneously in each presentation and there is no time constraint for the assessment period. Moreover, subjects are shown pairs of images (internally random) in a randomised order. However, the voting scales/sliders are placed exactly under the images to be assessed (in the same presentation) (Figure 6.1). Therefore, the subject has to rate the quality of both images (at least one of these images is a reference image) in each pair by choosing the slider corresponding to each image.

It has been stated in (ITU-R-BT.500-11, 2002) that : *"Prior to a session, the observers should be screened for (corrected-to-) normal visual acuity on the Snellen or Landolt chart, and for normal colour vision using specially selected charts (Ishihara, for instance)"*. Therefore, the screening of subjects was performed according to the guidelines described in ITU-R BT.500-11, the Snellen Eye Chart was used to test vision acuity and the Ishihara test was used to check the colour blindness of subjects. Accordingly, 32 non-expert subjects were screened for visual acuity and colour blindness. However, two of these subjects have been discarded as outliers since they do not have correct-to-normal visual acuity. Therefore, the statistical analysis is based on the scores from 30 subjects (26 males and 4 females), who assessed the quality of each reference and test image using a continuous quality scale.

The age of participants was between 21 and 45 years (Figure 6.2). Additionally, the number of hours which participants usually spend on watching digital media contents (broadcasting programs, movies, and game playing) is shown in Figure 6.3. The average time of each experiment session was 16.44 minutes per participant while the maximum and minimum times were 27.44 and 6.47 minutes respectively.

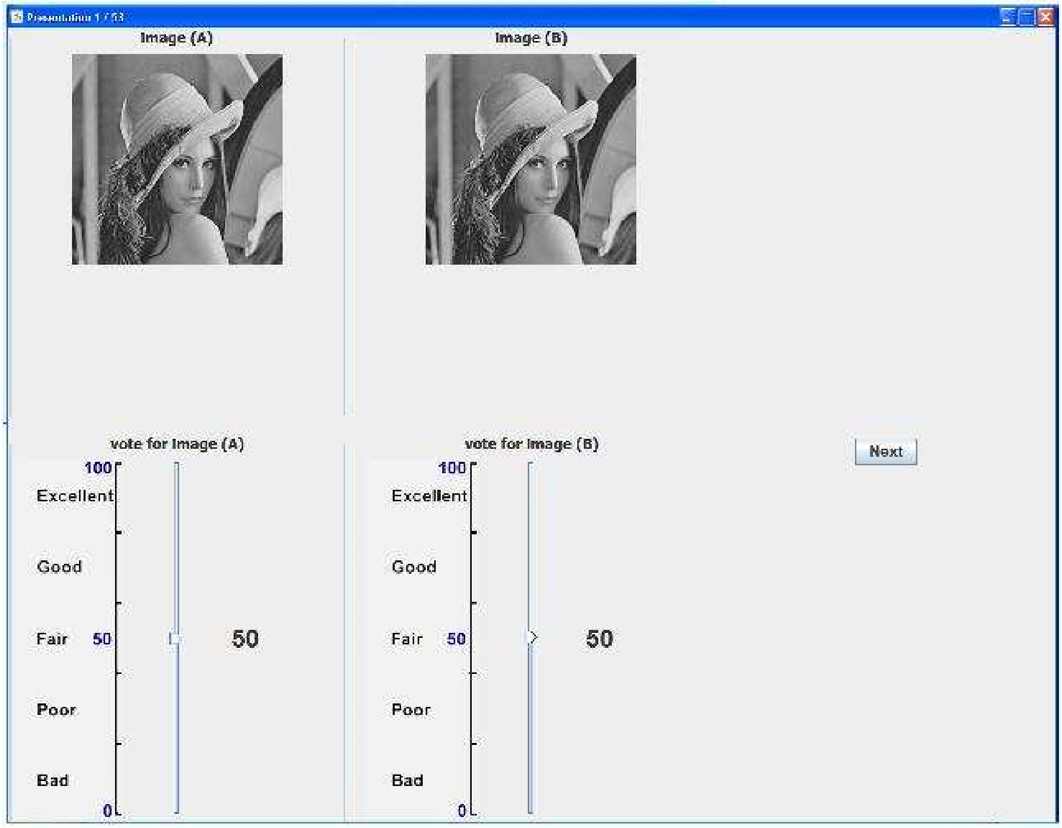


Figure 6.1: The Graphical User Interface of Our Adapted DSCQS Method

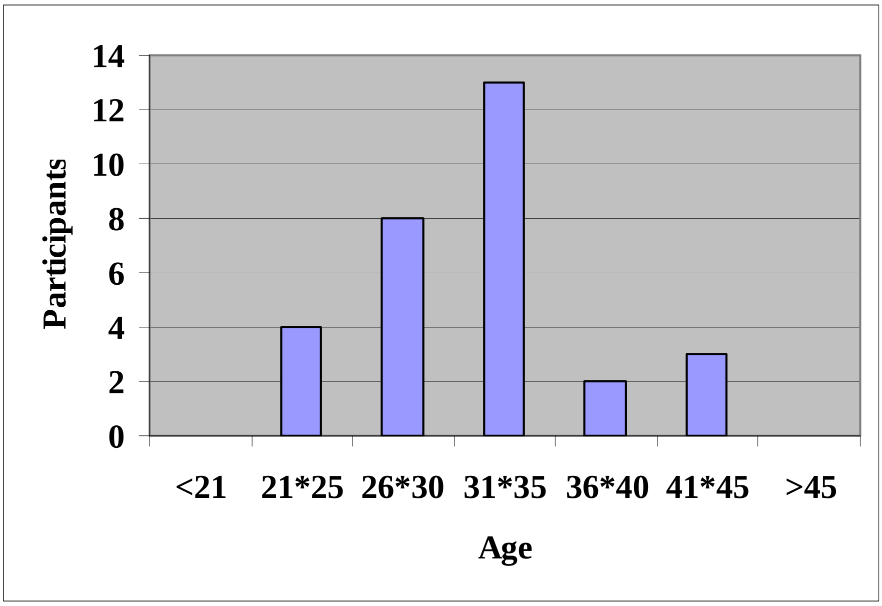


Figure 6.2: The Age of the Participants in our Subjective Experiment

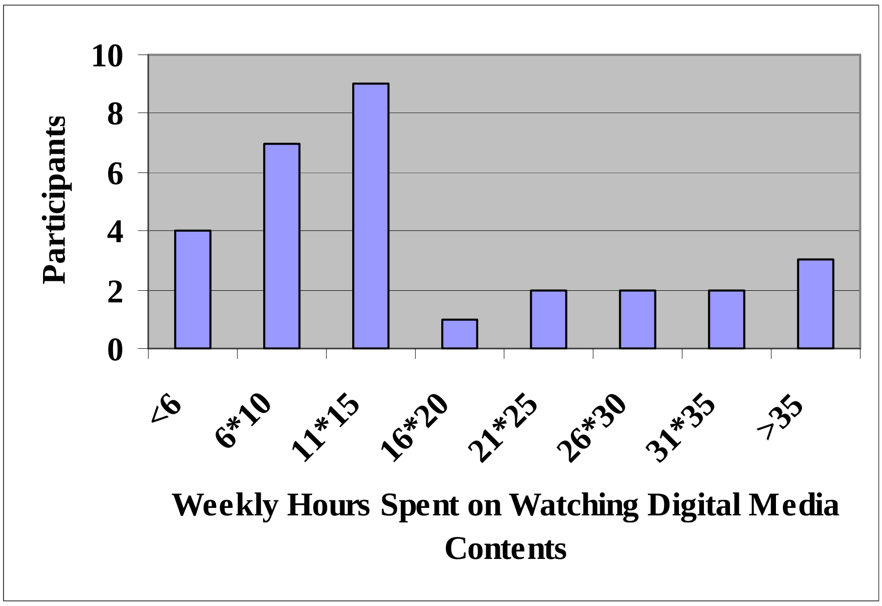


Figure 6.3: The Time which Participants Spent on Watching Digital Media

Subjects were introduced to the method of assessment and the continuous grading scale. Accordingly, at the beginning of each test session, one example presentation is introduced to familiarize the observer with the grading scale, evaluation process, and the graphical user interface (GUI) (Figure 6.4). The example image used (*"Cameraman.jpg"*) is different from those used in the actual test (Figures 4.5 and 5.1).

Additionally, three dummy presentations are introduced to stabilize the observer's opinion (Figure 6.4). However, the images used in these presentations are from those used in the actual test but data gathered during these three presentations are not included in the experiment results.

Every subject was asked to evaluate ten pairs (5 grayscale and 5 colour pairs) of reference-reference images. This procedure evaluates the reliability of subjects qualitatively (ITU-R-BT.500-11, 2002). In each test session and following the example and dummy presentations, a series of image pairs (actual test images) is presented to an assessor in a random order (Figure 6.4). Furthermore, these image pairs are randomly positioned on the screen either left or right in each presentation. Figure 6.1 shows the developed graphical user interface (GUI) that implements our subjective image quality evaluation method.

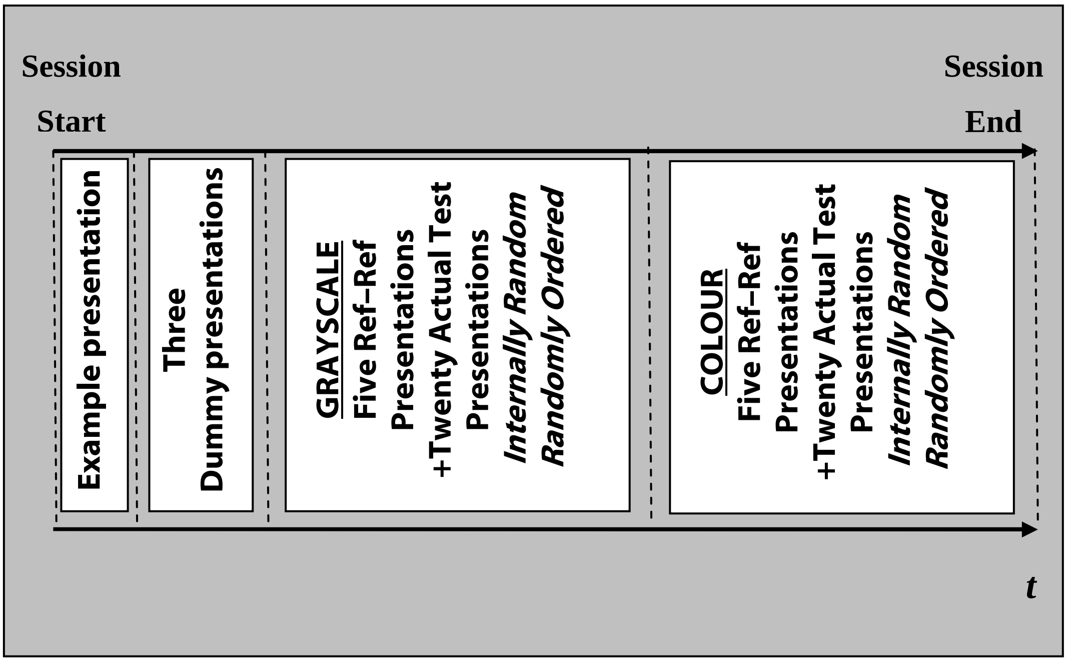

Session Start
Session End
Example presentation
Three Dummy presentations
GRAYSCALE
Five Ref–Ref Presentations
+Twenty Actual Test Presentations
Internally Random
Randomly Ordered
COLOUR
Five Ref–Ref Presentations
+Twenty Actual Test Presentations
Internally Random
Randomly Ordered
t

Figure 6.4: The Timescale of a Test Session in our Subjective Experiment

6.3.2 Test Images and Steganography Methods

In this chapter we tested the reliability and efficiency of PSNR and MSE as stego image quality measures. Moreover, five colour images and their grayscale versions are used as test images in our experiment. However, we divided the study into three parts. In the first part, we examined the relationship between the PSNR and subjective quality of stego images using four different steganography methods and five grayscale test images. In the second part, we used the five colour versions of test images and two steganography methods to hide data in different image components. As a result, we examined the relationship between the PSNR and subjective quality of these colour stego images. In the third part, we used the data collected from the first and second parts. Moreover, we compared the quality of both grayscale and colour stego versions of a given test image when a given steganography method is used to embed the same data in both image versions. Therefore, the relationship between PSNR and subjective quality was examined.

Firstly, we measured the quality of stego images from four different steganography methods and five grayscale test images using both subjective and objective (PSNR) measures. The four steganography methods tested are:

1- JSteg method.

2- JMQT method.

3- Our first proposed method (refer to 4.3.1).

4- Our hybrid proposed method (refer to 4.3.2).

Secondly, we measured the quality of stego images from two different steganography methods (JSteg and JMQT) and five colour test images using both the PSNR and the subjective evaluation method. We embedded the secret data once in the luminance component only and once again in both luminance and chrominance components of colour image as follows:

5- JSteg method in the luminance component (Y) only.

6- JSteg method in both luminance and chrominance components (YCbCr).

7- JMQT method in the luminance component (Y) only.

8- JMQT method in both luminance and chrominance components (YCbCr).

Thirdly, we are going to compare the method 1 (JSteg with grayscale images) with method 5 (JSteg with colour images). Thus, the same amount of data is embedded in both cases since method 5 only hides information in the Y component. Similarly, method 2 and method 7 will be compared in terms of stego image quality.

Each subject is shown one example presentation, 3 dummy presentations, 10 reference-reference presentations (5 grayscale and 5 colour presentations), and 40 actual test presentations. These 40 actual test presentations are: 20 grayscale presentations (4 methods (1-4) * 5 grayscale images) and 20 colour presentations (4 methods (5-8) * 5 colour images)). However, the example and dummy presentations are sequentially presented at the beginning of each session. Afterwards, the 5 grayscale reference-reference and 20 grayscale test presentations are randomly presented. Finally, the 5 colour reference-reference and 20 colour test presentations are also randomly presented.

The selection of test images to be used for evaluation is a quite important and difficult process. Therefore, five test images were selected and used as test images in our experiment: Lena, Peppers, Baboon, Girl, and Airplane (Figure 5.1), each of 256*256 pixels. As mentioned before, these images were arbitrarily selected and used as cover images since they are well-known and commonly used in the areas of digital image processing, compression and steganography. However, these test images have different spatial and frequency characteristics. Thus, the overall activity level in an image can be represented by the spatial frequency measure (SFM), defined as follows (Grgic et al., 2004):

$$SFM = \sqrt{R^2 + C^2} \tag{6.1}$$

$$R = \sqrt{\frac{1}{MN} \sum_{m=1}^{M} \sum_{n=2}^{N} \left(x\left(m,n\right) - x\left(m,n-1\right) \right)^2} \tag{6.2}$$

$$C = \sqrt{\frac{1}{MN} \sum_{m=2}^{M} \sum_{n=1}^{N} \left(x\left(m,n\right) - x\left(m-1,n\right) \right)^2} \tag{6.3}$$

where R is row frequency and C is column frequency, x(m,n) denotes the samples of grayscale image, M and N are numbers of pixels in horizontal and vertical directions respectively. The SFM of each grayscale test image was measured and the results were as follows: (Lena: 19.46, Peppers: 19.84, Baboon: 33.40, Girl: 13.18, Airplane: 24.63).

6.4 Experimental Results

The results of the experiment are organized, discussed, and viewed from three perspectives. Therefore, in order to investigate the relationship between the PSNR and the subjective evaluation we used these two measures to assess the quality of all stego images.

Initially, we evaluated the performance of the four steganography methods (methods 1-4) by measuring the quality of their grayscale stego images firstly by using the PSNR metric and secondly by using the subjective evaluation method proposed. Then, we evaluated the impact of using both luminance and chrominance components in colour images for steganography (methods 5-8) on the quality of stego images. Therefore, methods 5 and 7 use only the Y component for embedding while methods 6 and 8 use the Y, Cb, and Cr components for embedding. As a result, we measured the quality of these stego images using both PSNR and subjective assessment method. Finally, we evaluated the performance of both grayscale and colour versions of a given image when a similar data is embedded in both images using a given steganography method ((method (1 with 5) and (2 with 7)).

The mean opinion score (MOS) of each stego image and reference image was calculated according to equation (2.4). In each presentation, subjects are asked to evaluate both images (reference and stego), randomly presented on the screen. Since a given reference image (grayscale or colour) must be evaluated four times (once for each steganography method) by every subject, we used the ratio of (MOSr=MOS/MOS(ref)). Therefore, the MOS of each single stego image is divided by the MOS(ref) of the reference image (which appear together in the same presentation). Moreover, this ratio (MOSr) expresses the quality difference between these two images (the stego and reference).

The value of PSNR for a given image means nothing. Therefore, comparing the PSNR value of two images gives the measure of image quality (equation (2.2)). Since we used both stego and reference images together in each presentation of our subjective evaluation method, we have to use the MOS of stego images relatively to that of accompanied reference images. For this reason, we have to divide the PSNR of each stego image by the PSNR(ref) of its reference image (PSNRr=PSNR/PSNR(ref)) to validate the comparison between these two image quality measures. Basically, the reference image represents the compressed clean image of the source image while its PSNR(ref) is calculated against the original uncompressed image.

As mentioned before, every subject was asked to assess the quality of 10 pairs of reference-reference images. However, we used a continuous quality scale (0-100) rather than a 5-level scale (i.e. DSIS). Therefore, it is very unlikely for a subject to choose the same score for both images in a given presentation. Selecting scores close to each other is most likely to be seen in continuous scale methods. This is because the probability of selecting similar scores in the case of a five-level quality scale is higher than that of a continuous scale. Therefore, the rate of correct identification (assessment) for these reference-reference pairs was relatively low in our experiment: thus, the rate of hitting grayscale reference-reference pairs is 1.6/5 while it is 1.43/5 for colour pairs.

It is important to know the distribution of the data under analysis (normality of distribution) in order to perform the statistical analysis. Collected data (PSNRr and MOSr) (dependent variables) was analysed using the Kolmogorov-Smirnov statistic to verify the normality of its distributions. However, the assumption of normality was not verified for both (PSNRr) and (MOSr) (p<0.05). Therefore, non-parametric statistics were used to analyse the data collected.

Accordingly, a Kruskal-Wallis test revealed a statistically significant difference in these two measures (PSNRr and MOSr) across the eight different steganography methods;

$$x^2_{PSNRr}\text{ (7, n=1200) =636.347,} \tag{6.4}$$

$$x^2_{MOSr}\text{ (7, n=1200) =241.572,} \tag{6.5}$$

$$p_{PSNRr} < 0.05\,, \tag{6.6}$$

$$p_{MOSr} < 0.05\,. \tag{6.7}$$

In another data analysis, the relationship between the PSNRr and the MOSr was investigated using Spearman's correlation test. There was a poor positive correlation between these two variables, rho=0.2, n=1200, p<0.05. The values of correlation coefficients indicate that commonly used measure of visual quality (PSNR) can not be reliably used because it has a poor correlation with the MOS. Therefore, knowing the value of PSNR of a given stego image provides little-to-no assistance in predicting the actual quality of this image.

6.4.1 Using Different Steganography Methods

The first four steganography methods (1-4) were tested and evaluated in terms of stego image quality. Therefore, we measured the quality of all grayscale stego images (4 methods * 5 grayscale test images) using both PSNR and MOS. In the meantime, we will determine the best steganography method in terms of stego image quality. Thus, these steganography methods were comparatively evaluated once depending on the PSNR and once again depending on the MOS.

Additionally, we examined if there is any difference between these two evaluations.

Table 6.1 shows the relative objective quality of stego images (PSNR) to the quality of their reference images (PSNR(ref)). Therefore, this ratio (PSNRr) may reflect the amount of distortion added to the stego image by the used steganography method since it represents the relative objective quality. Moreover, this ratio helps us to compare the PSNR quality measure with the subjective evaluation method since the subjects are asked to assess the quality of both the reference and stego image every time. However, Figure 6.5 shows the PSNRr diagram of four steganography methods (1-4) and the five grayscale images used in the experiment.

Method	Lena	Peppers	Baboon	Girl	Airplane
JSteg (1)	0.93887	0.92951	0.51791	0.94772	0.93344
JMQT (2)	1.04193	1.05738	0.72932	1.04984	1.05018
Our First Method (3)	1.03107	1.04823	0.53087	1.04471	1.04608
Our Second Method (4)	1.05506	1.06134	0.58401	1.05448	1.02842

Table 6.1: The PSNRr of Grayscale Stego Images

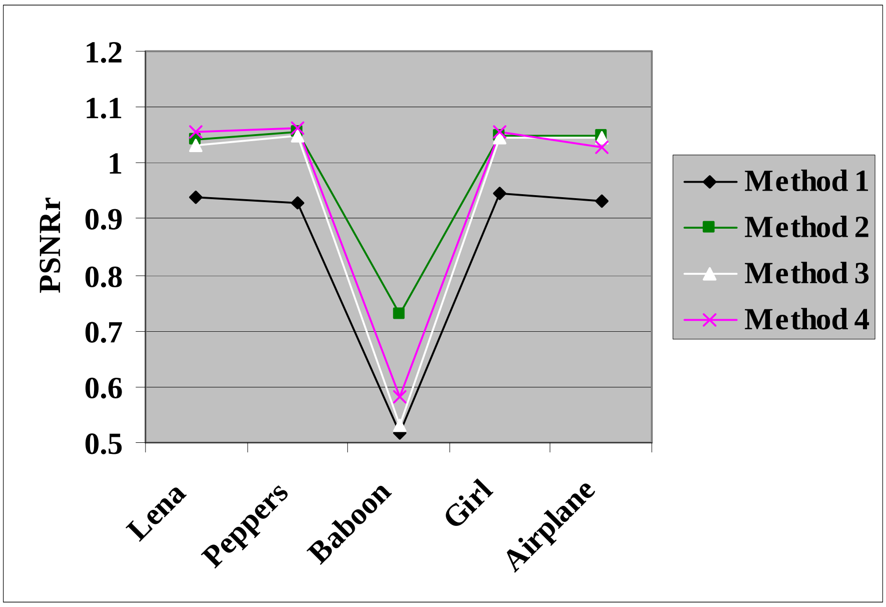


Figure 6.5: The PSNRr Diagram of Grayscale Stego Images

Table 6.2 shows the relative subjective quality score (MOS) of stego images to the quality score of the reference images (MOS(ref)). Additionally, Figure 6.6 shows the MOSr diagram of these four steganography methods (1-4) and the five grayscale images. As a result, a comparative evaluation between Figure 6.5 and Figure 6.6 reveals how different these two measures (PSNR and MOS) are.

Method	Lena	Peppers	Baboon	Girl	Airplane
JSteg (1)	0.69981	0.85126	0.8975	0.75632	0.86445
JMQT (2)	1.10211	1.02964	1.00576	1.07775	1.04847
Our First Method (3)	0.82191	0.95998	0.96904	0.85178	0.89704
Our Second Method (4)	0.79769	0.95245	0.98409	1.00208	0.93916

Table 6.2: The MOSr of Grayscale Stego Images

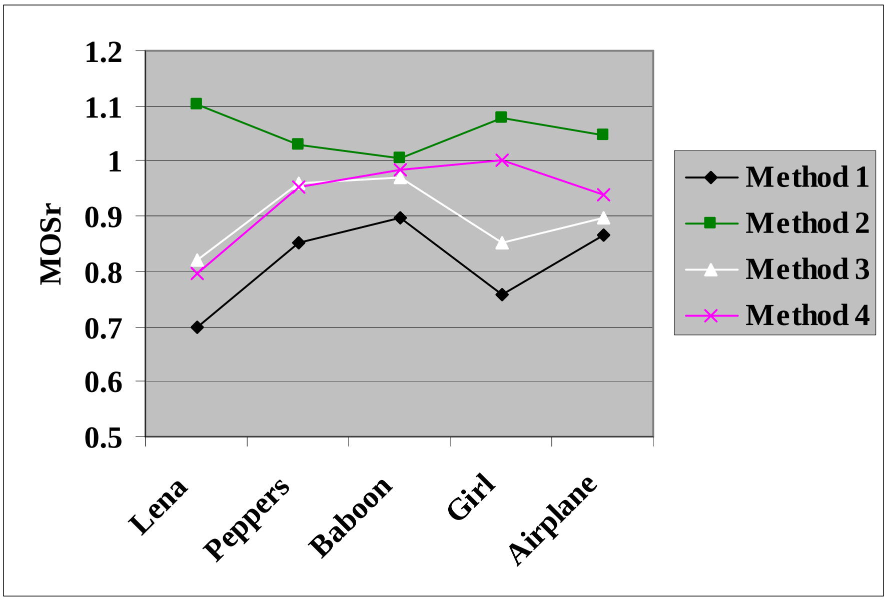


Figure 6.6: The MOSr Diagram of Grayscale Stego Images

In order to evaluate the impact of measuring the quality of stego images using PSNR, the performance of input steganography methods should be evaluated. Thus, we are going to illustrate what can happen if only the PSNR is used as an objective quality measure of stego image. Basically, if we consider only PSNRr, we can conclude that our hybrid method (Method 4), proposed in the Chapter 4, provides a better stego image quality than other three methods (Methods 1-3) for three test images (Table 6.1 and Figure 6.5). Moreover, this conclusion is clearly presented in (Almohammad et al., 2009) where we used another set of cover images (five 512x512 pixels images). The PSNR of all stego images tested for Method 4 was better than that of all other three steganography methods.

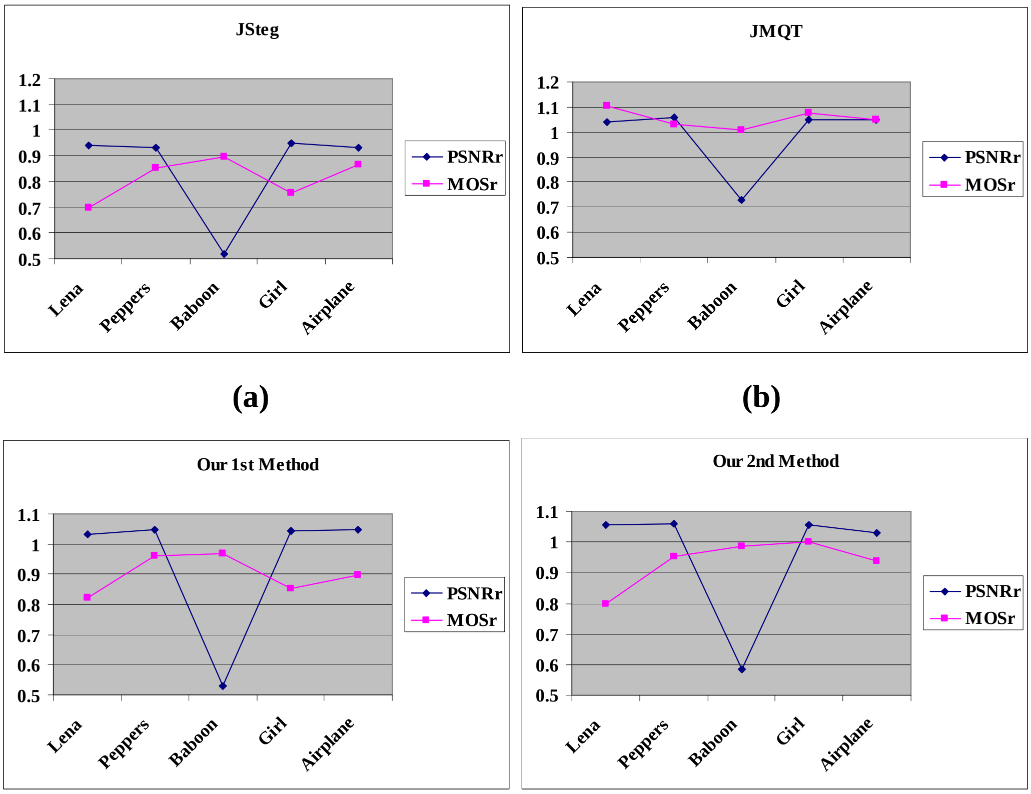
However, if we take into account the visual quality of these stego images measured by the MOS, the conclusions are quite different. For all test images, the JMQT produces better visual quality (MOSr) than the other three steganography methods (JSteg, our first method, and our second (hybrid) method) (Table 6.2 and Figure 6.6).

Additionally, comparing both our first and second steganography methods, MOS and PSNR provide totally different conclusions for three test images. Thus, the conclusion of using the PSNR measures and the subjective measure are similar only for two images out of five. However, PSNR shows that Method 4 is better than Method 3 for all test images except the Airplane image. Moreover, the conclusion of our study (Almohammad et al., 2009), where another set of cover images were used (five 512x512 pixels images), was that the PSNR value for all Method 4 stego images tested is larger (better quality) than that of Method 3.

Obviously, PSNR cannot be used as a definitive stego image quality measure. Therefore, evaluating steganography methods by measuring the quality of their stego images using the PSNR metric generates different conclusions than using MOS or subjective evaluation method.

Figure 6.7 shows both PSNRr and MOSr diagrams for each steganography method but for all test images. The PSNRr value of the Baboon image is the lowest among all five test images and four steganography methods. Basically, the Baboon image contains a lot of detail and has the largest (SFM) among all test images. This means that the Baboon image contains few redundancies, which is difficult for compression, and also contains numerous components in high frequency area. Additionally, there is a strong negative relationship between the PSNR and SFM (Grgic et al., 2004). Therefore, this explains the reason behind the low PSNRr of the Baboon test image.

Furthermore, the lowest value of PSNRr for all Baboon stego images compared to that of all other stego images indicates that these stego images (Baboon) are the images most degraded by steganography (Figure 6.7). However, this is not true for the MOS measure since most of the Baboon stego images have a larger MOSr value than the other stego images, which means that Baboon stego images are less degraded by steganography than many other stego images.



(a) **(b)**

(c) **(d)**

Figure 6.7: Comparing the PSNRr and MOSr of Grayscale Stego Images

If we consider each steganography method as a source of noise, steganography methods may change or manipulate the test images differently. Figure 6.7 shows that the shape of the PSNRr diagram is almost the same for all steganography methods. Moreover, apart from the Baboon image, all other stego images for each steganography method have almost the same PSNRr value. This reflects the statistical nature of the PSNR measure which calculates the image quality regardless the nature and the type of the added noise (steganography method).

However, the shape of the MOSr diagram is different from one steganography method to another. Additionally, using a particular steganography method for different test images has different impacts on the quality of the reconstructed stego images. Therefore, this is more expressed by the MOSr measure (considerable change in shape) rather than the PSNRr.

6.4.2 Chrominance Components as Cover Mediums

In this part of experiment, we tested the relationship between the PSNR and MOS when chrominance components (Cb and Cr) of colour images are used for steganography in addition to the luminance component (Y) (methods 5-8). Therefore, we measured the quality of all colour stego images (4 methods * 5 colour test images) using both PSNR and MOS. Furthermore, we determined the impact of using chrominance components for steganography on the quality of stego images. This quality was measured once by using PSNR and once again by using MOS in order to examine if there is any difference.

Method		Lena	Peppers	Baboon	Girl	Airplane
JSteg-Y	**(5)**	0.95529	0.96447	0.73598	0.96256	0.94826
JSteg-YCbCr	**(6)**	0.94069	0.93671	0.71722	0.94905	0.93292
JMQT-Y	**(7)**	1.02555	1.02093	0.95839	1.02771	1.03153
JMQT-YCbCr	**(8)**	1.05634	1.09932	0.96811	1.06393	1.06559

Table 6.3: The PSNRr of Colour Stego Images

Table 6.3 shows the PSNRr of all colour stego images tested. Moreover, Table 6.4 shows the MOSr of these stego images. Figure 6.8 shows the diagram of PSNRr while Figure 6.9 shows the diagram of MOSr of the 20 colour stego images used in the experiment.

If we consider only PSNRr values of JSteg stego images (method 5 and 6), we can conclude that the quality of all stego images was degraded when the chrominance components were used for steganography in addition to the luminance component. However, considering only the PSNRr values of the JMQT stego images (method

7 and 8), we can conclude that the quality of all stego images was improved when the chrominance components were used for steganography in addition to the luminance component (Table 6.3 and Figure 6.8).

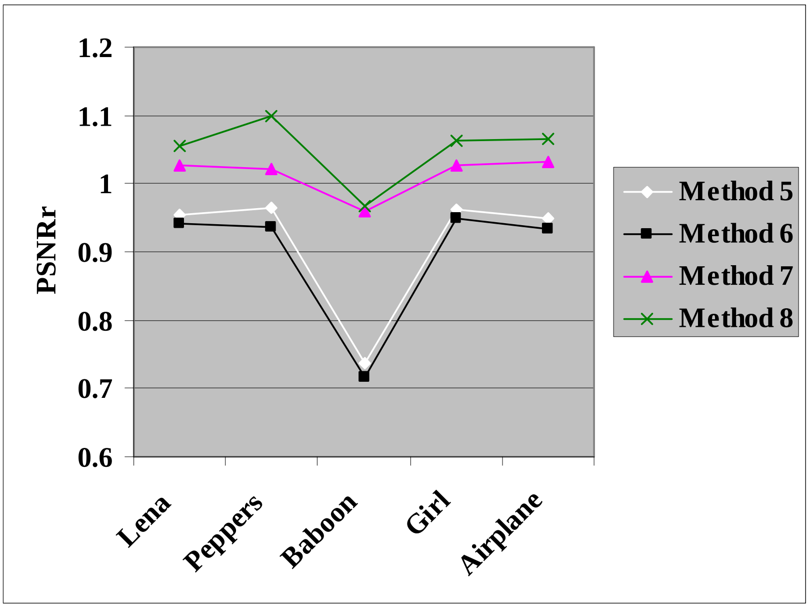


Figure 6.8: The PSNRr Diagram of Colour Stego Images

Method		Lena	Peppers	Baboon	Girl	Airplane
JSteg-Y	**(5)**	0.73086	0.9078	0.95747	0.78495	0.86429
JSteg-YCbCr	**(6)**	0.77518	0.90772	0.95873	0.67021	0.85995
JMQT-Y	**(7)**	1.08425	0.96875	1.00051	1.01037	1.08
JMQT-YCbCr	**(8)**	1.08208	0.98418	1.00643	0.98476	1.07254

Table 6.4: The MOSr of Colour Stego Images

If we take into account the visual quality of these stego images measured by the MOS, the conclusions are quite different, however. Thus, for JSteg, the MOSr of two stego images (Lena and Baboon) was improved when the chrominance components were used for steganography. Moreover, for the JMQT method, the MOSr of three stego images (Lena, Girl, and Airplane) was decreased, and quality degraded, when the chrominance components were used for steganography (Table 6.4 and Figure 6.9).

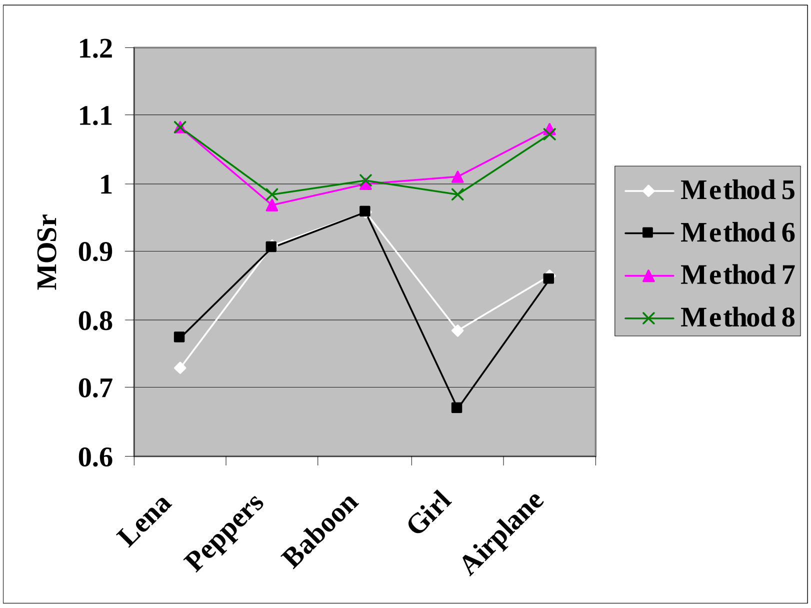


Figure 6.9: The MOSr Diagram of Colour Stego Images

As a result, these two image quality measures, PSNR and MOS, produced almost different conclusions, or at most dissimilar conclusions, when a given steganography method was used to hide secret data in the chrominance and luminance components of colour images.

6.4.3 Grayscale and Colour Images as Steganography Cover Images

In this part of experiment, we tested the relationship between the PSNR and MOS when both grayscale/colour versions of test images are used to hide the same data in the luminance component (Y) of each image version. Thus, we embedded the same amount of data and evaluated the performance of grayscale and colour versions (from the same source) in terms of stego image quality. Therefore, the quality of the colour stego image and the grayscale stego image (from the same source) are measured (Method 1 with 5 and 2 with 7).

Thus, we measured the quality of 10 colour stego images (2 methods * 5 colour images) and 10 grayscale stego images (2 methods * 5 grayscale images) using both PSNR and MOS. In the meantime, we will determine which kind of images (grayscale or colour) provides better stego image quality for a given steganography method. Therefore, stego image quality is measured once by PSNR and once again by MOS in order to examine if there is any difference in the conclusion.

Figure 6.10 shows the PSNRr diagrams of both colour and grayscale stego images tested. Moreover, Figure 6.11 shows the MOSr diagrams of these stego images.

If we consider only PSNRr values of JSteg stego images (Method 1 and 5), we can conclude that the quality of all colour stego images was better than that of grayscale stego images (Figure 6.10). Moreover, we can derive the same conclusion if we take into account the visual quality of these stego images measured by the MOS (Figure 6.11). Thus, the conclusions from both image quality measures are similar in this specific part of experiment.

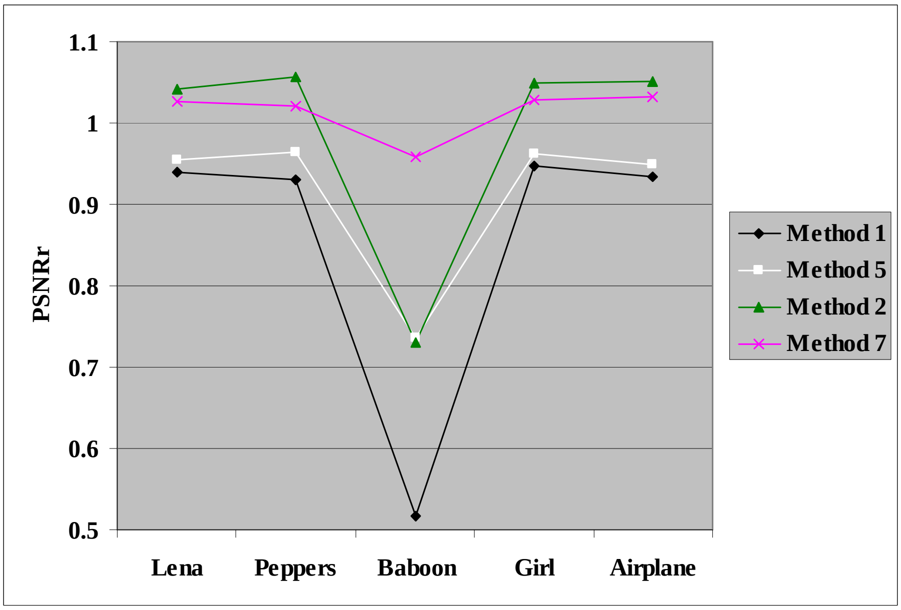


Figure 6.10: Comparing the PSNRr of Grayscale and Colour Stego Images

Additionally, considering only the PSNRr values of the JMQT stego images (Method 2 and 7) we can conclude that the quality of grayscale stego images was better than that of colour images except for the Baboon image (Figure 6.10). However, for the JMQT method, the MOSr of grayscale stego images were better than that of colour images except for Airplane image (Figure 6.11). Thus, the conclusions derived from these two image quality measures were similar for three test images out of five in this part of experiment.

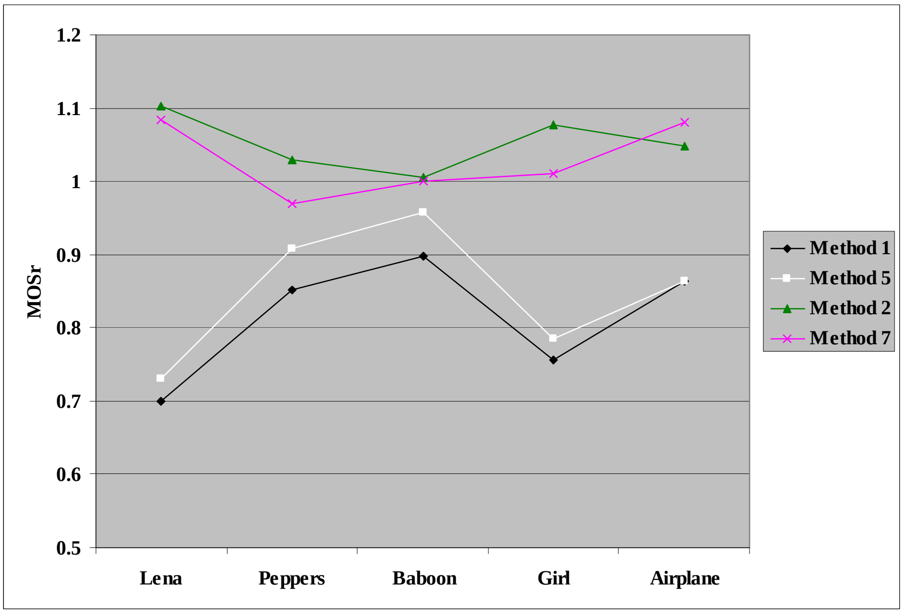


Figure 6.11: Comparing the MOSr of Grayscale and Colour Stego Images

As a result, these two image quality measures have produced somehow similar conclusions when a given steganography method was used to hide the same data in both grayscale and colour versions of a given image.

6.5 PSNR and Subjective Quality Methods

The computing of the PSNR, the widely-used image quality measure, is very easy and fast (Wang et al., 2003). Therefore, PSNR is the most common metric used to measure the quality of digital images in many image processing applications and considered as a reference model to evaluate the efficiency of other objective image quality evaluation methods (Wang et al., 2002b).

Mostly, digital image steganography uses the PSNR to evaluate the quality of stego images or the imperceptibility of steganography methods. Thus, PSNR measures the efficiency of a particular steganography method over another in terms of imperceptibility or stego image quality (See, for instance, (Agaian et al., 2006; Chang et al., 2002; Li and Wang, 2007; Liu and Liao, 2008; Noda et al., 2006). This influenced our choice to use the PSNR measure in order to evaluate the quality of stego images in Chapters 4 and 5.

PSNR has shown the best advantage almost over all objective image quality metrics under different image distortion environments and strict testing conditions. However, the PSNR does not offer good results, in terms of human perception, with colour images and it is not a reliable predictor of perceived quality (Stoica et al., 2003; Wu and Rao, 2006). Additionally, the subjective measures represent the most reliable method to determine the actual image quality and therefore the subjective test is the best method to evaluate the quality of images surely and reliably (Marini et al., 2007). This inspired us to use the subjective image quality methods in order to evaluate the quality of stego images in this chapter. We also investigated the relationship between the PSNR and MOS of stego images in this chapter.

Generally, there is a poor correlation between objective quality estimation and the actual subjective evaluation (Nyman et al., 2006). Thus, the PSNR represents a poor indicator of the subjective image quality (Wang et al., 2002a; Wang et al., 2002b). In this chapter, we examined the correlation between PSNR and MOS for stego images in specific. As a result, there is a poor positive correlation between these two measures: PSNR and MOS. This outcome would not affect our results shown in Chapters 4 and 5, however. Since the outcomes of both PSNR and MOS are not identical, we claimed that the PSNR can not be reliably used for stego images. Indeed, the conclusions derived from PSNR do not contradict those conclusions derived from the MOS. These conclusions are not identical also, so we claimed that the conclusions derived from the PSNR are different from those derived from the MOS. As a result, the MOS method is used to improve and ensure the results of our proposed steganography methods rather than to invalidate these methods since there is a positive correlation between the PSNR and MOS.

6.6 Conclusion

Different steganography methods introduce different types of degradation into reconstructed stego images. Moreover, the results of an evaluation concerning stego image quality are measured subjectively using an adapted double stimulus continuous quality scale method (DSCQS) and presented. Additionally, stego image quality is measured objectively using PSNR metric. We then examined the relationship between the PSNR and the subjective quality (MOS) of different stego images.

It has been found that the commonly used measure of stego image quality PSNR can not be reliably used because it has a poor correlation with MOS. Moreover, evaluating steganography methods by measuring the quality of their stego images using the PSNR metric resulted in different conclusions than using the MOS method. Using the JSteg method, the PSNR values of stego images when data was embedded in all image components were smaller than that of stego images when data was embedded only in the luminance component. However, this is not the case when MOS measure was used to evaluate the quality of these stego images. Using the JMQT method, the PSNR values of stego images when data was embedded in all image components were higher than that of stego images when data was embedded only in the luminance component. This is not the case when the quality of these stego images was measured using the MOS. Hiding the same data in both grayscale and colour versions of a given image using a particular steganography method, we almost get the same conclusion from both PSNR and MOS values.

MOS indicated that using a particular steganography method for different test images affects the quality of the reconstructed stego images differently. However, the PSNR did not express or show such conclusion. Thus, the PSNR is not a wrong indication of the perceived stego image quality but it is not also a strong predictor of the actual quality of stego images. There is a poor positive correlation between the PSNR and MOS and the subjective evaluation improves and enhances the results of PSNR rather than invalidate them. Thus our results in Chapters 4 and 5 will not be affected by the results of this chapter anyway.

Chapter 7: Information Hiding in SOAP

7.1 Introduction

The SOAP protocol is designed to enable the exchange of structured information (i.e. SOAP messages) over a variety of underlying protocols in decentralized and distributed environments. This lightweight protocol uses XML technologies to define a messaging framework that is independent of any specific programming languages or implementation semantics (Lubacz et al., 2010).

In this chapter, we intend to consider objective 5 of our study, which concerns information hiding within SOAP messages in an undetectable way. To this end, we will propose a novel steganography algorithm to be used with SOAP messages. This method manipulates the SOAP protocol by rearranging the order of the contents and attributes (sub-elements) of specific elements in a SOAP message. Basically, this method has a high imperceptibility since it leaves almost no trail because of using the communication protocol as a cover medium. Thus, it keeps the structure and size of the SOAP message intact.

In order to overcome the capacity issue, we can use as many SOAP messages as we need to hide our secret message. Hence, we can split the secret message into many sub-messages and then hide them in different SOAP messages to be sent subsequently to an intended recipient. Furthermore, this method has no effects on the SOAP message functionality and therefore is a secure method of information hiding. Also, this method has been experimentally tested and evaluated.

The structure of this chapter is as follows: section 7.2 provides a description for the main features and characteristics of SOAP messages. In section 7.3, we show the possibility and feasibility of hiding secret data depending on the contents and characteristics of SOAP messages. Then, the previous works related to XML and SOAP based steganography are summarised in section 7.4. The procedure of data hiding and then data extracting using our method were presented in section 7.5. However, information concerning the experimental approach used and the results of this experiment are presented in section 7.6. In section 7.7, we consider issues related to security aspects provided by SOAP steganography compared to other conventional security features. Finally, in section 7.8, the conclusion is presented.

7.2 SOAP Specifications and Processing

In this section, we intend to introduce the reader to the SOAP specifications and functionalities in order to show the possibility of using some of these features for data hiding. Since some SOAP rules are loose, we can rely on this to propose a novel steganography method. Thus, we explain and describe some features and aspects of SOAP protocol which must be met in order to get a valid communication over the Web. Moreover, issues of SOAP message handling and processing are also explained here.

SOAP supports XML messaging, defines the elements and the rules of XML messaging, and sends an XML message via Hyper Text Transport Protocol (HTTP) request; it may then receive a reply via HTTP response. Thus, a SOAP processor such as Apache or Microsoft Internet Information Server (IIS) is needed to handle XML messages correctly. Additionally, SOAP processors must be able to validate and understand the format of XML documents defined in the SOAP specification. Moreover, the Web service implementation must know how to understand the data within SOAP messages since SOAP makes the transmitted messages to be known as a SOAP message only.

SOAP nodes can be SOAP message senders, receivers, or both. Furthermore, a SOAP intermediary is a special case of a SOAP node located between sender and receiver for the purpose of handling special headers. Hence, a SOAP node can support one or more SOAP processors and is responsible for handling the received SOAP messages.

SOAP specification contains many rules that may or may not be imposed. Also, we may find two SOAP messages implementing a different collection of optional features. Therefore, there is a difference between required rules in a specification and widely acceptable or optional features. Unlike loose rules, the tight specification rules may discourage implementation. Hence, if a SOAP processor is compliant with v1.1 but received a message containing the v1.2 envelope namespace then it will generate a fault. Thus, SOAP versions must be changed when the envelope changes. There is an attribute within headers named *“mustUnderstand”* which can be used to ask SOAP processors to understand the header or to reject the message if they could not understand the header. The blocks of SOAP message can be processed in any order if the optional header did not constrain the processing order. Furthermore, SOAP messages that do not have correct namespaces will be discarded by the SOAP processor, however SOAP messages without namespaces can be processed.

SOAP messages represent a significant part of Web services since they offer a reliable link between SOAP processors of different organisations regardless of the operating system used, programming language, and object model. Thus, SOAP messages represent a suitable cover of secret communications and steganography. In this chapter, we propose a novel method of data hiding to be used with SOAP messages. Thus, this method will take into account all these specifications in order to get a valid SOAP message which can be processed correctly.

7.3 Information Hiding in SOAP messages

Communicating through SOAP messages requires the actual data in the sender endpoint to be converted into an XML stream. This converted (serialized) XML document is the SOAP message that needs to be de-serialized at the receiver endpoint to reconstruct the actual data. Figure 7.1 illustrates an example Java class “*Book*” and its XML serialized class instance.

```
public class BookOrder{
   private String isbn;
   private String author;
   private String bookName;
   private int numOfPages;
   private String publisher;
   private int year;
   private double price;

  public getters and setters
}
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
   <S:Body>
      <ns2:BookOrder
xmlns:ns2="http://service.bookorder.com/">
         <book>
         <isbn>1-11-111111-1</isbn>
         <author>Author_1</author>
         <bookName>Book_1</ bookName >
         <numOfPages>372</numOfPages>
         <publisher>Publisher_1</publisher>
         < year >2009</year>
         <price>29.99</price>
         </book>
      </ns2:BookOrder>
   </S:Body>
</S:Envelope>
```

Figure 7.1: Example Java Class and its Serialized Instance

An endpoint application normally employs a SOAP package to perform the serialization and de-serialization processes, as web applications and clients care mainly about the actual data transmitted and not the structure of the SOAP message. Hence, secret information can be smuggled into SOAP messages, which provide a perfect cover if the hidden secret message does not damage the SOAP messages or spoil the actual data.

Hiding secret information in a SOAP message means that the mule that is used to convey the secret message is the communication protocol that governs the actual data path over a network, instead of using the actual data itself as a cover. This idea can overcome many of the limitations that faced conventional steganography techniques. Traditional steganography techniques hide secret data inside digital files, which impose the threat of detecting the secret as these files are usually saved. Alternatively, a SOAP message leaves almost no trail as they are normally deleted after receiving the message and de-serializing the actual data. Additionally, a secret piece of information can be divided into multiple smaller messages and transmitted over several SOAP messages to overcome the size limitation and steganographic capacity problem.

In the literature, the number of studies targeting XML steganography is quite limited. However, out of all of these there is only one study regarding the SOAP steganography. According to XML and SOAP specifications, some of these techniques are infeasible and invalid since they do not comply with the compulsory aspects of the cover files or protocols. Thus, the steganography methods proposed for XML and SOAP message are described and discussed in the next section.

7.4 Previous Steganography Methods Analysed

There are only a couple of studies and examples of research regarding information hiding in XML files. Inoue et al. (2001) stated that there is almost no study on steganography methods in structured documents. Thus, they proposed five steganography methods to be used with XML files. We can summarise these steganography methods as follows (Figure 7.2):

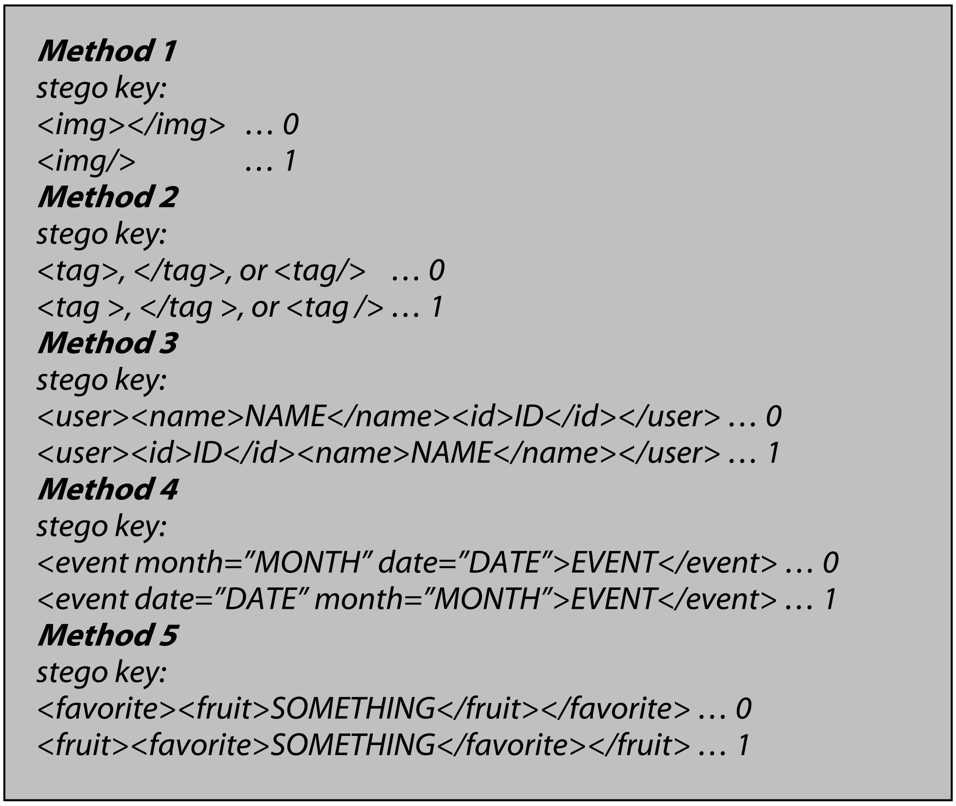
1- The empty elements are represented according to the secret bit; either a start-tag immediately followed by an end-tag (<img></img>), or an empty-element tag (<img/>). This technique can embed one bit per empty element.

2- According to the secret bit, we can either add a white space before the close bracket (<tag >), or delete (normal with no added spaces) this white space (<tag>). This technique can embed one bit per tag.

3- Two elements may or may not be exchanged according to the secret bit. Thus, one bit per an exchange of two elements can be hidden.

4- The order of attributes in an element can be exchanged to hide the secret data. Thus, one bit per an exchange of the attributes order can be hidden.

5- Elements that contain each other can be used to hide secret data by exchanging inner-tags and outer-tags. In this method, one bit per an exchange can be hidden.



```
Method 1
stego key:
<img></img>  … 0
<img/>       … 1
Method 2
stego key:
<tag>, </tag>, or <tag/>   … 0
<tag >, </tag >, or <tag /> … 1
Method 3
stego key:
<user><name>NAME</name><id>ID</id></user> … 0
<user><id>ID</id><name>NAME</name></user> … 1
Method 4
stego key:
<event month="MONTH" date="DATE">EVENT</event> … 0
<event date="DATE" month="MONTH">EVENT</event> … 1
Method 5
stego key:
<favorite><fruit>SOMETHING</fruit></favorite> … 0
<fruit><favorite>SOMETHING</favorite></fruit> … 1
```

Figure 7.2: Examples of the Five Methods Proposed by Inoue et al. (2001)

If an element has no content then empty-element tag can be used whether or not it is declared using the keyword EMPTY. However, the number of such elements in an XML document is limited and then the capacity of method (1) is limited too. Additionally, using two formats to represent empty elements in the same document will arouse the attention of observers. Also, the parser may use only one representation of empty elements rather than two, which invalidate this method.

Names of XML elements can't contain spaces but there can be space before the closing character ">" (<tag >). However, this process will increase the size of the XML file and the hidden data may be destroyed due to parsing which may discard these added spaces (secret data). Additionally, tags are case sensitive and therefore the tag <tag> is different from the tag <tag >. In other words, the end-tag's name has to exactly match the start-tag's name. Thus, the method (2) is practically infeasible since it uses a start-tag different from the end-tag (one tag may contain a white-space).

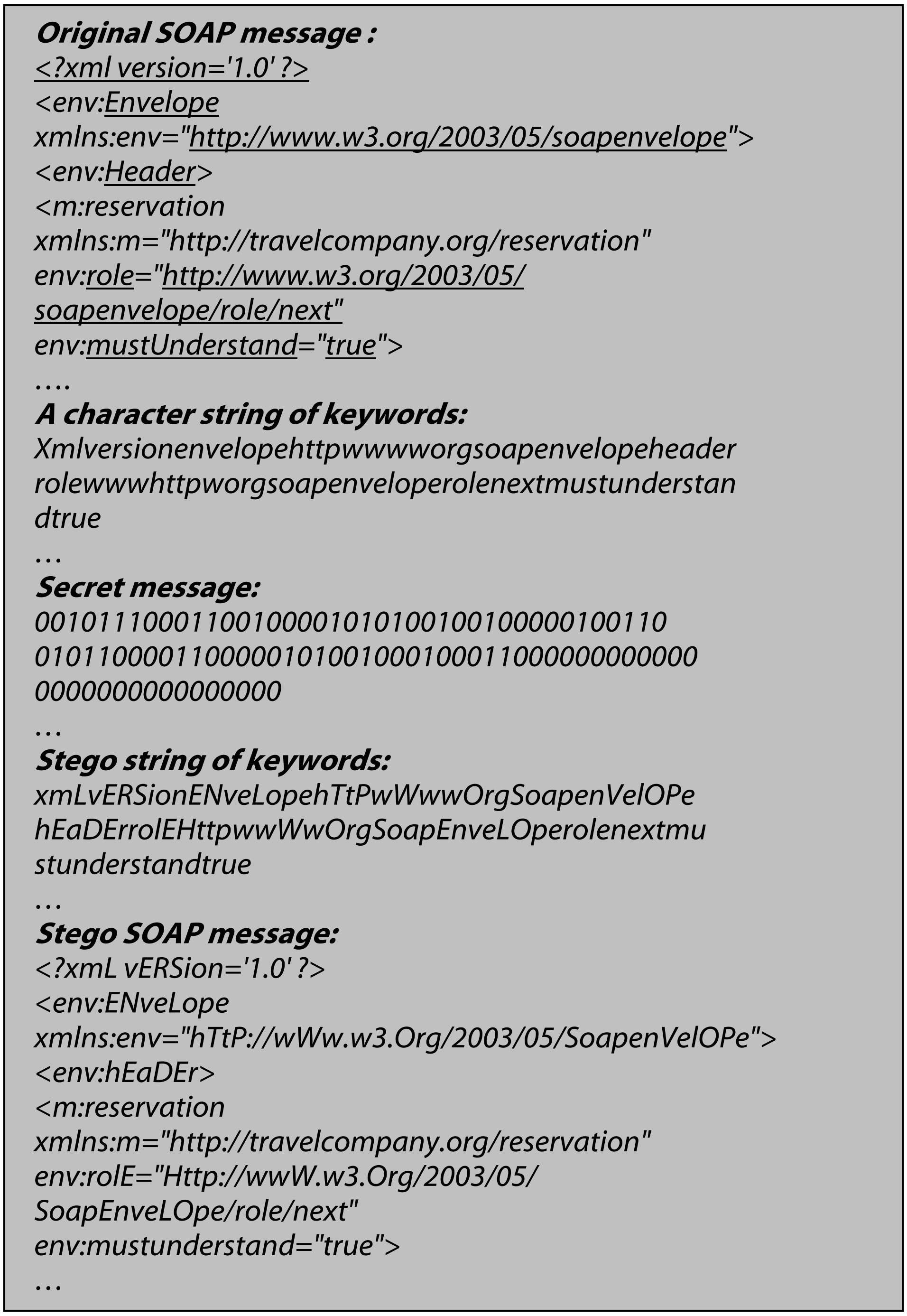
The order in which attributes are included on an element is not considered relevant. For example, if an XML parser encounters a specific order of an element attributes, it doesn't necessarily have to give us the attributes in the same order. As a result, method (4) above is infeasible in terms of validity and applicability even though its capacity is very limited. However, a certain order of information can be maintained in an XML document if we put this information into elements, rather than attributes. As a result, method (3) above is a valid and possible solution for steganography. Nevertheless, hiding only one secret bit per an exchange of two elements represents a very small capacity.

Finally, an XML document must have a top-level element and all the other elements are its children. Furthermore, one and only one root element must be included in each XML document even if this element has no content. However, each of these children elements may represent a parent element and therefore have some sub-elements. Thus, exchanging a parent element with a sub-element technically looks valid (method (5) above). However, it seems impractical since the semantics will not make sense and we will get a new and different parent element by such an exchange. Also, the steganographic capacity of this method is very limited.

Since XML documents are widely used for data exchange over different networks and exposed to different threats, XML security become a key concern of organisations. Thus, Memon et al. (2008) considered XML steganography as a new method and solution for secure communication. Furthermore, they proposed and designed four XML based steganography methods for the purpose of securing the cover file (XML document) rather than for the purpose of secret communication. The main aspects of these methods are as follows:

1- Random characters are inserted inside all tags and their values. So, after the 1st character of the 1st tag one random character is inserted, after the 2nd character of the 1st tag two random characters are inserted and so on. Thus, it mixes up the actual XML data with random fake characters and therefore increases the size of the stego XML file significantly.
2- XML tags are shuffled (sequentially) in such a way the position of the 1st tag and its value are swapped with that of the last tag and its value. The same process happens with the second and the second last tags, and so on. The large XML file is, the better this technique work.
3- Similar to method (2) above, but the correct order of shuffled tags is identified in the attribute value of the root element. Thus, the first tag is determined by the first character of attribute value while the second character is randomly generated. Also, this method works better with large XML files.
4- The sequence of characters in all tags and values are reversed. Thus, the order of tags' characters is reversed by moving the last character to become the first one while the second last one becomes the second character and so on. As a result, the XML file will look like an encrypted file since the characters are scrambled in an unreadable form.

Then, they suggested combining all these methods together in one hybrid method to provide better XML security. In conclusion, all these four methods aim to safeguard the stego XML document against actual XML content detection rather than against hidden information detection. Additionally, their goal is the XML content not the hidden data itself. Therefore, the goal of these methods is totally different from our steganography goal which is undetectable and covert communication. Nevertheless, the first and fourth methods are definitely infeasible for steganography since the stego XML arouses the suspicion of everyone (look like encrypted). The second method may hide a few bits only, while in the third method, the secret key is included in the stego file, which is more than enough to extract the hidden message.



```
Original SOAP message :
<?xml version='1.0' ?>
<env:Envelope
xmlns:env="http://www.w3.org/2003/05/soapenvelope">
<env:Header>
<m:reservation
xmlns:m="http://travelcompany.org/reservation"
env:role="http://www.w3.org/2003/05/
soapenvelope/role/next"
env:mustUnderstand="true">
....
A character string of keywords:
Xmlversionenvelopehttpwwwworgsoapenvelopeheader
rolewwwhttpworgsoapenveloperolenextmustunderstan
dtrue
...
Secret message:
001011100011001000010101001001000000100110
0101100001100000101001000100011000000000000
0000000000000000
...
Stego string of keywords:
xmLvERSionENveLopehTtPwWwwOrgSoapenVelOPe
hEaDErrolEHttpwwWwOrgSoapEnveLOperolenextmu
stunderstandtrue
...
Stego SOAP message:
<?xmL vERSion='1.0' ?>
<env:ENveLope
xmlns:env="hTtP://wWw.w3.Org/2003/05/SoapenVelOPe">
<env:hEaDEr>
<m:reservation
xmlns:m="http://travelcompany.org/reservation"
env:rolE="Http://wwW.w3.Org/2003/05/
SoapEnveLOpe/role/next"
env:mustunderstand="true">
...
```

Figure 7.3: SOAP-Based Steganography Method (Zhang et al., 2007)

SOAP parsers have been developed and they only process XML that conforms to the SOAP schema and associated structural rules. To the best of our knowledge, there is only one study regarding steganography with SOAP messages (Zhang et al., 2007). The method proposed uses the physical properties of SOAP keywords and namespaces (elements and attributes) as steganography covers. It has been assumed that users do care about actual data transmitted but do not care about SOAP keywords and namespaces. Thus, according to the secret bit, the coordinated letter from the cover string (built from the keywords and namespaces) will be set either as lower-case (to embed "0") or upper-case (to embed "1") (Figure 7.3).

As a result, the elements and namespaces of the stego SOAP message have both lower-case and upper-case letters. Essentially, this method does not comply with the case-sensitivity nature of XML documents. Additionally, the overall shape of the stego SOAP message is very suspicious due to the different and random letter-cases. Thus, this method can not be considered as a feasible steganography algorithm.

7.5 SOAP Steganography Method Proposed

The main concern of hiding secret information within a SOAP message is how we can hide this information and not be detected. Basically, end users care about the actual data transmitted but they do not care about other issues like SOAP namespace, keywords, or the attributes-order of elements. However, the transmitted message must be recognised by SOAP processors and must generate no errors; therefore it should not be discarded. Additionally, the final SOAP message should look normal (in terms of its form and size) in case it has been intercepted by an attacker.

As a result, we propose a novel steganography method that manipulates the SOAP protocol. This method is based on rearranging the order of the contents and attributes of specific elements in a SOAP message. Additionally, every permutation represents a specific status according to a secret key shared between the sender and the receiver. For example, there are 7 sub-elements within the element "book" in Figure 7.1. These sub-elements are arranged in a particular order (isbn, author, bookName, numOfPages, publisher, year, price). This order is not significant for the endpoint application, however. If the order of these sub-elements is rearranged, the SOAP message will still have the same meaning for the endpoint. For a set of (n) sub-elements, there are a maximum of (n!) (factorial of n) permutations. This means that (n!) different sequences of order can be presented.

7.5.1 Embedding and Extracting Procedures

Considering the previous concept, we have designed and implemented a data hiding method that monitors a SOAP message just after its serialization in the sender endpoint and before it is sent, analyses its elements and embeds a secret message accordingly. When the stego SOAP message arrives at the receiver endpoint, the secret message is extracted using a stego key that is shared between the sender and receiver. Figure 7.4 illustrates the general model of data hiding in SOAP messages.

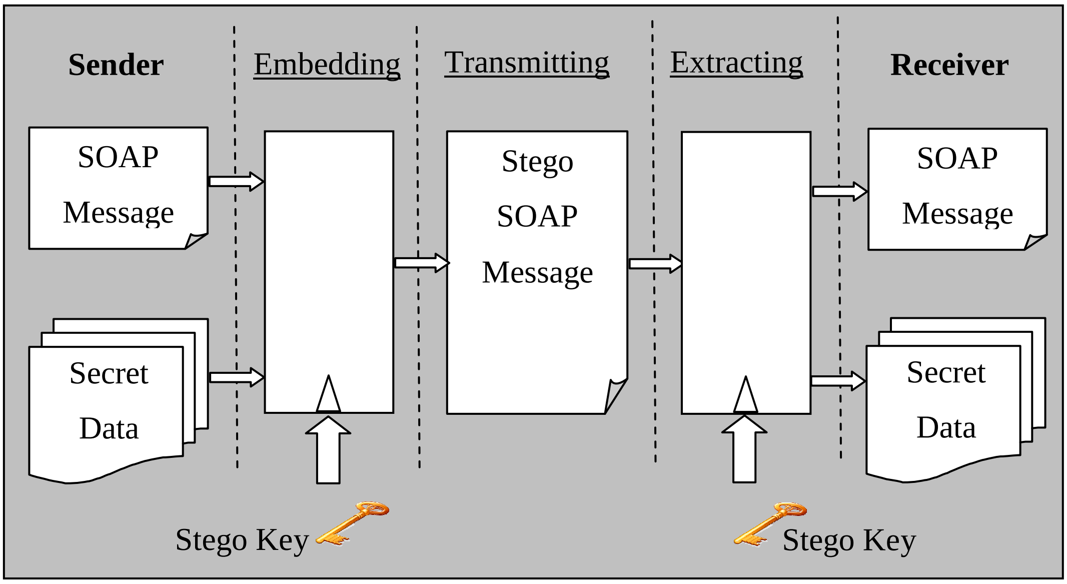


Figure 7.4: General SOAP Steganography Model

In our proposed method, the procedure of hiding a secret message within SOAP consists of six steps. Figure 7.5 explains this procedure while Figure 7.6 shows the diagram of our SOAP-based data embedding algorithm.

1. *Capturing the SOAP message after its serialization.*

2. *Analysing its contents to identify all the elements with contents that can be rearranged to determine if the SOAP message is suitable for embedding (i.e. has elements with contents that can be rearranged).*

3. *Calculating the number of elements that can be used to hide data (N).*

4. *Permuting every set of sub-elements to reflect a status of a symbol from the secret message.*

5. *If all the symbols of the secret message can be hidden in one SOAP message (the number of available sets N is greater than the length of the secret message M), then the sub-elements of the set M+1 will be rearranged to indicate the end of secret message.*

6. *Otherwise, if M>N, only a part of the secret message is sent in this SOAP message and the last set of sub-elements is rearranged to indicate that more hidden data are to arrive within the next received SOAP message.*

Figure 7.5: The Embedding Procedure of our SOAP-Steganography Method

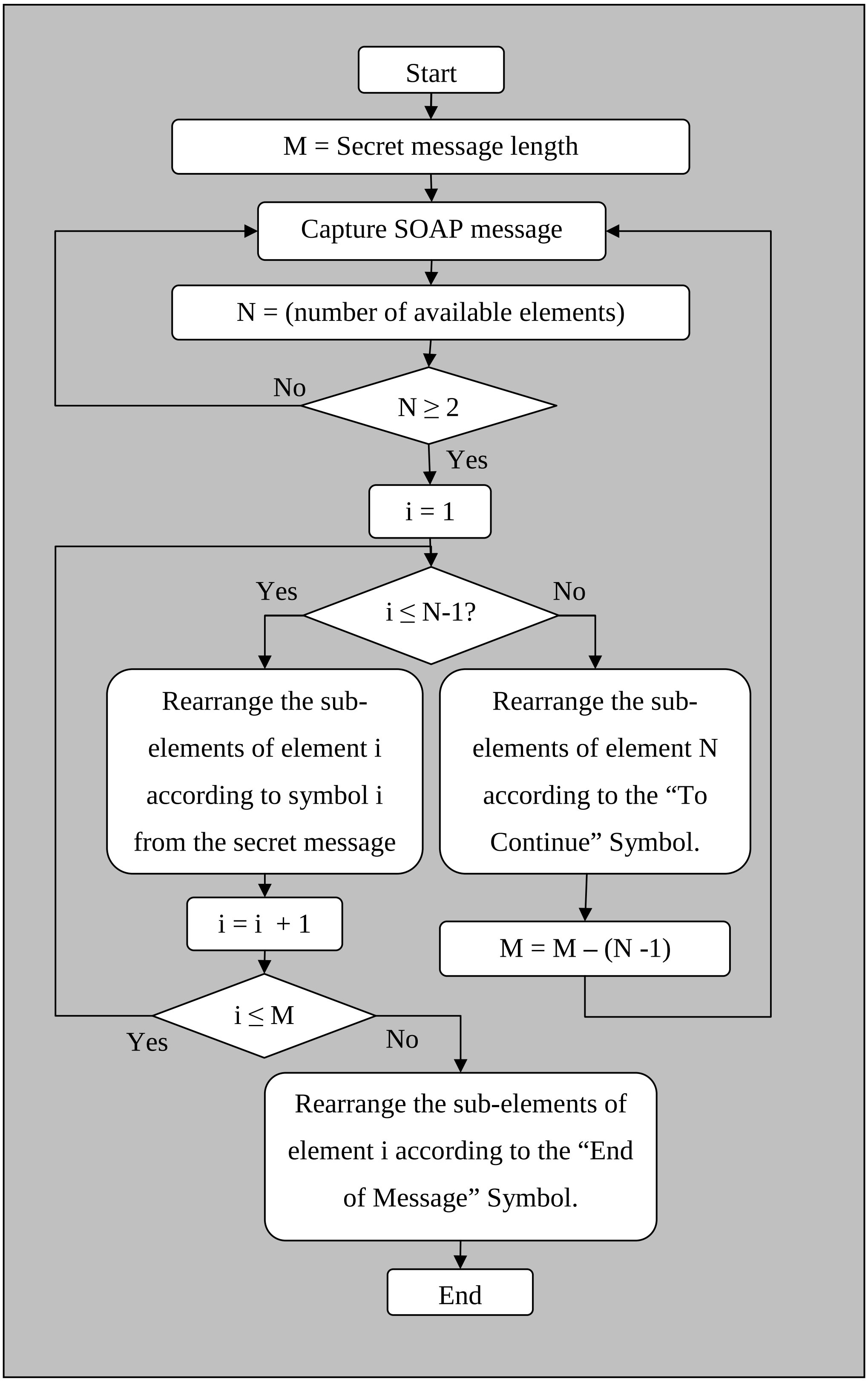

Start
M = Secret message length
Capture SOAP message
N = (number of available elements)
N ≥ 2
No
Yes
i = 1
i ≤ N-1?
Yes
No
Rearrange the sub-elements of element i according to symbol i from the secret message
Rearrange the sub-elements of element N according to the "To Continue" Symbol.
i = i + 1
M = M – (N -1)
i ≤ M
Yes
No
Rearrange the sub-elements of element i according to the "End of Message" Symbol.
End

Figure 7.6: Our SOAP Message-based Embedding Algorithm

1. *Capturing the SOAP message and checking its validity and capability to be a stego SOAP message.*

2. *Calculating the number of elements that might be used for data hiding (N).*

3. *Extracting the hidden symbols by analysing the sub-elements order of each element in the stego SOAP message.*

4. *Stop the process either if the extracted symbol indicates that the message is not a stego SOAP or if the extracted symbol means "End of Message".*

5. *If the extracted symbol means "To Continue", new SOAP message to be captured and analysed as in 1.*

6. *Otherwise, the next symbol will be extracted and so on until we get the entire secret message embedded.*

Figure 7.7: The Extracting Procedure of our SOAP-Steganography Method

The process of extracting the secret message from the SOAP messages is quite simple and easy. The receiver of stego SOAP message extracts hidden data by analysing the contents of each eligible element using the secret key to reveal the hidden symbol, as explained in Figure 7.7. Figure 7.8 shows the diagram of embedded data extracting procedure.

As a result, the procedure of data extracting from stego SOAP messages is the reverse process of the hiding procedure. Additionally, the receiver endpoint has to check the SOAP structure to determine whether it is a possible stego SOAP or not. The permutation status of each set of sub-elements has to be examined to see if this SOAP message is actual stego SOAP or not.

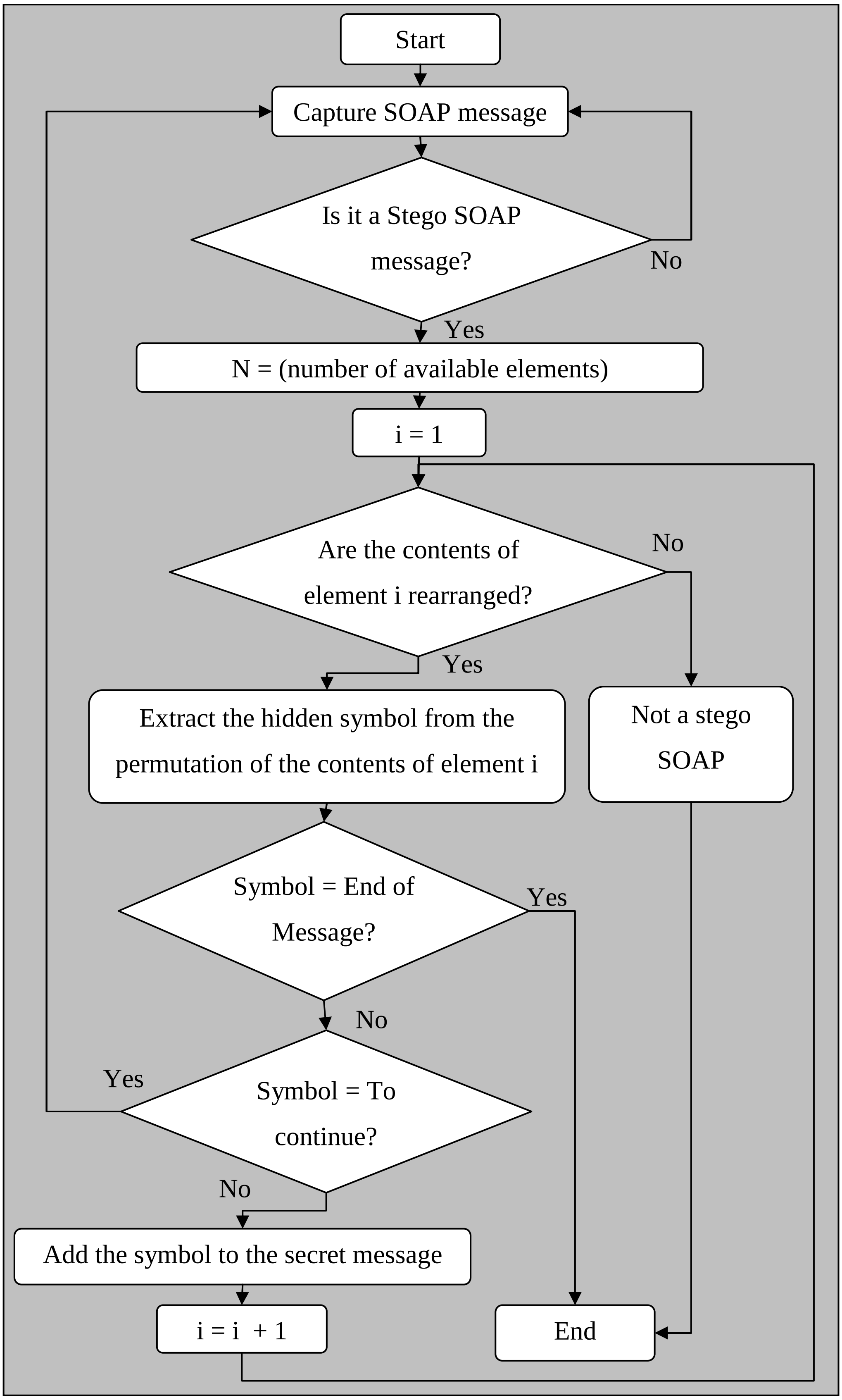

Start
Capture SOAP message
Is it a Stego SOAP message?
No
Yes
N = (number of available elements)
i = 1
Are the contents of element i rearranged?
No
Yes
Extract the hidden symbol from the permutation of the contents of element i
Not a stego SOAP
Symbol = End of Message?
Yes
No
Yes
Symbol = To continue?
No
Add the symbol to the secret message
i = i + 1
End

Figure 7.8: Our SOAP Message-based Extracting Algorithm

7.6 Experimental Approach and Results

Our novel method of SOAP steganography is empirically tested and validated. Thus, we demonstrate the data embedding and extracting algorithms using an example scenario (actual web service) which has been implemented and examined.

1. *The Book Buyer (Service Requester) selects the books he/she wants to order from the Book Seller website (service Provider).*

2. *The Book Order will be formatted as XML document and then an XML-based SOAP message will be generated in order to be sent to the Service Provider.*

3. *An application is used at the sender (Book Buyer) endpoint in order to capture each SOAP message before it has been sent (prevents the sending process of SOAP).*

4. *The "Embedding Procedure" of our SOAP steganography method (Figure 7.5) is applied on each captured SOAP message.*

5. *The output of the "Embedding procedure" (probably stego SOAP) will be sent to the Book Seller.*

6. *The Book Seller receives the SOAP message (probably stego SOAP) and a similar application to that used at the Book Buyer endpoint (see 3) will be used at the Book Seller endpoint to capture each received SOAP message.*

7. *The "Extracting Procedure" of our SOAP steganography method (Figure 7.7) is applied on each captured SOAP message in order to extract the secret message from the stego SOAP messages.*

Figure 7.9: Real Scenario for SOAP Steganography-Book Order Example

Figure 7.9 illustrates an actual scenario for the proposed SOAP steganography method in a realistic web service (Book Order). In this scenario, we assume that the person who wants to send secret data is the “Book Buyer” while the intended recipient of secret message is the “Book Seller”. However, the opposite scenario is true since the “Book Seller” can send a secret message to the “Book Buyer” using the same procedure.

Book Order (1 in Figure 7.9):

1-Book ‘A’: isbn=1000-11, author=Author_1,
bookName=Book_1, numOfPages=350,
publisher=Publisher_1, year=2009, price=29.99.
2-Book ‘B’: isbn=1000-12, author=Author_2,
bookName=Book_2, numOfPages=420,
publisher=Publisher_2, year=2006, price=44.99.

Cover SOAP (2+3 in Figure 7.9):

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/
envelope/">
   <S:Body>
      <ns2:BookOrder
xmlns:ns2="http://service.bookorder.com/">
      <Book_A>
       <isbn>1000-11</isbn>
       <author>Author_1</author>
       <bookName>Book_1</ bookName >
       <numOfPages>350</numOfPages>
       <publisher>Publisher_1</publisher>
       < year >2009</year>
       <price>29.99</price>
      </Book_A>
      <Book_B>
       <isbn>1000-12</isbn>
       <author>Author_2</author>
       <bookName>Book_2</ bookName >
       <numOfPages>420</numOfPages>
       <publisher>Publisher_2</publisher>
       < year >2006</year>
       <price>44.99</price>
      </Book_B>
      </ns2:BookOrder>
   </S:Body>
</S:Envelope>
```

Stego SOAP (4+5+6+7 in Figure 7.9):

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/
envelope/">
   <S:Body>
      <ns2:BookOrder
xmlns:ns2="http://service.bookorder.com/">
      <Book_A>
        <author>Author_1</author>
        <isbn>1000-11</isbn>
        <bookName>Book_1</ bookName >
        <numOfPages>350</numOfPages>
        <publisher>Publisher_1</publisher>
        < year >2009</year>
        <price>29.99</price>
       </Book_A>
       <Book_B>
         <price>44.99</price>
         < year >2006</year>
         <publisher>Publisher_2</publisher>
         <numOfPages>420</numOfPages>
         <bookName>Book_2</ bookName >
         <author>Author_2</author>
         <isbn>1000-12</isbn>
        </Book_B>
      </ns2:BookOrder>
   </S:Body>
</S:Envelope>
```

Secret Message:
H
Stego Key:
NO EMBEDDING = isbn, author, bookName, numOfPages, publisher, year, price.
Capital “H” Character = author, isbn, bookName, numOfPages, publisher, year, price.
End_of_message Symbol = price, year, publisher, numOfPages, bookName, author, isbn.

Figure 7.10: Information Hiding within “Book Order” SOAP Message

Figure 7.10 illustrates the steps in which this scenario work. it shows the customer's (buyer) order in the top, the secret message (H) to be embedded and sent to the seller along with the stego key (shared between the buyer and the seller) in the bottom, the serialized SOAP message used as steganography cover in the left and the stego SOAP containing the secret message received by the seller in the right of this figure.

Figure 7.11 and Figure 7.12 show two stego SOAP messages where the secret message *"Hello"* is smuggled by shuffling the sub-elements of each "record" element in these messages. The first message contains only part of the hidden message *"Hel"* and "To Continue" symbol (Figure 7.11), while the second message contains the rest of the message *"lo"* and the "End of Message" symbol (Figure 7.12).

Since each element has 5 sub-elements, 5! (120) different cases can be represented. That covers all the alphabetical characters (in small and capital caps), numbers and most of the printing characters. For the purpose of demonstration, we used a shifted version of the ASCII table as a secret key for data hiding. Nevertheless, more complex secret keys can be used in real implementations.

In our experimental design, the steganography method used to embed secret data within SOAP message and then get a stego SOAP was assumed the independent variable. Thus, in order to evaluate the efficiency of our proposed steganography method, three dependent variables were considered: the steganographic capacity, stego SOAP detectability, and the stego SOAP size. Since there are no actual SOAP steganography methods to compare with and because there is not a standard method to measure the undetectability of stego SOAP, we analysed and discussed our dependent variables qualitatively.

To ensure our SOAP steganography method validity and functionality, we have empirically applied this method on the example scenario. Thus, we embedded the secret message manually and then passed the stego SOAP over a parser to see if it can recognise the stego SOAP message or not. As a result, our proposed steganography method worked well without any errors and the secret message was extracted easily using the secret key selected for data embedding.

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
   <S:Body>
      <ns2:receiveOrder xmlns:ns2="http://service.project/">
         <order>
            <b>33.69</b>
            <d>StringD1</d>
            <e>StringE1</e>      b → d → e → c → a : "H"
            <c>StringC1</c>
            <a>75</a>
         </order>
         <order>
            <c>StringC2</c>
            <e>StringE2</e>
            <d>StringD2</d>    c → e → d → a → b : "e"
            <a>410</a>
            <b>16.6</b>
         </order>
         <order>
            <d>StringD3</d>
            <a>76</a>
            <e>StringE3</e>      d → a → e → c → b : "l"
            <c>StringC3</c>
            <b>18.88</b>
         </order>
         <order>
            <e>StringE4</e>
            <a>150</a>
            <b>47.83</b>         e → a → b → c → d :
            <c>StringC4</c>                        To Continue
            <d>StringD4</d>
         </order>
      </ns2:receiveOrder>
   </S:Body>
</S:Envelope>
```

Figure 7.11: Stego SOAP Message 1

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope
xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
<ns2:receiveOrder xmlns:ns2="http://service.project/">
        <order>
          <d>StringD5</d>
          <a>150</a>
          <e>StringE5</e>      d → a → e → c → b : "l"
          <c>StringC5</c>
          <b>47.83</b>
        </order>
        <order>
          <d>StringD6</d>
          <b>99.18</b>
          <c>StringC6</c>      d → b → c → a → e : "o"
          <a>65</a>
          <e>StringE6</e>
        </order>
        <order>
          <e>StringE7</e>
          <d>StringD7</d>
          <c>StringC7</c   >   e → d → c → b → a :
          <b>22.28</b>                    End of Message
          <a>75</a>
        </order>
      </ns2:receiveOrder>
  </S:Body>
</S:Envelope
```

Figure 7.12: Stego SOAP Message 2

In conclusion, our proposed steganography method provides a good solution to hide secret data in SOAP messages. In general, the capacity of text based steganography methods is very low but the capacity of our method is relatively high compared to other methods proposed for XML documents. In terms of functionality and applicability, our method avoids the flaws of many steganography methods since it totally complies with XML and SOAP specifications and features. Thus, it overcomes the limitations of other methods proposed for XML and SOAP steganography since some of these methods are infeasible and inapplicable. Unlike some XML steganography methods, our method does not add extra data to the SOAP message and therefore it does not increase the message size. Additionally, the stego SOAP looks normal and unsuspicious since our method slightly modifies the position of contents in the SOAP. Thus, this steganography method proposed can be considered as an imperceptible and undetectable way of secret communication.

7.7 SOAP Steganography and Security

Security is an ongoing process and as soon as developers fix one set of problems crackers will find yet another way to break these systems. Essentially, the applications must be flexible in order to add new security features as needed. Furthermore, anyone on the Internet can intercept the data transmitted between different sites. Thus, distributed applications require higher security levels than internal applications.

Encryption can be used to preserve data security but the technologies required for encryption cause problems with firewalls and they don’t work very well on the Internet. Encryption has another problem; if both communication parties don’t have the same platform then the receiver can’t decrypt the sender’s message. Thus, even a common encryption scheme usually can only work on a limited number of platforms (Mueller, 2001). As a result, our SOAP based steganography method could be a reasonable solution for transmitted data security. It can be used as a secret communication channel over different kinds of networks regardless of the applications used at the distributed endpoints.

As a kind of communication security, the process of surely knowing the identity of the other communicating party (on the other end of a channel) is known as Authentication. Additionally, associated HTTP Authentication Framework with HTTP 1.1 provides better authentication means between communicating parties. Thus, the HTTP Authentication Framework secures only the authentication portion of the communication. Furthermore, Secure/Multipurpose Internet Mail Extensions (S/MIME) and Secure Socket Layer (SSL) use digital certificates to provide security which relies on the use of public key cryptography. Usually, using static keys provides the crackers more chance to break the system than using dynamic keys (Mueller, 2001).

As a result, we can use the proposed SOAP steganography method to convey information of authentication which necessary to authenticate the communicating parties. Additionally, encryption keys can be embedded and transmitted in order to get dynamic keys instead of static keys, and therefore add another layer of system security.

Basically, encryption algorithms represent a conventional solution of information security but the encrypted data is still there and everyone can observe it over the network. Thus, our SOAP steganography algorithm provides a way of secret communications over the Internet. It can overcome the limitations and challenges of encryption as well as it can be used with encryption to provide a double layer of security.

In conclusion, the proposed SOAP steganography method can be used for a variety of applications such as; authentication, proof of identity, watermarking, digital signature and message hash.

7.8 Conclusion

In this chapter, we have provided a communication protocol-based steganography method that manipulates the SOAP protocol. This method monitors a SOAP message just after its serialization in the sender endpoint and before it is sent. It analyses the SOAP elements and embeds a secret message by rearranging the order of the contents and attributes of specific elements in a SOAP message, where every permutation represents a specific symbol according to a secret key shared between the sender and the receiver.

As a result, the provided method has a high resistance against detection since it uses the communication protocol as a cover medium rather than the traditional digital files. Furthermore, the stego SOAP message has the same size of the original message. The method is tested and validated using an actual scenario so as to demonstrate its utility and applicability.

Chapter 8: Conclusions and Future Research

8.1 Overview

The steganographic capacity and imperceptibility represent the most important aspects of any steganography technique. Thus, this thesis addresses and improves these two fundamental aspects of digital steganography methods: steganographic capacity and stego image quality. This research proposed novel steganography methods in order to increase the steganographic capacity and enhance the imperceptibility (i.e. stego image quality). Additionally, it examined the reliability of PSNR, a principal measure used to evaluate the performance of image based steganography methods through measuring the quality of their stego images.

This chapter summarizes the research conclusions and presents future research directions. It starts by summarizing the research and then shows the research findings. However, this research summary is organised based on the research chapters. Afterwards, the research contributions are discussed. In the end, significant future research avenues that would provide further development to this important area of research are suggested.

8.2 Research Overview

The aim of this research was to increase the steganographic capacity and enhance the stego image quality (imperceptibility). Also, this research aimed to find out a way or method to hide secret data within communication protocols such as SOAP messages. This thesis was organised in eight chapters. This section provides a summary of the previous seven chapters.

Chapter One:

The introduction chapter of this thesis presented and explored the main motivations for conducting this research. The relationship between cryptography, steganography and watermarking was analysed and discussed. Furthermore, the advantages of steganography as an information security technique were shown. The main aspects that have a very important impact on the performance of steganographic systems were identified and explained. Thereafter, the research problem based on the limitations of these fundamental aspects, steganographic capacity and imperceptibility, was formed and established. Depending on this research problem, the research aim was clearly identified and explained. This chapter was ended up by describing the structure of the thesis.

Chapter Two:

The literature review chapter presented an overview of digital steganography basics and explained the different file types that can be used as cover files. The main components of steganographic systems in addition to the procedures of hiding and sending a secret message from one party to another were illustrated. Thereafter, the different kinds of attacks that meet digital steganography were described. This chapter presented the main classification methods of steganography and the key techniques used to embed secret data. Additionally, it provided an explanation of the network steganography techniques. Thus, the basics of SOAP messages and Web Services were explained in this chapter. This chapter also described in-detail the main aspects of steganographic systems: steganographic capacity and imperceptibility. Finally, it explained the main techniques of steganalysis or steganography breaking.

Chapter Three:

The focus of our research was essentially on JPEG based steganography and SOAP based steganography. Hence, this chapter discussed and explained in-depth the basics of DCT-based JPEG compression and JPEG based steganography techniques. Thus, the main procedures of JPEG encoding and decoding were illustrated and the common methods of JPEG based steganography were identified. Then, the impact of the quantisation tables of JPEG on the efficiency of compression was presented. Additionally, this chapter investigated and critically analysed the studies related to five directions: improving JPEG steganographic capacity and stego image quality, enhancing the quality of JPEG images, using chrominance components for steganography, evaluating the quality of stego images and information hiding in text and SOAP messages.

Chapter Four:

It is quite difficult to increase the steganographic capacity and simultaneously maintain the imperceptibility of stego images. Our study aimed to increase the steganographic capacity and enhance the quality of JPEG stego images. This was achieved through two directions: approaches based on JPEG quantisation tables and approaches based on image chrominance components. In this chapter, we explained and discussed the first kind of these approaches and therefore two steganography methods were proposed in this regard. The first method used 16x16 non-overlapping blocks and quantisation table instead of 8x8 to get high steganographic capacity. However, the second method (hybrid method) used optimised JPEG quantisation tables instead of the default tables to get enhanced-quality stego images. Also, this method embedded data using both of our first method and JSteg method to get high steganographic capacity. Consequently, in this chapter, we considered objective 1 and 2 of our research, which concerned the usage of quantisation tables to improve the image steganographic capacity and stego image quality.

Chapter Five:

In this chapter, we investigated the second kind of approached that used to achieve our research aim: improving the steganographic capacity and stego image quality. Firstly, the impact of using grayscale and colour versions of a given cover image was examined in terms of the main steganography requirements. Then, image chrominance components were used for steganography in addition to the luminance component in order to increase the steganographic capacity. Consequently, in this chapter, we considered objective 3 of our research, which concerned the usage of chrominance components of images for steganography in addition to the luminance component to get high steganographic capacity.

Chapter Six:

For security and imperceptibility reasons, it is very important for stego images not to show any detectable artefacts or distortions. Thus, the quality of stego images or imperceptibility represents a key factor of steganographic systems.

Generally, there are two primary ways to measure image quality; objective quality methods (e.g. PSNR) and subjective quality methods (human-based). In this chapter, we examined and evaluated the reliability of PSNR as an objective quality measure used to evaluate the quality of stego images. Besides, the relationship between the PSNR and subjective quality (MOS) was tested and discussed. Therefore, this chapter considered objective 4 of our research, which concerned the quality evaluation of many JPEG stego images using a subjective quality evaluation method.

Chapter Seven:

In this chapter, we considered objective 5 of our study, which concerned information hiding within SOAP messages in an undetectable way. For that reason, we proposed a novel steganography algorithm to be used with SOAP messages. This method embeds secret data in SOAP protocol by rearranging the order of the contents and attributes (sub-elements) of specific elements in a SOAP message. Additionally, the effects of this method on the SOAP message functionality were analysed and discussed. Also, this method was experimentally tested and evaluated.

8.3 Research Findings

This research set out to meet a number of objectives described in Chapter 3, which were accomplished as follows:

- **Objective 1: Increasing the capacity of JPEG steganography using 16x16 quantisation tables**.

We proposed a steganography method based upon JPEG compression and DCT transformation. Furthermore, non-overlapping blocks of 16x16 pixels and a modified 16x16 quantisation table were utilised in this method. The 2-LSBs of each quantised (middle frequency) DCT coefficient was modified to embed two secret bits. As a result, our steganography method provided larger steganographic capacity than other common JPEG steganography methods tested; JSteg, F5 and JMQT. Generally, the stego image quality of our method was good and acceptable. It was better than that of JSteg and F5 but roughly similar to that of JMQT method. The size of the stego images of our method was slightly larger than that of JMQT method. Moreover, our method did not increase the computational time even though it uses 16x16 non-overlapping blocks and quantisation table. The computational time of our method was almost half of that of other methods which use 8x8 blocks and quantisation tables: JSteg and JMQT.

- **Objective 2: Enhancing the quality of JPEG stego images and increasing the steganographic capacity using optimised quantisation tables.**

We suggested using optimised JPEG quantisation tables instead of the default tables in order to improve the quality of stego images. Thus, the quality of all stego images was enhanced significantly by using optimised quantisation tables. The capacity of JSteg and F5 was increased, and the size of all stego images was slightly increased by using optimised quantisation tables. Thus, we took advantage of this quality improvement of stego images to embed more secret data in these images. Accordingly, we proposed a hybrid steganography method based on our first method, the JSteg method, and optimised quantisation tables. The steganographic capacity of this hybrid method was larger than that of other methods tested (including our first method). Also, the quality of stego images of our hybrid method was better than that of other methods which use default JPEG quantisation tables.

- **Objective 3: Investigating the impact of using chrominance components for steganography.**

Firstly, we examined the performance of both grayscale and colour versions of test images when used as steganography cover images. Thus, the same amount of data was embedded in both grayscale and colour versions of a given cover image using a given steganography method. Then, the other steganography aspects (stego image quality and size) were measured. In conclusion, using colour images was better than using grayscale images for data hiding since using colour images for steganography provided better stego image quality and smaller stego image size than using grayscale versions of the same images. Secondly, we investigated the impact of using each single component of cover image for data hiding. Hence, Cb and Cr were used for data hiding and they were more efficient than (Y) for small steganographic capacity. Thereafter, we used all colour image components for data hiding. As a result, steganographic capacity was significantly increased and the quality of stego images were quite good (PSNR>35 db). Finally, the steganography method or technique used to embed secret data in the luminance component had the most effect on the properties of the resultant stego image.

- **Objective 4: Evaluating the reliability of PSNR as a quality measure of stego image.**

We evaluated the reliability and consistency of PSNR as a measure of stego image quality. Thus, an adapted double stimulus continuous quality scale (DSCQS) method was used to evaluate the quality of stego images. Thus, we examined the relationship between the PSNR and the subjective quality (MOS) of different stego images. As a result, there was a poor correlation between PSNR and MOS and therefore the PSNR can not be reliably used to measure the quality of stego images. Additionally, conclusions derived from PSNR values were different from those derived from MOS values by comparing different steganography methods (in terms of their stego image quality). Also, PSNR and MOS produced different conclusions when we evaluated the impact of using all components of colour images (luminance and chrominance) for steganography. However, when hiding the same data in both grayscale and colour versions of a given image using a particular steganography method, we obtained almost the same conclusion from both PSNR and MOS values.

- **Objective 5: Examining the capability of using SOAP message for steganography.**

We proposed a novel steganography method based on SOAP message. This method embeds secret data in a SOAP message just after its serialization in the sender endpoint and before sending. The embedding procedure was based on rearranging the order of the contents and attributes of specific elements in a SOAP message. Hence, every permutation represents a specific symbol according to a secret key shared between the sender and the receiver. As a result, the provided method was highly resistant against detection since it used the communication protocol as a cover medium rather than the traditional digital files. Additionally, the size of stego SOAP messages was the same size as the original message.

Besides, this method had a high imperceptibility since no trail was left and the structure and size of the SOAP message were kept intact. This method can use many SOAP messages in order to hide the whole message and therefore overcome the capacity matter. A secret message was split into many sub-messages and then embedded in different SOAP messages. Finally, this method had no effects on the SOAP message functionality and is therefore a secure method of information hiding.

8.4 Research Contributions

This thesis adds value to research and practice communities concerned with data hiding, image steganography, watermarking, and JPEG compression in addition to those interested in Web services and SOAP based steganography. The novel integration of these relevant research domains also enhanced the value of contributions made in this research. The contributions of this thesis are discussed below under four different but closely related areas.

8.4.1 Increased JPEG Steganographic Capacity

The steganographic capacity represents one of the two key requirements of JPEG steganography, and all steganographic systems in general. Furthermore, increasing the steganographic capacity of JPEG steganography while maintaining the stego image quality (good stego image quality) is considered as a significant contribution.

In the first steganography method proposed, the steganographic capacity was increased by using 16x16 non-overlapping blocks and quantisation table. Thus, using larger blocks and quantisation tables provided more quantised middle-frequency DCT coefficients which were used for data hiding. This increase in the steganographic capacity had no effect on the quality of stego images since the energy of images are concentrated in the low-frequency coefficients.

In the hybrid steganography method proposed, the steganographic capacity was increased by using two different steganography techniques for both low and middle frequency coefficients. The JSteg method was used for data hiding in quantised low frequency coefficients while our first method was used for data hiding in the quantised middle frequency coefficients. Additionally, in this hybrid method, the quality of stego images was maintained by using optimised quantisation tables instead of the default JPEG tables.

Some steganography methods have a fixed capacity for a given cover image size (e.g. JMQT). Thus, such methods have the same capacity regardless the quantisation table used. However, the steganographic capacity of steganography methods that have non-fixed capacity (e.g. JSteg and F5) was increased when optimised quantisation tables were used instead of the default JPEG quantisation tables.

Generally, the luminance component of images is used for data hiding. As a way of increasing the steganographic capacity of digital images, chrominance components of colour images were used for data hiding in addition to the luminance component. Thus, depending on the size of secret message, some or all image components can be used as cover media of steganography.

8.4.2 Enhanced Imperceptibility of JPEG Steganography

The two most important aspects of any image based steganographic system are the quality of the stego image and the capacity. Basically, enhancing the stego image quality (imperceptibility) while maintaining the steganographic capacity also represents a contribution.

Since the quantisation table is not a part of the JPEG standard, it has been suggested in this research to use optimised quantisation tables instead of the default JPEG tables. This has been proposed to enhance and improve the quality of stego images. It has been proved in this research that the quality of all stego images that use optimised quantisation tables is better than the quality of stego images that use default JPEG quantisation tables.

Our hybrid steganography method did not use the quantised DC coefficients for data hiding in order to maintain the stego image quality. For the same reason, it used the JSteg method for quantised low frequency coefficients since JSteg slightly modifies the LSB of these coefficients without having a significant affect on the stego image quality.

It has been proved that using a colour version of a given cover image to embed a specific amount of secret data provides better stego images in terms of quality than using the grayscale version of the same image. Thus, using colour images as steganography cover images enhanced, or degraded less, the quality of stego images when compared to grayscale images.

8.4.3 Proved Unreliability of PSNR for Stego Images

Measuring the quality of digital images represents a significant issue in many image processing applications. Mostly, PSNR and MSE are used to evaluate the quality of stego images since they are easy, fast and cost-effective methods. However, JPEG steganography adds another type of distortion to stego images in addition to that added by image compression.

This research analysed and evaluated the relationship between PSNR and subjective quality (MOS), which represent the most reliable measure of image quality, as quality measures of JPEG stego images. Thus, it examined the reliability of PSNR as a measure of stego images quality and therefore a measure of steganography methods performance.

There is a poor positive correlation between PSNR and subjective evaluation (MOS) and these two measures may provide different results. Thus, PSNR can't be used as a reliable and definitive quality measure for stego images. Moreover, evaluating steganography methods by measuring the quality of their stego images using PSNR metric resulted in different conclusions than using MOS.

8.4.4 Proposed Secure SOAP Steganography Method

The feasible steganography methods that can be used with Web services and specifically SOAP messages are very few and therefore, there is a need for such methods to be used for different applications such as watermarking and authentication. Thus, designing or finding out a secure steganography method for SOAP messages is a very important issue.

In our SOAP based steganography method, the order of contents or sub-elements of SOAP main elements was rearranged to embed the secret message. The imperceptibility of this method is very high and it is undetectable since it leaves almost no trail and keeps the structure and size of the SOAP message intact. Additionally, this method overcomes the capacity problem by using more than one SOAP message to embed a secret message.

Using SOAP messages as cover media of steganography is still in its infancy. As a result, this research provides a secure data hiding method to be used with SOAP messages. Additionally, the steganographic capacity of this method is very good compared to the other steganography methods based on text or XML files.

8.5 Research Limitations

This research has investigated how main aspects of JPEG steganography could be improved. In addition to the important contributions made in this research highlighted in the previous section, this research has some limitations.

The main focus of this research was on JPEG based steganography. Experimental design was used as a methodology of our research. Additionally, only five similar-size images were used as cover images in the experiments. Using a larger number of different-size cover images in the experiments may enhance the results consistency.

Our first and hybrid steganography methods proposed use 16x16 non-overlapping blocks and quantisation tables. However, the stego images of these two methods can not be viewed normally since they do not comply with the JPEG standard (8x8 blocks). Thus, this presents a limitation to the usefulness of our steganography methods that use 16x16 blocks and quantisation table. In chapter 3, some studies investigated the advantages of using larger or adaptive-size blocks and quantisation tables on the performance of JPEG compression. Therefore, our work investigated the impact of the larger blocks on the performance of JPEG steganography. Thus, it provides a suggestion and a valid direction to improve JPEG steganography aspects. Additionally, this research along with other studies in this direction may help and lead to propose or amend the available standard to include larger blocks and quantisation tables as optional choice of user.

Additionally, this research could analyse and evaluate the impact of using other large non-overlapping blocks and quantisation tables (e.g. 32x32) on the steganography performance. In this research, only one method of quantisation table optimisation was used to enhance the quality of stego images and therefore the imperceptibility of JPEG steganography. Since the majority of optimisation methods were designed for 8x8 quantisation tables, the impact of such methods on the quality of stego images could be examined and evaluated.

This study has also investigated how secret data could be embedded in SOAP messages. Thus, a novel steganography method was proposed in this regard. However, this method can not be used with all SOAP messages but it needs cover

SOAP messages with special requirements (SOAP elements have sub-elements to be rearranged).

As this research focused on improving both steganographic capacity and imperceptibility, it considered the visual attacks only (stego image quality). Like the majority of steganography methods proposed in the literature, it considered the visual quality of stego images as an indication of their security. Thus, this research did not consider other kinds of possible attacks or steganalysis methods that might detect or compromise the proposed steganography methods.

8.6 Future Research

In addition to the limitations of this research described in the previous section, this research highlights and provides many important directions for future research.

The JPEG standard uses only 8x8 blocks and quantisation tables. Thus, in order to improve the main aspects of steganography, using larger blocks and quantisation tables can be investigated with the DWT (discrete wavelet transform) based JPEG2000 steganography.

This research used an optimisation method for JPEG quantisation tables in order to enhance the quality of stego images. Indeed, designing optimisation methods for quantisation tables that take into consideration the aspects and features of a given steganography method represent another direction of future research.

Additionally, it has been noticed that the contents of cover images may affect the quality of its stego image. So, it is quite a good idea to investigate the relationship between the characteristics and contents of cover images on the one side and the quality of stego images and the steganographic capacity on the other.

Generally, objective image quality measures are easy, fast and cost-effective compared to subjective measures. Furthermore, PSNR is the most common measure used to evaluate the quality of stego images. However, this measure is not reliable to be used with stego images. Thus, designing or finding out an objective image quality measure that can predict the perceived quality and provide reliable results with stego images still represent a challenge. Additionally, the reliability of other available methods of objective measures can be tested and examined with stego images.

Determining the key aspects of steganography, measuring method of each aspect or attribute, and evaluating the performance of steganography methods are all still suffering from a lack of standardisation. Thus, the performance of a given steganography method is evaluated by comparing the values of its main aspects with that of other methods. If a given steganography method is better than the other in terms of only one aspect, how we can decide which method is better than the other? Thus, using a decision making algorithm that can consider each aspect of steganography represent an important direction of future research also.

References

Acharya, T. & Tsai, P.-S. (2005) *JPEG2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures*, John Wiley & Sons, Inc., Hoboken, New Jersey.

Agaian, S. S., Cherukuri, R., Schneider, E. C. & White, G. B. (2006) A New JPEG-Based Steganographic Algorithm for Mobile Devices. *Mobile Multimedia/Image Processing for Military and Security Applications,* 6250**,** 62500F1-62500F11.

Almohammad, A., Ghinea, G. & Hierons, R. M. (2009) JPEG steganography: a performance evaluation of quantization tables. *International Conference on Advanced Information Networking and Applications, AINA '09***,** 471-478.

Alturki, F. & Mersereau, R. (2001) A Novel Approach for Increasing Security and Data Embedding Capacity in Images for Data Hiding Applications. *Proceedings of International Conference on Information Technology: Coding and Computing***,** 228-233.

Anderson, R. J. & Petitcolas, F. A. P. (1998) On the Limits of Steganography. *IEEE Journal on Selected Areas in Communications,* 16**,** 474-481.

Artz, D. (2001) Digital Steganography: Hiding Data within Data. *IEEE Internet Computing,* 5**,** 75-80.

Baba, S., Krekor, L., Arif, T. & Shaaban, Z. (2009) Watermarking scheme for copyright of digital images. *International Journal of Computer Science and Network Security,* 9**,** 1-9.

Bailey, K. & Curran, K. (2006) An Evaluation of Image Based Steganography Methods Using Visual Inspection and Automated Detection Techniques. *Multimedia Tools and Applications,* 31**,** 55-88.

Bailey, K., Curran, K. & Condell, J. (2004) Evaluation of Pixel-Based Steganography and Stegodetection Methods. *The Imaging Science Journal,* 52**,** 131-150.

Baroncini, V. (2006) New Tendencies in Subjective Video Quality Evaluation. *IEICE Trans Fundam Electron Commun Comput Sci (Inst Electron Inf Commun Eng),* E89-A**,** 2933-2937.

Bender, W., Gruhl, D., Morimoto, N. & Lu, A. (1996) Techniques for data hiding. *IBM Systems Journal,* 35**,** 313-336.

Bethel, D. M., Monro, D. M. & Sherlock, B. G. (1997) Optimal Quantisation of the Discrete Cosine Transform for Image Compression. *The Sixth International Conference on Image Processing and Its Applications,* 1**,** 69-72.

Bracamonte, J., Ansorge, M. & Pellandini, F. (1997) Adaptive Block-Size Transform Coding for Image Compression. *IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP-97,* 4**,** 2721-2724.

Cachin, C. (1998) An Information-Theoretic Model for Steganography. *The Second International Workshop on Information Hiding, IH'98,* 1525**,** 306-318.

Chang, C.-C., Chen, T.-S. & Chung, L.-Z. (2002) A Steganographic Method Based Upon JPEG and Quantization Table Modification. *Information Sciences,* 141**,** 123-138.

Chang, C.-C., Chen, Y.-H. & Lin, C.-C. (2009) A data embedding scheme for color images based on genetic algorithm and absolute moment block truncation coding. *Soft Computing - A Fusion of Foundations, Methodologies and Applications,* 13**,** 321-331.

Chang, C.-C., Lin, C.-C., Tseng, C.-S. & Tai, W.-L. (2007) Reversible Hiding in DCT-Based Compressed Images. *Information Sciences,* 177**,** 2768-2786.

Chang, C.-C., Lin, C.-Y. & Wang, Y.-Z. (2006) New Image Steganographic Methods Using Run-Length Approach. *Information Sciences,* 176**,** 3393-3408.

Chang, C.-C. & Tseng, H.-W. (2004) A Steganographic Method for Digital Images Using Side Match. *Pattern Recognition Letters,* 25**,** 1431-1437.

Chang, L.-W., Wang, C.-Y. & Lee, S.-M. (1999) Designing JPEG Quantization Tables Based on Human Visual System. *The International Conference on mage Processing. ICIP 99,* 2**,** 376-380.

Cherukuri, R. C. & Agaian, S. S. (2007) Switching Theory-Based Steganographic System for JPEG Images. *Mobile Multimedia/Image Processing for Military and Security Applications,* 6579**,** 65790C1-65790C12.

Chu, R., You, X., Kong, X. & Ba, X. (2004) A DCT-Based Image Steganographic Method Resisting Statistical Attacks. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '04,* 5**,** V953-V956.

Cole, E. (2003) *Hiding in Plain Sight: Steganography and the Art of Covert Communication,* Indiana, John Wiley & Sons Inc.

Coolican, H. (1994) *Research methods and statistics in psychology. (2nd edition),* Hodder and Stoughton.

Costa, L. F. & Veiga, A. C. P. (2005) Identification of the Best Quantization Table Using Genetic Algorithms. *2005 IEEE Pacific Rim Conference on Communications, Computers and signal Processing. PACRIM***,** 570-573.

Cox, I. J., Miller, M. L., Bloom, J. A., Fridrich, J. & Kalker, T. (2008) *Digital Watermarking and Steganography-Second Edition,* Burlington, MA, USA, Elsevier Inc.

Crouse, M. & Ramchandran, K. (1995) Joint Thresholding and Quantizer Selection for Decoder-Compatible Baseline JPEG. *International Conference on Acoustics, Speech, and Signal Processing. ICASSP-95,* 4**,** 2331-2334.

Fard, A. M., Akbarzadeh-T, M.-R. & Varasteh-a, F. (2006) A New Genetic Algorithm Approach for Secure JPEG Steganography. *2006 IEEE International Conference on Engineering of Intelligent Systems, ICEIS'2006***,** 1-6.

Fridrich, J., Goljan, M. & Du, R. (2001) Detecting LSB Steganography in Color and Gray-Scale Images. *IEEE Multimedia and Security,* 8**,** 22-28.

Fridrich, J., Goljan, M. & Hogea, D. (2002) Attacking the OutGuess. *The ACM Workshop on Multimedia and Security 2002*.

Garg, M. (2008) Performance Analysis of Chrominance Red & Chrominance Blue in JPEG. *World Academy of Science, Engineering and Technology, WASET,* 43**,** 110-113.

Grgic, S., Grgic, M. & Mrak, M. (2004) RELIABILITY OF OBJECTIVE PICTURE QUALITY MEASURES. *Journal of ELECTRICAL ENGINEERING,* 55**,** 3-10.

Hamamoto, K. (1999) Standardization of JPEG Quantization Table for Medical Ultrasonic Echo Images. *The 6th IEEE International Conference on Electronics, Circuits and Systems. ICECS '99,* 2**,** 683-686.

Hinkelmann, K. & Kempthorne, O. (2008) *Design and Analysis of Experiments: Introduction to Experimental Design,* John Wiley & Sons, Inc., Hoboken, New Jersey.

Hirata, K., Yamada, J.-I. & Shinjo, K. (1999) Applying the Hamiltonian Algorithm to Optimize JPEG Quantization Tables. *Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts IV,* 3648**,** 344-351.

Inoue, S., Makino, K., Murase, I., Takizawa, O., Matsumoto, T. & Nakagawa, H. (2001) A Proposal on Information Hiding Methods using XML. *The First NLP and XML Workshop*.

Iso-Dis (1992) Digital Compression and Coding of Continuous-Tone Still Images - Requirements and Guidelines. *CCITT Recommendation T.81.*

Itu-R-Bt.500-11 (2002) Methodology for the subjective assessment of the quality of television pictures. International Telecommunication Union/ITU Radiocommunication Sector.

Johnson, N. F. & Jajodia, S. (1998) Steganalysis: the investigation of hidden information. *IEEE Information Technology Conference***,** 113-116.

Kahn, D. (1996) The History of Steganography. *The First Workshop on Information Hiding,* 1174**,** 1-5.

Katzenbeisser, S. & Petitcolas, F. A. P. (2000) *Information Hiding Techniques for Steganography and Digital Watermarking,* Artech House.

Kharrazi, M., Sencar, H. T. & Memon, N. (2006) Performance study of common image steganography and steganalysis techniques. *Journal of Electronic Imaging,* 15**,** 1-16.

Kipper, G. (2004) *Investigator's Guide to Steganography,* Florida, CRC Press LLC.

Kong, X., Chu, R., Ba, X., Zhang, T. & Yang, D. (2003) A Perception Evaluation Scheme for Steganography. IN LIU, J., CHEUNG, Y. & HUJUNYIN (Eds.) *Intelligent Data Engineering and Automated Learning.* Springer Berlin / Heidelberg, LNCS, 2690.

Lee, Y.-K. & Chen, L.-H. (2003) Secure Error-Free Steganography for JPEG Images. *International Journal of Pattern Recognition and Artificial Intelligence,* 17**,** 967-981.

Lee, Y. K. & Chen, L.-H. (2000) High Capacity Image Steganographic Model. *IEE Proceedings of Vision, Image and Signal Processing,* 147**,** 288-294.

Li, Q., Yu, C. & Chu, D. (2006) A Robust Image Hiding Method Based on Sign Embedding and Fuzzy Classification. *The Sixth World Congress on Intelligent Control and Automation, WCICA 2006,* 2**,** 10050-10053.

Li, X. & Wang, J. (2007) A Steganographic Method Based Upon JPEG and Particle Swarm Optimization Algorithm. *Information Sciences,* 177**,** 3099-3109.

Liu, C.-L. & Liao, S.-R. (2008) High-Performance JPEG Steganography Using Complementary Embedding Strategy. *Pattern Recognition,* 41**,** 2945-2955.

Liu, M., Guo, Y. & Zhou, L. (2009) Text Steganography Based on Online Chat. *The Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing***,** 807-810

Lou, D.-C. & Liu, J.-L. (2002) Steganographic Method for Secure Communications. *Computers and Security,* 21**,** 449-460.

Lubacz, J., Mazurczyk, W. & Szczypiorski, K. (2010) Vice Over IP. *IEEE Spectrum.*

Marini, E., Autrusseau, F., Callet, P. L. & Campisi, P. (2007) Evaluation of standard watermarking techniques. *Security, Steganography, and Watermarking of Multimedia Contents IX,* 6505**,** 1-10.

Marvel, L. M., Jr, C. G. B. & Retter, C. T. (1998) Reliable Blind Information Hiding for Images. *Second International Workshop on Information Hiding, IH'98,* 1525**,** 48-61.

Mastronardi, G., Castellano, M. & Marino, F. (2001) Steganography effects in various formats of images. A preliminary study. *International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications***,** 116-119.

Memon, A. G., Khawaja, S. & Shah, A. (2008) Steganography: A New Horizon for Safe Communication Through XML. *Journal of Theoretical and Applied Information Technology,* 4**,** 187-202.

Miano, J. (1999) *Compressed Image File Formats: JPEG, PNG, GIF, XBM, BMP,* Addison-Wesley, Longman, Inc.

Monro, D. M. & Sherlock, B. G. (1996) Optimal Quantisation Strategy for DCT Image Compression. *IEE Proceedings of Vision, Image and Signal Processing,* 143**,** 10-14.

Mueller, J. P. (2001) *Special Edition Using SOAP,* USA, QUE.

Munirajan, V. K., Cole, E. & Ring, S. (2004) Transform Domain Steganography Detection Using Fuzzy Inference Systems. *The IEEE Sixth International Symposium on Multimedia Software Engineering***,** 286-291.

Newcomer, E. (2002) *Understanding Web Services: XML, WSDL, SOAP and UDDI,* Addison Wesley.

Noda, H., Niimi, M. & Kawaguchi, E. (2006) High-Performance JPEG Steganography Using Quantization Index Modulation in DCT Domain. *Pattern Recognition Letters,* 27**,** 455-461.

Nyman, G., Radun, J., Leisti, T., Oja, J., Ojanen, H., Olives, J.-L., Vuori, T. & Häkkinen, J. (2006) What do users really perceive - probing the subjective image quality. *Image Quality and System Performance III,* 6059-605902**,** 1-7.

Ogihara, T., Nakamura, D. & Yokoya, N. (1996) Data embedding into pictorial images with less distortion usingdiscrete cosine transform. *Proceedings of the 13th International Conference on Publication Pattern Recognition,* 2**,** 675-679.

Pinson, M. H. & Wolf, S. (2003) Comparing subjective video quality testing methodologies *Visual Communications and Image Processing,* 5150**,** 573-582.

Ponomarenko, N., Lukin, V., Egiazarian, K., Astola, J., Carli, M. & Battisti, F. (2008) Color image database for evaluation of image quality metrics. *IEEE 10th Workshop on Multimedia Signal Processing***,** 403-408.

Por, L. Y. & Delina, B. (2008) Information Hiding: A New Approach in Text Steganography. *7th WSEAS International Conference on APPLIED COMPUTER & APPLIED COMPUTATIONAL SCIENCE (ACACOS '08)***,** 689-695.

Provos, N. (2001) Defending Against Statistical Steganalysis. *The 10th conference on USENIX Security Symposium,* 10.

Provos, N. & Honeyman, P. (2003) Hide and Seek: An Introduction to Steganography. *IEEE Security & Privacy Magazine,* 1**,** 32-44.

Rabah, K. (2004) Steganography- The Art of Hiding Data. *Information Technology Journal,* 3**,** 245-269.

Rongrong, J., Hongxun, Y., Shaohui, L., Liang, W. & Jianchao, S. (2006) A New Steganalysis Method for Adaptive Spread Spectrum Steganography. *The International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP '06***,** 365-368.

Sallee, P. (2003) Model Based Steganography. *The Second International Workshop on Digital Watermarking, IWDW 2003,* 2939**,** 154-167.

Seitz, J. (2005) *Digital Watermarking for Digital Media,* Information Science Publishing (an imprint of Idea Group Inc.).

Sheikh, H. R., Sabir, M. F. & Bovik, A. C. (2006) A Statistical Evaluation of Recent Full Reference Image Quality Assessment Algorithms. *IEEE Transactions on Image Processing,* 15**,** 3440-3451.

Sherlock, B. G., Nagpal, A. & Monro, D. M. (1994) A Model for JPEG Quantization. *The International Symposium on Speech, Image Processing and Neural Networks, 1994, ISSIPNN '94,* 1**,** 176-179.

Shirali-Shahreza, M. (2008) Text Steganography by Changing Words Spelling. *10th International Conference on Advanced Communication Technology***,** 1912-1913.

Shirali-Shahreza, M. H. & Shirali-Shahreza, M. (2008) A New Synonym Text Steganography. *The Fourth International Conference on Intelligent Information Hiding and Multimedia Signal Processing***,** 1524-1526.

Shohdohji, T., Hoshino, Y. & Kutsuwada, N. (1999) Optimization of Quantization Table Based on Visual Characteristics in DCT Image Coding. *Computers & Mathematics with Applications,* 37**,** 225-232.

Shuttleworth, M. (2008) Experiment Resources. Accessed: July 13, 2010. URL: http://www.experiment-resources.com.

Simmons, G. J. (1983) The Prisoners’ Problem and the Subliminal Channel. *CRYPTO’83***,** 51-67.

Simone, F. D., Goldmann, L., Baroncini, V. & Ebrahimi, T. (2009) Subjective evaluation of JPEG XR image compression *Applications of Digital Image Processing XXXII,* 7443-744301**,** 1-12.

Stanescu, D., Stratulat, M., Groza, V., Ghergulescu, I. & Borca, D. (2007) Steganography in YUV color space. *The International Workshop of Robotic and Sensors Environments. ROSE 2007***,** 1-4.

Stoica, A., Vertan, C. & Fernandez-Maloigne, C. (2003) Objective and subjective color image quality evaluation for JPEG 2000 compressed images. *International Symposium on Signals, Circuits and Systems, SCS 2003,* 1**,** 137-140.

Tan, K. T., Ghanbari, M. & Pearson, D. E. (1998) An objective measurement tool for MPEG video quality. *Signal Processing,* 70**,** 279-294.

Toutant, J.-L., Puech, W. & Fiorio, C. (2006) Minimizing Data-Hiding Noise in Color JPEG Images by Adapting the Quantization. *The Third European Conference on Color in Graphics, Imaging and Vision, CGIV 2006***,** 387-391.

Tseng, H.-W. & Chang, C.-C. (2004) Steganography Using JPEG-Compressed Images. *The Fourth International Conference on Computer and Information Technology, CIT '04*, 12-17.

Upham, D. JPEG-Jsteg-v4, <http://www.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz>.

Venkatraman, S., Abraham, A. & Paprzycki, M. (2004) Significance of Steganography on Data Security. *The International Conference on Information Technology: Coding and Computing. ITCC 2004*, 2**,** 347-351.

Wallace, G. K. (1991) The JPEG Still Picture Compression Standard. *Communications of the ACM*, 34**,** 30-44.

Wang, H. & Wang, S. (2004) Cyber Warfare: Steganography vs. Steganalysis. *Communications of The ACM*, 47**,** 76-82.

Wang, R.-Z. & Chen, Y.-S. (2006) High-Payload Image Steganography Using Two-Way Block Matching. *IEEE Signal Processing Letters*, 13**,** 161-164.

Wang, Z., Bovik, A. C. & Lu, L. (2002a) Why is image quality assessment so difficult? *IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP '02)*, 4**,** 3313-3316.

Wang, Z., Sheikh, H. R. & Bovik, A. C. (2002b) No-reference perceptual quality assessment of JPEG compressed images. *Proceedings of the International Conference on Image Processing*, 1**,** 477-480.

Wang, Z., Sheikh, H. R. & Bovik, A. C. (2003) Objective Video Quality Assessment. *The Handbook of Video Databases: Design and Applications.* CRC Press.

Watters, P. A., Martin, F. & Stripf, S. H. (2005) Visual Steganalysis of LSB-Encoded Natural Images. *The Third International Conference on Information Technology and Applications, ICITA 2005*, 1**,** 746-751.

Wayner, P. (2002) *Disappearing Cryptography, Information Hiding: Steganography & Watermarking-Second Edition*, San Francisco, CA, USA, Elsevier Science.

Westfeld, A. (2001a) F5-A Steganographic Algorithm: High Capacity Despite Better Steganalysis. *The 4th International Workshop on Information Hiding, IH 2001*, 2137**,** 289-302.

Westfeld, A. (2001b) <http://os.inf.tu-dresden.de/~westfeld/publikationen/f5r11.zip>.

Wong, P. H. W. & Wong, J. W. C. (2001) A Data Hiding Technique in JPEG Compressed Domain. *The SPIE Conference on Security and Watermarking of Multimedia Contents III,* 4314**,** 309-340.

Wu, H. R. & Rao, K. R. (2006) *Digital Video Image Quality and Perceptual Coding,* CRC Press.

Wu, N.-I. & Hwang, M.-S. (2007) Data Hiding: Current Status and Key Issues. *International Journal of Network Security,* 4**,** 1-9.

Wu, S.-W. & Gersho, A. (1993) Rate-Constrained Picture-Adaptive Quantization for JPEG Baseline Coders. *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-93,* 5**,** 389-392.

Yildiz, Y. O., Panetta, K. & Agaian, S. (2007) New Quantization Matrices for JPEG Steganography. *Mobile Multimedia/Image Processing for Military and Security Applications,* 6579**,** 65790D1-65790D11.

Yu, L., Zhao, Y., Ni, R. & Zhu, Z. (2008) PM1 steganography in JPEG images using genetic algorithm. *Soft Computing - A Fusion of Foundations, Methodologies and Applications,* 13**,** 393-400.

Yu, Y.-H., Chang, C.-C. & Hu, Y.-C. (2005) Hiding Secret Data in Images via Predictive Coding. *Pattern Recognition,* 38**,** 691-705.

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,* 107**,** 183-194.

Zeng, W., Yu, H. & Lin, C.-Y. (2006) *Multimedia Security Technologies for Digital Rights Managment,* Elsevier Inc.

Zhang, X., Wang, H. & Sun, J. (2007) An Information Hiding Method based on SOAP. *The Third International Conference on International Information Hiding and Multimedia Signal Processing (IIH-MSP 2007),* 01**,** 453-456.

Zhang, X. & Wang, S. (2005) Steganography Using Multiple-Base Notational System and Human Vision Sensitivity. *IEEE Signal Processing Letters,* 12**,** 67-70.