**SUDAN UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**COLLEGE OF GRADUATE STUDIES**

**COLLEGE OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

# Multi-Level Image Steganography by Using Pixel Intensity

# إخفاء المعلومات متعدد المستويات في الصورة بإستخدام كثافة البسكل

**February 2015**

SUDAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

COLLEGE OF GRADUATE STUDIES

COLLEGE OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

# Multi-Level Image Steganography by Using Pixel Intensity

# إخفاء المعلومات متعدد المستويات في الصورة بإستخدام كثافة البسكل

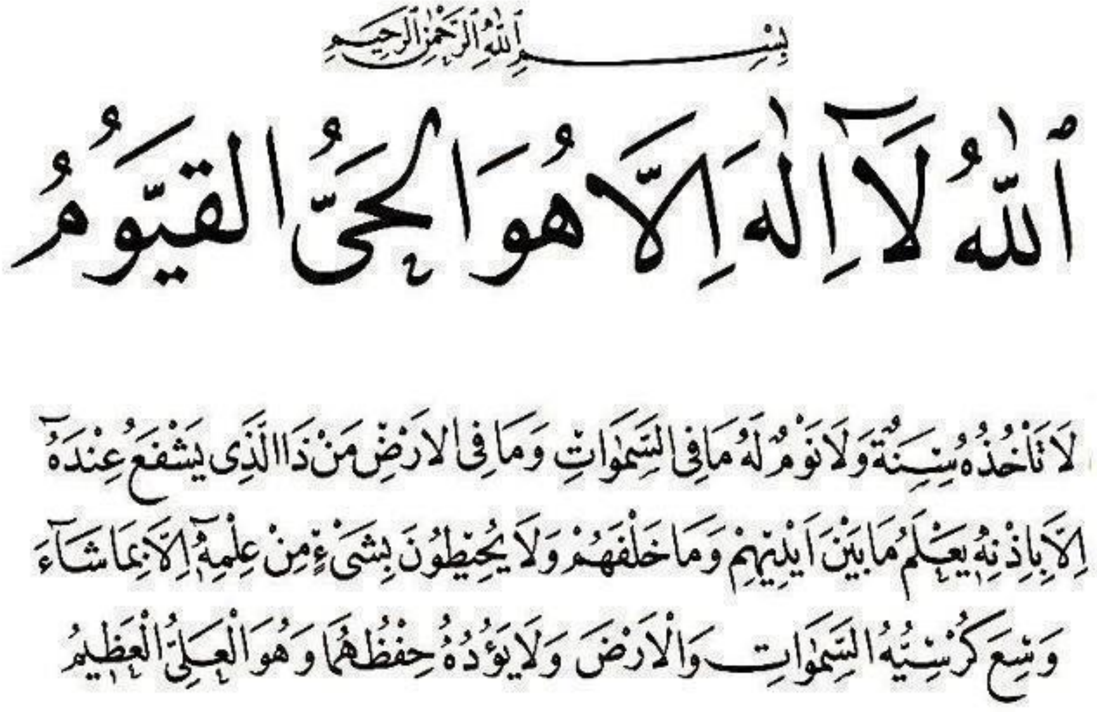
**A Thesis Submitted in Partial Fulfillment of the Requirements of Master Degree in Computer Science**

BY:
HUSSEIN ABDELLATIEF HUSSEIN

Supervisor:
DR. TALAAT WAHBY

**February 2015**

# آیـــــــــــــــــــــه



بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

اَللّٰهُ لَآ اِلٰهَ اِلَّا هُوَ اَلْحَیُّ الْقَیُّوْمُ

لَا تَاْخُذُهٗ سِنَةٌ وَّلَا نَوْمٌ لَهٗ مَا فِی السَّمٰوٰتِ وَمَا فِی الْاَرْضِ مَنْ ذَا الَّذِیْ یَشْفَعُ عِنْدَهٗٓ

اِلَّا بِاِذْنِهٖ یَعْلَمُ مَا بَیْنَ اَیْدِیْهِمْ وَمَا خَلْفَهُمْ وَلَا یُحِیْطُوْنَ بِشَیْءٍ مِّنْ عِلْمِهٖٓ اِلَّا بِمَا شَآءَ

وَسِعَ كُرْسِیُّهُ السَّمٰوٰتِ وَالْاَرْضَ وَلَا یَئُوْدُهٗ حِفْظُهُمَا وَهُوَ الْعَلِیُّ الْعَظِیْمُ

# الحمد

الحمد لله اللهم ربنا لكـ الحمد بما خلقتنا ورزقتنا وهديتنا وعلمتنا وأنقذتنا وفرجت عنا ،لك الحمد بالإيمان ولك الحمد بالإسلام ولك الحمد بالقرآن ولك الحمد بالأهل والمال والمعافاة ، اللهم لك الحمد بكل نعمة أنعمت بها علينا في قديم أو حديث أو سر أو علانية أو خاصة أو عامة أو حي أو ميت أو شاهد أو غائب.

نحمد الله تبارك وتعالى ان تفضل علينا بأنـ زودنا بأدواتـ العلمـ من السمعـ والبصر والفؤاد فعلمنا مالم نكن نعلم وزادنا من العلم بسطة بفضله مما أعاننا على إخراجـ هذا البحث ، لكـ الحمد حتى ترضى ولكـ الحمد إذا رضيت ولك الحمد بعد الرضـ.

وصلي اللهم وسلم على سيدنا محمد وسلم تسليما كثيرا.

# DEDICATION

I dedicate this research for

The soul of my father

My beloved mother

My siblings

My friends

And to all the people who helped me bring this project to life.

# TABLES OF CONTENTS

# LIST OF TERMS

| | |
|---|---|
| **LSB** | Least Significant Bit |
| **MSB** | Most Significant Bit |
| **MLS** | Multi-Level Steganography |
| **PNG** | Portable Network Graphic |
| **JEPG** | Joint Photographic Experts Group |
| **MRC** | MSB of Red Color |
| **GreenIndex** | Index of Green Color in level two |
| **BlueIndex** | Index of Blue Color in level two |
| **RGB** | Red Green Blue |
| **STEGCRYP** | Steganography and Cryptography |
| **StegImg** | Stego Image |
| **SSCE** | Secret steganography code for embedding |
| **PSNR** | Peak Signal to Noise Ratio |
| **MSE** | Mean Square Error |
| **DCT** | Discrete Cosine Transform |
| **PMM** | Pixel Mapping Method |
| **RSA** | Rivest, Shamir Adleman |

# LIST OF TABLES

# LIST OF FIGURES

# ABSTRACT

In a world of digital technology, maintaining the security of the secret data has become a great challenge. One way to achieve this is to encrypt the message before it is sent .but encryption draws the attention of third parties, which may cause the third party to seek breaking the encryption and detecting the original message. Another way is steganography, steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message.

In this thesis, a new concept for performing hidden secret data, called Multi-Level Steganography for image steganography, was presented. MLS consists of at least two stenographic methods utilized respectively. Two-levels of stenography have been applied; The first level is called (the upper-level), and it has been applied using enhance LSB (HS_LSB) image steganography, the secret data in this level is English text, and the cover is gray scale or RGB image, the output is a stego_image called (intermediate image).

The second level is called (the lower-level); it has been applied using pixel intensity based image steganography. In this level another RGB image has been used as a cover image and embeds (the RGB or gray scale image output from level one) as a secure data and generates the new RGB image as stego image.

After completing the proposed method implementation, many experiments have been conducted. Different sizes of secret messages and different sizes of cover images have been experimented. Finally comparative analysis has been applied to experiments results and presented good result for proposed method.

# المستخلص

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

فيهذا البحث، تم تطبيق مفهوم جديد لإخفاء البيانات السرية،يسمي بعلم إخفاء الصور متعدد المستويات .الاخفاء متعدد المستويات يحتوي علي طريقتين علي الاقل من طرق إخفاء البيانات، ويتم تطبيقهم علي التوالي .في هذا البحث تم استخدام مستويان . المستوي الاول يسمي ب(المستوي العلوي) و يتم تطبيقه بتحسين طريقة LSB وتم تسميتها ب (HS_LSB) لإخفاء الصور، البيانات السرية عبارة عن نص باللغة الانجليزية يتم إخفائه في صورة غير ملونة (رمادية) أو في صورة ملونة(RGB) ، و الناتج يكون عبارة عن صورة بها نص مخفي (stego_image) تسمي ب(الصورة الوسيطة).

المستوي الثاني يسمي ب(المستوي الادني) وتم تطبيقه بإستخدام طريقة كثافة البسكل للإخفاء في الصور.في هذا المستوي يتم استخدام صورة RGBاخري كغطاء يتم فيها اخفاء الصورة الناتجة من المستوي الأولي و الناتج يكون عبارة عن صورة RGB جديدة (stego_image).

هنالك العديد من التجارب تم إجرائها بعد الإنتهاء من تطبيق النظام المقترح ، وهذه التجارب تم إجرائها بإستخدام أحجام مختلفة من الرسائل السرية و أيضا أحجام مختلفة من الصور التي يتم الإخفاء فيها. واخيرا تم اجراء عملية مقارنة للنتائج و قد اظهرت نتائج جيدة للنظام المقترح .

# CHAPTER 1

# (INTRODUCTION)

# 1.1 Introduction

Security of information is one of the most important factors of information technology and communication. Security of information often lies in the secrecy of its existence and/or the secrecy of how to decode it. Cryptography, watermarking and Steganography can be used in information security. The cryptography techniques hide secret information by encrypts it using encryption key(s), the output of encryption is chipper text or the secret information in unreadable format, and this may draw the attention of attackers to the existence of confidential information. The digital watermarking is the process of embedding information into digital multimedia content such that the information (the watermark) can later be extracted or detected for a variety of purposes including copy prevention and control. The proposed method of information security in the research is steganography [1].

Steganography is an art and a science of communicating in a way, which hides the existence of the communication. It is also called as “covered writing”, because it uses a “cover” of a message for sending any important secret message. Steganography serves as a means for private, secure and sometimes malicious communication.

Steganography can be classified into image, text, audio and video steganography based on the cover media used to embed secret data. Images are the most popular cover objects used for steganography. In the domain of digital images many different image file formats exist, most of them for specific applications. For these different image file formats, different stenographic algorithms exist [2].

Any steganography technique has to satisfy two basic requirements. The first requirement is perceptual transparency, cover object (object not containing any additional data) and stego-object (object containing secret message) must be perceptually indiscernible. The second constraint is high data rate of the embedded data. All the stego-applications, besides requiring a high bit rate of the embedded data, have need of algorithms that detect and decode hidden bits without access to the original multimedia sequence (blind detection algorithm)[3].

Steganography, watermarking and encryption techniques are used to ensure data confidentiality however the main difference between them is that with encryption anybody can see that both parties are communicating in secret. While Steganography and watermarking hides the existence of a secret message and in the best case nobody can see that both parties are communicating in secret. This makes steganography suitable for some tasks for which encryption aren’t, such as copyright marking. [4]

Image steganography is steganography technique using image as cover object. There are many kinds image type can used for cover. Examples: jpg, png, bmp etc.

Least Significant Bit (LSB) image stenography one of the earliest techniques studied in the information hiding of digital image (as well as other media types) is

Least Significant Bit modification coding technique. In this technique LSB of binary sequence of each sample of digitized image file is replaced with binary equivalent of secret message. The advantage of this techniques it is the simplest way to embed information in a digital audio file. It allows large amount of data to be concealed within an image file. This method is more widely used as modifications to LSBs usually not create audible changes to the sounds. The LSB has disadvantage, it has considerably low robustness against attacks [5].

Pixel intensity based image steganography is the other method of image transform domain, in this method, all the three color planes will be converted in to binary values. For each pixel in the image, the plane which has the minimum number of ones in its MSB will act as index plane and the other two color planes are considered as data planes.

Multi-Level Steganography is a new concept of information hiding in telecommunication networks that uses features of an existing steganography method (the upper level method) to create a new one (the lower-level method). Multi-Level Steganography (MLS) was originally proposed by Al-Najjar for image steganography. MLS is based on combining two or more steganography methods in such a way that one method (the upper-level) is a carrier for the other method (the lower-level) [2].

## 1.2 Research Scope

The scope of this research will be steganography technique especially multilevel steganography (MLS) focusing on Image steganography. The hiding of secret information (text) will be achieved by two levels of image steganography; level one uses modified least significant bit (HS_LSB) image steganography to hide the secret information into image. While level two employs Pixel Intensity based image steganography to hide the image output from level one in another image.

The secret text massage used here is English language text, in NotePad text document under MS Windows. The images used are RGB images (colored images). In ".PNG and .JEPG" extensions. Figure 1.1 explains the scope of proposed method in details.

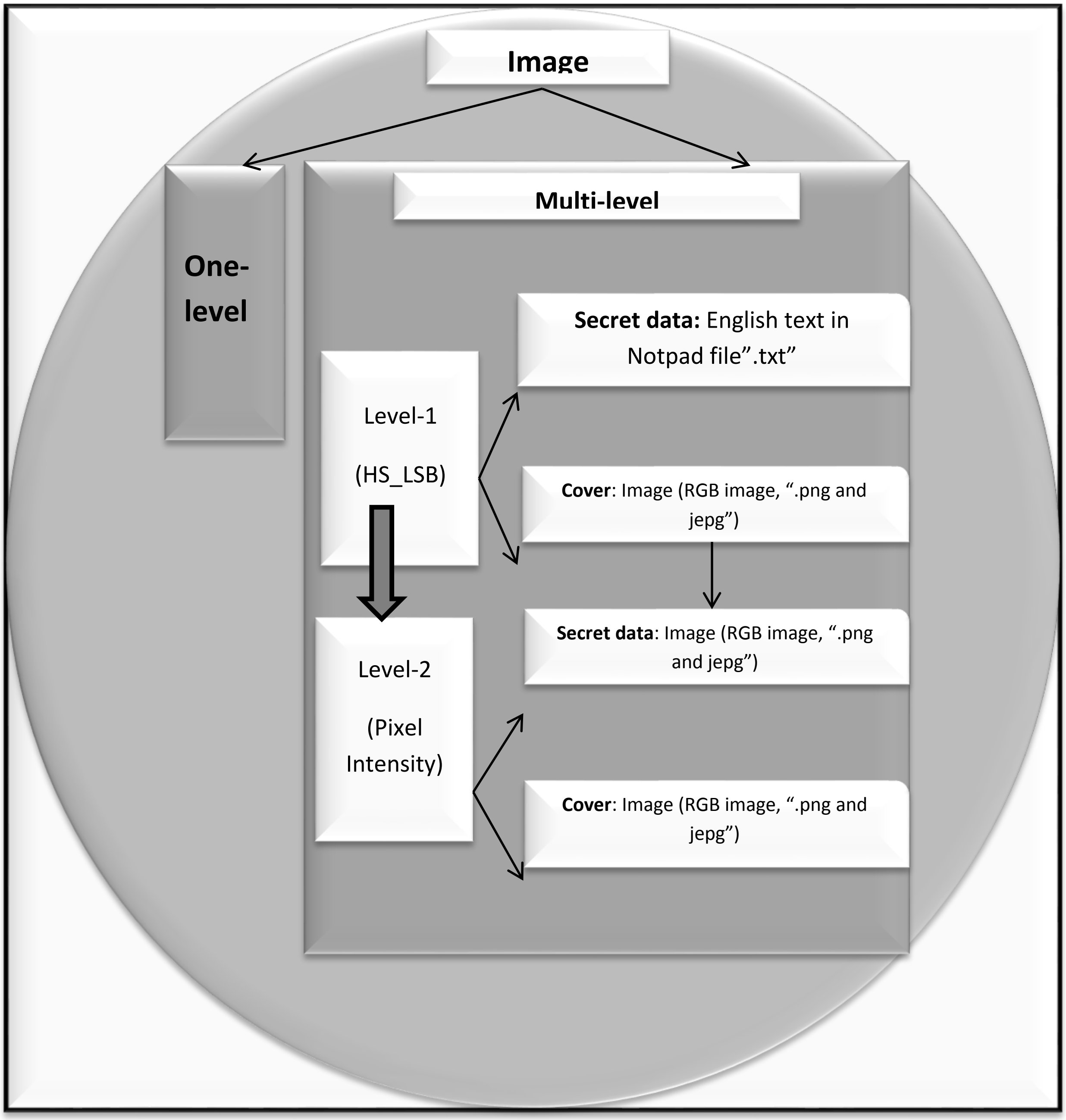


Image
One-level
Multi-level
Secret data: English text in Notpad file".txt"
Level-1
(HS_LSB)
Cover: Image (RGB image, ".png and jepg")
Secret data: Image (RGB image, ".png and jepg")
Level-2
(Pixel Intensity)
Cover: Image (RGB image, ".png and jepg")

Figure 1.1: the scope of proposed method

# 1.3 Problem Statement

Systems that use only one level of Steganography are usually more vulnerable, due to the fact that they lack the complexity to keep the data secure.

Furthermore the most commonly used Steganography algorithm which is the normal (LSB) algorithm is proved to be weak and the secret data is easy to retrieve. For this reason the proposed system uses a modified more secure version of LSB called the (HS_LSB). The second Steganography level also employs another strong algorithm called (Pixel Intensity).

# 1.4 Objective of the Research

The main objective is developing a system that applies multilevel image Steganography to concealing secret data into image by applying two-levels of image steganography. Level one HS_LSB image steganography, and level two Pixel Intensity based image steganography. It then extracts the secret data safely from the stego-object. Additionally the proposed method has some sub-objectives:

- Enhancing the confidentiality of the secret information by using two level image steganography in one the system.
- Add more complexity to the Steganography process through applying it in two levels.
- Develop a System that provides a high degree of data confidentiality.
- Try to balance the capacity of embedded data (secret text) and the changed pixels value especially in level one.
- Measure the performance of the proposed algorithm.

# 1.5 Research Questions

- How to use two levels of image steganography with different techniques to hide the secret information?
- How to extract the image (intermediate cover object) from image (cover object) and extract secret information (text) from image (intermediate cover object)?
- How the proposed method helps in hiding the secret information (text) to protect it from unauthorized disclosure?
- How to make changes in the stego-image invisible to protect it from detection by human visual system (HVS)?

- How to measure the performance of the proposed system?

# 1.6 Research Methodology and Tools

By applying deep study in one-level LSB image steganography techniques , We discovered the existence of vulnerabilities in LSB image steganography ,multi-level steganography can meet the vulnerabilities founded in LSB by adding another level of image steganography using Pixel Intensity based image steganography.

# 1.7 Research Organization

Chapter one gives introduction and brief history about the steganography, defining the types of steganography and multilevel steganography. Recently literatures review and related works will be explained in chapter two. Chapter three explains the proposed algorithm, tools and techniques used in the project. The analysis of the proposed algorithm and discussion of the results appears in chapter four and finally Chapter five contains the conclusion, recommendations and future work.

# CHAPTER 2

# LITERATURE REVIEW AND RELATED WORK

# 2.1 Introduction

Since the rise of the Internet one of the most important factors of information technology and communication has been the security of information. Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep the existence of the message secret. The technique used to implement this, is called steganography.

Steganography is the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information. The word steganography is derived from the Greek words "*stegos*" meaning "cover" and "*grafia*" meaning "writing" defining it as "covered writing". In image steganography the information is hidden exclusively in images. [6]

Steganography and encryption are both used in transfer secure data to ensure data confidentiality. However the main difference between them is that with encryption anybody can see that both parties are communicating in secret. Steganography hides the existence of a secret message and in the best case nobody can see that both parties are communicating in secret. This makes steganography suitable for some tasks for which encryption aren't, such as copyright marking. Adding encrypted copyright information to a file could be easy to remove but embedding it within the contents of the file itself can prevent it being easily identified and removed.

The main goal of steganography is to hide information in the other cover media so that other person will not notice the presence of the information. This is a major distinction between this method and the other methods of covert exchange of information because, for example, in cryptography, the individuals notice the information by seeing the coded information but they will not be able to comprehend the information [4]. However, in Steganography hides the existence of a secret message and in the best case nobody can see that both parties are communicating in secret [6].

Most steganography jobs have been carried out on images, video clips, texts, music and sounds .Nowadays, using a combination of steganography and the other methods, information security has improved considerably. This makes steganography suitable for some tasks for which encryption aren't, such as copyright marking. Adding encrypted copyright information to a file could be easy to remove but embedding it within the contents of the file itself can prevent it being easily identified and removed.

The below table show the main differences between steganography and cryptography:

| | Steganography | Cryptography |
|---|---|---|
| **Techniques** | LSB, Spatial Domain, Jsteg, Outguess | Transposition, Substitution, RSA |
| **Naked eye Identification** | No, as message is Hide within other carrier (cover image) | Yes, as message is convert in Other way, which sough something is hidden |
| **Capacity** | Differs as different Technology usually low hiding capacity | Capacity is so high, but as message is long it chances to be decrypt |
| **Detection** | Not easy to detect because to find stenographic image is hard. | Not easy to detect ,depend on technology used to generate |
| **Strength** | Hide message without altering the message, it conceals information | Hide message by altering the message by assigning key |
| **Imperceptibility** | High | High |
| **Robust** | Yes | Yes |

**Table2.1: the main differences between steganography and cryptography**

Two other technologies that are closely related to steganography are watermarking and fingerprinting. These technologies are mainly concerned with the protection of intellectual property, thus the algorithms have different requirements than steganography. These requirements of a good stenographic algorithm will be discussed below. In watermarking all of the instances of an object are “marked” in the same way. The kind of information hidden in objects when using watermarking is usually a signature to signify origin or ownership for the purpose of copyright protection. With fingerprinting on the other hand, different, unique marks are embedded in distinct copies of the carrier object that are supplied to different customers. This enables the intellectual property owner to identify customers who break their licensing agreement by supplying the property to third parties. In watermarking and fingerprinting the fact that information is hidden inside the files may be public knowledge– sometimes it may even be visible. While in steganography the imperceptibility of the information is crucial [6].

Steganography have many terminologies must be defined:

Cover image: It is defined as the original image into which the required information is embedded. It is also termed as carrier image. The information should be embedded in such a manner that there are no significant changes in the statistical properties of the cover image. [1] Stego-image: refers to the object which is carrying a hidden message. So given a cover object, and a messages the goal of the stenographer is to produce a stego object which would carry the message. [9]

Perceptibility: It describes the ability of a third party (not the intended recipient) to visually detect the presence of hidden information in the stego image. The embedding algorithm is imperceptible when used on a particular image if an innocent third party, interested in the content of the cover image, is unaware of the existence of the payload. Essentially this requires that the embedding process not degrade the visual quality of the cover image.

Robustness: It characterizes the ability of the payload to survive the embedding and extraction process, even in the face of manipulations of the stego image such as filtering, cropping, rotating and compression.

Stenographic capacity: refers to the maximum amount (rate) of information that can be embedded into a cover-object and then can be reliably recovered from the stego-object (or a distorted version), under the constraints of undetectability, perceptual intactness and robustness. Compared to data hiding systems, stegosystems have the added core requirement of undetectability. Therefore, the stenographic embedding operation needs to preserve the statistical properties of the cover-object, in addition to its perceptual quality [9].

Almost all digital file formats can be used for steganography, but the formats that are more suitable are those with a high degree of redundancy. Redundancy can be defined as the bits of an object that provide accuracy far greater than necessary for the object’s use and display [6]. The redundant bits of an object are those bits that can be altered without the alteration being detected easily. Image and audio files especially comply with this requirement, while research has also uncovered other file formats that can be used for information hiding. Figure 2.1 shows the four main categories of file formats that can be used for steganography.

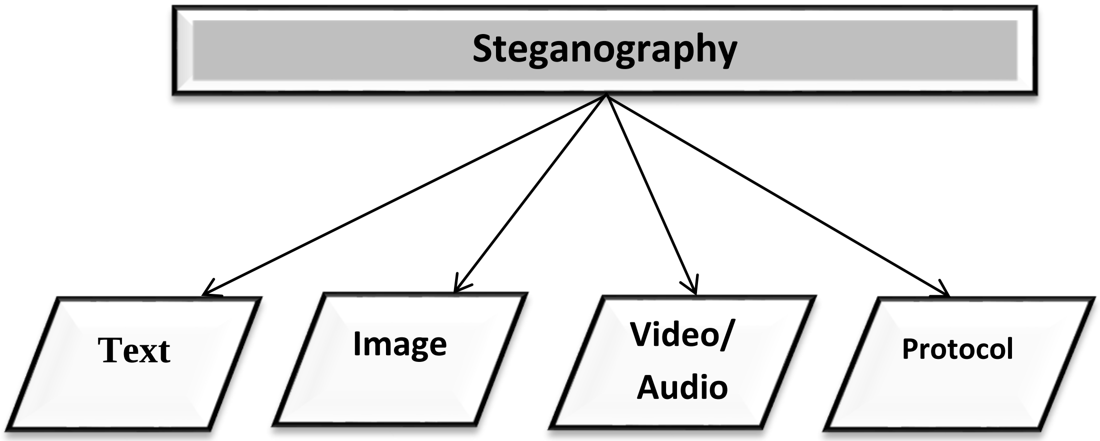


**Figure 2.1: Categories of steganography**

Steganography can be split into two types, these are fragile and robust:

Fragile: Fragile steganography involves embedding information into a file which is destroyed if the file is modified. This method is unsuitable for recording the copyright holder of the file since it can be so easily removed, but is useful in situations where it is important to prove that the file has not been tampered with, such as using a file as evidence in a court of law, since any tampering would have removed the watermark. Fragile steganography techniques tend to be easier to implement than robust methods [10].

Robust: Robust marking aims to embed information into a file which cannot easily be destroyed. Although no mark is truly indestructible, a system can be considered robust if the amount of changes required to remove the mark would render the file useless. Therefore the mark should be hidden in a part of the file where its removal would be easily perceived. There are two main types of robust marking [10].

# 2.2 Image Steganography

Image Steganography has many applications, especially in today's modern, high-tech world. Privacy and anonymity is a concern for most people on the internet. Image Steganography allows for two parties to communicate secretly and covertly. It allows for some morally-conscious people to safely whistle blow on internal actions; it allows for copyright protection on digital files using the message as a digital watermark. One of the other main uses for Image Steganography is for the transportation of high-level or top-secret documents between international governments. While Image Steganography has many legitimate uses, it can also be quite nefarious. It can be used by hackers to send viruses and Trojans to compromise machines, and also by terrorists and other organizations that rely on covert operations to communicate secretly and safely.

Image steganography techniques can be divided into two groups the spatial Domain or Image and Transform Domain or frequency domain.

## 2.2.1 Spatial Domain Embedding

The best widely known steganography algorithm is based on modifying the least significant bit layer of images, hence known as the LSB technique. This technique makes use of the fact that the least significant bits in an image could be thought of random noise and changes to them would not have any effect on the image. Although the image seems unchanged visually after the LSBs are modified, the statistical properties of the image changes significantly.

In the LSB technique, the LSB of the pixels is replaced by the message to be sent. The message bits are permuted before embedding, this has the effect of distributing the bits evenly, thus on average only half of the LSB's will be modified. Popular stenographic tools based on LSB embedding, vary in their approach for hiding information. Some algorithms change LSB of pixels visited in a random walk, others modify pixels in certain areas of images, or instead of just changing the last bit they increment or decrement the pixel value.

Spatial domain techniques encompass bit-wise methods that apply bit insertion and noise manipulation and are sometimes characterized as “simple systems” [7]. The image formats that are most suitable for image domain steganography are lossless and the techniques are typically dependent on the image format [8].

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image .The least significant bit (in other words, the 8th bit) of some or all of the bytes inside an image is changed to a bit of the secret message. When using a 24-bit image, a bit of each of the red, green and blue color.

LSB is the lowest bit in a series of numbers in binary. e.g. in the binary number: 10110001, the least significant bit is far right 1.The LSB based Steganography is one of the stenographic methods, used to embed the secret data in to the least significant bits of the pixel values in a cover image. e.g. 240 can be hidden in the first eight bytes of three pixels in a 24 bit image.

PIXELS: (00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

240: (011110000)

RESULT: (00100110 11101001 11001001)
(00100111 11001001 11101000)
(11001000 00100110 11101000)

Here number 240 is embedded into first eight bytes of the grid and only 6 bits are changed.

Pixel intensity based image steganography is the other method of image transform domain, in this method, all the three color planes will be converted in to binary values. For each pixel in the image, the plane which has the minimum number of ones in its MSB will act as index plane and the other two color planes are considered as data planes. Compared to method 1 and method 2 in the existing work, this method will help us to embed more number of message bits in the cover medium.

After determining the index and data channel using any of the method which is specified above then follow these steps, For each index plane its two LSB is considered .If the value of the LSB is 00 or 11 then the embedding process will be on both data plane or if the value is 01 then data will hide in data plane2 alone or if the value is 10 then data will hide in data plane1 alone. The number of bits gets embed in the data plane is equal to the number of ones in its MSB of the data plane. These above steps are similar for all the three methods. [15]

### 2.2.2 Transform Domain Embedding

Transform coding constitutes an integral component of contemporary image/video processing applications. Transform coding relies on the premise that pixels in an image exhibit a certain level of correlation with their neighboring pixels.

Another category for embedding techniques for which a number of algorithms have been proposed is the transform domain embedding category. Most of the work in this category has been concentrated on making use of redundancies in the DCT (discrete cosine transform) domain, which is used in JPEG compression. But there have been other algorithms which make use of other transform domains such as the frequency domain. Embedding in DCT domain is simply done by altering the DCT coefficients, for example by changing the least significant bit of each coefficient. One of the constraints of embedding in DCT domain is that many of the 64 coefficients are equal to zero, and changing two many zeros to non-zeros values will have coefficients on the compression rate. That is why the number of bit one could embed in DCT domain, is less that the number of bits one could embed by the LSB method. Also the embedding capacity becomes dependent on the image type used in the case of DCT embedding, since de- pending on the texture of image the number of non-zero DCT coefficients will vary. Although changing the DCT coefficients will cause unnoticeable visual artifices, they do cause detectable statistical changes.

Steganography in the transform domain involves the manipulation of algorithms and image transforms .These methods hide messages in more significant areas of the cover image, making it more robust. Many transform domain methods are independent of the image format and the embedded message may survive conversion between lossy and lossless compression [8].

## 2.3 Multi-Level Steganography (MLS)

Multi-Level Steganography can be utilized to achieve various aims – it all depends on how it will be used. Here we present several of the most interesting MLS applications, in our opinion. The benefits of MLS for hidden data exchange are summarized in the below Table. [16]

| MLS benefit | Described MLS application |
|---|---|
| **Increased stenographic bandwidth for user data** | Using two or more stenographic methods increases the total stenographic bandwidth achieved for user data compared with a single stenographic method. |
| **Increased undetectability** | An upper-level method controlled by information carried by the lower-level method |
| **Steganogram transmission reliability** | Lower-level method carrying information for steganogram Integrity verification (Sec. 3.1). |
| **Harder steganogram extraction and analysis** | 1. Cryptographic key carried by lower-level method and upper-level method steganogram ciphered (Sec. 3.1).<br>2. Parts of the steganogram sent using the upper-level and others by the lower-level method.<br>3. Steganogram carried only by the lower-level method; upper-level steganogram only for masking (Sec. 3.2). |
| **Steganography cost unchanged** | In best-case scenario, depends on the upper- and lower-level methods used, but can be the same as for utilization Of the upper-level method alone. |

**Table 2.2: MLS benefits and possible applications [16].**

Let us consider the abovementioned MLS applications based on where the steganogram is inserted. There are three possible cases:

- Steganogram is carried only by upper-level method
- Steganogram is carried only by lower-level method
- Steganogram is carried by both upper- and lower-level methods

# 2.4 Related works

Section two in this chapter presents the related work in image steganography both LSB techniques and pixel intensity techniques and present compression table between them in the latter part of this section.

In [11] Dr.AL-NAJJAR presented “Multi-Level Digital Multimedia Steganography Mode”. This paper focuses on two folds: to develop an abstract multi-level model and to illustrate the model by hiding text represented using a black and white image into a gray decoy image and then into a color image in the RGB format. Four objects are defined, the message-object (M), the intermediate-object (I), the cover-object (C), and the stego-object (S). The elements of M are given by the set {M} of size |M|, similarly, {I} of size |I| and so forth.

The message {M} is passed through the transformation T1 that can include many possibilities. It can be compression, private-key or public-key encryption, or a combination of techniques, as required by the particular application. The same can be said about the other transformations T2 and T3. Figure 2.6 demonstrate the proposed model.

Embedding and recovery is controlled by the embedding/recovery function pairs f/g. The embedding function from {M} to {I} (or {D}) and from {D} to {C} can be different, improving one or more of the three steganography attributes: Capacity, Robustness, and Transparency.

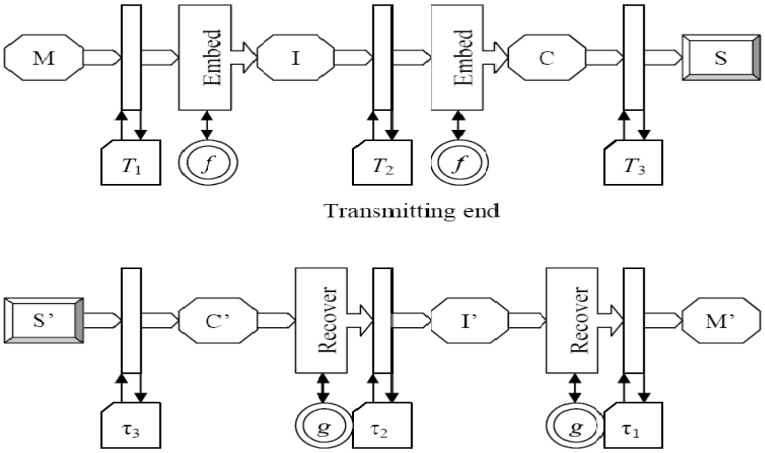


**Figure 2.2: multi-level steganography model [11].**

In [12] Gurmeet Kaur present comparative analysis is made to demonstrate the effectiveness of the proposed methods. The effectiveness of the proposed methods has been estimated by computing Mean square error (MSE) and Peak Signal to Noise Ratio (PSNR) and processing time.

The algorithm of steganography is divided in two section, section one focus in embed the text message using LSB steganography, this section have six steps, in step one read the cover image and text message which is to be hidden in the cover image, in step two convert the color image into grey image, after that in step three convert text message in binary then in step four calculate LSB of each pixels of cover image and in step five Replace LSB of cover image with each bit of secret message one by one , finally in step six Write stego image.

The section two in steganography algorithm is retrieve text message from stego image, this section have eight steps, in step one read the stego image, and then in step two calculate LSB of each pixels of stego image, after that in step three retrieve bits and convert each 8 bit into character and the cover image is broken into 8×8 block of pixels and then in step four working from left to right, top to bottom subtract 128 in each block of pixels after that in step five DCT is applied to each block, in step six each block is compressed through quantization table, additionally in step seven calculate LSB of each DC coefficient and replace with each bit of secret message, Finally in step eight write stego image.

In [12] another proposed algorithm is embed text message based on DCT steganography in this algorithm in step one is read cover image and in step two read secret message and convert it in binary.

The analysis of LSB based and DCT based steganography has been done on basis of parameters like PSNR, MSE, Processing time, security. PSNR computes the peak signal to noise ratio, in decibels, between two images. This ratio is used as a quality measurement between two images. If PSNR ratio is high then images are best of quality

| **Method** | **PSNR** | **MSE** | **PROCESSING TIME** | **SIZE OF COV IMAGE** |
|---|---|---|---|---|
| **LSB** | 51.1 109 | 0.50 35 | 0.133777 seconds | 256x256 |
| **LSB** | 51.1 109 | 0.49 93 | 0.084754 sec | 256x256 |
| **DCT** | 40.6735 | 5.56 84 | 1.0140 sec | 256x256 |
| **DCT** | 39.3 983 | 7.4687 | 1.3260 sec | 256x256 |

**Table 2.3: Simulation results for LSB & DCT Method**

In this paper analysis of LSB & DCT methods has been successfully implemented and results are delivered. The MSE and PSNR of the methods are also compared and also this paper presented a background discussion and implementation on the major algorithms of steganography deployed in digital imaging. From the results it is clear that as PSNR in LSB is the best but as we know that security is much more important in today's communication system. So security wise DCT is the best.

In [13] Souvik Bhattacharyya is presented "Data Hiding through Multi Level Steganography and SSCE". They proposed that a stenographic model combining the features of both text and image based steganography technique for communicating information more securely between two locations. The authors incorporated the idea of secret key for authentication at both ends in order to achievehigh level of security. As a further improvement of security level, the information has been encoded through SSCE values and embedded into the cover text using the proposed text steganography method to form the stego text. This encoding technique has been used at both ends in order to achieve high level of security. Next the stego text has been embedded through PMM method into the cover image to form the stego image. At the receiver side different reverse operation has been carried out to get back the original information.

Figure 2.3 below show the block diagram of the proposed stenographic model. The input message is first encoded through SSCE (Secret steganography code for embedding) values and embedded into the cover text using the proposed text steganography method. This encrypted message generates the secret key. The encrypted message is then embedded in the cover text using the mapping technique method to form the stego text which in turn embedded in to the cover image through PMM (Pixel Mapping Method) to form the stego image and transmit to the receiver side.At the receiver side, the stego image will be tested first for a specific feature. If that feature matches, the extraction process starts by extracting the stego text from the stego image. Next the stego text goes through the text extraction and decryption method and finally the receiver may be able to see the embedded message with the help of same secret key generated at the sender side.

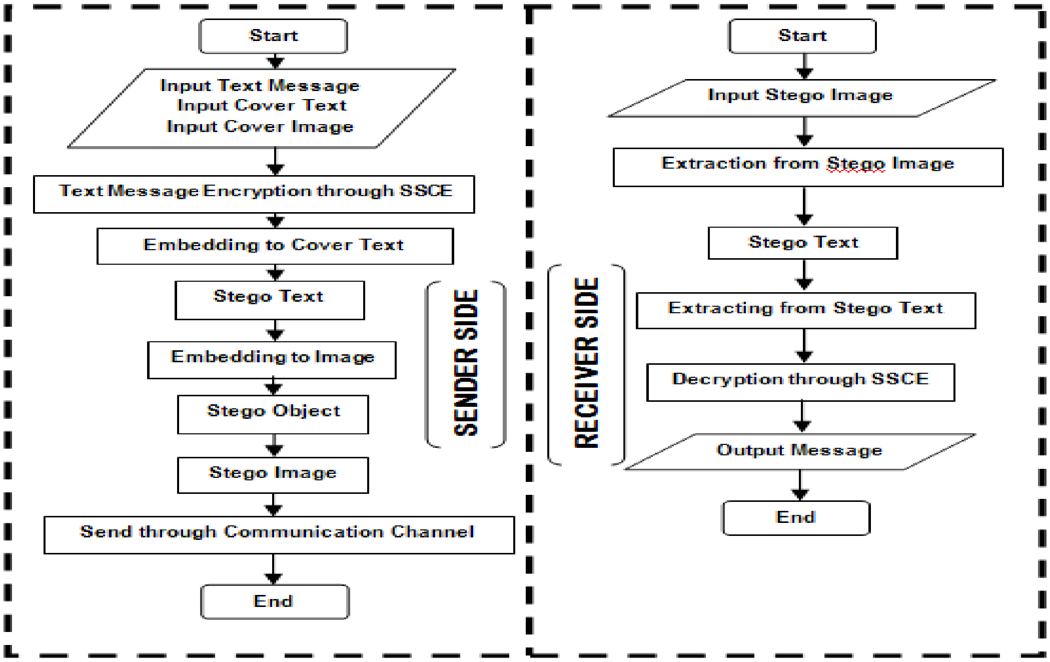


**Figure 2.3: proposed algorithm for stenographic model [13]**

In [14] Mohammad Tanvir Parvez presents a new algorithm for RGB image based steganography. The algorithm introduces the concept of storing variable number of bits in each channel (R, G or B) of pixel based on the actual color values of that pixel: lower color component stores higher number of bits. Our algorithm offers very high capacity for cover media compared to other existing algorithms.

The proposed is splitting pixel value to three channels (Red, Green and Blue) Use one of the three channels as the indicator. The indicator sequence can be made random, based on a shared key between sender and receiver, and then in the embedding process the data is stored in one of the two channels other than the indicator. The channel, whose color value is lowest among the two channels other than the indicator, will store the data in its least significant bits. Instead of storing a fixed no of data-bits per channel, no of bits to be stored will depend on the color value of the channel. The lower value higher data-bits to be stored. Therefore a partition of the color values is needed. Through experimentations, we show that optimal partition may depend on the actual cover image used. To retrieve the data in this algorithm, we need to know which channel stores the data-bits. This is done by looking at the least significant bits of the two channels other than the indicator, if the bits are same, then the channel following the indicator in cyclic order stores the data, otherwise, the channel which precedes the indicator in cyclic order stores the data [14]. The Flowing is Flow charts explain the encoding and decoding parts of this algorithm:

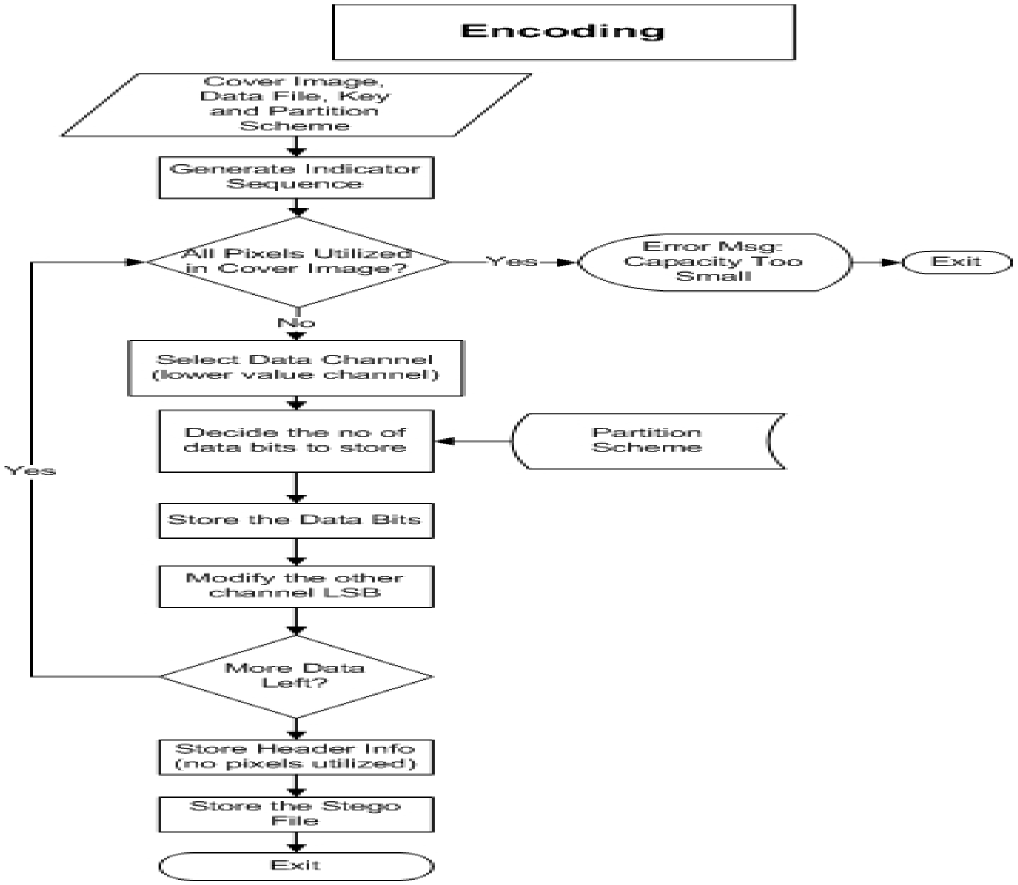

Encoding
Cover Image, Data File, Key and Partition Scheme
Generate Indicator Sequence
All Pixels Utilized in Cover Image?
Yes
Error Msg: Capacity Too Small
Exit
No
Select Data Channel (lower value channel)
Decide the no of data bits to store
Partition Scheme
Yes
Store the Data Bits
Modify the other channel LSB
More Data Left?
Store Header Info (no pixels utilized)
Store the Stego File
Exit

**Figure 2.4: Flow charts of the encoding part of algorithm [14].**

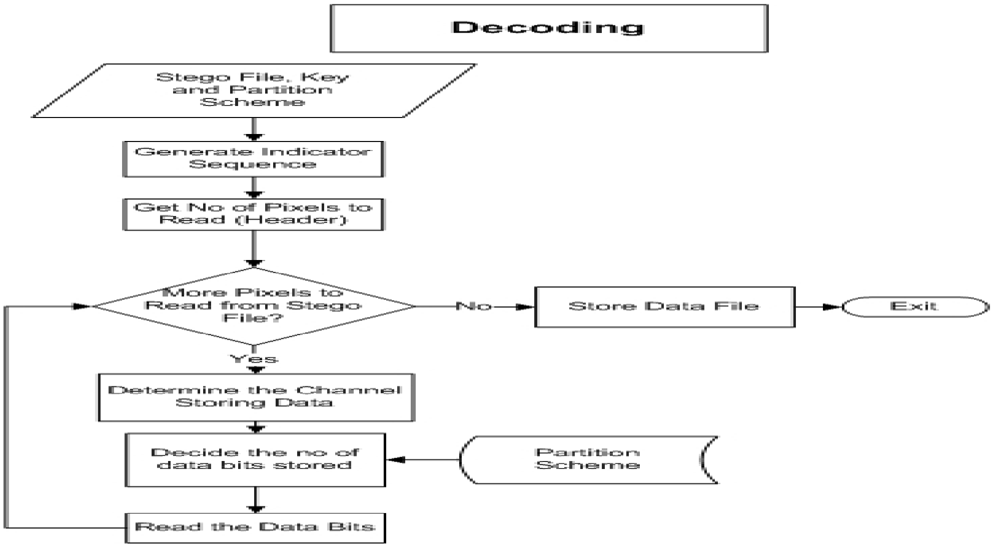

Decoding
Stego File, Key and Partition Scheme
Generate Indicator Sequence
Get No of Pixels to Read (Header)
More Pixels to Read from Stego File?
No
Store Data File
Exit
Yes
Determine the Channel Storing Data
Decide the no of data bits stored
Partition Scheme
Read the Data Bits

**Figure 2.5: Flow charts of the decoding part of algorithm [14].**

| Paper name | Message object | Number Of Level | Cover object | Techniques |
|---|---|---|---|---|
| **The Decoy: Multi-Level Digital Multimedia Steganography Model** | Text represented by black and white images | Two - Levels | A Gray scale image and RGB image | LSB in both levels |
| **A Steganography Implementation based on LSB & DCT** | text | one - Level | Image | LSB & DCT |
| **Hiding through Multi Level Steganography and SSCE** | Text message | Two-levels | Image | PMM (Pixel Mapping Method)- Inserting non-specific or non-particular nouns in English |
| **RGB Intensity Based Variable-Bits Image Steganography** | Text Message | one - Level | Image | Pixel intensity |

**Table 2.4: the summarization of related work.**

# CHAPTER THREE

# WORK ENVIROMENT AND PROPOSED SYSTEM ANALYSIS

# 3.1 Overview

This chapter describes the proposed method (multilevel image steganography) and explains the diagrams that clarify the proposed method. In level one modified Least Significant Bit (HS_LSB) Image steganography, RGB image is used as a cover image with a secure data (text) converted to long bit-stream before concealing, while level two (pixel intensity based image steganography) another RGB image is used as a cover image with a secure data (the RGB image output from level one) also converted to long bit-stream before concealing.

The programming language will used in implementation of the two levels (level one and level two) is java programming language, since it contains appropriate and more suitable methods to read from file, write in file, manipulate and modify the pixels that belong to an image then save the modified image.

MATLAB (R2010a) is also used to evaluate the results of the proposed method by calculating the PSNR and MSE of image, the MATLAB is suitable for the evaluation because it’s a high-level technical computing language and an interactive environment for algorithm development; data visualization, data analysis, and numeric computation.

# 3.2 Proposed Method

The proposed method is using multilevel image steganography (two levels) level one will be done by embedding the secret message (text) into cover image (cover one) which is a colored image (RGB image) using Least Significant Bit (LSB) image steganography.

The output from level one is stego image referred to as (intermediate image), the intermediate image will be converted into binary text and will work as input in level two.

Level two conceals the binary text in another image referred to as (cover two) which is also a colored image (RGB image), the binary text is concealed in this image using pixel intensity based image steganography and the output of this level is new RGB image ( stego image) .

In level one LSB hiding technique hides the secret message directly in the least (one-two or three) Significant bits in the image pixels, the variation between the number of embedding in least significant bits will be based on the specific indicators will explain later in level one embed process.

In level two the proposed techniques is pixel intensity image steganography in this techniques the image will be split into two section or channel, the first channel will use as index (index channel or index plane), the first channel will contain one color from pixel value (Red color).and the second channel will used for embedding data (data channel or plane), the second channel will contain two colors from pixel value (Green and Blue colors).

The new proposed technique that able to make the secret message more secure and make balance between the quality and capacity of the image. Figure 3.1 below explain the general overview of the proposed method (embedding process)

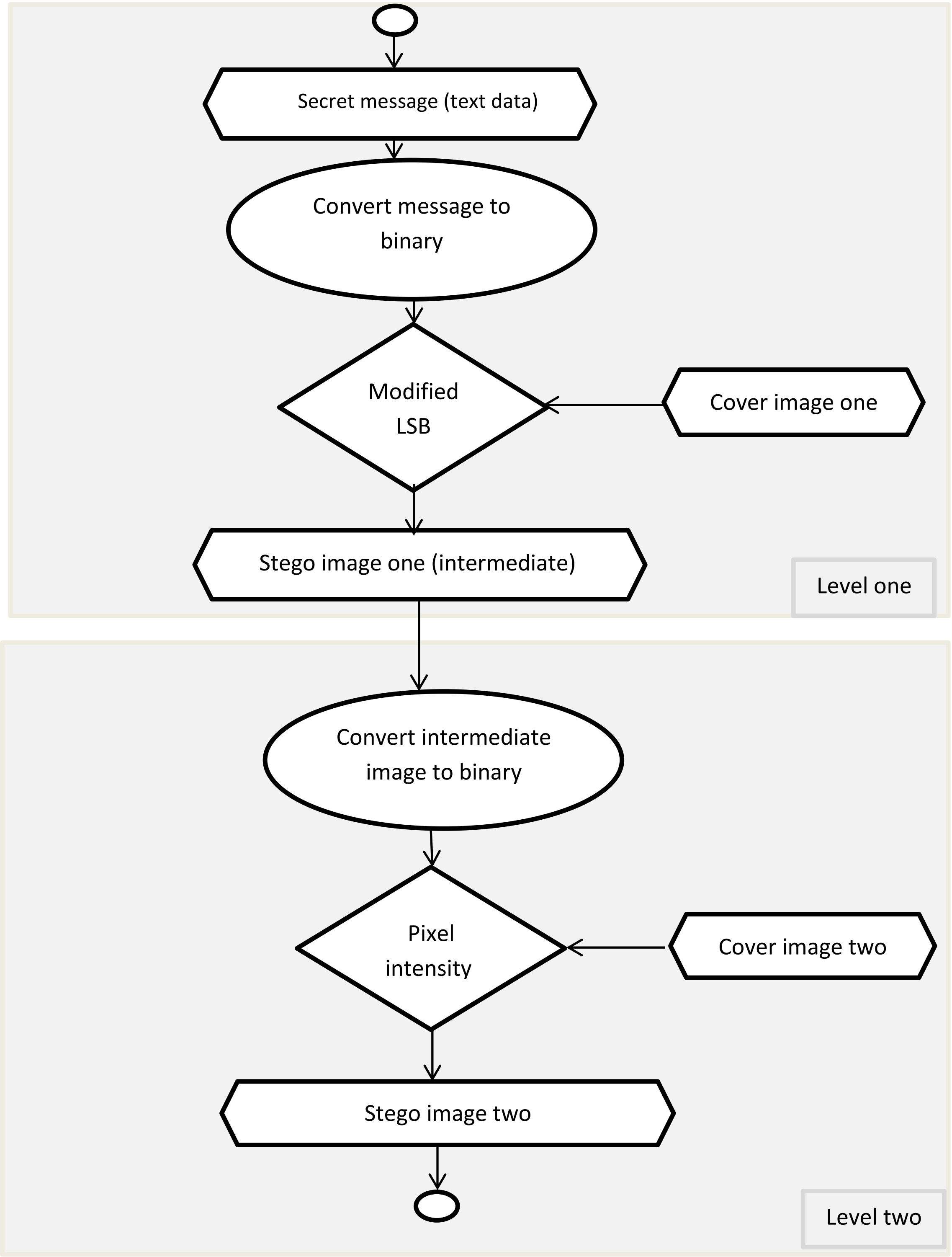

Secret message (text data)
Convert message to binary
Modified LSB
Cover image one
Stego image one (intermediate)
Level one
Convert intermediate image to binary
Pixel intensity
Cover image two
Stego image two
Level two

**Figure 3.1: the general overview of the proposed method**

### 3.2.1 The Embedding Process in Level One Using HS_LSB

1- Read the secret message from file and Convert it to binary.

2- Read the pixels of cover image and split it to (Red, Green and Blue) (cover image one).

3- Check the first two MSB bits of each color, assume the first two MSB bits of red assigned to variable (MRC):

```
If (MRC==”00”): Jump the red color in this pixel.
Else
If (MRC=”01”): Apply one-LSB by put one bit from secret message in the LSB of Image.
Else
If (MRC=”10”): Apply two-LSB by put two bit from secret message in the LSB of Image.
Else
If (MRC=”11”): Apply three-LSB by put three bits from secret message in the LSB of Image.
```

4. After complete the bits in the binary secret text, save the resulting stego image (intermediate image).

Table 3.1 explain example of embedding process using HS_LSB image steganography.

| Secret message | Value of color before stego | Value of color After stego | Applied method |
|---:|---:|---:|---:|
| 1001 | **00**100110 | 00100110 | Jump color |
| 1001 | **01**100110 | 0110011**1** | One-LSB |
| 1001 | **10**100110 | 101001**01** | Two-LSB |
| 1001 | **11**001100 | 11100**001** | Three-LSB |

**Table 3.1: example of embedding using HS_LSB**

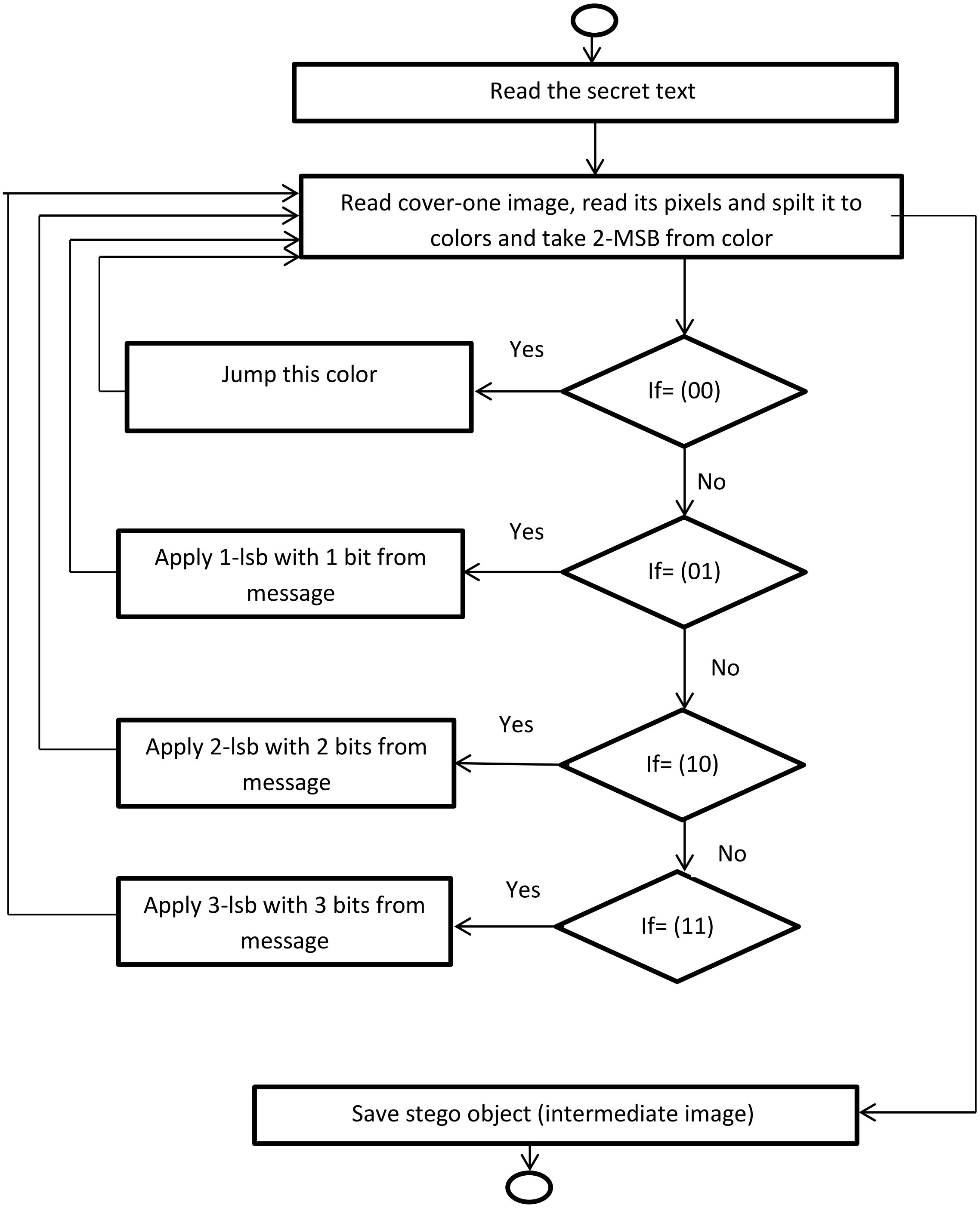

Read the secret text
Read cover-one image, read its pixels and spilt it to colors and take 2-MSB from color
Yes
Jump this color
If= (00)
No
Yes
Apply 1-lsb with 1 bit from message
If= (01)
No
Yes
Apply 2-lsb with 2 bits from message
If= (10)
No
Yes
Apply 3-lsb with 3 bits from message
If= (11)
Save stego object (intermediate image)

**Figure 3.2: level one embedding process (HS_LSB)**

### 3.2.2 The Retrieving Process in Level One Using HS_LSB

1- Read the stego image from level one (intermediate image).

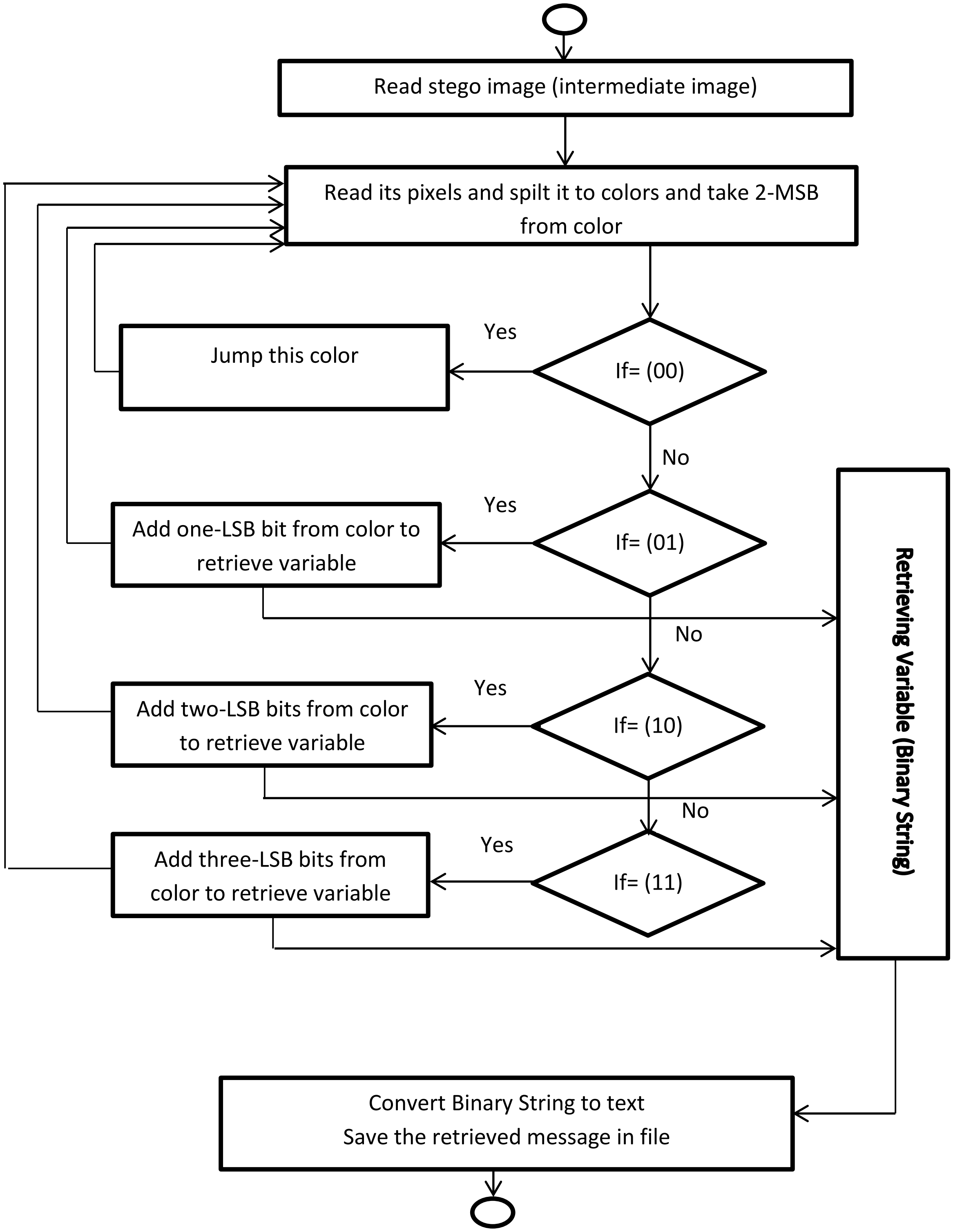
2- Read the pixels of cover image and split it to (Red, Green and Blue).

3- Check the first two MSB bits of each color, assume the first two MSB bits of red assigned to variable (MRC) and assume the retrieving binary String(Retrieve):

```
        If (MRC==”00”): Jump the red color in this pixel.
        Else
        If(MRC=”01”): copy one bit from LSB of the red color and add it to (Retrieve) variable.
        Else
        If(MRC=”10”): copy two bits from LSB of the red color and add it to (Retrieve) variable.
        Else
        If(MRC=”11”): copy three bits from LSB of the red color and add it to (Retrieve) variable.
```

4- After reading all bits convert it to string and write the secret message in file.

Figure 3.3 explain the retrieving process using HS_LSB image steganography.



Read stego image (intermediate image)
Read its pixels and spilt it to colors and take 2-MSB from color
If= (00)
Yes
Jump this color
No
If= (01)
Yes
Add one-LSB bit from color to retrieve variable
No
If= (10)
Yes
Add two-LSB bits from color to retrieve variable
No
If= (11)
Yes
Add three-LSB bits from color to retrieve variable
Retrieving Variable (Binary String)
Convert Binary String to text
Save the retrieved message in file

**Figure 3.3: level one retrieving process (HS_LSB)**

### 3.2.3 The Embedding Process in Level Two Using Pixel Intensity Based Image Steganography

1- Convert secret message to binary.
2- Read the pixels of cover image and split it to (Red, Green, Blue) RGB.
3- Convert the values of colors (Red, Green and blue) to binary string.
4- Assume the Red color as index plane, Green and Blue as data plane.
5- Take two bits from secret message and apply XOR operation with first two bits from green or blue color (two-LSB) bits:
    - If the result equal “00” that means the two bits from secret message match the two bits from first two bits from green or blue color and put it as index in red color (index plane).
    - Else if result not equal “00” that means don’t matches, shift one bit to lift in green or blue color to take the second and third bits then apply XOR operation with the two bits from secret message.
    - If matches assign the value in red color (index plane), if don’t matching shift one bit in green or blue color (data plane).
    - If matches assign the value in red color (index plane), if don’t matching jump this byte green or blue.

Table 3.2 explains example of embedding process using pixel intensity based image steganography and figure 3.4 explain its diagram.

| **Index Plane (Red Color)** | **Data Plane(Green Or Blue)** |
|---|---|
| XXXXXXXX<br>If data plane green: XXXXXX00<br>If data plane blue : XXXX00XX | Value Of color: XXXXXX11<br>2 bits from sec: 11<br>The Value Of XOR : 00 |
| XXXXXXXX<br>If data plane green: XXXXXX01<br>If data plane blue : XXXX01XX | Value Of color: XXXXX11X<br>2 bits from sec: 11<br>The Value Of XOR : 00 |
| XXXXXXXX<br>If data plane green: XXXXXX10<br>If data plane blue : XXXX10XX | Value Of color: XXXX11XX<br>2 bits from sec: 11<br>The Value Of XOR : 00 |
| XXXXXXXX<br>If data plane green: XXXXXX11<br>If data plane blue : XXXX11XX | Value Of color: XXX11XXX<br>2 bits from sec: 11<br>The Value Of XOR : 00 |

**Table 3.2: example of embedding using pixel intensity**

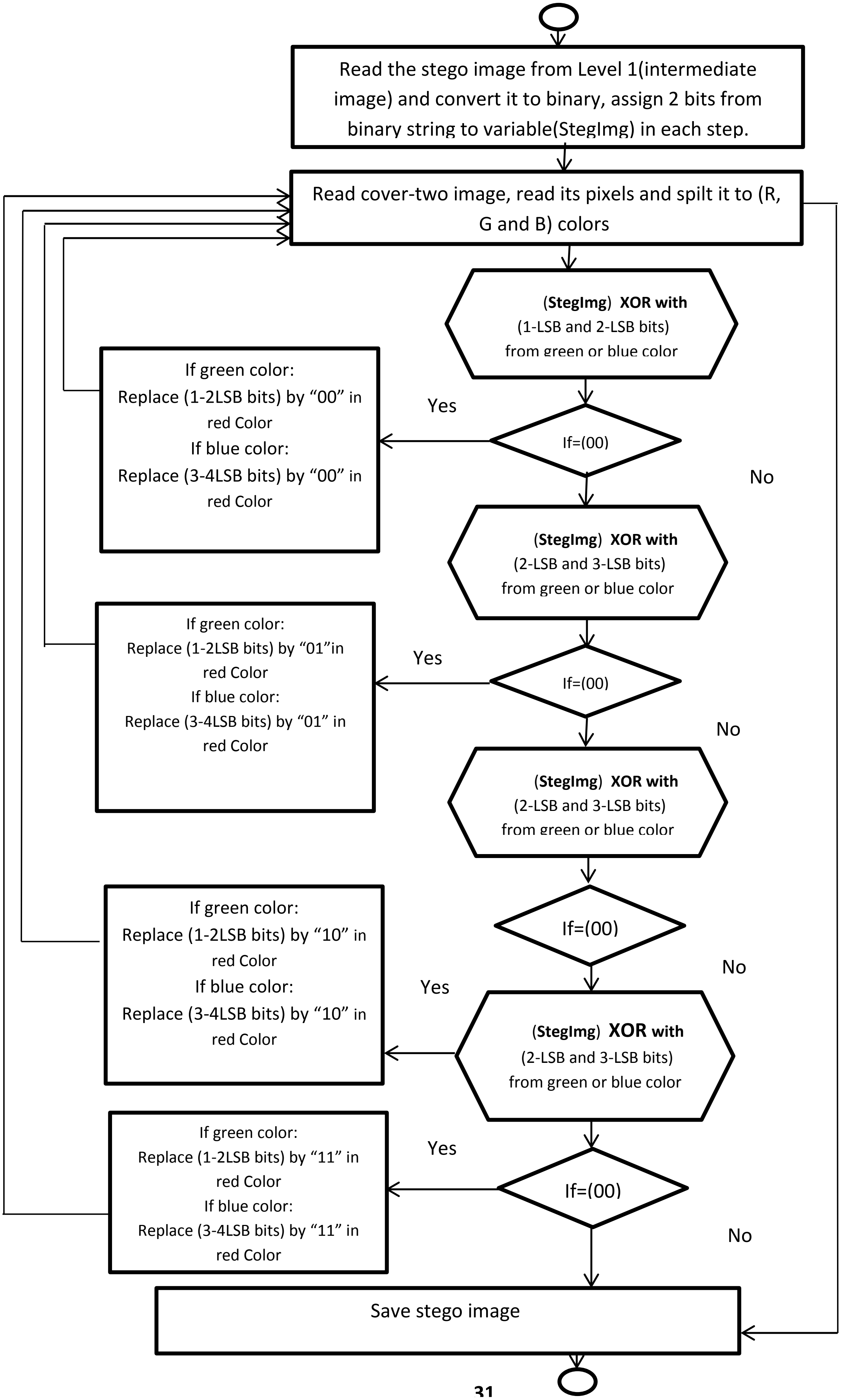

Read the stego image from Level 1(intermediate image) and convert it to binary, assign 2 bits from binary string to variable(StegImg) in each step.
Read cover-two image, read its pixels and spilt it to (R, G and B) colors
(StegImg) XOR with (1-LSB and 2-LSB bits) from green or blue color
If=(00)
Yes
No
If green color: Replace (1-2LSB bits) by "00" in red Color If blue color: Replace (3-4LSB bits) by "00" in red Color
(StegImg) XOR with (2-LSB and 3-LSB bits) from green or blue color
If=(00)
Yes
No
If green color: Replace (1-2LSB bits) by "01"in red Color If blue color: Replace (3-4LSB bits) by "01" in red Color
(StegImg) XOR with (2-LSB and 3-LSB bits) from green or blue color
If=(00)
No
Yes
If green color: Replace (1-2LSB bits) by "10" in red Color If blue color: Replace (3-4LSB bits) by "10" in red Color
(StegImg) XOR with (2-LSB and 3-LSB bits) from green or blue color
Yes
If=(00)
No
If green color: Replace (1-2LSB bits) by "11" in red Color If blue color: Replace (3-4LSB bits) by "11" in red Color
Save stego image

**Figure 3.4: level two embedding process (Pixel intensity)**

## 3.6 The Retrieving Process in Level Two Uses Pixel Intensity Based Image Steganography

Read the stego image from level two:

1- Read the pixels of stego image and split it to (Red, Green and Blue).

2- Convert the values of colors (Red, Green and blue) to binary string.

3- put the first two LSB bits of Red color(1- LSB and 2- LSB) in variable (GreenIndex) put the second two LSB bits of Red color(3- LSB and 4- LSB) in variable (BlueIndex) and assume the retrieving binary String(Retrieve):

   Check the GreenIndex variable:

   If (GreenIndex=”00”): Copy two bits from LSB (1-LSB and 2-LSB) of the green color and add it to (Retrieve) variable.

   Else

   If (GreenIndex=”01”): Copy two bits from LSB (2-LSB and 3-LSB) of the green color and add it to (Retrieve) variable.

   Else

   If (GreenIndex=”10”): Copy two bits from LSB (3-LSB and 4-LSB) of the green color and add it to (Retrieve) variable.

   Else

   If (GreenIndex=”11”):

   Copy two bits from LSB (4-LSB and 5-LSB) of the green color and add it to (Retrieve) variable.

   Check the BlueIndex variable:

   If (BlueIndex =”00”): Copy two bits from LSB (1-LSB and 2-LSB) of the blue color and add it to (Retrieve) variable.

   Else

   If (BlueIndex =”01”): Copy two bits from LSB (2-LSB and 3-LSB) of the blue color and add it to (Retrieve) variable.

   Else

   If (BlueIndex =”10”): Copy two bits from LSB (3-LSB and 4-LSB) of the blue color and add it to (Retrieve) variable

   Else

   If (BlueIndex =”11”)

   Copy two bits from LSB (4-LSB and 5-LSB) of the blue color and add it to (Retrieve) variable.

4- Convert the binary variable (Retrieve) to image and save the image (intermediate image).

Figure 3.5 explains the diagram of retrieving process using pixel intensity based image steganography.

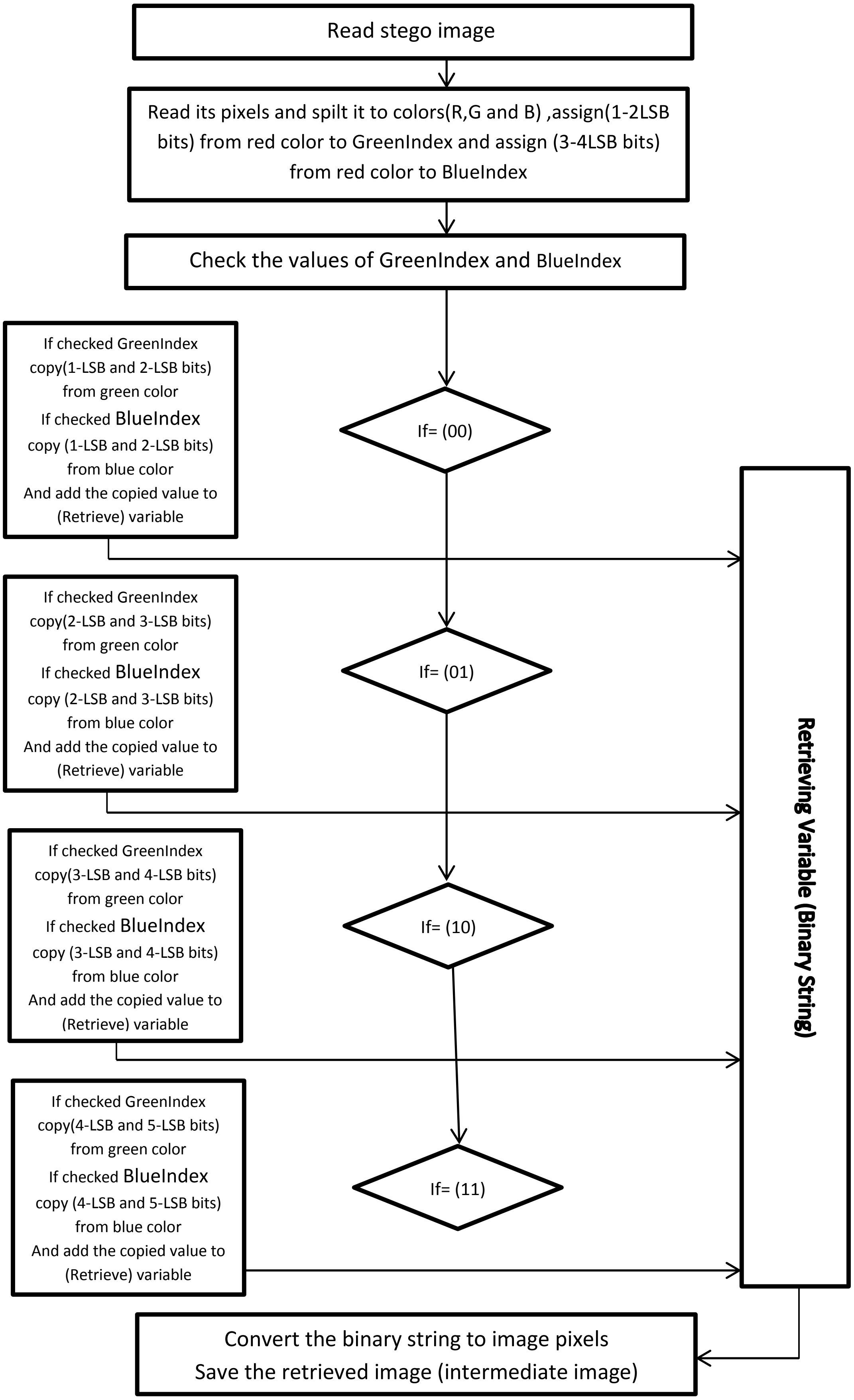

Read stego image
Read its pixels and spilt it to colors(R,G and B) ,assign(1-2LSB bits) from red color to GreenIndex and assign (3-4LSB bits) from red color to BlueIndex
Check the values of GreenIndex and BlueIndex
If checked GreenIndex copy(1-LSB and 2-LSB bits) from green color If checked BlueIndex copy (1-LSB and 2-LSB bits) from blue color And add the copied value to (Retrieve) variable
If= (00)
If checked GreenIndex copy(2-LSB and 3-LSB bits) from green color If checked BlueIndex copy (2-LSB and 3-LSB bits) from blue color And add the copied value to (Retrieve) variable
If= (01)
If checked GreenIndex copy(3-LSB and 4-LSB bits) from green color If checked BlueIndex copy (3-LSB and 4-LSB bits) from blue color And add the copied value to (Retrieve) variable
If= (10)
If checked GreenIndex copy(4-LSB and 5-LSB bits) from green color If checked BlueIndex copy (4-LSB and 5-LSB bits) from blue color And add the copied value to (Retrieve) variable
If= (11)
Retrieving Variable (Binary String)
Convert the binary string to image pixels
Save the retrieved image (intermediate image)

**Figure3.5: level two retrieving process (Pixel Intensity)**

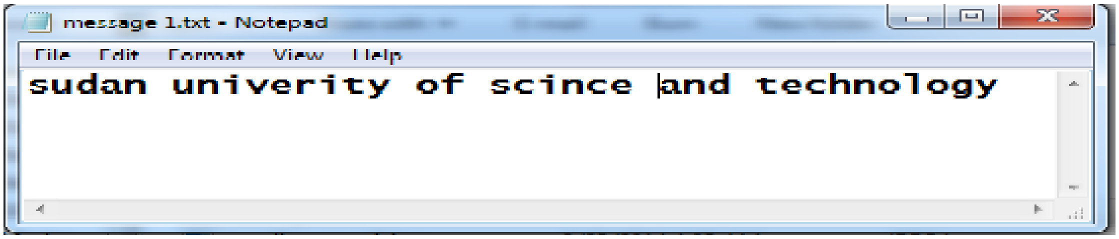
# CHAPTER 4

# RESULT AND DISCUSION

# 4.1 Results

Comparative analysis of multilevel image steganography (HS- LSB and pixel intensity based image steganography) has been done on basis of parameters like PSNR, MSE and embeds data size. Both grayscale and colored images (RGB) have been used for experiments.  Peak signal to noise ratio (PSNR) is used to compute how well the methods perform. PSNR computes the peak signal to noise ratio, in decibels, between two images. This ratio is used as a quality measurement between two images. If PSNR ratio is high then images are best of quality.in addition Mean Squared Error is the average squared difference between a reference image and a modified image (stego image). It is computed pixel-by-pixel by adding up the squared differences of all the pixels and dividing by the total pixel count.

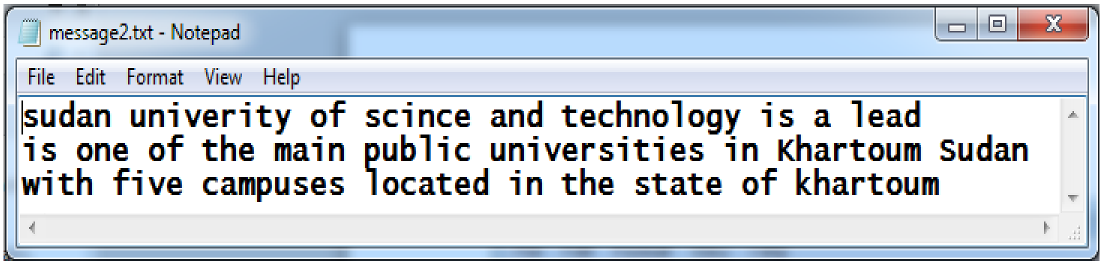
There are three different messages size have been used to embed them in different image size in the upper level of image steganography, the first message (first secret message) will be use shown in figure 4.1.



sudan univerity of scince and technology

**Figure 4.1: the first secret message (message1)**

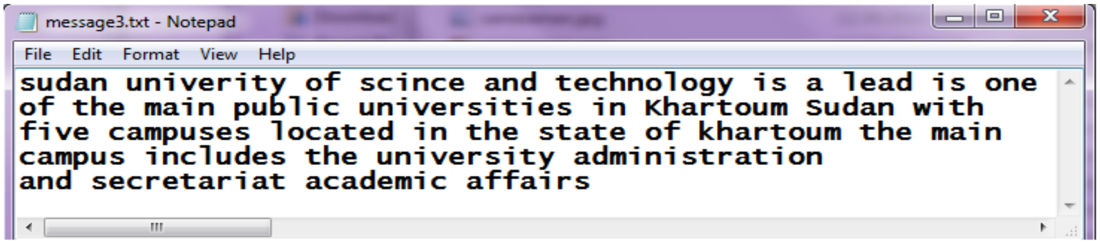
The size of first secret message is 320 bits and the size will be increase in the next secret message, the second secret message is shown in figure 4.2.



sudan univerity of scince and technology is a lead
is one of the main public universities in Khartoum Sudan
with five campuses located in the state of khartoum

**Figure 4.2: the second secret message (message2)**

The size of second secret message is 1328 bits and the size will be increase in the next secret message, the third secret message is shown in figure 4.3.



```
sudan univerity of scince and technology is a lead is one
of the main public universities in Khartoum Sudan with
five campuses located in the state of khartoum the main
campus includes the university administration
and secretariat academic affairs
```

**Figure 4.3: the third secret message (message3)**

The size of third secret message is 196608 bits (almost the maximum capacity).

After the upper level (level one - HS_LSB) is applied to the above secret messages the output is four stego images. each image concealing one of the secret messages. The first cover image is the monaliza image and is concealing (message1) as secret data; the size of the stego image is 442,368 bits. Figure 4.4 shows the monaliza _stego image.

The second cover image is Lenna image and is concealing (message2) as secret data; Figure 4.5 shows the Lena_stego image, the size of which is 7025459 bits.

Figure 4.6 shows cameraman_stego image with (message3) as embedded secret data and the size of which is 11536384 bits.

Finally the fourth stego image is the small icon image, in this image the maximum capacity of the secret message is attempted to be embedded, message3 seems to almost be the maximum capacity, the output is shown in Figure 4.7 icon_stego image and its size is 23760 bits.

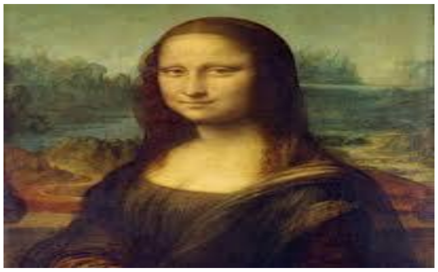


Figure 4.4: monaliza _stego1 from level one with secret message1

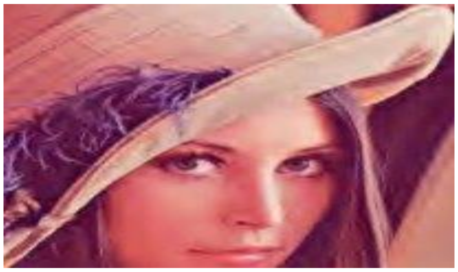


Figure 4.5: Lenna _tego2 from level one with secret message2

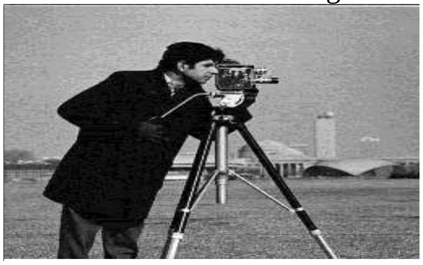


Figure 4.6: cameraman_stego3 from level one with secret message3

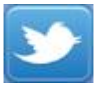


Figure 4.7: icon_image _stego4 from level one with secret message3

In the lower level (level two pixel intensity based image stegnography ) three images with different sizes and dimmentions have been used. the first image is the monaliza image with dimmention 360 × 397 used as a cover image, the secret data to be embedded in this cover image are the stego images which were the ouput of the upper level.firstly the stego image in fiqure 4.4 is used as secret data and is concealed in the monaliza cover image, the new stego image shows in figure 4.9.secondlay the stego image in fiqure 4.5 is used as secret data and is concealed in the monaliza cover image. the new stego image shows in figure 4.10.

in figure4.11 the monaliza cover image conceales image in figure 4.6 as the secret data.finally monaliza cover image concealed image in figure 4.7 as the secret data.

Fiqure 4.8: monaliza orginal image

Fiqure 4.9: monaliza _stego3 (embedded Data : monaliza _stego1)

Fiqure4.10. :monaliza _stego4 (embedded (embedded Data : Lenna _tego2)

Fiqure 4.11: monaliza _stego5 Data : cameraman_stego3)

Table 4.1 shows the experiment results of the monaliza _stego images and contains the PSNR and MSE values of stego images above. Figure 4.12 is a Diagram showing its PSNR values

| Secret message | Size of Secret message in bits | Embedded image | Size Embedded image | PSNR | MSE |
|---|---|---|---|---|---|
| Message3 | 196608 | icon_image _stego4 | 23760 | 66.245 | 0.0156 |
| message1 | 320 | monaliza _stego1 | 442,368 | 62.339 | 0.0599 |
| message2 | 1,328 | Lenna_stego2 | 7025459 | 61.2977 | 0.0735 |
| Message3 | 196608 | cameraman_stego3 | 11536384 | 56.8179 | 0.1860 |

**Table 4.1 Experimental results-1**

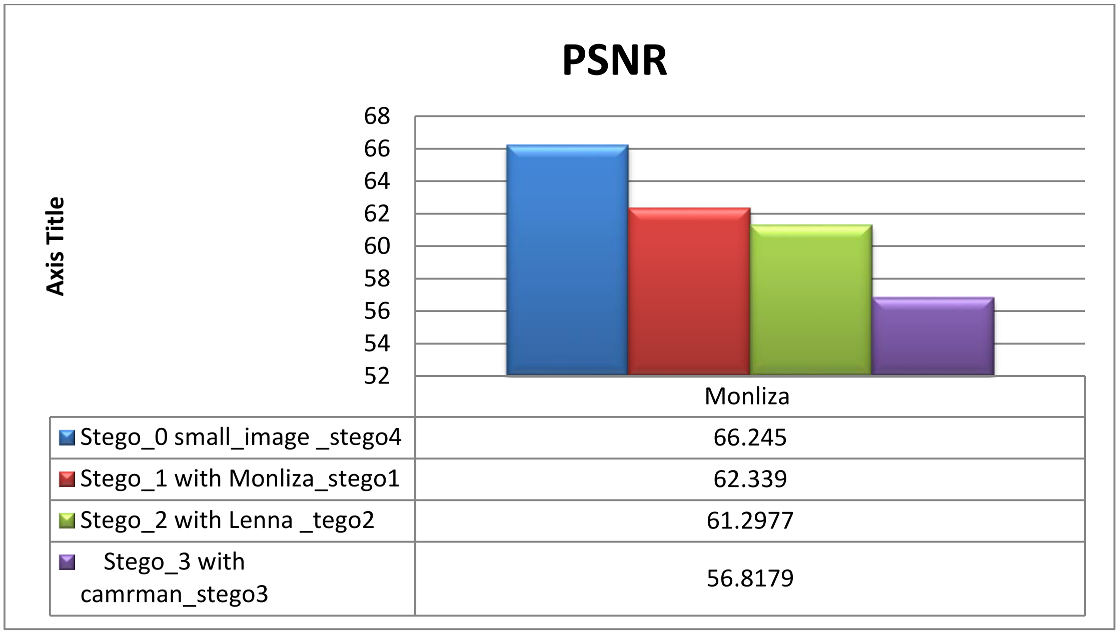


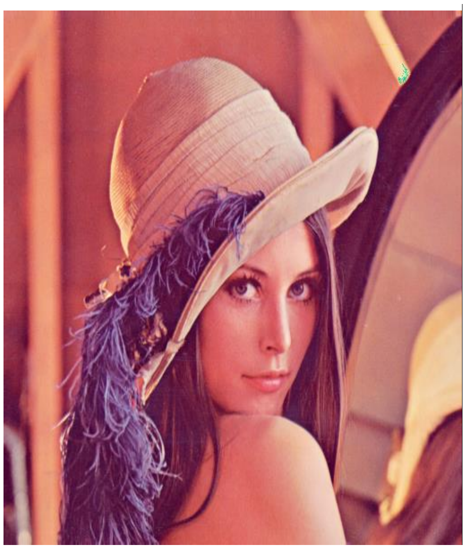
**Figure 4.12: PSNR values for monaliza_stego images**

the second image used in the lower level (level two pixel intensity based image stegnography ) is the Lenna image with dimmention 512 × 512 as a cover image. the secret data to be embedded in this cover image are the stego images ouputed from the upper level.

The image in figure 4.4 is embeded in the (first stego image) as secret data and the output stego image shown in figure 4.14.

In figure 4.15 the lenna cover image embeds image in figure 4.5 as secret data. Additionally Lenna cover image concealed image showed in figure 4.6 as secret data and the output stego image show in figure 4.16.

Finally Lenna cover image conceal image showed in figure 4.7 as a secret data.



Fiqure 4.13: Lenn orginal image

Fiqure 4.14: Lenna_stego6 (embedded Data : monaliza _stego1)

Fiqure 4.15 Lenna_stego7 (embedded Data : Lenna _tego2)

Fiqure 4.16: Lenna_stego8 (embedded Data : cameraman_stego3)

Table 4.2 shows the experiment result of monaliza _stego images and contains the PSNR and MSE values of stego images above. Figure 4.16 shows the Diagram for its PSNR values

| Secret message | Size of Secret message in bits | Embedded image | Size Embedded image | PSNR | MSE |
|---|---|---|---|---|---|
| message3 | 196608 | icon_image_stego4 | 23760 | 68.0012 | 0.0211 |
| message1 | 320 | monaliza _stego1 | 442,368 | 65.9313 | 0.0317 |
| message2 | 1,328 | Lenna_stego2 | 7025459 | 62.7867 | 0.3408 |
| message3 | 196608 | cameraman_stego3 | 11536384 | 59.5993 | 0.5219 |

**Table 4.2 Experimental results-2**

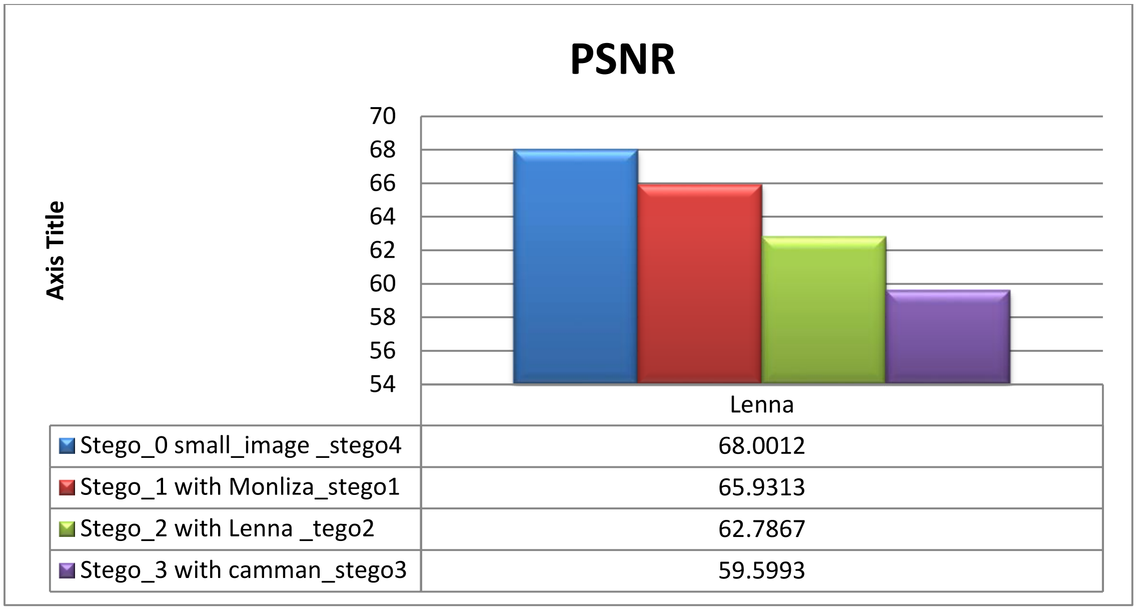


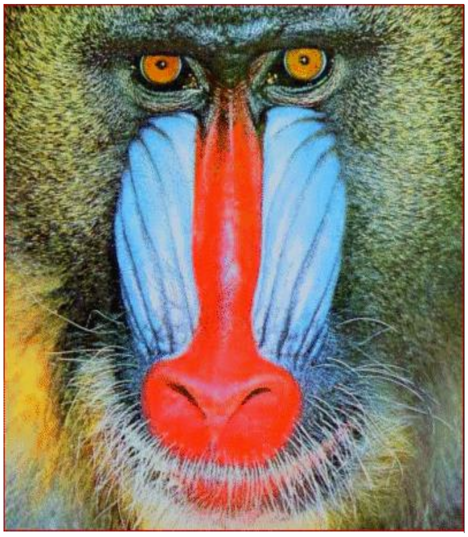
**Figure 4.17: PSNR values for Lenna stego images**

the third image is the baboon_face image with dimmentions 400 × 400 used as a cover image. the secret data to be embedded in this cover image are the stego images ouputed from the upper level.

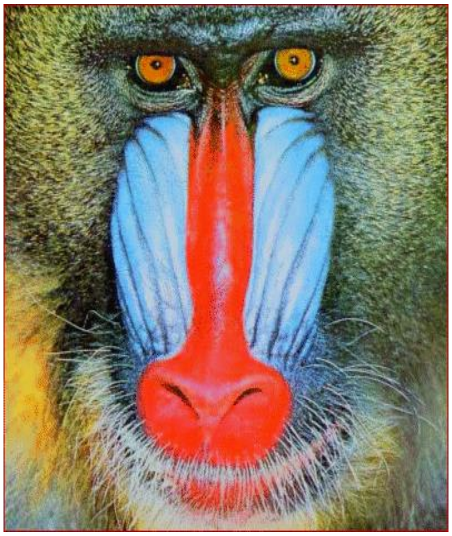
firstly the baboon_face cover image concealed stego image in fiqure 4.4 as the secret data, the new stego image shows in figure 4.18.

secondlay the baboon_face cover image concealed stego image in fiqure 4.5 as the secret data, the new stego image shows in figure 4.19.

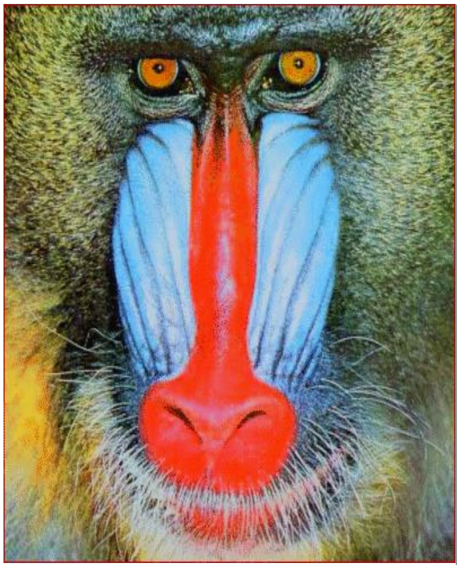
in figure 4.20 the monaliza cover image concealed image in figure 4.6 as a secret data, finally monaliza cover image concealed image in figure 4.7 as a secret data.



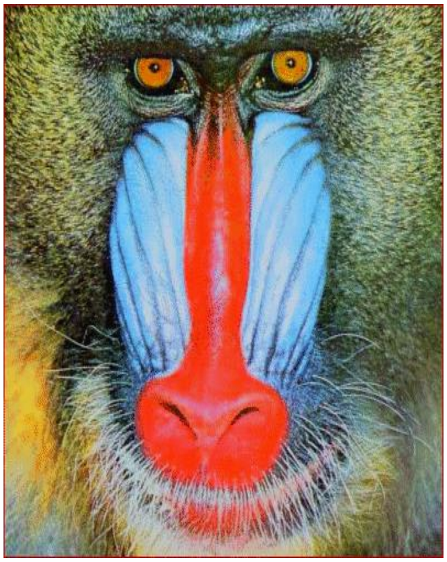
Fiqure 4.18:baboon_face orginal image



Fiqure 4.19: baboon_face _stego9 (embedded Data : monaliza _stego1)



Fiqure 4.20:baboon_face _stego11 (embedded Data : Lenna _tego2)

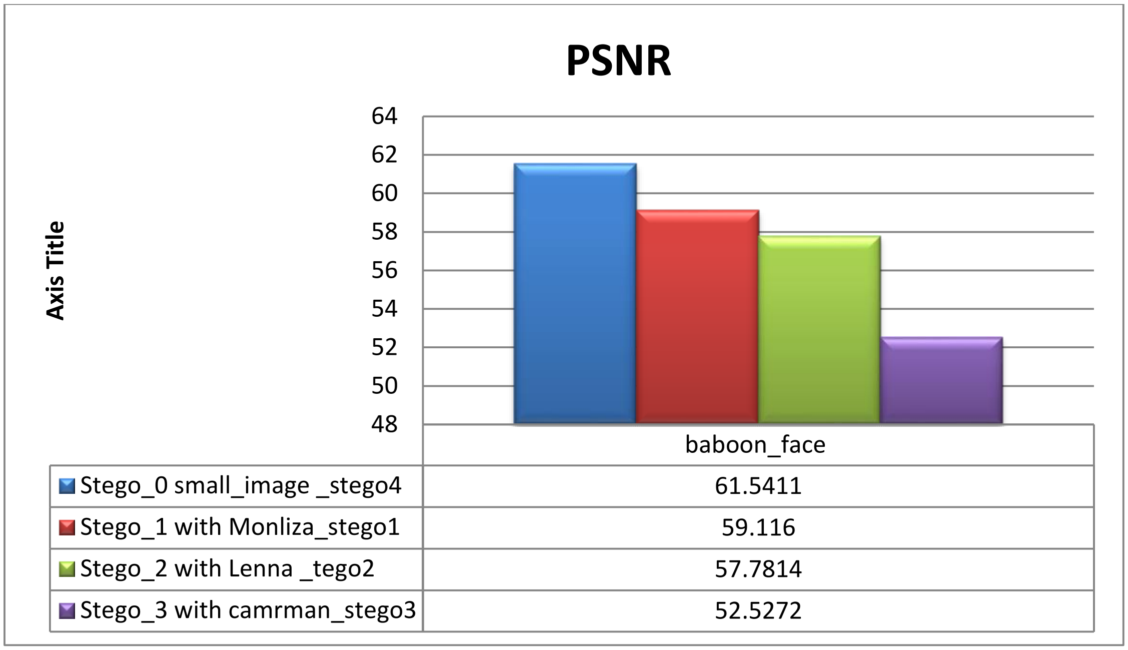


Fiqure 4.21:baboon_face _stego12 (embedded Data : cameraman_stego3)

Table 4.3 shows the experimental results of baboon_face stego images and contains the PSNR and MSE values of stego images above. Figure 21 show the Diagram for its PSNR values.

| Secret message | Size of Secret message in bits | Embedded image | Size Embedded image | PSNR | MSE |
|---|---|---|---|---|---|
| message3 | 196608 | icon_image_stego4 | 23760 | 61.5411 | 0.7151 |
| message1 | 320 | Monliza_stego1 | 442,368 | 59.116 | 0.9019 |
| message2 | 1,328 | Lenna_stego2 | 7025459 | 57.7814 | 0.9421 |
| message3 | 196608 | cameraman_stego3 | 11536384 | 52.5272 | 1.2914 |

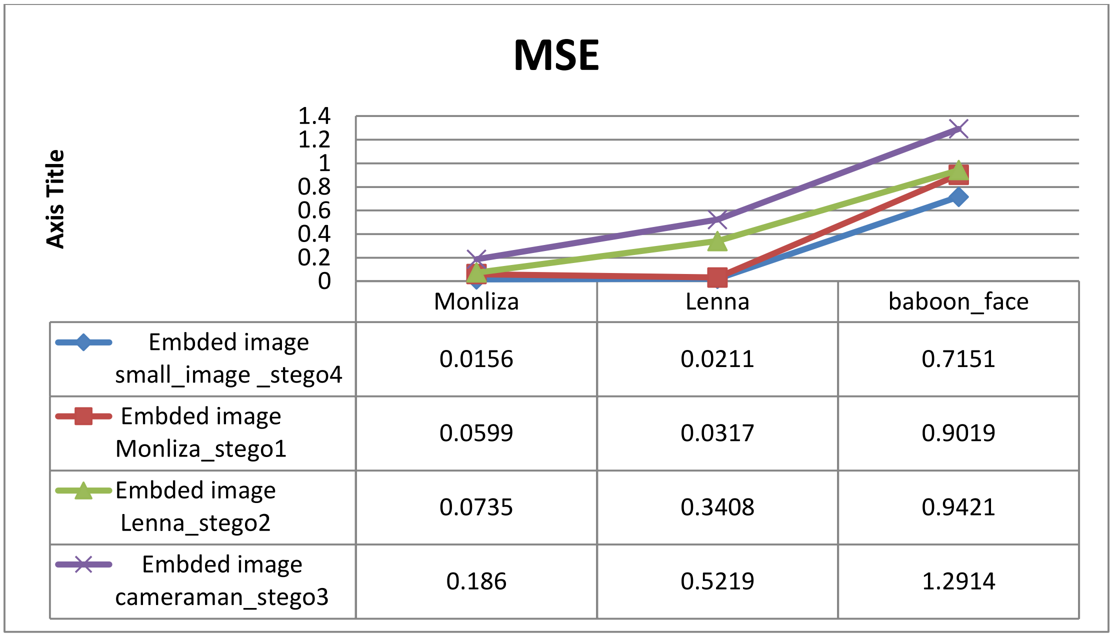
**Table 4.3 Experimental results-3**



| | baboon_face |
|---|---|
| 🟦 Stego_0 small_image _stego4 | 61.5411 |
| 🟥 Stego_1 with Monliza_stego1 | 59.116 |
| 🟩 Stego_2 with Lenna _tego2 | 57.7814 |
| 🟪 Stego_3 with camrman_stego3 | 52.5272 |

**Figure 4.22: PSNR values for baboon_face stego images**

After the completion of all experiments, the results of each experiment were taken and compared. Figure 4.22 shows the MSE values of the monaliza, Lenna and Baboon_face experiments.



| | Monliza | Lenna | baboon_face |
|---|---|---|---|
| ◆ Embded image small_image _stego4 | 0.0156 | 0.0211 | 0.7151 |
| ■ Embded image Monliza_stego1 | 0.0599 | 0.0317 | 0.9019 |
| ▲ Embded image Lenna_stego2 | 0.0735 | 0.3408 | 0.9421 |
| × Embded image cameraman_stego3 | 0.186 | 0.5219 | 1.2914 |

**Figure 4.23: MSE values for Monaliza, Lenna and Baboon_face experiments.**

Figure 4.24 shows the best PSNR values of each stego image monaliza, Lena and Baboon_face in this diagram appears the Lena cover image with figure 4.7 as a secret data has a best PSNR value in experiments, also monliza and baboon_face got a best PSNR value when it’s concealed image in figure 4.7 as a secret data. From digram 4.23 we can deduce the output stego image depend on size of concealed secret data, the size of level two cover image and lastly the variation between pixels values and secret data values.

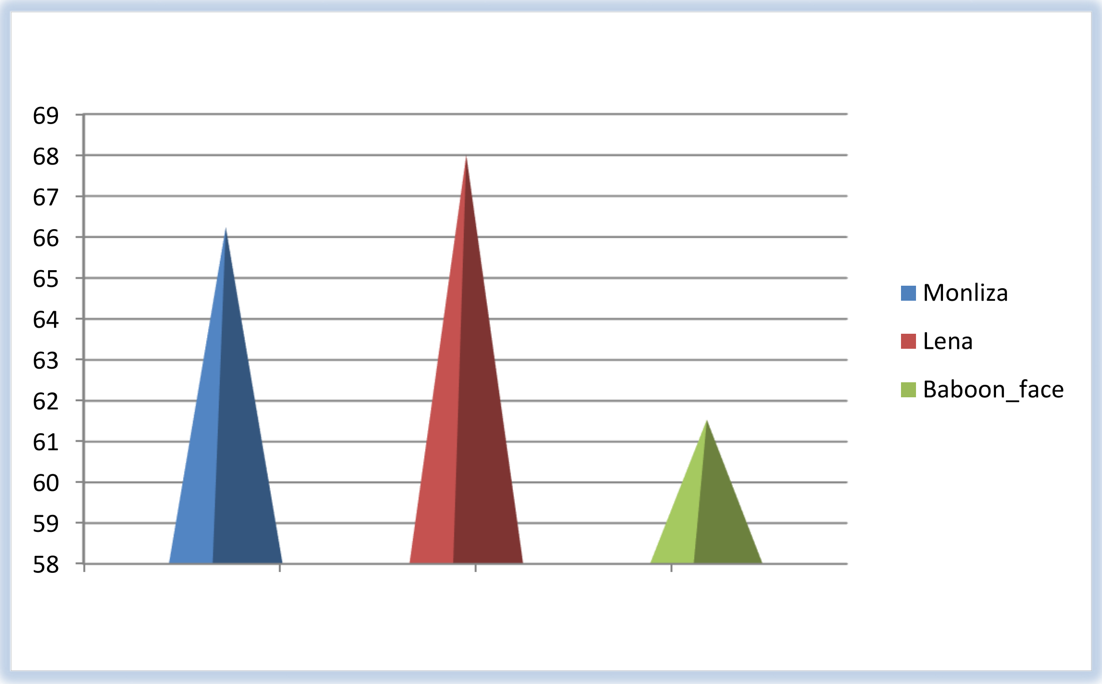


**Figure 4.24: best PSNR values of experiments stego images.**

# CHAPTER 5

# CONCLUSION AND FUTURE WORKS

# 5.1 Conclusion

The proposed model adds a level of security through the main theme of steganography: “hiding information in plain sight”. The cover object usually does not invite suspicion, since it looks similar to the original object for the general observer.

In this thesis, a new concept for performing hidden secret data, called Multi-Level Steganography for image steganography, was presented. MLS consists of at least two stenographic methods are utilized respectively, in such a way that one method (called the upper-level) as a carrier for the second one (called the lower-level).

The proposed method is two levels of image steganography, level one is applied by using modified Least Significant Bit (HS_LSB) Image steganography and it has been applied to enhance the performance of basic LSB algorithm and increase the algorithm strength because the basic LSB has considerably low robustness against attacks, in this level RGB and gray scale images(with .png and jepg extensions) have been used as a cover image and it’s conceal a secure data (English text) saved in .txt file to generate a stego image called (intermediate object) , level one called (upper level) in the proposed method regarding to MLS.

Level two has been applied using (pixel intensity based image steganography) algorithm to add another level of security to proposed method, in this level another RGB image (with .png and jepg extensions) has been used as a cover image and embed (the RGB or gray scale image output from level one) as a secure data and generate the new RGB image as stego image, level two in this prosed method called (lower level) regarding to MLS.

Multilevel Steganography have potential benefits, as it may enhance the confidentiality of the secret information by using two level image steganography in one the system and add more complexity to the steganography process through applying it in two levels.

Measuring the performance of proposed algorithm has been applied using many experiments and calculate two values of each experiment, the first value is Peak signal to noise ratio (PSNR) , this ratio is used as a quality measurement between two images. If PSNR ratio is high then images are best of quality, the second measurement value is Mean Squared Error is the average squared difference between a reference image and a modified image (stego image).

There are many experiments have been conducted through different size of secret messages (secret message one, two and three) utilized as a secret data in level one (upper level) and concealed in four RGB or gray scale images with different size as a cover image in this level, the output is stego images or (intermediate object) and it’s used as a secret data in level two (lower level).

The cover images have been used in level two are three RGB images (Monliza, Lena and baboon_face) with different size and dimmentions and each of them embed four stego images (intermediate object) from level one to produce stego images for level two.

After the completion of all experiments the PSNR and MSE values have been calculated for each experiment, the results of each experiment were taken and compared with each other's, after summarization of results the best PSNR value is 68.0012for Lenna stego_image , in addition the best MSE value is 0.0211and it got also from lenna stego_image.

Finally some results have been concluded from experimental results which explain the factors affecting in image quality after applying the proposed method, the most important factors are the size of level one stego image (intermediate image), whenever increased the quality of stego image probably affected negatively, in addition the size of level two cover image, whenever increased the quality effected positively, and lastly the variation between pixels values and secret data values.

# 5.2 Recommendations

- The proposed method can be used in military application for secure communications.

- Try to check the result of proposed algorithm using gray scale image in both levels to compare the performance results.

# 5.3 Future Work

1- Adding encryption algorithm to in Upper level to encrypt the secure text message to increase the security to proposed method.
2- Adding compression algorithm to compress the intermediate image after level one and before level two to enhance the performance of the proposed method.
3- Increase the System functionality to hide all other data types like audio, video not only text data and images.
4- Trying to enhance the performance of algorithms in both levels to increase the system capacity.

# References

[1] Singla, Deepak, and Rupali Syal. "Data Security Using LSB & DCT Steganography In Images." Editorial Board (2013): 359.

[2] Bandyopadhyay, Samir Kumar, and Barnali Gupta Banik. "Multi-Level Steganographic Algorithm for Audio Steganography using LSB Modification and Parity Encoding Technique." International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 1.2 (2012).

[3] Cvejic, Nedeljko, and Tapio Seppanen. "Reduced distortion bit-modification for LSB audio steganography." Signal Processing, 2004. Proceedings. ICSP'04. 2004 7th International Conference on. Vol. 3. IEEE, 2004.

[4] Channalli, Shashikala, and Ajay Jadhav. "Steganography an art of hiding data." arXiv preprint arXiv:0912.2319 (2009).

[5] Sridevi, R., A. Damodaram, and S. V. L. Narasimham. "EFFICIENT METHOD OF AUDIO STEGANOGRAPHY BY MODIFIED LSB ALGORITHM AND STRONG ENCRYPTION KEY WITH ENHANCED SECURITY." Journal of Theoretical & Applied Information Technology 5.6 (2009).

[6] T Morkel, JHP Eloff and MS Olivier, "An Overview of Image Steganography," in Proceedings of the Fifth Annual Information Security South Africa Conference ,Sandton, South Africa, June/July 2005).

[7] Johnson, Neil F., and Sushil Jajodia. "Steganalysis of images created using current steganography software." Information Hiding. Springer Berlin Heidelberg, 1998.

[8] Venkatraman, Abraham and Paprzycki “Significance of Steganography on Data Security”, Proceedings of the International Conference on Information Technology.

[9] Kharrazi, Mehdi, Husrev T. Sencar, and Nasir Memon. "Image steganography: Concepts and practice." Lecture Note Series, Institute for Mathematical Sciences, National University of Singapore (2004).

[10] Varghese, Jithy. Image Encryption and Compression using Embedding Technique. Diss. Christ University, 2010.

[11] Al-Najjar, Atef Jawad. "The decoy: multi-level digital multimedia steganography model." WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering. No. 12. World Scientific and Engineering Academy and Society, 2008..

[12] Kaur, Gurmeet, and Aarti Kochhar. "A steganography implementation based on LSB & DCT." International Journal for Science and Emerging (2012).

[13] Walia, Dr Ekta, Payal Jain, and Navdeep Navdeep. "An analysis of LSB & DCT based steganography." Global Journal of Computer Science and Technology 10.1 (2010).

[14] Parvez, Mohammad Tanvir, and AA-A. Gutub. "RGB intensity based variable-bits image steganography." Asia-Pacific Services Computing Conference, 2008. APSCC'08. IEEE. IEEE, 2008.

[15] Shobana, M., and R. Manikandan. "Efficient method for hiding data by pixel intensity." International Journal of Engineering and Technology 5.1 (2013): 75-81.

[16] Boneh, Dan, and Matthew Franklin. "Identity-based encryption from the Weil pairing." SIAM Journal on Computing 32.3 (2003): 586-615.

[17] Bhattacharyya, Souvik. "Data hiding through multi level steganography and SSCE." Journal of Global Research in Computer Science 2.2 (2011).