LOSSLESS DATA HIDING USING HISTOGRAM MODIFICATION TECHNIQUE FOR IMAGE AUTHENTICATION

A Project Report Submitted
in Partial Fulfilment of the Requirements
for the Degree of

BACHELOR OF TECHNOLOGY

By

Avneet Kaur (140210)
Shivani Santoshi (140248)
Smita Kumari (140252)
Sonali Negi (130252)

Under the Supervision of
Dr. Bhumika Gupta
Assistant Professor
G.B. Pant Institute of Engineering and Technology

Department of Computer Science and Engineering

G.B. PANT INSTITUTE OF ENGINEERING AND TECHNOLOGY

PAURI GARHWAL

June, 2018

CANDIDATE'S DECLARATION

We hereby declare that the work which is being presented in this project report entitled, **"Lossless data hiding using Histogram Modification Technique for Image Authentication"** submitted towards the partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology (Computer Science and Engineering)** is an authentic record of our own work carried out under the supervision and guidance of Dr. Bhumika Gupta, Assistant Professor, Department of Computer Science and Engineering, Govind Ballabh Pant Institute of Engineering and Technology, Ghurdauri, Pauri Garhwal.

We have not submitted the matter embodied in this thesis report for the award of any other degree.

Avneet Kaur	**Shivani Santoshi**	**Smita Kumari**	**Sonali Negi**
(140210)	**(140248)**	**(140252)**	**(130252)**

Date:

Department of Computer Science and Engineering

G.B. Pant Institute of Engineering and Technology

Ghurdauri, Pauri Garhwal-246194(Uttarakhand)

(A State Government Autonomous Institute)

CERTIFICATE

This is to certify that this dissertation entitled, "**LOSSLESS DATA HIDING USING HISTOGRAM MODIFICATION FOR IMAGE AUTHENTICATION"** submitted by **Avneet Kaur (140210), Shivani Santoshi (140248), Smita Kumari (140252), Sonali Negi (130252)** to the Department of Computer Science and Engineering, in partial fulfillment of requirements for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering,** during final year **(Aug 2017- June 2018)** is an authentic record of the work carried by them under our supervision and guidance.

We wish them success for their future endeavours.

Dr. Bhumika Gupta	Mr. V.M. Thakkar	Dr. Ashish Negi
(Project Guide)	(Project Coordinator)	(H.O.D. CSED)

ABSTRACT

Digital formats are gradually replacing their classical analog counterparts since they are easy to edit and modify. Also, it is very easy to maliciously modify digital media and create forgeries. Alternations to content may be malicious and the changes may affect the interpretation of the content. For example, malicious tampering of criminal evidence may result in a wrong verdict. Techniques that help us provide the authenticity of digital images are very vital whenever problems are raised about the integrity of an image. Thus, there is a need for image authentication for applications where we must be certain an image has not been modified. Authentication based on data hiding embeds the authentication information in the image, which makes it capable of authenticating itself. The embedded message is able to supply additional information about image, such as an authentication code, author's signature, and so on. Only when the embedded authentication information matches the extracted message, the image is deemed authentic. Inevitably, hiding information destroys the host image even though the distortion generated by hiding is imperceptible to eyes. In many authentication data hiding schemes, the distortion cannot be completely removed even when the image is deemed authentic. However, there are some applications for which any modification made to the image is intolerable, such as medical images, military images or images with a high strategic importance. Like medical images, where even slight changes are not accepted for a potential risk of a physician misjudging an image. Thus, it is desired to remove the embedding distortion if the image is deemed authentic. Lossless data hiding technique gives a solution to the problem of how to embed a large message in digital images in a lossless way so that after the embedded message is extracted, the image can be completely restored to its original state before the embedding occurred.

In this report, we present a lossless data hiding technique based on histogram modification for image authentication that is lossless in the sense that if the marked image is deemed authentic, the embedding distortion can be completely removed from the marked image after the embedded message has been extracted. This technique uses characteristics of the pixel difference to embed more data than other histogram based lossless data hiding algorithms.

ACKNOWLEDGEMENT

It is our proud privilege and duty to acknowledge the kind of help and guidance received from several people in preparation of this report. It would not have been possible to prepare this report in this form without their valuable help, cooperation and guidance.

We express our deepest sense of gratitude towards our supervisor **Dr. Bhumika Gupta**, Department of Computer Science & Engineering, G.B Pant Institute of Engineering and Technology, for her patience, inspirational guidance, constant encouragement, moral support and keen interest in our work. This work would not have been possible without her support and valuable suggestions.

My sincere thanks to **Mr. V.M. Thakkar**, Department of Computer Science and Engineering, G.B Pant Institute of Engineering and Technology, for his valuable suggestions and guidance throughout the period of this report. Our numerous discussions with him were extremely helpful. We hold him in esteem for guidance, encouragement and inspiration received from him.

We would like to thank our parents who patiently helped us as we went through our work. We would also like to thank all our friends for their help during our project work. We also have been lucky to have some of classmates who never hesitated to render their help at the time of some critical difficulties and spared their time whenever required.

At last I would like to thank God for his grace upon us.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Data hiding

Data hiding is the practice of concealing information or files within non secret data. The file containing the secret data is called the carrier. The modified carrier looks like original carrier. Best's carriers are images, audio, video files since everybody can send, receive, download them. The data is hidden not encrypted.

Data hiding, also called information hiding, plays an important role in multimedia security. The main purpose is to conceal messages in the original medium to protect intellectual property rights, to share secret message, or for content authentication. Nevertheless, the original medium will be permanently altered and cannot be completely reconstructed after the secret message is extracted if the recovering information is not provided. In some applications, such as medical imaging, remote sensing, and military imaging, a slight distortion is not allowed. Therefore, reversible data hiding techniques have become an important research topic in recent years.

Data hiding is a scheme in digital media to embed any secondary data into original information. It has found a variety of applications such as access control, annotation, authentication, etc. Data hiding is also found to be useful in sending secondary information in multimedia communication for achieving additional functionalities. A fundamental problem of all data hiding techniques is the embedding capacity i.e. the no of bits that can be embedded into the original signal. Data hiding algorithms can be classified into two categories: irreversible data hiding and reversible data hiding. In irreversible algorithms the host signal cannot be completely recovered. These algorithms are not suitable for the medical and military applications. In the reversible data hiding algorithms, original information can be completely recovered. The embedded data in the cover media data may be related to the image such as authentication data or author information. Data hiding is the process to hide data within a cover media. Therefore, the data hiding process contains two types of data, embedded data and cover media data. The data is transmitted by embedding it within images, which improves data security. The data hiding method in which the reversibility can be achieved is called Reversible data hiding. This technique is mainly used to improve the security of the cover image in encryption. Reversible image data hiding (RIDH) is one

method of data hiding technique, which makes sure that the cover image is reconstructed perfectly after the extraction of the embedded message. The reversibility of this method makes the data hiding approach attractive in the critical scenarios, e.g., military and remote sensing, law forensics, medical image sharing and copyright authentication, where the original cover image is required after reconstruction.

The Need for Data Hiding

- Covert communication using images (secret message is hidden in a carrier image)
- Ownership of digital images, authentication, copyright
- Data integrity, fraud detection, self-correcting images
- Traitor-tracing (fingerprinting video-tapes)
- Adding captions to images, embedding subtitles or audio tracks to video.
- Intelligent browsers, automatic copyright information, viewing a movie in a given rated version
- Copy control (secondary protection for DVD)

1.2 Steganography

1.2.1 What is Steganography?

Steganography comes from the Greek and literally means, "covered or secret writing". Although related to cryptography, they are not the same. Its intent is to hide the existence of the message, while cryptography scrambles a message so that it cannot be understood.

Steganography is one of various data hiding techniques, which aims at transmitting a message on a channel where some other kind of information is already being transmitted. This distinguishes Steganography from covert channel techniques, which instead of trying to transmit data between two entities that were unconnected before. The goal of Steganography is to hide messages inside other "harmless" messages in a way that does not allow any "enemy" to even detect that there is a second secret message present. The only missing information for the "enemy" is the short easily exchange able random number sequence, the

secret key, without the secret key, the “enemy” should not have the slightest chance of even becoming suspicious that on an observed communication channel, hidden communication might take place.

1.2.2 Introduction to Terms used

In the field of steganography, some terminology has developed. The term “cover” is used to Describe original data like audio, video and so on. When referring to audio signal steganography, the cover signal is sometimes called the host signal. The information to be hidden in the cover data is known as the embedded data. The “stego” data is the data containing both the cover signal and the embedded information. Logically the processing of putting the hidden or embedded data, into the cover data, is sometimes known as embedding. Occasionally, especially when referring to image Steganography, the cover image is known as the Container.

1.2.3 Steganography under Various Media

In the following three sections we will try to show how steganography can and is being used through the media of text, image, and audio. Often although it is not necessary, the hidden message will be encrypted. This meets a requirement posed by the “Kerckoff principle” in cryptography. This principle states that the Security of the system has to be based on the assumption that the enemy has full knowledge of the design and implementation details of the steganography system. The only missing information for the enemy is a short, easily exchangeable random number sequence, the secret key. Without this secret key, the enemy should not have the chance to even suspect that on an observed communication channel, hidden communication is taking place. Most of the software that we will discuss later meets this principle. When embedding data, it is important to remember the following restrictions and features:

1. The cover data should not be significantly degraded by the embedded data, and the embedded data should be as imperceptible as possible. (This does not mean the embedded data needs to be invisible; it is possible for the data to be hidden while it remains plain slight.)
2. The embedded data should be directly encoded into the media, rather than into a header or wrapper, to maintain data consistency across formats.

3. The embedded data should be as immune as possible to modifications from intelligent attacks or anticipated manipulations such as filtering and resampling.
4. Some distortion or degradation of the embedded data can be expected when the cover Data is modified. To minimize this, error correcting codes should be used.

The embedded data should be self-clocking or arbitrarily re-entrant. This ensures that the embedded data can still be extracted when only portions of the cover data are available. For example, if only a part of image is available, the embedded data should still be recoverable.

1.2.4 Steganography in Images

Steganography in images means to hide data in the image. Any plain text, ciphertext, images, or anything that can be embedded in a bit stream can be hidden in an image. Image steganography has come quite far in recent years with the development of fast, powerful graphical computers, and steganographic software is now readily available over the Internet for everyday users.

Some Guidelines to Image Steganography:

To a computer, an image is an array of numbers that represent light intensities at various points, or pixels. These pixels make up the image's raster data. An image size of 640 by 480 pixels, utilizing 256 colors(8 bits per pixel) is fairly common. Such an image would contain around 300 kilobits of data. Digital images are typically stored in either 24-bit or 8-bit per pixel files. 24-bit images are sometimes known as true color images. Obviously, a 24-bit image provides more space for hiding information; however, 24-bit images are generally large and not that common. A 24-bitimage 1024 pixels wide by 768 pixels high would have a size in excess of 2 megabytes. For large files, image compression is desirable. However, compression brings with it other problems.

Alternatively, 8 bit color images can be used to hide information. In 8 bit color images (such as GIF files), each pixel is represented as a single byte. Each pixel merely points to a color index table, or palette, with 256 possible colors. The pixel’s value then is between 0 and 255. The image software merely needs to paint the indicated color on the screen at the selected pixel position. If using an 8 bit-image as the cover image, many steganography experts recommend using images featuring 256 shades of grey as the palette, for reasons that will

become apparent. Grayscale images are preferred because the shades change very gradually between palette entries. This increases the image's ability to hide information. When dealing with 8-bit images, the steganographer will need to consider the image as well as the palette. Obviously, an image with large areas of solid color is a poor choice, as variances created by embedded data might be noticeable. Once a suitable cover image has been selected an image encoding technique needs to be chosen.

1.3 Image Steganography Techniques

Image steganography is the process of hiding the sensitive information into the cover image with no degradation of the image and providing better security so that unauthorized user cannot access the hidden information. Image steganography techniques are broadly classified into following:

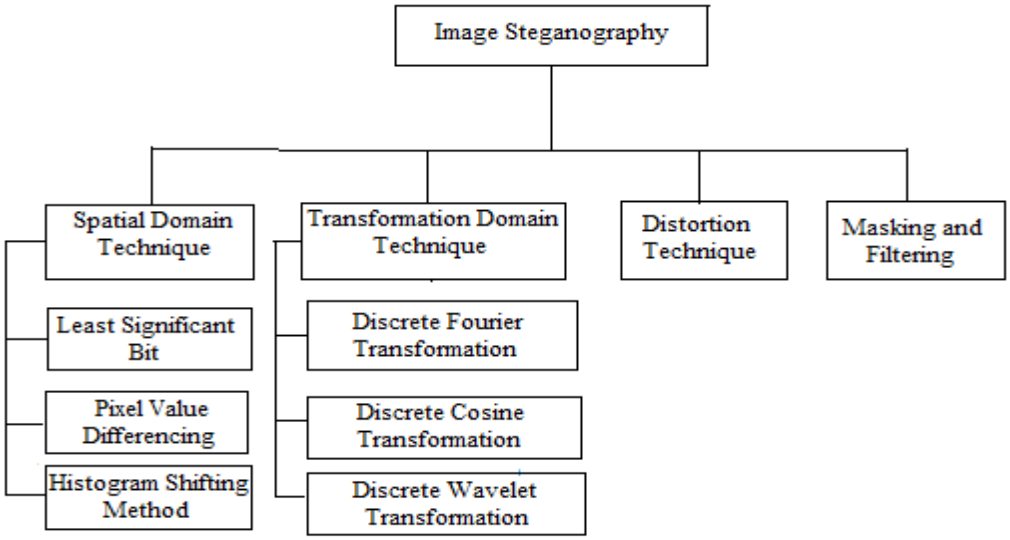


Figure 1.1 Image steganography techniques

In spatial domain steganography method, for hiding the data some bits are directly changed in the image pixel values. Most used method in this category is least significant bit. Spatial domain techniques are classified into following-

1.3.1 Least Significant Bit (LSB)

LSB insertion is a common and simple approach for embedding information in a cover file. Digital images used as cover file are mainly of two types- 24-bit images and 8-bit images. In 24-bit images we can embed three bits of information in each pixel. In 8-bit images, one bit

of information can be hidden into images. After applying the LSB algorithm the image obtained having secret message is called stego-image.

The least significant bit insertion method is probably the most well-known image steganography technique. It is a common, simple approach to embedding information in a graphical image file. Unfortunately, it is extremely vulnerable to attacks, such as image manipulation. A simple conversion from a GIF or BMP format to a lossy compression format such as JPEG can destroy the hidden information in the image. When applying LSB techniques to each byte of a 24-bit image, three bits can be encoded into each pixel. (As each pixel is represented by three bytes) Any changes in the pixel bits will be indiscernible to the human eye. For example, the letter A can be hidden in three pixels. Assume the original three pixels are represented by the three 24-bit words below:

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)
The binary value for the letter A is (101101101). Inserting the binary value of A into the three pixels, starting from the top left byte, would result in:
(00100111 1110100***0***11001001***1***)
(0010011111001000 11101001)
(1100100***1***00100111***0***11101001)
The emphasized bits are the only bits that actually changed. The main advantage of LSB Insertion is that data can be hidden in the least and second to least bits and still the human eye would be unable to notice it. When using LSB techniques on 8-bit images, more care needs to be taken, as 8-bitformats are not as forgiving to data changes as 24-bit formats are. Care needs to be taken in the selection of the cover image, so that changes to the data will not be visible in the stego-image. Commonly known images, (such as famous paintings, like the Mona Lisa) should be avoided.
In fact, a simple picture of your dog would be quite sufficient. When modifying the LSB bits in 8-bit images, the pointers to entries in the palette are changed. It is important to remember that a change of even one bit could mean the difference between a shade of red and a shade of blue. Such a change would be immediately noticeable on the displayed image, and is thus unacceptable. For this reason, data-hiding experts recommend, using, gray-scale palettes, where the difference between shades are not as pronounced. Alternatively images

consisting mostly of one color, such as the so-called Renoir palette, named because it comes from a 256 color version of Renoir's “Le Moulin de la Galette”.

1.3.2 Pixel Value Differencing (PVD)

In PVD method, gray scale image is used as a cover image with a long bit-stream as the secret data. It was originally proposed to hide secret information into 256 gray valued images. The method is based on the fact that human eyes can easily observe small changes in the smooth areas but they cannot observe relatively larger changes at the edge areas in the images. PVD uses the difference between the pixel and its neighbour to determine the number of embedded bits. The larger the difference amount is, the more secret bits can be embedded into the cover image.

The pixel-value differencing (PVD) scheme uses the difference value between two consecutive pixels in a block to determine how many secret bits should be embedded.. Most of the related studies focus on increasing the capacity using LSB and the readjustment process, so their approach is too conformable to the LSB approach. The pixel-value differencing (PVD) scheme provides high imperceptibility to the stego image by selecting two consecutive pixels and designs a quantization range table to determine the payload by the difference value between the consecutive pixels. Besides, it offers the advantage of conveying a large number of payloads, while still maintaining the consistency of an image characteristic after data embedding.

In recent years, several studies have been proposed to improve the PVD method.
There are two types of the quantization range table in Wu and Tasi's method. The first was based on selecting the range widths of [8, 8, 16, 32, 64, 128], to provide large capacity. The second was based on selecting the range widths of [2, 2, 4, 4, 4, 8, 8, 16, 16, 32, 32, 64, 64], to provide high imperceptibility Wu et al.’s presented a method combining pixel-value differencing and the LSB replacement method. Yang and Weng proposed a multipixel differencing method that uses three difference values in a four-pixel block to determine how many secret bits should be embedded, and Jung et al.’s proposed an image data hiding method based on multipixel differencing and LSB substitution. Liu and Shih proposed two extensions of the PVD method, the block-based approach and Haar-based approach, and Yang et al. proposed an information hiding technique based on blocked PVD. Liao et al.’s

proposed a four-pixel differencing and modified LSB substitution, and Yang et al.'s proposed a data hiding scheme using the pixel-value differencing in multimedia images.
Some studies focused on increasing the capacity using LSB or a readjusted process to improve the embedding capacity or image quantity. Few studies focus on the range table design. Besides, it is intuitive to design it using the width of the power of two.

1.3.3 Histogram Shifting Method

Histograms are used for graphical representation of image. It represents the pixel value and density at a particular pixel. It plots the pixel for each part of the image. A histogram is useful to identify pixel distribution, density of colors and tonal distribution. A histogram provides the highest and lowest pixel values in graph. Histogram shifting is the technique which is used to modify or to extract a certain group of pixels from a image. In histogram the highest value is called maxima and the lowest value is called minima. When the pixel value is modified for embedding process it should not cross the minima and maxima limit. There are several algorithm which supports histogram functionality in order to manipulate the image. The number of the pixels constituting the peak in the histogram of a cover image is equal to the hiding capacity because a single peak in a cover image is used.

Several histogram shifting techniques are enhanced by dividing the cover image into blocks to generate a respective peak for each block which provides more hiding capacity into the multiple blocks. Transformation domain methods hide message in the significant areas of the cover image which makes them more robust against various image processing operations like compression, cropping and enhancement. Many transformation domain methods exist. The basic approach used for hiding information is to transform the cover image, tweak the coefficients and then insert the transformation. Transformation domain techniques are broadly classified into following:

1.3.4 Discrete Fourier Transformation (DFT) Technique

In DFT all the insertion of hidden message is done in the frequency domain. It is a more complex way of hiding message into frequency domain of the image. This technique is important as it separates an image into the sine and cosine values. It converts space and time dependent information into the frequency based information. It is useful for a number of applications including image filtering and reconstruction as well as image compression. It does not include all frequencies that result to form an image but constitutes of only the set of

those samples which are sufficient to describe the original image. The DFT for the vector x having length n is some other vector y having length n

$y_{p+1}=\sum_{j=0}^{n-1} \omega^{jp} x_{j+1}$

Where ω signifies root n^{th} for unity

$$\omega = e^{-2\pi i/n}$$

1.3.5 Discrete Cosine Transformation (DCT) Technique

The DCT transforms the image from spatial to frequency domain and separates the image into spectral sub-bands with respect to visual quality of the image, i.e. low, middle and high frequency components. This transformation technique is useful for separating an image into different parts of differing significance (which is associated with the image's quality). It resembles the Fourier Transform Technique as it converts an image from its spatial domain into frequency domain. In this technique, for every color constituent, the JPEG format of image makes use of cosine transform to convert consecutive pixel blocks of size 8 x 8 into a count of 64 cosine coefficients each. For each 8x8 block having pixel value f(x,y), the coefficients f(u,v) are given as

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

Where

$$C(u)=\begin{cases}\frac{1}{\sqrt{2}}, & \text{if } u \leq 0\\ 1, & \text{if } u > 0\end{cases}$$

1.3.6 Discrete Wavelet Transformation (DWT) Technique

The Discrete Wavelet Transformation Technique is the new idea in the applications of the wavelets. The standard technique of storing in the least significant bit of pixel still applies but the only difference is the information is stored into the wavelet coefficients, instead of changing the bits of actual pixels in the image.

Wavelets are described as the functions obtained over a fixed interval and have zero as an average value. This transformation is an extremely necessary way to be used for signal

investigation as well as image processing, mainly for multi-resolution demonstration. It may crumble a signal into a number of constituents in frequency domain. 1-D DWT segments a cover image further into two major components known as approximate component and detailed component . A 2-D DWT is used to segment a cover image into mainly four sub components: one approximate component (LL) and the other three include detailed components represented as (LH, HL, HH).

1.3.7. Distortion Technique

In distortion techniques the information is stored by signal distortion. These techniques require the knowledge of the original cover image during the decoding process. The encoder applies series of modifications to the cover image and the decoder functions to check for the various differences between the original cover image and distorted cover image to recover the secret message. Using this technique, a stego object is created by the sender by applying a sequence changes to the cover image. This sequence of modification corresponds to a specific secret message required to transmit. The message is encoded at pseudo- randomly chosen pixels in the image. If the stego-image differ from the cover image at the given message pixel, the message bit is a “1” otherwise “0”. The sender can modify the “1” value pixels in such a way that the statistical properties of the image should not affected. The receiver must have access to the original cover for retrieving the message; it limits the benefits of this technique. In every steganography techniques, the cover image should never be used more than once. If an attacker has access to the cover image the secret message can be easily detected by attacker from the stego-image by cropping, scaling or rotating it. In some cases, if the message is encoded with error correcting information, the change can even be reversed and the original message can be recovered.

1.3.8. Masking and Filtering

This technique is usually applied on 24 bits or grayscale images, uses a different approach to hiding a message. It hides information by marking an image, similar to paper watermarks. This technique actually extends an image data by masking the secret data over the original data as opposed to hiding information inside of the data. These techniques embed the information in the more significant areas of the image than just hiding it into noise level. Watermarking techniques can be applied on the image without the fear of its destruction due to lossy compression as they are more integrated into the image.

This method is more robust than LSB modification with respect to compression and different kinds of image processing since the information is hidden into the visible parts of the image. The main drawback of this technique is that it can only be used on gray scale images and restricted to 24-bit images.

1.4 Compression

In recent years, the development and demand of multimedia product grows increasingly fast, contributing to insufficient bandwidth of network and storage of memory device. Therefore, the theory of data compression becomes more and more significant for reducing the data redundancy to save more hardware space and transmission bandwidth. In computer science and information theory, data compression or source coding is the process of encoding information using fewer bits or other information-bearing units than an un encoded representation. Compression is useful because it helps reduce the consumption of expensive resources such as hard disk space or transmission bandwidth. Image compression is an application of data compression that encodes the original image with few bits. The objective of image compression is to reduce the redundancy of the image and to store or transmit data in an efficient form. The main goal of such system is to reduce the storage quantity as much as possible, and the decoded image displayed in the monitor can be similar to the original image as much as can be.

Two kinds of image compression are lossless and lossy compression. Both methods save storage space but have differing effects on any uncompressed hidden data in the image. Lossy compression, as typified by JPEG (Joint Photographic Experts Group) format files, offers high compression, but may not maintain the original image's integrity. This can impact negatively on any hidden data in the image. This is due to the lossy compression algorithm, which may “lose” unnecessary image data, providing a close approximation. To high-quality digital images, but not an exact duplicate. Hence, the term “lossy”compression. Lossy compression is frequently used on true-color images, as it offers high compression rates.

Lossless compression maintains the original image data exactly; hence it is preferred when the original information must remain intact. It is thus more favoured by steganographic techniques. Unfortunately, lossless compression does not offer such high compression rates as lossy compression. Typical examples of lossless compression formats are CompuServe’s GIF(Graphics Interchange Format) and Microsoft's BMP (Bitmap) format.

Lossy Compression	Lossless Compression
1. The technique involves some loss of information.	1. Involves no loss of information
2. Data that has been compressed using this technique can't be recovered and reconstructed exactly	2. If data has been (lossless) compressed, the original data can be recovered from the compressed data.
3. Used for application that can tolerate difference between the original and reconstructed data.	3. Used for application that can't tolerate any difference between original and reconstructed data.
4. In return for accepting this distortion in reconstructed data we obtain high compression rate	4. No loss in information so compression rate is small.
5. Sound and Image compression uses lossy compression.	5. Text compression uses lossless compression.
6. More data can be accommodated in channel.	6. Less data can be accommodated in channel.
7. Distortion	7. Distortion less
8. E.g. Telephone System, Video CD	8. E.g. Fax Machine, Radiological Imaging

Table 1.1 Differences between lossy and lossless compression

1.5 Image authentication

Authentication is the process of recognizing a user's identity. It is the mechanism of associating an incoming request with a set of identifying credentials. The credentials provided are compared to those on a file in a database of the authorized user's information on a local operating system or within an authentication server

Image authentication has recently gained great attention due to its importance for a large number of multimedia applications. Digital images are increasingly transmitted over non-secure channels such as the Internet. Therefore, military, medical and quality control images must be protected against attempts to manipulate them; such manipulations could tamper the decisions based on these images.

To protect the authenticity of multimedia images, several approaches have been proposed. These approaches include conventional cryptography, fragile and semi-fragile watermarking and digital signatures that are based on the image content. Methods are classified according

to the service they provide, that is strict or selective authentication, tamper detection, localization and reconstruction capabilities and robustness against different desired image processing operations. Furthermore, we introduce the concept of image content and discuss the most important requirements for an effective image authentication system design.

Image authentication can be divided in two groups: strict and selective authentication. Strict authentication is used for applications where no modifications in the protected image are allowed. On the other hand, selective authentication is used especially when some image processing operations must be tolerate such as compression, different filtering algorithms and/or even some geometrical transformations. For strict authentication, solutions including conventional cryptography and fragile watermarking provide good results that satisfy users, even though some researches still need to be done in order to enhance localization and reconstruction performances of the image regions that were tampered. Selective authentication on the other hand, uses techniques based on semi-fragile watermarking or image content signatures to provide some kind of robustness against specific and desired manipulations. Results are satisfying, but the problem is far from being solved. Researches are now more concentrated in the area of image content signatures and the number of proposed solutions has increased rapidly in last year's due to the large number of applications. Nevertheless, more sophisticated solutions that allow combinations of several desired modifications are still to be discovered.

CHAPTER 2

LITERATURE REVIEW

2.1 Lossless Data Hiding Based on Histogram Modification for Image Authentication

Chin-Chen Chang, Kuo-Nan Chen & Wei-Liang Tai [2008], IEEE/IFIP International Conference on Embedded and Ubiquitous Computing.

Abstract

Lossless data hiding enables the embedding of messages in a host image without any loss of content. In this paper, a lossless data hiding technique is presented based on histogram modification for image authentication that is lossless in the sense that if the marked image is deemed authentic, the embedding distortion can be completely removed from the marked image after the embedded message has been extracted. This technique uses characteristics of the pixel difference to embed more data than other histogram based lossless data hiding algorithms. A histogram shifting technique is used to prevent overflow and underflow problems. Performance comparisons with other existing lossless data hiding schemes are provided to demonstrate the superiority of the proposed scheme.

Summary

Ni et al. firstly introduced a novel histogram based reversible data hiding technique where the message is embedded into the histogram bin. Pairs of peak points and zero points are used to achieve low embedding distortion but low hiding capacity. Fallahpour et al. presented the block-based histogram modification scheme. Lee et al. proposed a reversible hiding scheme based on histogram modification of difference images. To increase hiding ability, they presented an efficient extension of the histogram modification technique by considering the pixel difference instead of simple pixel value. They also exploit a histogram shifting technique to prevent problems that are raised about overflow and underflow. As a result, characteristics of the pixel difference enable the proposed algorithm to obtain the higher peak point to embed a large amount of message. It also briefly describes prior relevant Ni et al.'s histogram modification technique. It contains a detailed process of the proposed algorithm. They experimentally studied the relationship between the capacity and distortion, and the effect of variant images on the capacity. Performance comparison with existing histogram-based lossless hiding schemes is also given. Finally, the paper is concluded outlining the future investigations.

Conclusion

We present an efficient extension of the histogram modification technique by considering the difference between adjacent pixels instead of simple pixel value. Characteristics of the pixel difference that is almost Laplacian distributed are used to accommodate a large payload.

Thus, the proposed scheme is able to provide high capacities at invertible distortion. Also, it can be easily modified for compressed image formats, such as JPEG, MPEG, and JPEG2000. The distribution of frequency coefficients may be almost Laplacian distributed due to quantization since the embedding must be performed in the transform domain. As a result, the proposed scheme can be generalized to other data types than images.

2.2 Efficient data hiding scheme using lossless data compression and image steganography

Naresh Kumar, Rahul Jain [2012], International Journal of Engineering Science and Technology (IJEST).

Abstract: Steganography is an art of hidden communication in which secret message is embedded into a cover image. It has many applications like Online transactions, military communication etc. In this paper, they have proposed a data hiding scheme using image steganography and compression. This scheme can be applied to gray scale as well as colour images. This scheme improves the data hiding capacity of the image as compared to other existing image steganography methods while retaining the quality of the image after embedding the secret message into it. The improved embedding capacity of the image is possible due to pre-processing the secret message in which a lossless data compression technique is applied.

Summary: Image steganography is a process that hides the message into cover-image and generates a stego-image. That stego-image then sent to the receiver without anyone else knowing that it contain the hidden message. The receiver can extract the message with or without stego-key that depends on the hidden scheme. Image steganography techniques can be divided into two groups: Image Domain also called spatial domain and Transform Domain also called frequency domain. Spatial domain techniques embed information in the intensity of the original image pixels directly. Basically least significant bit (LSB) method is used where it replaces the least significant bit of original pixel with the message bit. Transform domain also known as frequency domain where images are first transformed then the message is embedded in the image. Discrete cosine transformation (DCT) technique is used in JPEG images to achieve compression. DCT is a lossy compression transform where the cosine values cannot be generated as original, because DCT alter values (e.g. 8.667 to 9) to hide the information.

Conclusion: In this paper existing image steganography techniques were explored. The researchers proposed an efficient image steganography technique. In image steganography, image is used as a carrier for transmission of the secret information or data. The image used can be either gray scale or colour image. In this technique data is firstly pre-process. This pre-processing reduces the size of the data by a significantly great amount. This pre-processed data is then embedded into the LSBs of the pixels of the image depending upon the intensity

of the pixel values. The algorithm is targeted to achieve very high image embedding capacity into the cover image and more security of the secret data. The proposed technique performs better than MKA. It has high PSNR value and low MSE value as compared to MKA. This pre-processing reduces the size of the secret data by a significant amount and thus permits more data into the same image. The embedding capacity of the proposed technique is very high as compared to MKA. This method has good imperceptibility, sufficient payload and has high security. Data security and high embedding capacity is there due to the pre-processing of the data before embedding into the cover image. This method does not require the original image while extracting the secret data from stego image. The work can be extended to provide an alternative strategy for data compression which can further reduce the size of the data. Efficient cryptographic techniques can also be used along with the steganographic techniques to provide more security to the data.

2.3 Hiding the Military Secret Message by Reversible Data Hiding

Alekhya Orugonda, S.Rajan [2013], International Journal of Engineering and Innovative Technology (IJEIT)

Abstract: A data hiding is a technique that is used for embedding the important information into images. In reversible data hiding, the degradation of the original image is not allowed, such as medical imagery and military imagery. The secret data is embedded in the compression domain and the receiver wants to store the image in a compression mode to save storage space. An encoding message can be compressed and encrypted by the secrete key. A decode message consists of secret data that can be viewed by the encrypted key. This paper proposed a MSM (Military Secret Message) method which restores the important data.

Summary: The main aim of data hiding is to enhance communication security by embedding secret messages into an inconspicuous carrier and there by transmitting them to receiver. The embedding process will usually introduce permanent distortion and reconstructed from the marked image. The uncompressed image using an encryption key to produce an encrypted image and then a data hider embeds additional data into the encrypted image using a data-hiding key. To apply reversible data hiding to encrypted images by wishing to remove the embedded data before the image decryption. The information is embedded data that it is perceptually and statistically undetectable .Data embedding also provides an embedding important control and information. Reversible data embedding, which is often referred to as lossless data embedding, is a technique that embedding the data into an image in a reversible manner. The original uncompressed image using an encryption key to produce an encrypted image and then a data hider embeds additional data into the encrypted image. To reduce transmission time the data compression is necessary. The encrypted image can be compressed by using several techniques. In Lossy compression of an encrypted image flexible compression ratio is done. The data exchange involves transmission of different types of data format such as medical images, texts, and graphs. Data hiding techniques can be also used for authentication. As an effective means for security protection, encryption converts the

ordinary signal into rough data, so that the traditional signal processing usually takes place before encoding or after decoding.

Conclusion: In this paper, a Military Secret Message method was proposed for transmitting the secured message.

2.4 Histogram Shifting based reversible data hiding

Lincy Rachel Mathews1, Arathy C. Haran V [2014], International Journal of Engineering Trends and Technology (IJETT)

Abstract: This paper presents a reversible data hiding scheme based on histogram modification. Distribution of pixel differences used to achieve high hiding capacity. In order to solve the issue of communicating the multiple peak points to the recipients, a binary tree structure is adopted. Data embedding performed after block division facilitates the marked image quality. Histogram shifting technique prevents overflow and underflow problems.

Summary: Most of the data hiding techniques are not reversible completely. The well known LSB technique is not completely reversible due to bit replacement without "memory". Tian et al proposed the data hiding technique in which data bits are embedded by computing the difference between the pixel pairs. Location map which is essential for image restoration is embedded along with the data bits. Coltuc et al proposed a data hiding technique in which embedding of data bits is done by taking the RCM transform of the image. Reversible contrast mapping is an integer transform which is invertible even if some LSB bits are lost. Histogram based data hiding technique embeds the data in the cover media by shifting the histogram of the image. Histogram technique finds peak or zero points in the histogram and data embedding is done by shifting these peak and zero points. This technique yields higher data hiding capacity with low distortion. Histogram based reversible data hiding method was introduced by Ni *et al*. in [8], where message is embedded within the histogram. Embedding is done by shifting the peak and zero points of the histogram. Also histogram shifting technique prevents overflow and underflow problem. Overflow is the condition that the gray value exceeds above 255. Underflow is the condition that the gray value falls below 0. One of the major issues associated with all these techniques is that the peak and zero points needs to be embedded along with the data during image embedding for the complete restoration of the image. The proposed algorithm in this paper solves this problem of communicating the peak and zero points by introducing a binary tree structure.

Conclusion: This paper proposes a new algorithm for data hiding in which histogram modification technique is done by considering the pixel difference rather than a single pixel. One of the main drawbacks of all the histogram modification techniques is the issue of communicating the multiple peak and zero points. This drawback is overcome in this work using the binary tree structure. Number of peak points is determined by the tree level L. Number of bits that can be embedded is determined by the number of pixels associated with the peak points. Also, in this work data embedding is performed after dividing the image into

blocks. This helps to distribute the message bits along the whole image and also improves the hiding capacity. This work can be extended to color images.

2.5 An Improvised Lossless Data-hiding Mechanism for Image Authentication Based Histogram Modification

Shaik Shaheena, B. L. Sirisha[2016], International Journal Of Professional Engineering Studies

Abstract: Digitalization has implemented in all security based applications ranging from home based security to data hiding. Although tremendous progress has been made in past years but still achieving lossless data hiding without any loss of information is concerned area. An optimized lossless data hiding algorithm based on histogram modification technique is proposed in this paper for authentication and the proposed algorithm handles the embedding distortion in effective way. Embedding distortion removal from image is unresolved issue in data hiding and it is handled in efficient manner in this work by taking pixel differences (underflow and overflow) into consideration. Underflow and overflow is prevented by histogram shifting technique and experimental results shows better performance over conventional state-of-art methods.

Summary: To increase hiding ability, in this paper an efficient extension of the histogram modification technique by considering the pixel difference instead of simple pixel value is proposed along with the histogram shifting technique to prevent problems that are raised about overflow and underflow. As a result, characteristics of the pixel difference enable the proposed algorithm to obtain the higher peak point to embed a large amount of message.

Conclusion: Pixel difference between adjacent pixels is implemented in this paper for histogram modification technique to attain simple pixel value. Hiding large amount of information in image is concerned area in digital image processing. Laplacian distributed is used in proposed method to hide more payload than traditional methods which help to increase the data hiding capacity at invertible distortion and this process adaptively modify according to compressed image formats in effective way. Two operations namely embedding and extraction are carried out to support different domains to achieve good authentication.

2.6 Reversible Data Hiding In Image- A Literature Survey

Arun Kumar.M.N, Krishnapriya K.R [2017], International Journal of Advanced Research in Computer Science

Abstract: Security must be provided for the transmission of confidential and sensitive data over the network. To increase the security of data transmission, data hiding can be performed in encrypted image .Therefore the security of image and embedded data is maintained. The

hidden data and the cover image can be restored thereby reversibility can be achieved, which is termed as Reversible Data Hiding. By using combined lossless and reversible data hiding, the embedded data and cover image can be retrieved. This paper focus on the various works in the area of reversible data hiding and various RDH techniques are discussed.

Summary: In this paper we studied about various works that have been proposed in the area of combination of steganography and cryptography. For high security of data several approaches like steganography, Data Hiding and cryptography can be used. In Cryptography the study of various mathematical methods and various aspects of Information Security like confidentiality and authentication of data. In cryptography a plain text is encrypted into cipher text and that can be look like a meaningless string of character whereas in case of steganography, cover media contains the hidden data that looks like normal image.

Conclusion: Reversible data hiding in encrypted image is getting more attention these days because of security maintaining requirements. Reversible data hiding in encrypted image is a powerful technique to improve the security of data. Data hiding in encrypted images provides more security for the data as cryptography and steganography are performed. By combining lossless and reversible data hiding techniques, more efficient data embedding can be done in encrypted images. The concept of data hiding and their applications in the security of digital data communication across network is studied in this paper and technical survey of recent methods in reversible data hiding is presented.

CHAPTER 3

PROBLEM DEFINITION

3.1 Problem

Now days, it is very easy to modify digital media and create forgeries. Alternations to content may be malicious and the changes may affect the interpretation of the content. Techniques that help us provide the authenticity of digital images are very vital whenever problems are raised about the integrity of an image. Thus, there is a need for image authentication for applications where we must be certain an image has not been modified. Authentication based on data hiding embeds the authentication information in the image, which makes it capable of authenticating itself. The embedded message is able to supply additional information about image, such as an authentication code, author's signature, and so on. Only when the embedded authentication information matches the extracted message, the image is deemed authentic.

However hiding information destroys the host image even though the distortion generated by hiding is imperceptible to eyes. In many authentication data hiding schemes, the distortion cannot be completely removed even when the image is deemed authentic. However, there are some applications for which any modification made to the image is intolerable, such as medical images, military images or images with a high strategic importance. Thus, it is desired to remove the embedding distortion if the image is deemed authentic. The problem is to embed the message in the image at the sender side and to recover both the embedded data and the image in its original form at the receiver side without any distortion. The image should be deemed authentic that is the embedded message should be equal to the extracted message. In this project we have implemented the lossless data hiding algorithm which is based on histogram modification technique.

3.2 Solution

Ni et al firstly introduced a lossless data hiding technique based on histogram modification where we generate the image histogram of a given image and seek a peak point and a zero point. A peak point corresponds to the grayscale value which the maximum number of pixels in the given image assumes, whereas a zero point corresponds to the grayscale value which no pixel in the given image assumes. The amount of message that can be embedded into an image equals to the number of pixels which are associated with the peak point. But this technique does not work well whenever an image has an equal histogram.

To improve hiding capacity, we present an efficient extension of the histogram modification technique by considering the difference between adjacent pixels instead of simple pixel value. Since the pixel grayscale values in a local area are often highly correlated and spatial redundancy, the distribution of pixel difference has a prominent maximum which shows that the difference value is expected to be very close to zero, thus, there are a lot of

candidates for embedding data. This observation was used in the making of lossless data hiding algorithm. Thus lossless data hiding algorithm using histogram modification technique is the solution of this problem.

CHAPTER 4

LOSSLESS DATA HIDING ALGORITHM AND ITS IMPLEMENTATION

4.1. Histogram Shifting Technique

1. First of all, we generate the image histogram of a given image.

2. In the histogram, we first seek a peak point and a zero point. A peak point corresponds to the grayscale value which the maximum number of pixels in the given image assumes, whereas a zero point corresponds to the grayscale value which no pixel in the given image assumes. Like the grayscale "Lena" image (512×512×8), its image histogram is shown in Fig. 1, where the peak point is at 154 and the zero point is at 255.

Let P be the value of peak point and Z be the value of zero point. We shift the range of the histogram, [P+1, Z-1], to the right-hand side by 1 to leave the zero point at P+1. Such as the "Lena" image, the pixels within the range [155, 254] are increased by 1, thus, the pixel value 155 in the histogram is empty. Once a pixel with value P is encountered, if the message bit to be embedded is "1," increase the pixel value by 1. Otherwise, no change is made. Obviously, the amount of message that can be embedded into an image equals to the number of pixels which are associated with the peak point.

3. The data extraction is the reverse process of data hiding. When a pixel with value P+1 is met, message bit "1" is extracted and its value is decreased by 1. Also, when a pixel with value P is met, bit "0" is extracted. After all message bits have been extracted, the range of the histogram, [P+2, Z], is shifted to the left-hand side by 1. Another point we should mention is that, zero point defined above may not exist in some image histograms. Thus, a minimum point that is defined as the grayscale value which the minimum number of pixels in the given image assumes is often used in place of the zero point. However, the grayscale value and coordinate of the pixel that is associated with the minimum point need to be recorded as overhead bookkeeping information. Also, the overhead information must be included in the image itself with payload.

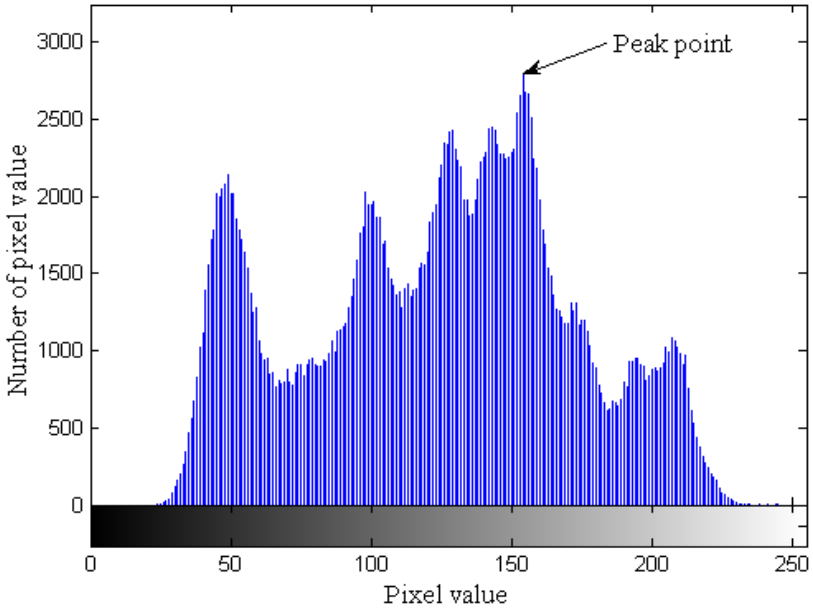


Figure 4.1 Histogram of the Lena image

Histogram Modification Technique

To improve hiding capacity, we present an efficient extension of the histogram modification technique by considering the difference between adjacent pixels instead of simple pixel value. Since the pixel grayscale values in a local area are often highly correlated and spatial redundancy, the distribution of pixel difference has a prominent maximum. Fig. 2 shows that the difference value is expected to be very close to zero, thus, there are a lot of candidates for embedding data. This observation leads toward designs in which the embedding is done in pixel differences. The experimental results have also supported this observation.

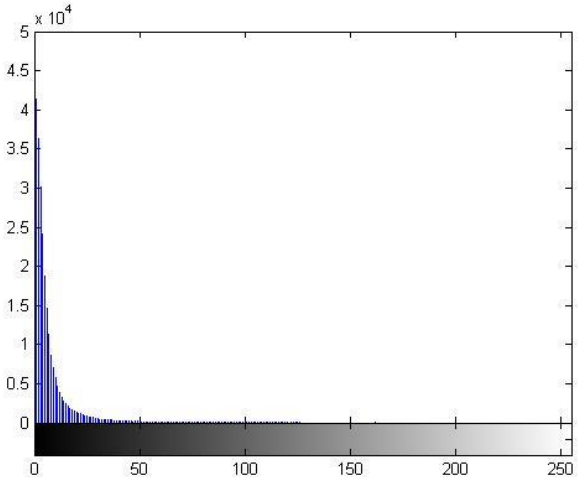


Figure 4.2 Histogram of the pixel difference

4.2. Lossless Data Hiding Algorithm

4.2.1 Embedding process

Consider a given N-pixel grayscale host image, we assume that the pixel value xi denotes the grayscale value of the ith pixel, $0 \leq i \leq N-1$, $x_i \varepsilon [0, 255]$. Let M be the message to be embedded and M = {0, 1, 2, 3}.

1) Scan the image in an inverse s-order as shown in Fig. Calculate the pixel difference d_i between pixels x_{i-1} and x_i by

$$d_i = \begin{cases} x_i & \text{if } i=1 \\ |x_{i-1} - x_i| & \text{otherwise} \end{cases}$$

2) Seek the peak point P from the pixel differences.

3) Scan the whole image in the same inverse s-order. If d_i>P, shift x_i by 3 units:

$$y_i = \begin{cases} x_i, & \text{if } i=0 \text{ or } d_i<P \\ x_i + 3, & \text{if } d_i>P \text{ and } x_i >= x_{i-1} \\ x_i - 3, & \text{if } d_i>P \text{ and } x_i < x_{i-1} \end{cases}$$

where y_i is the marked value of pixel i.

4) If d_i = P, modify x_i according to the message:

$$y_i = \begin{cases} x_i + M, & \text{if } d_i=P \text{ and } x_i >= x_{i-1} \\ x_i - M, & \text{if } d_i=P \text{ and } x_i < x_{i-1} \end{cases}$$

The above steps complete the data hiding process where one peak point is used. We note that large hiding capacities can be obtained by repeated data embedding processes.

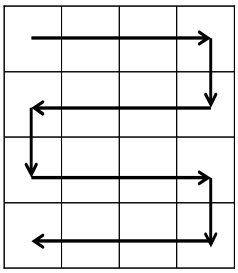


Figure 4.3 inverse s-order scanning

4.2.2. Extraction process

At the receiving end, the recipient extracts the embedded message from the marked image and losslessly recovers the host image. Consider an *N*-pixel grayscale marked image, we denote the grayscale value of the ith pixel in the image as y_i, $0 \leq i \leq N-1$, $y_i \in [0, 255]$.

1) Scan the marked image in the same order as during the embedding.

2) Set $x_0=y_0$, where x_0 denotes the restored value of y_0.

3) Extract message M by

$$M=\begin{cases} 0, & \text{if } |y_i - x_{i-1}| = P \\ 1, & \text{if } |y_i - x_{i-1}| = P+1 \\ 2, & \text{if } |y_i - x_{i-1}| = P+2 \\ 3, & \text{if } |y_i - x_{i-1}| = P+3 \end{cases}$$

Where x_{i-1} denotes the restored value of y_{i-1}.

4) Restore the original value of host pixel x_i by

$$x_i = \begin{cases} y_i + (|y_i - x_{i-1}| - P), & \text{if } P < |y_i - x_{i-1}| \le P + 3 \text{ and } y_i < x_{i-1}, \\ y_i - (|y_i - x_{i-1}| - P), & \text{if } P < |y_i - x_{i-1}| \le P + 3 \text{ and } y_i > x_{i-1}, \\ y_i + 3, & \text{if } |y_i - x_{i-1}| > P + 3 \text{ and } y_i < x_{i-1}, \\ y_i - 3, & \text{if } |y_i - x_{i-1}| > P+3 \text{ and } y_i < x_{i-1}, \\ y_i, & \text{otherwise.} \end{cases}$$

5) Go to Step 3 until the embedded message is completely extracted.

Thus, the exact copy of the original host image is obtained.

155	156	155	158
159	158	156	157
160	158	158	160
155	157	156	159

Figure 4.4 Host image

155	156	153	161
162	161	155	154
162	155	158	163
152	157	153	158

Figure 4.5 Marked image

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
155	156	155	158	157	156	158	159	160	158	158	160	159	156	157	155

x_i

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
155	1	1	3	1	1	2	1	1	2	0	2	1	3	1	2

d_i

P=1 MESSAGE TO BE EMBEDDED: 02313210

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
155	156	153	161	154	155	161	162	162	155	158	163	158	153	157	152

y_i

4.3 Implementation

Screenshots of the embedding and extraction algorithm discussed previously:

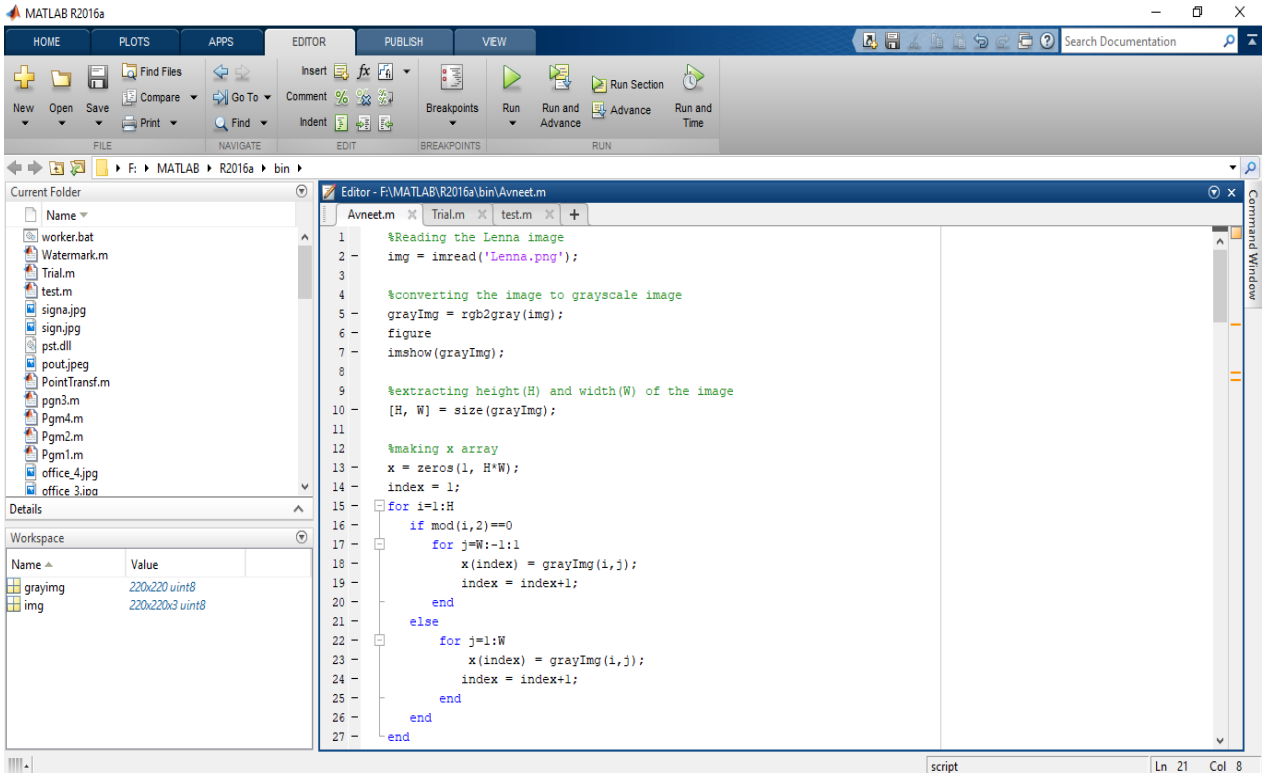


Figure 4.6 Converting the Lenna image into 1-d array i.e x

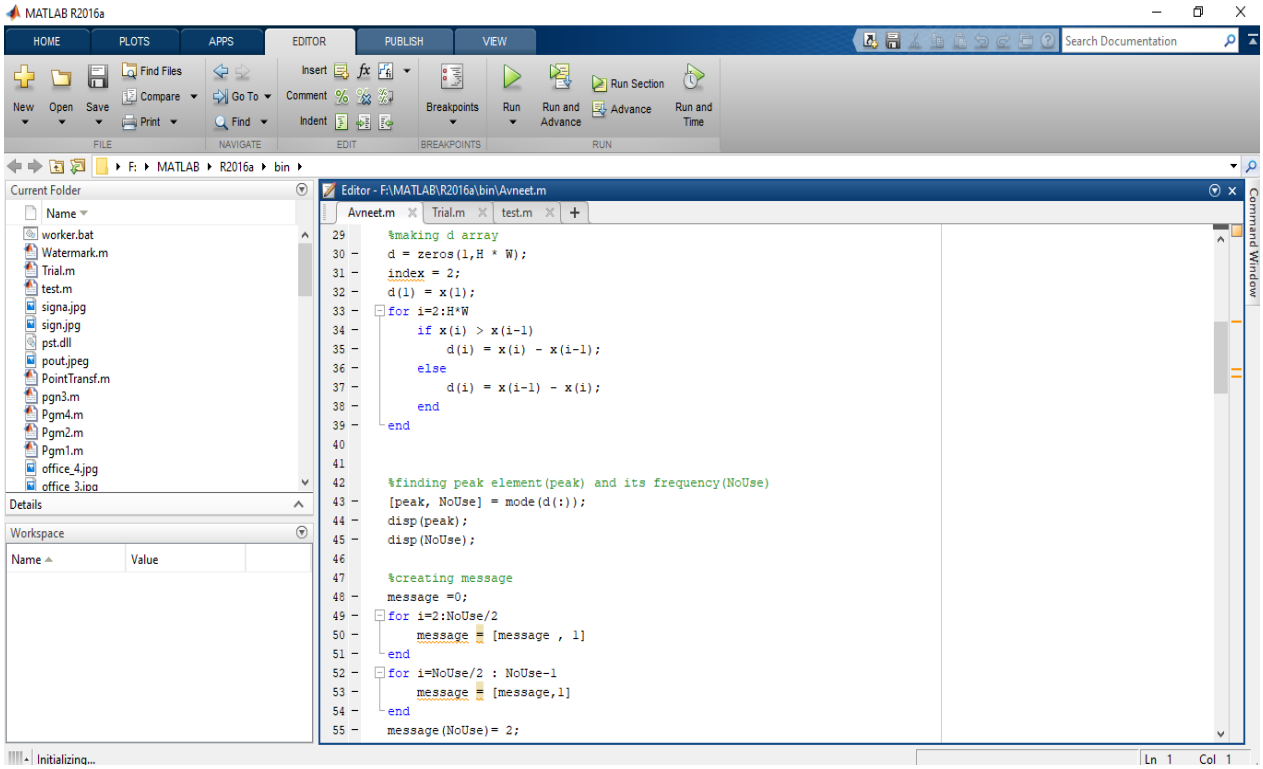


Figure 4.7 Creating message

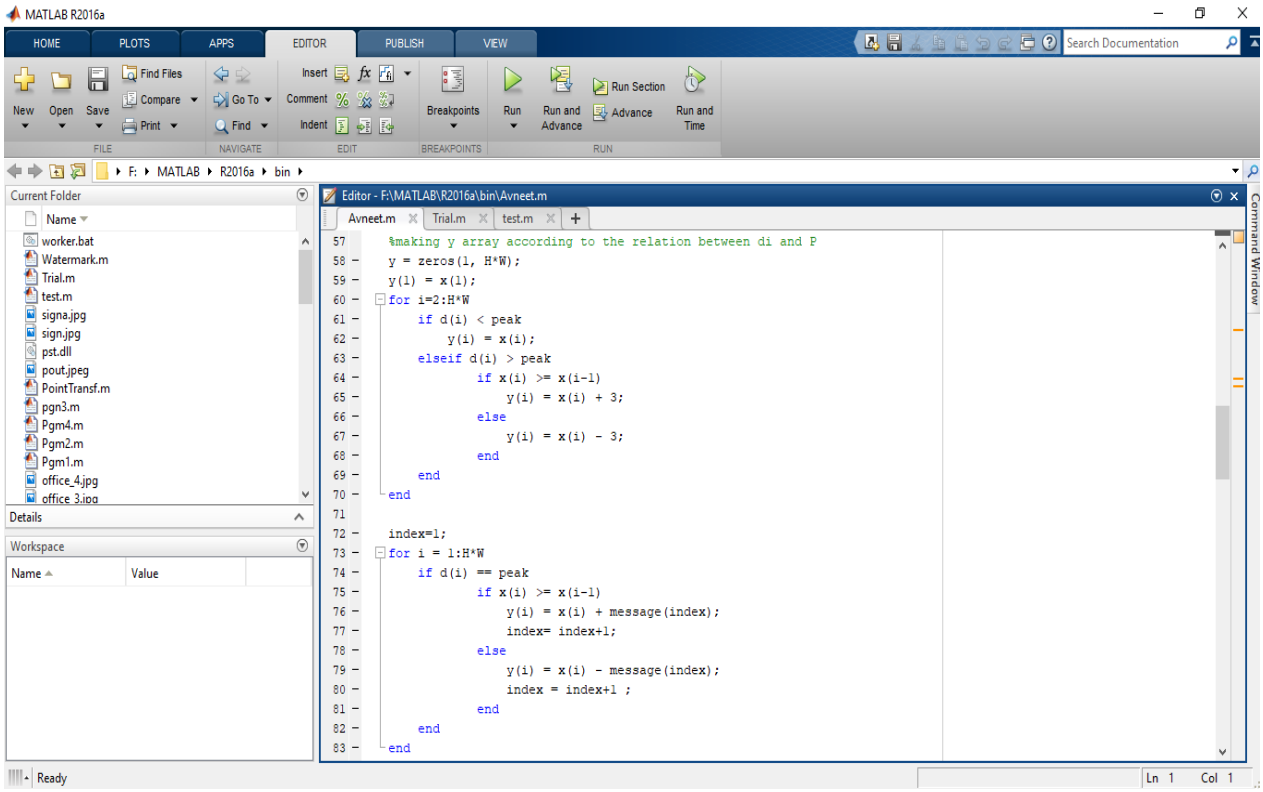


Figure 4.8 Embedding process of lossless data hiding algorithm

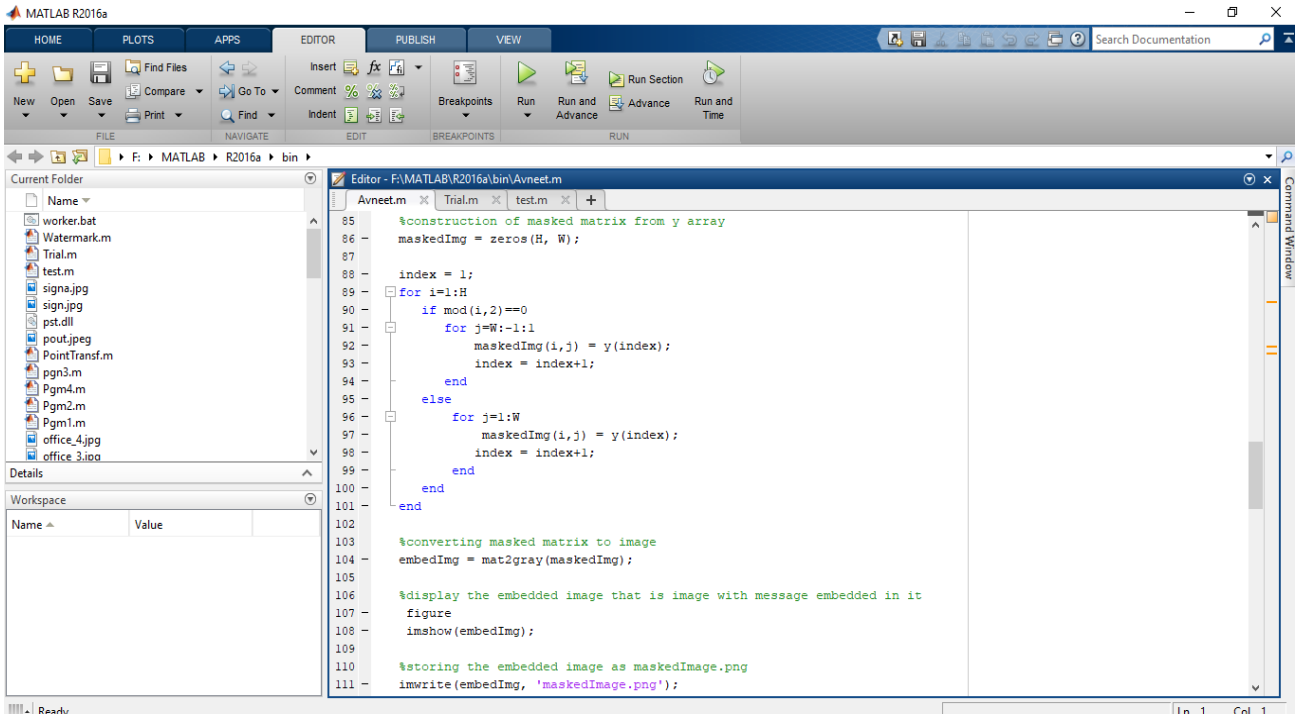


Figure 4.9 Making of masked(embedded) image

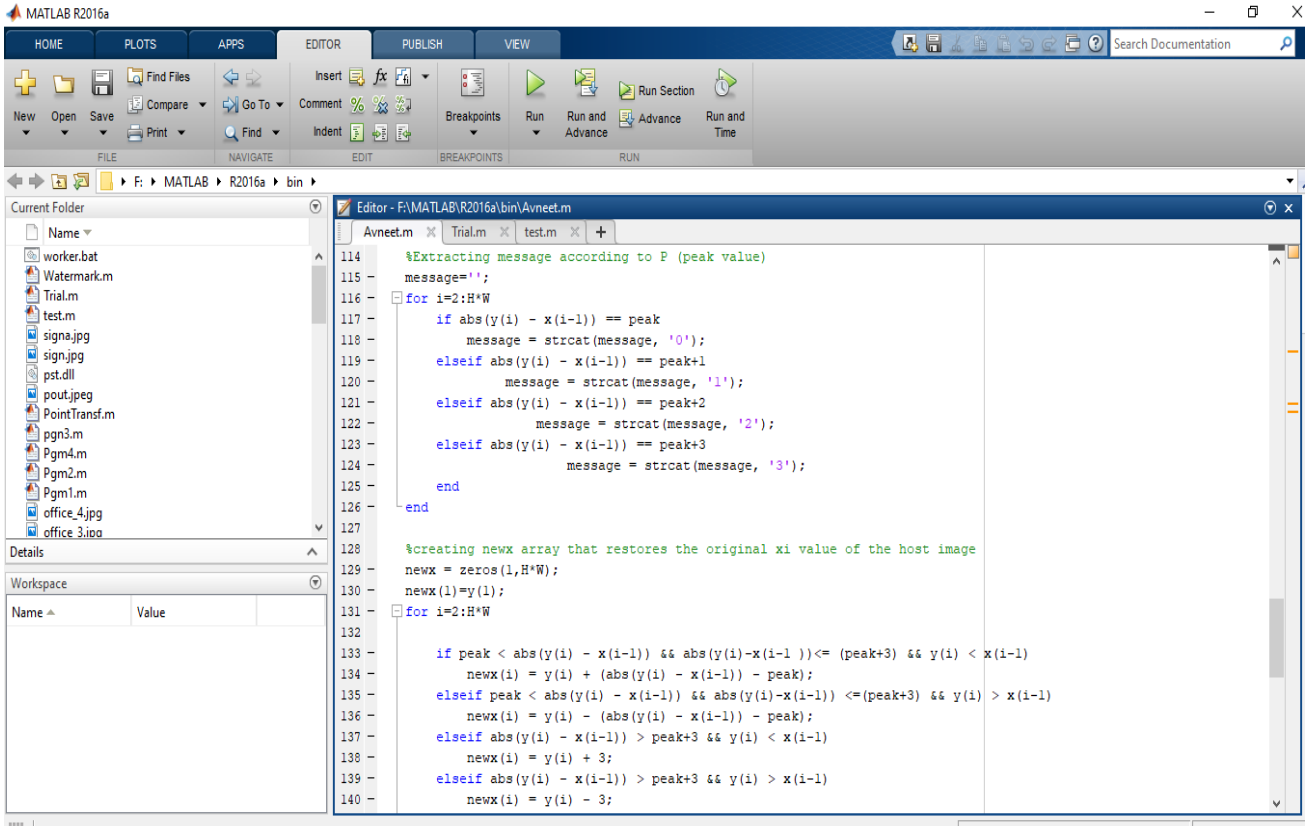


Figure 4.10 Extracting message

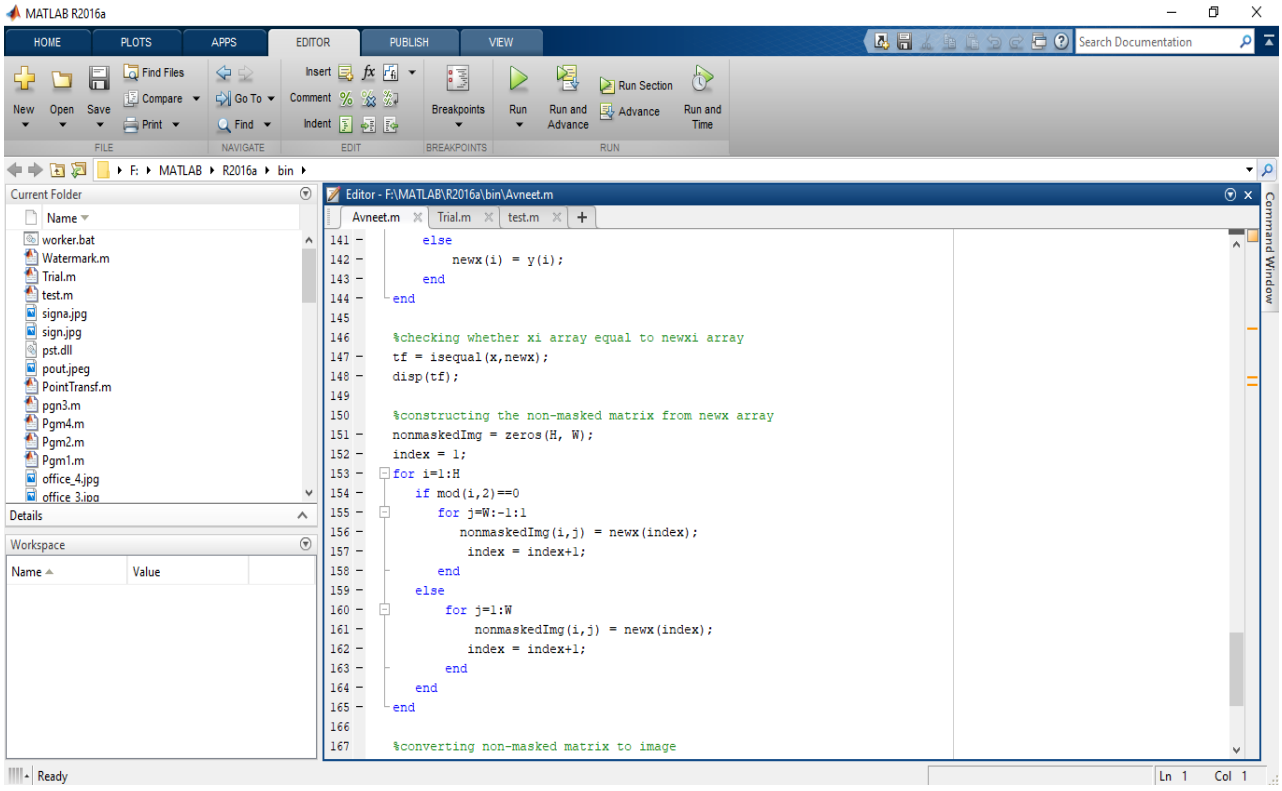


Figure 4.11 Constructing the original image

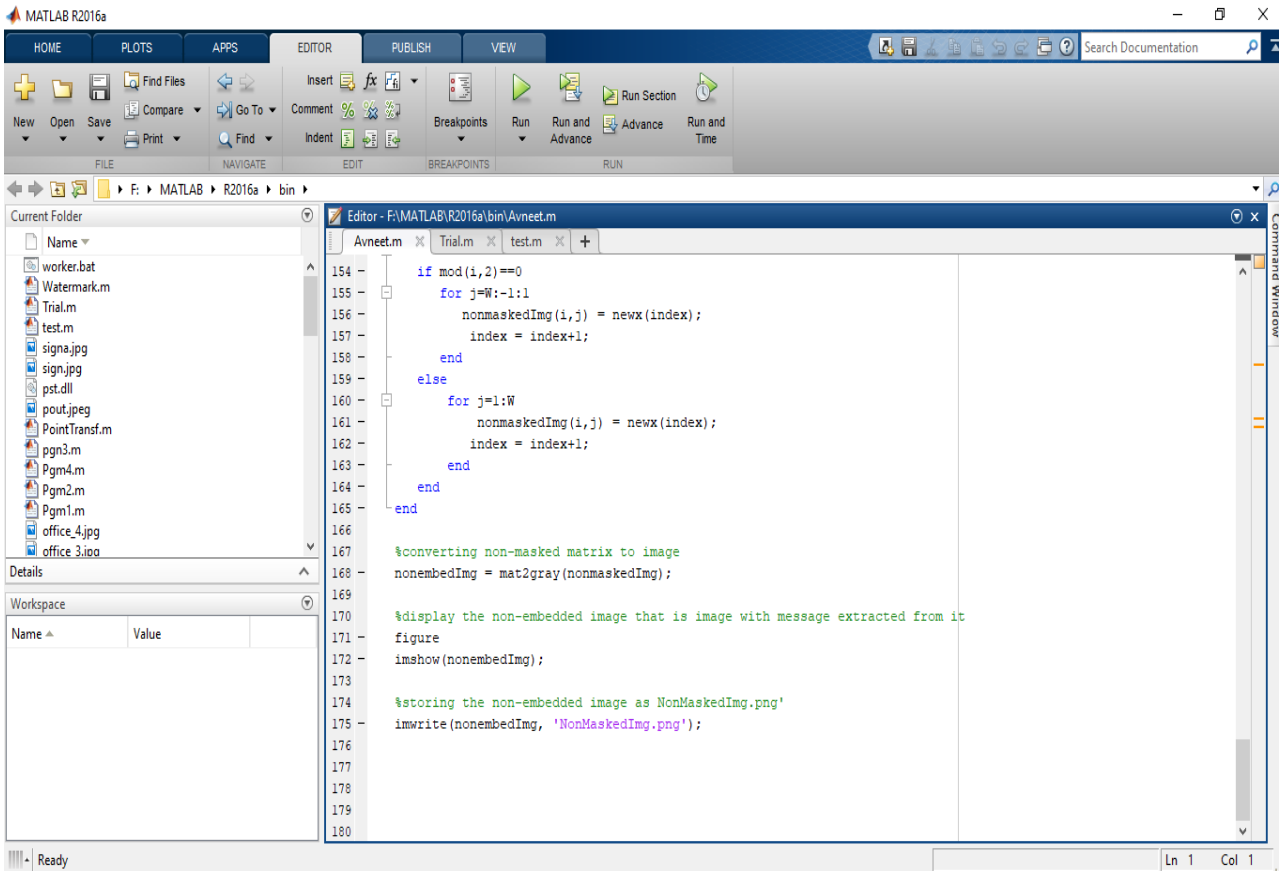


Fig 4.12 Recovering the original image (NonMaskedImg)

4.4 What is MATLAB?

MATLAB is a programming language developed by MathWorks. It started out as a matrix programming language where linear algebra programming was simple. It can be run both under interactive sessions and as a batch job.

MATLAB (matrix laboratory) is a fourth-generation high-level programming language and interactive environment for numerical computation, visualization and programming. It allows matrix manipulations; plotting of functions and data; implementation of algorithms; creation of user interfaces; interfacing with programs written in other languages, including C, C++, Java, and FORTRAN; analyze data; develop algorithms; and create models and applications. It has numerous built-in commands and math functions that help you in mathematical calculations, generating plots, and performing numerical methods.

MATLAB's Power of Computational Mathematics

MATLAB is used in every facet of computational mathematics. Following are some commonly used mathematical calculations where it is used most commonly −Dealing with Matrices and Arrays, 2-D and 3-D Plotting and graphics, Linear Algebra, Statistics,Data Analysis, Calculus and Differential Equations.

Features of MATLAB

Following are the basic features of MATLAB −

1. It is a high-level language for numerical computation, visualization and application development.
2. It also provides an interactive environment for iterative exploration, design and problem solving.
3. It provides vast library of mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration and solving ordinary differential equations.
4. It provides built-in graphics for visualizing data and tools for creating custom plots.
5. MATLAB's programming interface gives development tools for improving code quality maintainability and maximizing performance.
6. It provides tools for building applications with custom graphical interfaces.
7. It provides functions for integrating MATLAB based algorithms with external applications and languages such as C, Java, .NET and Microsoft Excel.

Uses of MATLAB

MATLAB is widely used as a computational tool in science and engineering encompassing the fields of physics, chemistry, math and all engineering streams. It is used in a range of applications including −

- Signal Processing and Communications
- Image and Video Processing

- Control Systems
- Test and Measurement
- Computational Finance
- Computational Biology

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 Result of the lossless data hiding algorithm

After running the lossless data hiding algorithm, we got the following result in Matlab.

Here are the screenshots of the results on “Lenna image”:

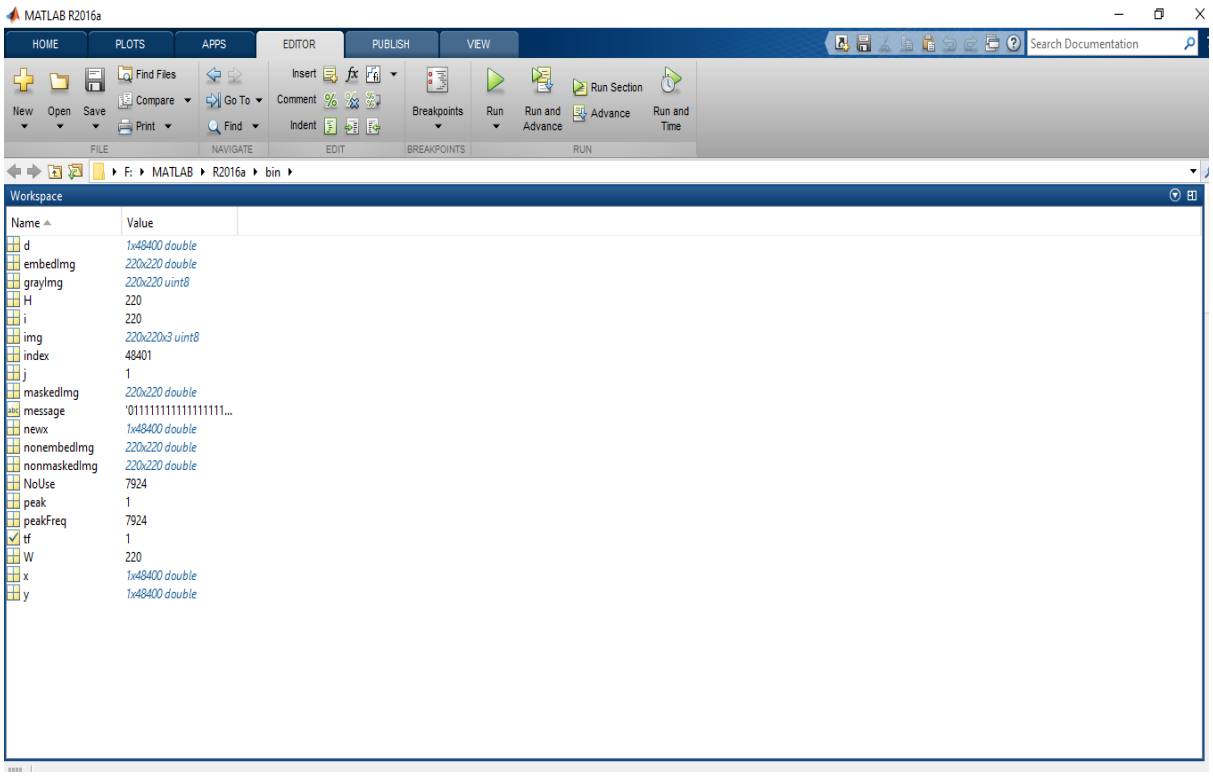


Figure 5.1 Screenshot of the workspace-depicts the values of the variables used in the implementation code.

H denotes height of the image and W denotes width of the image

peak denotes the most frequently occurring value in the array d i.e 1

peakFreq denotes the frequency of the Peak value i.e. 7924

Array x(1 row ,48400 columns) denotes the pixel values of the original image in the form of array

Array d(1 row,48400 columns) denotes the difference array obtained from x

Array y(1 row,48400 columns) denotes the new pixel values after embedding the message

Message=0111111….2 (size of the message is equal to the size of the variable “peakFreq”)

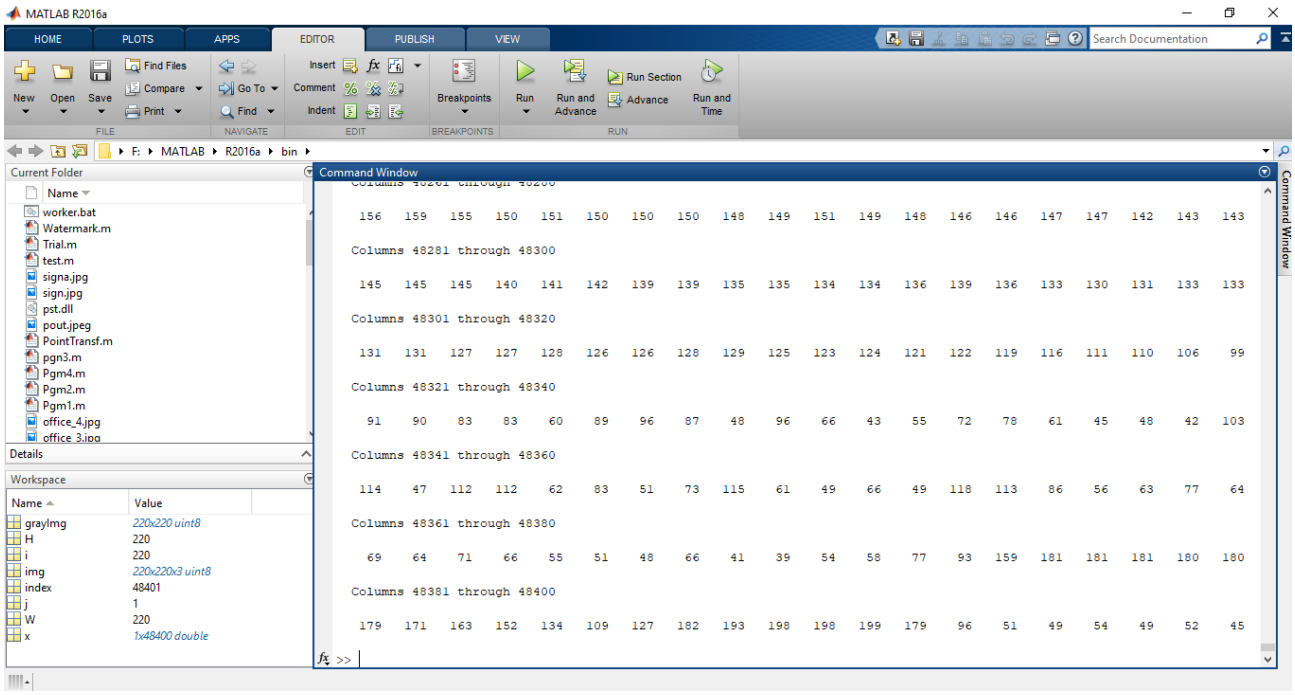


Figure 5.2 Depicts x array pixel values from columns 1 to 48400.

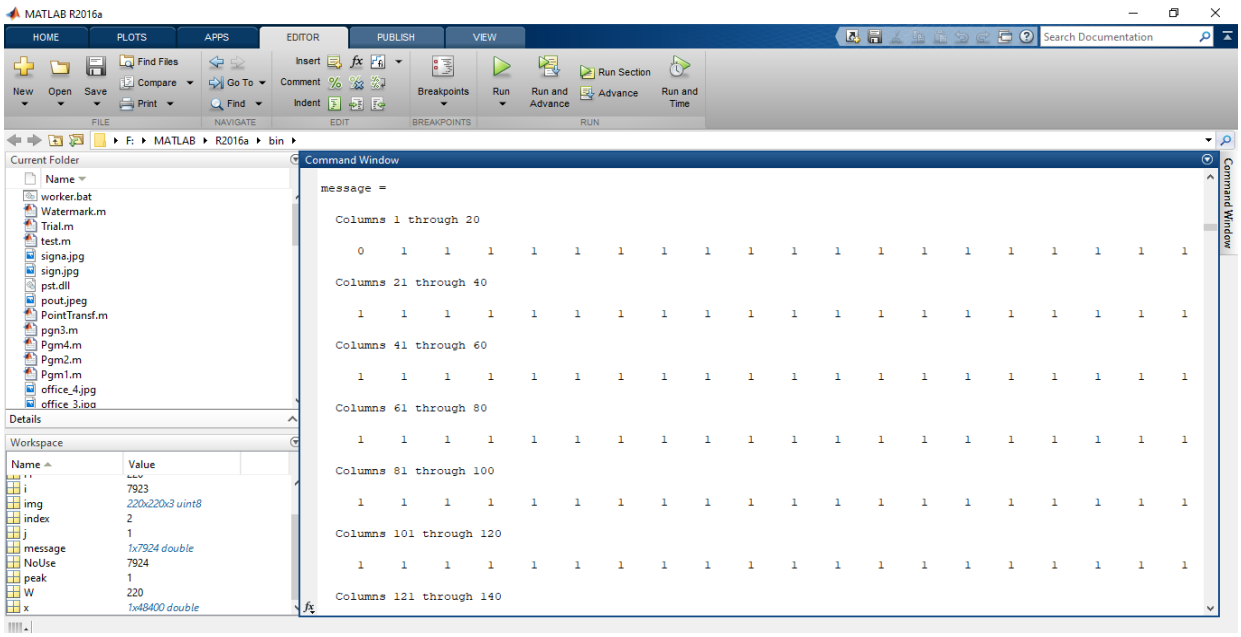


Fig 5.3 Depicts the message that is to be embedded=0111….112

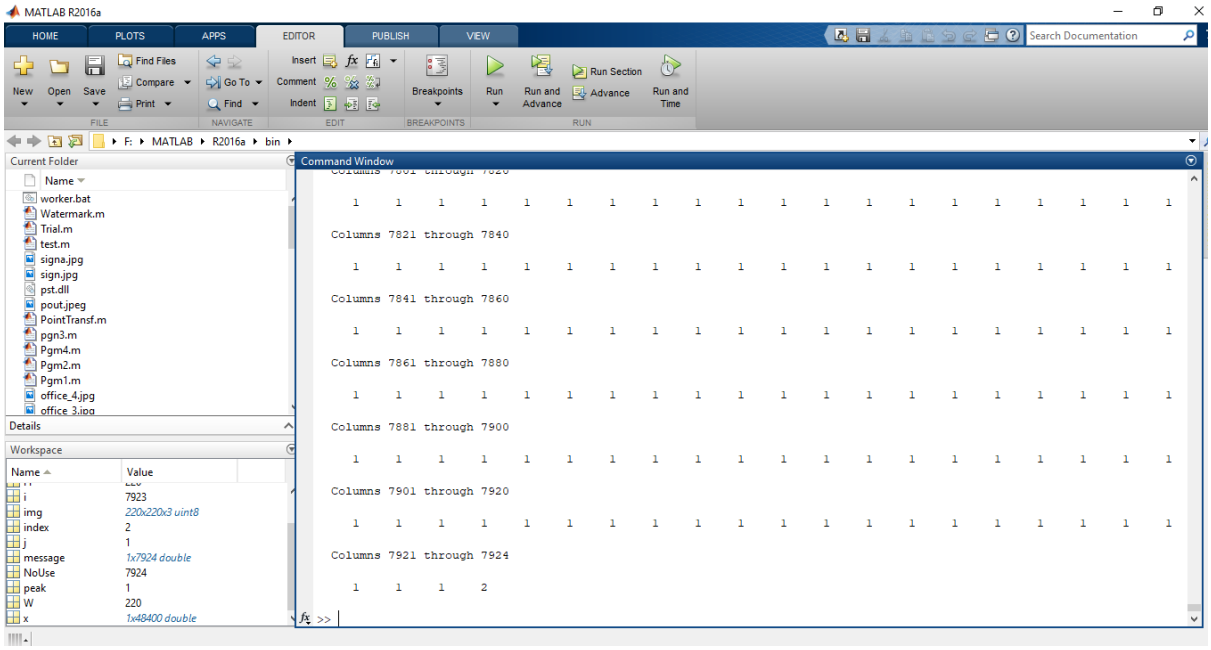


Fig 5.4 Depicts the message that is to be embedded=0111….112(in continuation to Figure 5.3)

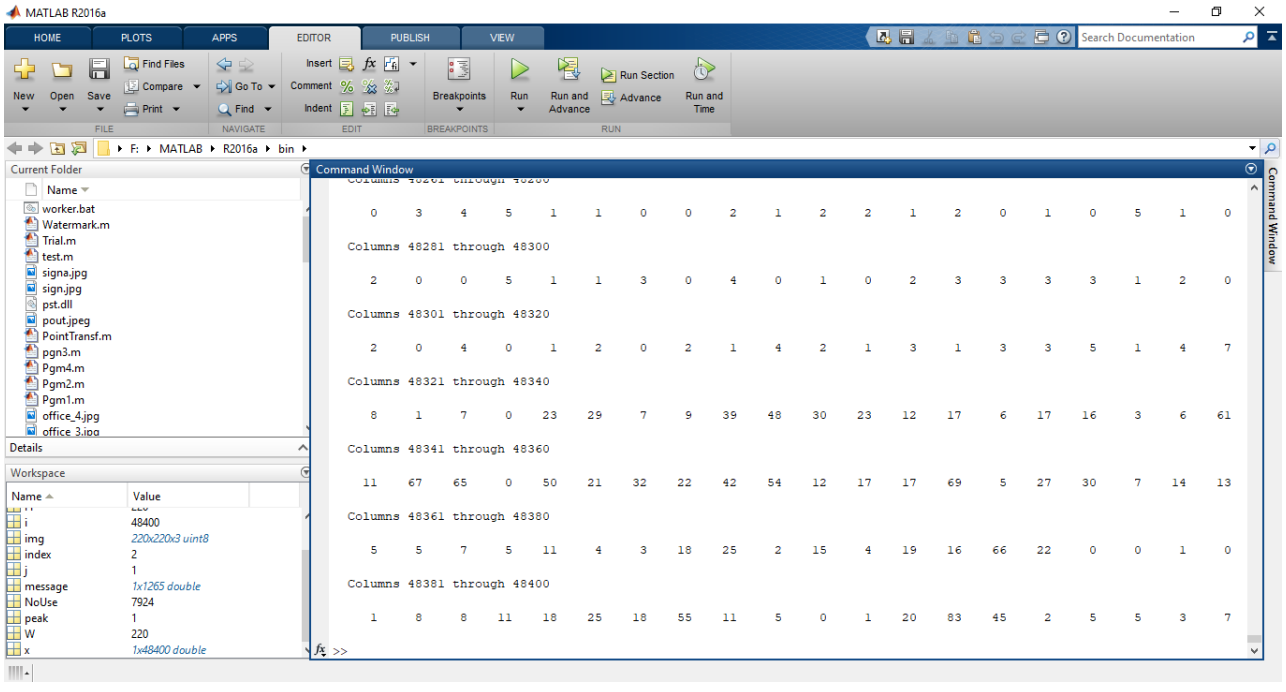


Fig 5.5 Depicts d array

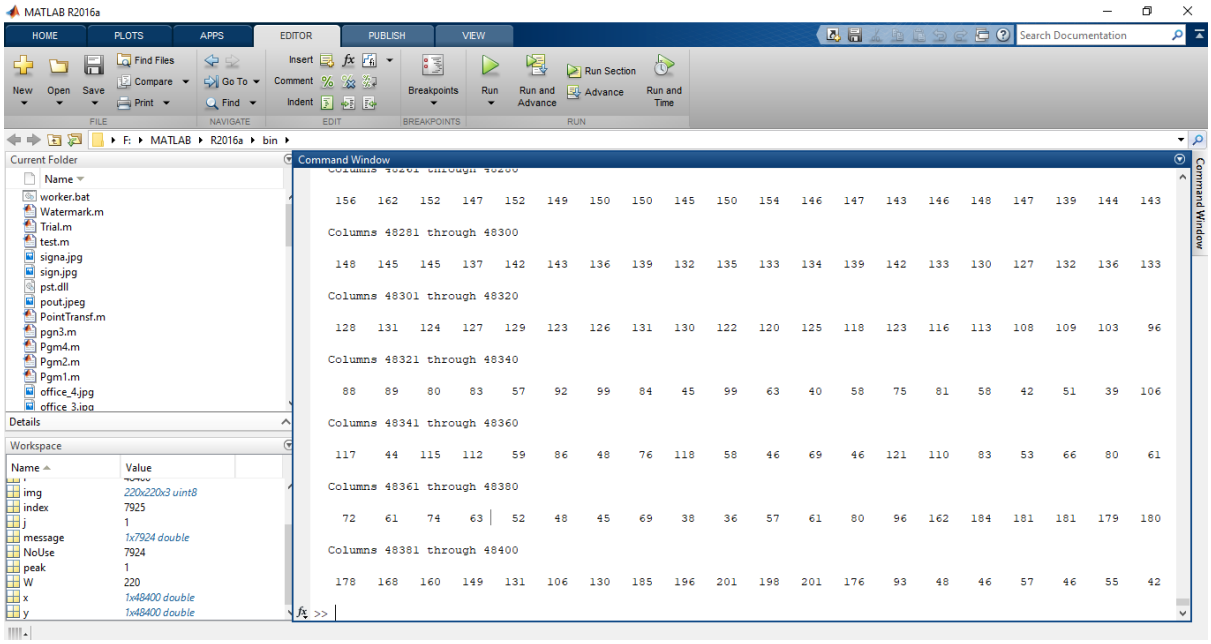


Fig 5.6 Depicts y array

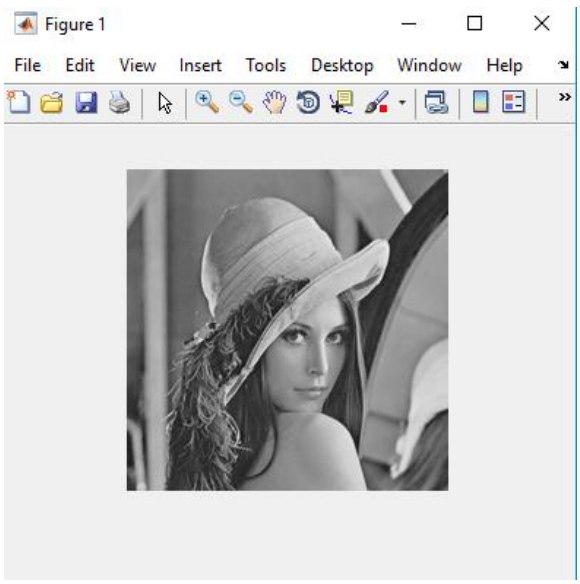


Figure 5.7 Original image

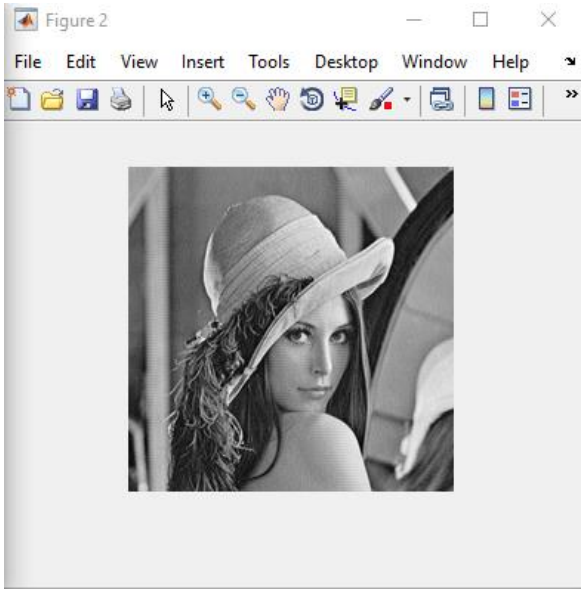


Figure 5.8 Embedded image

(created from y Array)

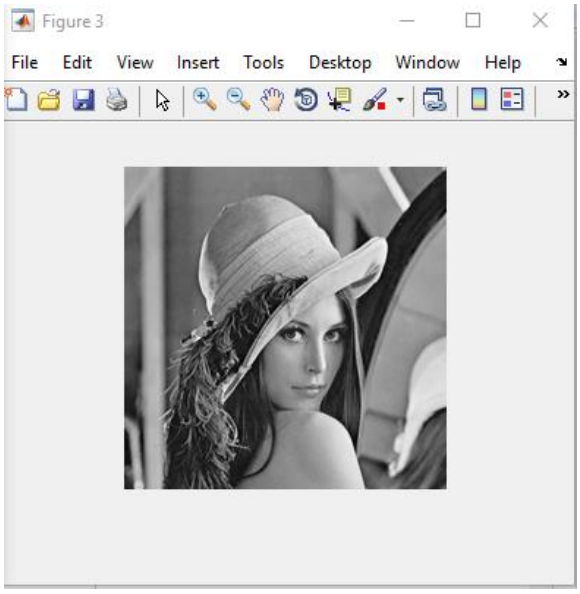


Figure 5.9 Regained image (which is same as original image)

Some screenshots of other images:

Fig 5.10 Original image

Fig 5.11 Embedded image

Figure 5.12 Regained image

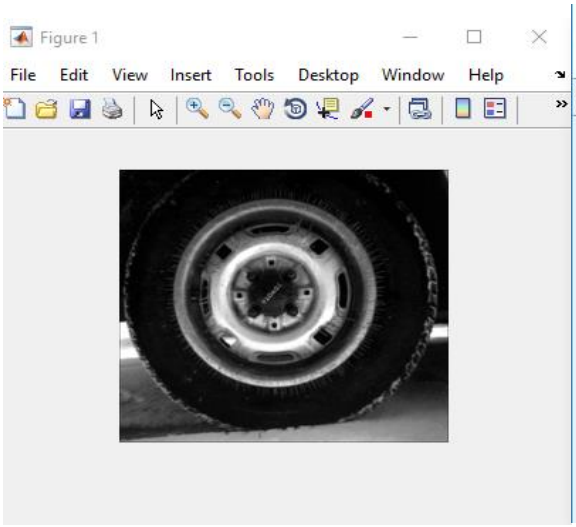


Figure 5.13 Original image

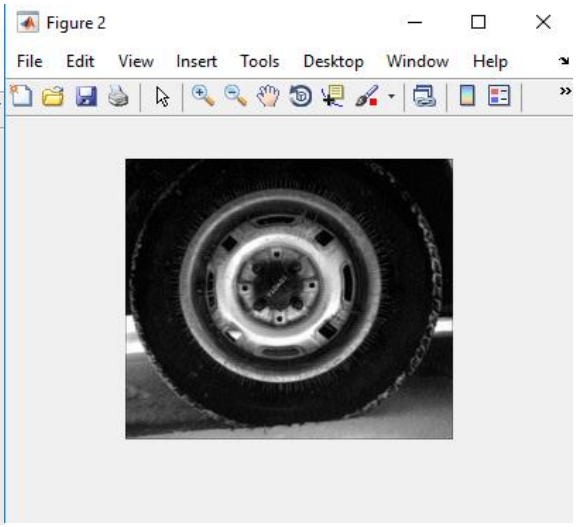


Figure 5.14 Embedded image

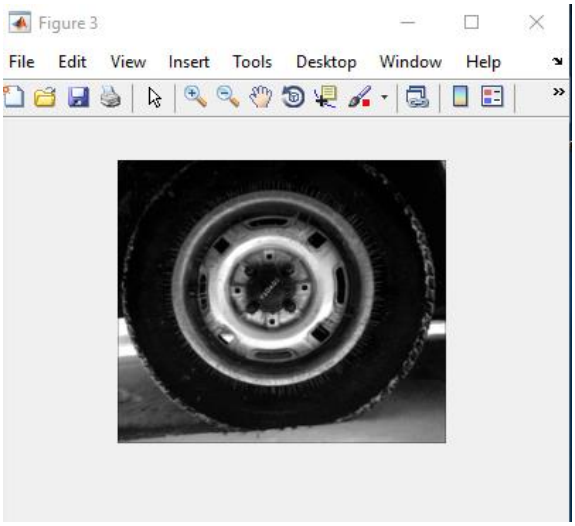


Figure 5.15 Regained image

Figure 5.16 Original image

Figure 5.17 Embedded image

Figure 5.18 Regained image

5.2 Discussions

5.2.1 Preventing overflow and underflow

Since the image format we perform is grayscale, the pixel value is within the range [0, 255]. Modification to a pixel may not be allowed whenever its value is saturated (0 or 255). As mentioned in the embedding process, the maximum modification of a pixel is 3. Using the histogram shifting technique shifting the histogram from both sides by 3 units enables us to avoid occurring overflow and underflow. After narrowing down the histogram to the range [3,252], we need to record the histogram shifting information as overhead bookkeeping information. To record overhead information, we create a one-bit map as the location map. If a pixel whose grayscale value is within the range [3, 252], we assign a value 0 in the location map; otherwise, we assign a value 1. Since pixels without the range [3, 252] are few and almost contiguous, we use the run-length coding algorithm that enables a large increase in compression ability to losslessly compress the location map. Note that the overhead information has to be embedded into the host image together with the embedded message. Since the message to be embedded is M = {0, 1, 2, 3}, each message symbol is represented by two bits. Thus, the pure payload Pur that is referred to real capacity is $Pur = 2 \times Np - |O|$, where Np is the number of pixels which are associated with peak points and |O| is the length of the overhead information.

5.2.2 Detecting noise

Peak signal-to-noise ratio (PSNR) is used to measure the distortion introduced by hiding.

The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is often used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed, or reconstructed image.

The Mean Square Error (MSE) and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare image compression quality. The MSE represents the cumulative squared error between the compressed and the original image, whereas PSNR represents a measure of the peak error. The lower the value of MSE, the lower the error.

$$\text{PSNR(dB)}=10\times\log_{10}\left(\frac{255^2}{MSE}\right),$$

where MSE is the mean squared error.

Fig 5.19 shows the performance comparison of pure payload in bpp versus image quality in PSNR of the proposed scheme with that of the existing histogram based lossless hiding schemes for the "Lena" image. Note that all schemes were presented based on histogram modification. We can see that Ni et al.'s scheme has low hiding capacity compared to the others. Fallahpour et al. and Lee et al. have improved Ni et al.'s work and have derived higher payload. However, they did not include the overhead information in their experiments. As shown in figure, the proposed scheme is able to achieve relatively higher pure capacity than existing schemes.

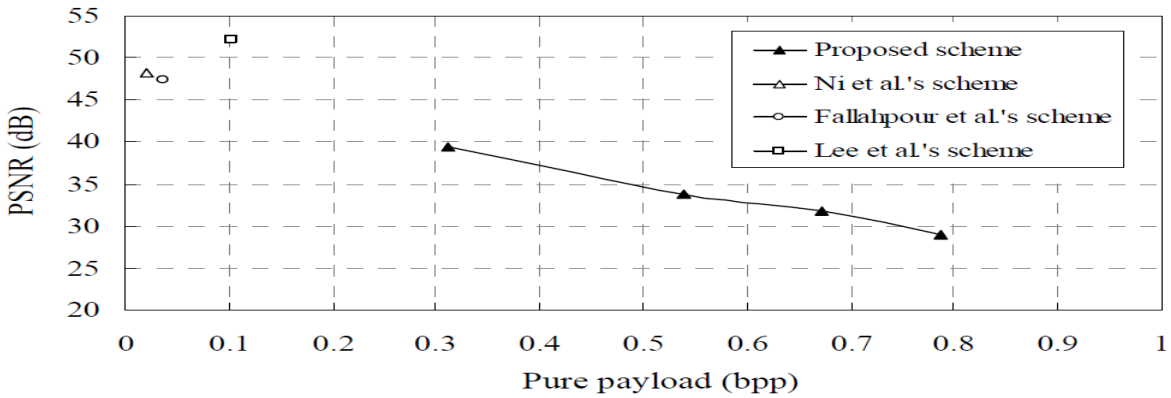


Figure 5.19 Performance comparison for the "Lenna" image with existing reversible schemes based on histogram modification

CHAPTER 6

CONCLUSIONS AND FUTURE SCOPE

6.1 Conclusions

Authenticity of digital images is very vital. Techniques that help us provide the authenticity of digital images are very vital whenever problems are raised about the integrity of an image. Thus, there is a need for image authentication for applications where we must be certain an image has not been modified. But in many authentication data hiding schemes, the distortion cannot be completely removed even when the image is deemed authentic and there are some applications for which any modification made to the image is intolerable, such as medical images, military images or images with a high strategic importance.

Lossless data hiding technique gives a solution to the problem of how to embed a large message in digital images in a lossless way so that after the embedded message is extracted, the image can be completely restored to its original state before the embedding occurred. An efficient extension of the histogram modification technique is used in lossless data hiding algorithm by considering the difference between adjacent pixels instead of simple pixel value. Using this technique, the difference values are expected to be very close to zero, thus, there are a lot of candidates for embedding data because the amount of data that can be emedded into an image equals to the number of pixels which are associated with the peak point.Thus, the proposed scheme is able to provide high capacities at invertible distortion.

Also, it can be easily modified for compressed image formats, such as JPEG, MPEG, and JPEG2000.

6.2 Future scope

The distribution of frequency coefficients may be almost Laplacian distributed due to quantization since the embedding must be performed in the transform domain. As a result, the proposed scheme can be generalized to other data types than images. The future work focuses on lossless authentication for video files and further improvement of the proposed scheme.

APPENDIX: CODE

IMPLEMENTATION OF THE ALGORITHM ON THE EXAMPLE GIVEN IN FIG.4.4

```
%initializing x array
x =[155,156,155,158,157,156,158,159,160,158,158,160,159,156,157,155];
disp(x);

%initializing the height(H) and width(W) of the image
H = 4;
W = 4;

%making d array
d = zeros(1,H*W);
index = 1;
d(1)= x(1);
d = [155,1,1,3,1,1,2,1,1,2,0,2,1,3,1,2];
disp(d);

%finding peak element(peak) and its frequency(peakFreq)
[peak, peakFreq] = mode(d(:));
disp(peak);
disp(peakFreq);

%creating message
message = [0,2,3,1,3,2,1,0];
message =0;
for i=2: peakFreq /2
   message = [message , i-1]
end
for i= peakFreq /2 : peakFreq -1
   message = [message,NoUse-i-1]
end
disp(message);

%making y array according to the relation between di and P
y = zeros(1, H*W);
y(1) = x(1);
for i=2:H*W
   if d(i) < peak
      y(i) = x(i);
   elseif d(i) > peak
         if x(i) >= x(i-1)
            y(i) = x(i) + 3;
         else
            y(i) = x(i) - 3;
         end
   end
end
```

```
index=1;
for i = 1:H*W
   if d(i) == peak
         if x(i) >= x(i-1)
            y(i) = x(i) + message(index);
            index= index+1;
         else
            y(i) = x(i) - message(index);
            index = index+1 ;
         end
   end
end
disp(y);

%construction of masked matrix from y array
maskedImg = zeros(H, W);
index = 1;
for i=1:H
  if mod(i,2)==0
     for j=W:-1:1
        maskedImg(i,j) = y(index);
        index = index+1;
     end
  else
      for j=1:W
         maskedImg(i,j) = y(index);
        index = index+1;
      end
  end
end
%converting masked matrix to image
embedImg = mat2gray(maskedImg);
%display the embedded image that is image with message embedded in it
figure
imshow(embedImg);
%storing the embedded image as maskedImage.png
imwrite(embedImg,'maskedImage.png')



%Extracting message according to P (peak value)
message='';
for i=2:H*W
   if abs(y(i) - x(i-1)) == peak
      message = strcat(message, '0');
   elseif abs(y(i) - x(i-1)) == peak+1
          message = strcat(message, '1');
```

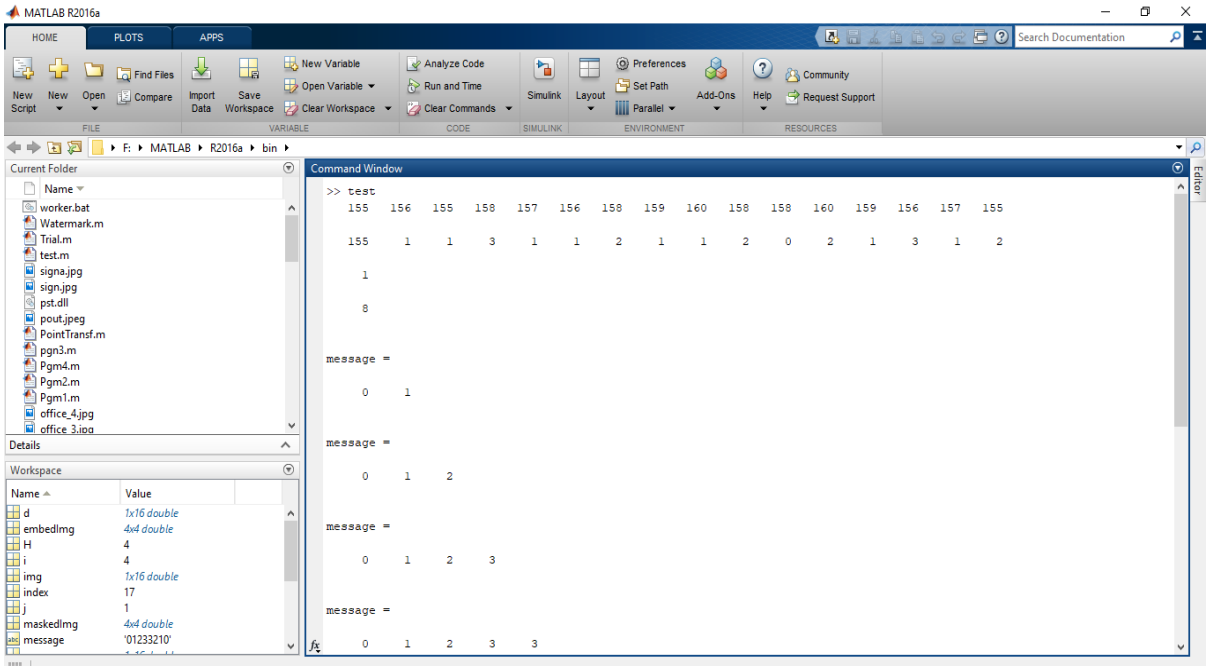
```
   elseif abs(y(i) - x(i-1)) == peak+2
            message = strcat(message, '2');
   elseif abs(y(i) - x(i-1)) == peak+3
              message = strcat(message, '3');
   end
end

disp(message);

%creating newx array that restores the original xi value of the host image
newx = zeros(1,H*W);
newx(1)=y(1);
for i=2:H*W

   if peak < abs(y(i) - x(i-1)) && abs(y(i)-x(i-1 ))<= (peak+3) && y(i) < x(i-1)
      newx(i) = y(i) + (abs(y(i) - x(i-1)) - peak);
   elseif peak < abs(y(i) - x(i-1)) && abs(y(i)-x(i-1)) <=(peak+3) && y(i) > x(i-1)
      newx(i) = y(i) - (abs(y(i) - x(i-1)) - peak);
   elseif abs(y(i) - x(i-1)) > peak+3 && y(i) < x(i-1)
      newx(i) = y(i) + 3;
   elseif abs(y(i) - x(i-1)) > peak+3 && y(i) > x(i-1)
      newx(i) = y(i) - 3;
   else
      newx(i) = y(i);
   end
end

disp(newx);

%checking whether xi array equal to newxi array
tf=isequal(x,newx);
disp(tf);

%constructing the non-masked matrix from newx array
nonmaskedImg = zeros(H, W);
index = 1;
for i=1:H
  if mod(i,2)==0
     for j=W:-1:1
       nonmaskedImg(i,j) = newx(index);
        index = index+1;
     end
  else
      for j=1:W
         nonmaskedImg(i,j) = newx(index);
        index = index+1;
      end
  end
end
```

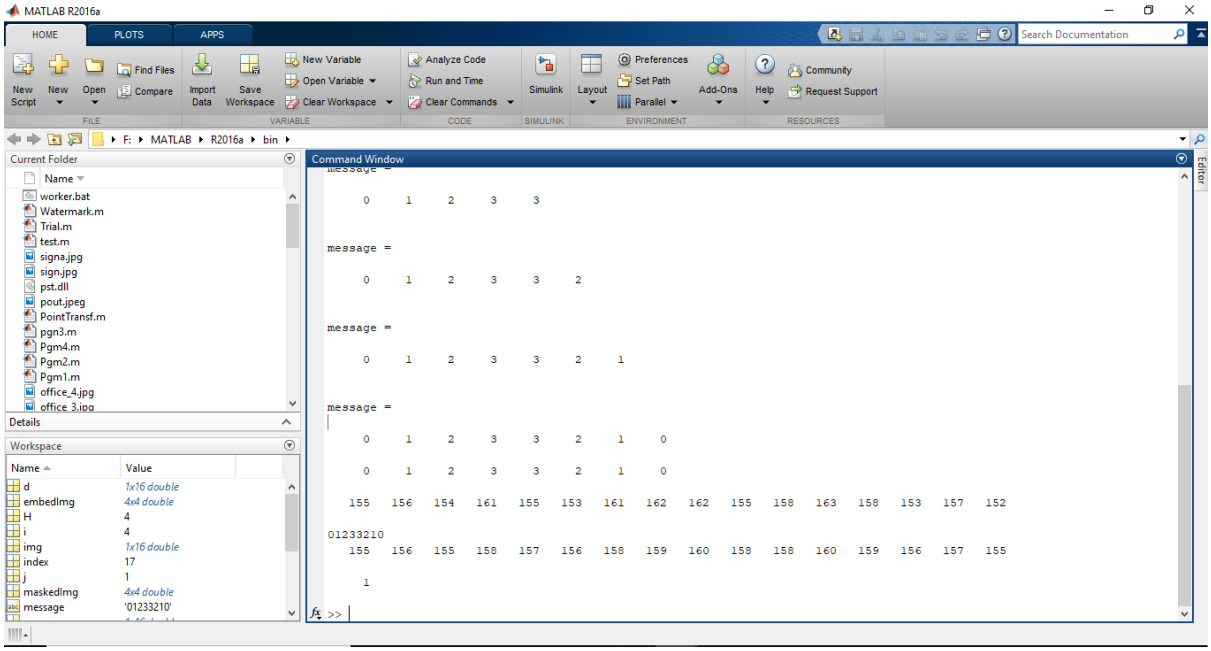
%converting non-masked matrix to image
nonembedImg = mat2gray(nonmaskedImg);

%display the non-embedded image that is image with message extracted from it
figure
imshow(nonembedImg);

%storing the non-embedded image as NonMaskedImg.png'
imwrite(nonembedImg, 'NonMaskedImg.png');

Screenshots of the results of the above code:





REFERENCES

[1] Alekhya Orugonda, S.Rajan (2013), "Hiding the Military Secret Message by Reversible Data Hiding",International Journal of Engineering and Innovative Technology (IJEIT).

[2] Arun Kumar.M.N, Krishnapriya K.R (2017), "Reversible Data Hiding In Image- A Literature Survey", International Journal of Advanced Research in Computer Science.

[3] Chin-Chen Chang, Kuo-Nan Chen & Wei-Liang Tai (2008), "Lossless Data Hiding Based on Histogram Modification for Image Authentication", IEEE/IFIP International Conference on Embedded and Ubiquitous Computing.

[4] DinuColtuc and Jean-Marc Chassery (2007), "Very fast watermarking by reversible contrast mapping", IEEE signal process .Lett .14 (4):255-258.

[5] Goldjan M., Fridrich J (2001), "Distortion-free data embedding", in Proc. 4th Information Hiding Workshop.

[6] Honsinger C, Jone P, Rabbani M, Stoffel J (2001), "Lossless recovery of an original image containing embedded data", US Patent: 6,278,791 B1.

[7] J. Fridrich, M. Goljan, and R. Du (2001), "Invertible authentication," Proceedings of the SPIE, Security and Watermarking of Multimedia Contents III, vol. 3971, San Jose, California, Jan. pp. 197-208

[8] Kekre H.B, Athawale A., Halarnkar P.N (2009) "Performance Evaluation of Pixel Value Differencing and Kekre's Modified Algorithm for Information Hiding in Images", International Conference on Advances in Computing, Communication and

Control.

[9] K Suresh Babu etal. (2005) “Authentication of secret information in image steganography”, Computer Journal.

[10] Lee J, Hwang S, Jeong S, Yoon K, Park C, Ryou J (2003),” A DRM framework for distributing digital contents through the Internet”, ETRI Journal, 423–436.

[11] Lincy Rachel Mathews1, Arathy C. Haran V (2014), “Histogram Shifting based reversible data hiding”, International Journal of Engineering Trends and Technology (IJETT).

[12] Mazurczyk W., Smolarczyk S., Szczypiorski K (2009), “Hiding Information in Retransmissions”, In: Computing Research Repository (CoRR), abs/0905.0363, arXiv.org E-print Archive, Cornell University, Ithaca, NY (USA).

[13] M. Fallahpour and M. H. Sedaaghi (2007), “High capacity lossless data hiding based on histogram modification,” IEICE Electronics Express, vol. 4, no. 7, pp. 205-210.

[14] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber (2005) “Lossless generalized-LSB data embedding,” IEEE Trans. Image Process, vol. 14, no.2, pp. 253–266.

[15] Naresh Kumar, Rahul Jain (2012), “Efficient data hiding scheme using lossless data compression and image steganography”, International Journal of Engineering Science and Technology (IJEST).

[16] Shaik Shaheena, B. L. Sirisha (2016), ”An Improvised Lossless Data-hiding Mechanism for Image Authentication Based Histogram Modification”, International Journal Of Professional Engineering Studies.

[17] S. K. Lee, Y. H. Suh, and Y. S. Ho (2006), “Reversible image authentication based on watermarking,” Proceedings of IEEE International Conference on Multimedia and Expo, Toronto, Ontario, Canada, pp. 1321-1324.

[18] T. Kalker and F. M. J. Willems (2003), “Capacity bounds and constructions for reversible data hiding,” Security Watermarking Multimedia Contents V, vol. 5020, pp. 604–611.

[19] Xuan G, Zhu J, Chen J, Shi Y , Ni Z, Su W (2002), “Distortionless data hiding based on integer wavelet transform”, IEE Electrionics Letters, 1646–1648 (2002).