ÖZET

Son yıllarda bilgi saklamak çok etkili hale gelmiştir. Pratik olarak bilgi saklamak için günümüze kadar birçok algoritma geliştirilmiştir. Bu çalışmada resimler içerisine counterlet transformasyonu kullanılarak bilgi saklamak için bir sistem önerisi yapılmıştır. Önerisi yapılan sistemin algoritması sabit resim içerisindeki resmin boyutlarını saklamaktadır. Buna ilave olarak resmi saklamazdan önce quartet try division tekniği kullanılarak saklanacak olan resim parçalara bölünmüştür. Tezde üç tane algoritma kullanılmıştır. Her üç algotitma için de değişik boyutlarda resimler kullanılarak pratik çalışmalar yapılmıştır. Bu çalışmalarda korelasyon oranının işlem öncesi ve işlem sonrasında 0.99 u aştığı gözlemlenmiştir. Çalışmada işlemin ve ölçümlerin, saklanan ve esas resim arasındaki ilişkinin PSNR, SNR, MSE ve korelasyona bağlı olduğu gözlemlenmiştir. Araştırmada program dili olarak Matlab kullanılmıştır.

Anahtar Kelimeler: bilgi saklamak, veri saklamak, resim saklamak, couterlet transformasyonu

ABSTRACT

In the last years the subject of hiding information about approving property right has been effective. Many algorithms appeared to work on developing efficient techniques of practical hiding of property right. In this research, a system is suggested for hiding watermarks embedded inside the coefficients of images decomposed by contourlet transformations which have qualities that offer an additional support to the power and practical security of the hiding process. The algorithms of the suggested system (non-blind) of hiding watermarking led to the probability of changing the watermarking size with the fixed cover size and vice versa. In addition, the quartet try division technique is followed in dividing the watermark before embedding it inside the cover-image and dividing it as well and then distributing the parts of the watermark inside the cover-image according to the quartet try technique. Three algorithms are used in this thesis for hiding information. The practical application, on the three proposed algorithm by using the cover image and watermarks of different sizes revealed that the ratio of the correlation factor before and after the embedding of the cover-image exceeds 0.99 and almost the ratio is in the watermark before and after the process. The measurement depended on the PSNR, SNR, MSE and correlation for measuring the similarity closeness between the cover-image and the input watermark and the retrieved one. Matlab was used as the programing language for building all of the programs in this research.

Keywords: Watermark, hiding information, hiding data, hiding picture, contourlet transformation

CONTOURLET TRANSFORMATION FOR DATA HIDING

**A THESIS SUBMITTED TO THE GRADUATE
SCHOOL OF APPLIED SCIENCES OF
NEAR EAST UNIVERSITY**

By

DIYAR QADER SALEEM ZEEBAREE

**In Partial Fulfillment of the Requirements for
the Degree of Master of Science
in
Computer Information Systems**

NICOSIA 2014

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: DIYAR QADER SALEEM ZEEBAREE

Signature:

Date:

ACKNOWLEDGMENTS

My deepest gratitude goes to Assoc. Prof. Dr. Nadire Cavus my supervisor, for her constant encouragement and guidance. She has walked me through all the stages of the writing of my thesis. Without her consistent and illuminating instruction, this thesis could not have reached its present form.

Also I would like to express unlimited thanks to Prof. Dr. Dogan Ibrahim.

My unlimited thanks and heartfelt love would be dedicated to my dearest family for their loyalty and their great confidence in me. Iam greatly indebted to my father Mr.Qader Saleem Zeebaree who is indeed my inspiration and the man who led me to the treasures of knowledge.

Last, but not least, gratitude is extended to my mother for their support and blessings.

ABSTRACT

In the last years the subject of hiding information about approving property right has been effective. Many algorithms appeared to work on developing efficient techniques of practical hiding of property right. In this research, a system is suggested for hiding watermarks embedded inside the coefficients of images decomposed by contourlet transformations which have qualities that offer an additional support to the power and practical security of the hiding process. The algorithms of the suggested system (non-blind) of hiding watermarking led to the probability of changing the watermarking size with the fixed cover size and vice versa. In addition, the quartet try division technique is followed in dividing the watermark before embedding it inside the cover-image and dividing it as well and then distributing the parts of the watermark inside the cover-image according to the quartet try technique. Three algorithms are used in this thesis for hiding information. The practical application, on the three proposed algorithm by using the cover image and watermarks of different sizes revealed that the ratio of the correlation factor before and after the embedding of the cover-image exceeds 0.99 and almost the ratio is in the watermark before and after the process. The measurement depended on the PSNR, SNR, MSE and correlation for measuring the similarity closeness between the cover-image and the input watermark and the retrieved one. Matlab was used as the programing language for building all of the programs in this research.

Keywords: Watermark, hiding information, hiding data, hiding picture, contourlet transformation

ÖZET

Son yıllarda bilgi saklamak çok etkili hale gelmiştir. Pratik olarak bilgi saklamak için günümüze kadar birçok algoritma geliştirilmiştir. Bu çalışmada resimler içerisine counterlet transformasyonu kullanılarak bilgi saklamak için bir sistem önerisi yapılmıştır. Önerisi yapılan sistemin algoritması sabit resim içerisindeki resmin boyutlarını saklamaktadır. Buna ilave olarak resmi saklamazdan önce quartet try division tekniği kullanılarak saklanacak olan resim parçalara bölünmüştür. Tezde üç tane algoritma kullanılmıştır. Her üç algotitma için de değişik boyutlarda resimler kullanılarak pratik çalışmalar yapılmıştır. Bu çalışmalarda korelasyon oranının işlem öncesi ve işlem sonrasında 0.99 u aştığı gözlemlenmiştir. Çalışmada işlemin ve ölçümlerin, saklanan ve esas resim arasındaki ilişkinin PSNR, SNR, MSE ve korelasyona bağlı olduğu gözlemlenmiştir. Araştırmada program dili olarak Matlab kullanılmıştır.

Anahtar Kelimeler: bilgi saklamak, veri saklamak, resim saklamak, couterlet transformasyonu

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CT:	Contourlet Transformations
DB:	Decibel
DRM:	Digital Rights Management Systems
DFB:	Directional Filter Bank
HDFB:	Horizontal DFB
IHW:	Information Hiding Workshop
ICT:	Inverse Contourlet Transformations
IDCT:	Inverse Discrete Cosine Transform
LP:	Laplacian Pyramid
NC:	Normalized Correlation Coefficient
PDFB:	Pyramidal Directional Filter Bank
QFB:	Quincunx Filter Bank
SVD:	Singular Value Decomposition
TCP:	Transmission Control Protocol
VDFB:	Vertical DFB
FCLT:	Fast Contour Let Transformation
ICLT:	Invers Contour Let Transformation
CLT:	Contour Let Transformation
RGB:	Reed Green Blue
JPG:	Joint Photographic Graphics
ASCII:	American Standard Code for Information Interchange
XOR:	logical Exclusive OR (Boolean)
TCP:	Transmission Control Protocol
GUI:	Graphical User Interface
GIF:	Graphics Interchange Format
PNG:	Portable Network Graphics image format

CHAPTER 1

INTRODUCTION

1.1 Overview

The development of information processing technologies and the rapid growth of the communication through the internet resulted in transferring the information sources easily (Ramana et al., 2011). Due to the electronic espionage, data security became the core of interest especially within the government and army sectors and other communications fields (Alam et al., 2011). As the internet is considered an open environment, a need emerged to make efficient ways available which prevent the data from being copied or manipulated illegally and amongst those techniques is an Encryption technique that is considered one of the traditional data security techniques.

The encryption techniques protect the secret data by transforming it into an unclear format before transferring it between the two transmitting parties (Ramana et al., 2011). But the form of those unclear encrypted data provoke doubts and drag the attention of the intruder who attempts to decode the information sent or destroy it. From the other hand, the development of information hiding offered another solution to protect the data by employing certain technologies that rests on the basis of transferring the secret data in a way that conceals the existence of a secret communication (Lee and Tsai, 2011).

1.2 The Problem

In the last years the subject of hiding information about approving the property right has been effective. Many algorithms have appeared to work on developing efficient techniques of practical hiding of property right. Also, in the recent years many technologies are becoming depended for protecting its possessiveness and proving its back profit and due to the recent development in the field of digital documentation just the science of watermarking is developed enormously for the embedding the information this kind of field proving of the property. Thus the process of hiding the owning data within document is accelerated with the development that's gaining in the methods of representation documents. It's clear that many transformations are appeared in the recent years to represent as well, the digital images including wavelet and curved and finally contourlet transformations.

1.3 The Aim of the Study

This research is aiming to improve the techniques of the watermark by means of achieving the merge between those techniques and the algorithms of processing the digital images. The contour transforms which represent a new representation of the two-dimensional digital images in analyzing the grey cover-image to embed the watermark in its frequency field is an attempt to make the suggested algorithm high solidity and increase the embedding capacity and at the same time preserving the quality of the cover-image and enabling to extract the watermark with the highest percentage of accuracy by studying the restored cover-image in addition to the watermark extracted and also the level of the effect on the quality and measuring the level of security of the algorithm using a secret which is expected to add another level of privacy.

1.4 Limitations of the Study

This thesis has the following limitations

1. This study is limited by the date that starts from August 2013 until May 2014.
2. Training on Matlab software before build the codes for the project.
3. Test image for the applied examples downloaded from the internet.
4. GUI build to make the proposed algorithms friendly used by the user.

1.5 Overview of the Thesis

Chapter One: Introduction of the depended system within the thesis.

Chapter Two: It deals with some previous works that are done before on the same topic of thesis like the descriptions of some researchers about hiding data and what they have been providing in this direction. Also includes the most important specifications contourlet transformations, as well as previous works which have been done to represent the digital image characterized through contourlet transformations.

Chapter Three: This chapter involves an introduction about the information and its branches with a review about the concept of the watermark and its classifications and mentioning the most important fields of applications in addition to the general structure of the watermark system; also a representation of the contourlet transforms in details in its two main stages the stage of the pyramid analysis and the directional analysis stage in addition to a presentation of the filters used in the analysis process.

Chapter Four: It tackled the design and the execution of the hiding system and the presentation of the empirical part of the research which is based on the watermark algorithm within the scope of contourlet transforms.

Chapter Five: The chapter presented the results and discussion with the conclusions that were reached by applying the embedding and extraction algorithms.

Chapter Six: This chapter included the conclusions and the future works suggested in this respect.

1.6 Summary

One of the challenge problems in the modern life is how to protect your personality and the security of your data, So many researcher did all their best to apply available algorithms with the available transforms. In this chapter a scope view try to be shown in order to warm up the later chapters in this thesis. In the literature survey the researcher looks for the present work in addition to the last few years.

CHAPTER 2

RELATED RESEARCH

2.1 Overview

Information privacy has become one of the important issues in information technology and its techniques developed vastly in the direction of image processing. The image transforms have a great portion in this respect. This thesis covers two important aspects which are the steganography and the contourlet transforms, therefore a portion of the researchers' works in both directions to include the works of some researchers which were published in the previous period of time.

2.2 Related Research in Data Hiding

Many researchers submitted several works in the subject of data security and in particular in the subject of steganography. The following are some of the current works in the field of the Steganography inside the digital images:

Fridrich (1998) suggested a new technique in the field of steganography to embed the message in the image based on the Palette type such as GIF files. According to the technique proposed, the embedding of the message in the image is performed through embedding each dual cell of the message data in a pixel in the image that is randomly selected using the generator of a pseudo-random number generator depending on a secret key. In each pixel in the image a dual cell of the message is embedded inside it and then the process of searching for the color which is nearest to the dual cell of the message and replaces it with the original color.

Yang and Chen (2008) developed a new method to hide the secret message by means of animation effects in the power point file. The animation effects were designed so that the person who makes the presentation can use them to emphasize the main points and to attract the attention during the presentation. The proposed method uses various animation effects to stand for different letters. A codebook was designed to record the letters and the correspondent animation effects and it is used to transform the letters of the secret letter into animation effects or vice versa. The proposed algorithm consists of two parts: the embedding process and the restoration processes. In both processes, the same code book is used which acts as a table lookup and this method does not distort the PowerPoint file.

Sudeep (2009) submitted in which he tackled in improving the storing capacity in the medical images that the capacity could reach one megabit of the data in grey images size 512×512. The researcher used Least Significant Digit LSD method to replace a dual cell of the secret message with the least significant number in the image.

Sarmah and Neha (2010) developed a new system of privacy which applies a proposed technique by the two researchers which uses steganography with cryptography the AES algorithm was used to encrypt the message Part of the encoded message is embedded in the coefficient of the Discrete Cosin Transforms DCT of the image and the other part of the image is used to generate two secret keys which increase the privacy of the system.

Mary (2010) submitted and proposed a new algorithm to hide the information in the real time using a compact video. In this algorithm, there is an embedding process and recovery that is performed in the compression domain without the need to the decompression process.

Mohan and Anurenjan (2011) researched and suggested a technique for hiding the information, adding another level of privacy by encrypting the ASCII of that data using an algorithm of encryption and then including it within the less important site of the contourlet transform transactions of the cover-image.

NitinJain (2012) submitted which they dealt with how to use the image edges to hide the secret message. Grey images were used in this research to try to find the dark areas the black points in the grey image are changed into dual images to categorize each object in the image into eight connected dual cells. The resulting images are changed into RGB images to find the dark areas. If the grey image was very bright, then we can change the histogram manually to find the dark areas. In the last step, each eight pixels in the dark areas are considered one octagonal block 1byte. The dual value of each letter of the secret message is concealed in the least significant dual cell for each octagonal block which is constructed manually from the dark areas to increase the privacy.

Saradha and Thamaraiselvan (2012) submitted and proposed a new algorithm for steganography using the spacing word-enter and spacing paragraph inter as a hybrid method. The unique characteristic of the proposed method is forming a text as a cover relying on the length of the secret message of the user.

2.3 Contourlet Properties Related Research

Because the contourlet transforms were recently employed in image processing for the purpose of information compression and recovering them, but the researcher in the field of steganography conducted some of their researches to include the application of steganography algorithms in the image processing which are disassembled using the contourlet transforms. The following are some of these works.

Farhad and Rabani (2010) suggested that an efficient technique to include the binary mark in the contourlet transforms. The first step was reformulating the image of the digital mark to one-dimensional matrix and then conducting a (XOR) to the resulting matrix with the secret key to get a single matrix with random data which is included in the contourlet transform coefficients.

Khalighi (2010) researched that used the Non-blind watermark scheme to include the fingerprint with the extension of bmp after dealing with it by the contourlet transforms using one analytical level. But, in the high frequency directional bands of the contourlet transform for the grey image of the cover that has high levels of energy.

Rahimi and Rabbani (2011) stated that due to the increasing importance of storing and transferring the digital medical images in a secure environment to preserve the privacy of the patient, in researched that presented a research in this field which aims at promoting the privacy and the reliability of transferring the data by using the binary watermark technique which relies on the contourlet transforms. The secret data with the binary format was included in SVD vectors of the low-frequency directional bands of the contourlet transforms after dividing them into blocks with the dimension of d×d.

Das and Kundu (2011) submitted a research that aims at overcoming the inclusion problems within the high and low frequencies in the transformations of image transforms. The researcher suggested a hybrid algorithm which combines the discrete cosin transforms DCT and the contourlet transforms as the watermark was encrypted using a certain encryption algorithm and then the resulting data were inputs of the discrete cosin transforms. While the cover-image was first analyzed using the contourlet transforms and then the discrete cosin transforms were applied on it and finally the data of the watermark was included in those final transformations of the cover-image.

Mahesh (2011) researched and proved that the accuracy and transparency of recognizing the cover-image after adopting the technique of non-blind watermark which depends on the contourlet transforms. The grey-colored image of the cover was analyzed using the contourlet transforms and the watermark was included after encrypting it using Knapsack algorithm in the fourth-level coefficients the high-frequency directional bands of the cover-image.

Kaviani (2011) suggested that an algorithm that is based on the contourlet transforms to include the watermark in an attempt to achieve high solidity. The first step was analyzing the cover-image using the contourlet transforms and then the transactions resulting are analyzed using SVD and finally the binary watermark data are included directly within it.

Mahesh (2011) decided that the non-blind watermark technique to include the grey image with dimensions of 64×64 in the field of contourlet transforms of the cover-image with the dimensions 512×512. After transforming the cover-image into four levels the watermark and the directional bands were divided into 8×8 blocks and the secret data are included in the high pass sub bands depending on the highest degree of similarity between the blocks of the watermark and the blocks of the directional bands. After that the inverse of the contourlet transforms is applied to get the image of the watermark.

Vidhyalakshmi and Vennila (2012) suggested that a new algorithm to reach an optimum method that accomplish the correspondence between the watermark and the transactions of the contourlet transforms depending on the entropy scale to select certain blocks in the cover-image and then dealing with it by the contourlet transforms and choosing the high-frequency directional sub bands to include the watermark data with the binary format.

2.4 Summary

This chapter can be summarized, that it explains that the data hiding plus watermark technique which applied in the last years and the ideas of some researchers which widely explained by academic thesis. Although the related works for the contourlet transformation applications in general field, with some specialization about its use in the direction of data hiding and security. Hybrid both of data hiding and contourlet transformation also shown.

CHAPTER 3

THEORITICAL FRAMEWORK

3.1 Overview

The development of computer science resulted in a rapid growth in the field of information technology and the wide spread of the networks (Tewari and Saxena, 2010). The distribution of the digital multimedia became an important means to deliver services all over the world throughout the marketing campaigns, electronic trade sites and as a result of the increasing use of the content of the multimedia many issues emerged including forging, illegal copy and hacking (Dua, 2012).

The emergence of hiding science offered several solutions within the scope of information security as it employs technologies which are characterized with credibility in the field of the digital communication that became recently the most used digital technology with the increase of the importance of the digital multimedia (Guyeux and Bahi, 2010). The need to protect these multimedia and it was necessary to reach certain means to provide the intellectual property protection to the innovators and distributors and it was found that the optimal solution to overcome these problems that are related to providing the reliability to the digital content for the two parties the producer and the consumer by including visible or invisible watermark in the multimedia. The copyrights and broadcast rights are considered the golden key for making this multimedia.

The watermark technology opened the door to the authors and publishers to protect their rights in those media. These technologies are represented by including certain information that prevents the illegal copy, the violation of the copyrights and broadcast discovering the manipulation with data. The purpose of the watermark is to prove the reliability, the digital content and the control the illegal access to data and not limiting the access to the digital multimedia.

3.2 Data Hidding

It is the science of hiding the secret communication and the term hiding refers to keeping the secret message unrecognized in case of discovering the secret communication by unauthorized person during transmitting the secret message included in the digital. Multimedia and it is considered the optimum solution when transferring the data secretly and safely through the internet (Tai and Chang, 2009). The Figure 3.1 shows a number of branch technologies used to hide the information and these ways and through the course of time occupied a considerable attention in the field of providing data reliability and privacy (Sikarwar, 2010).

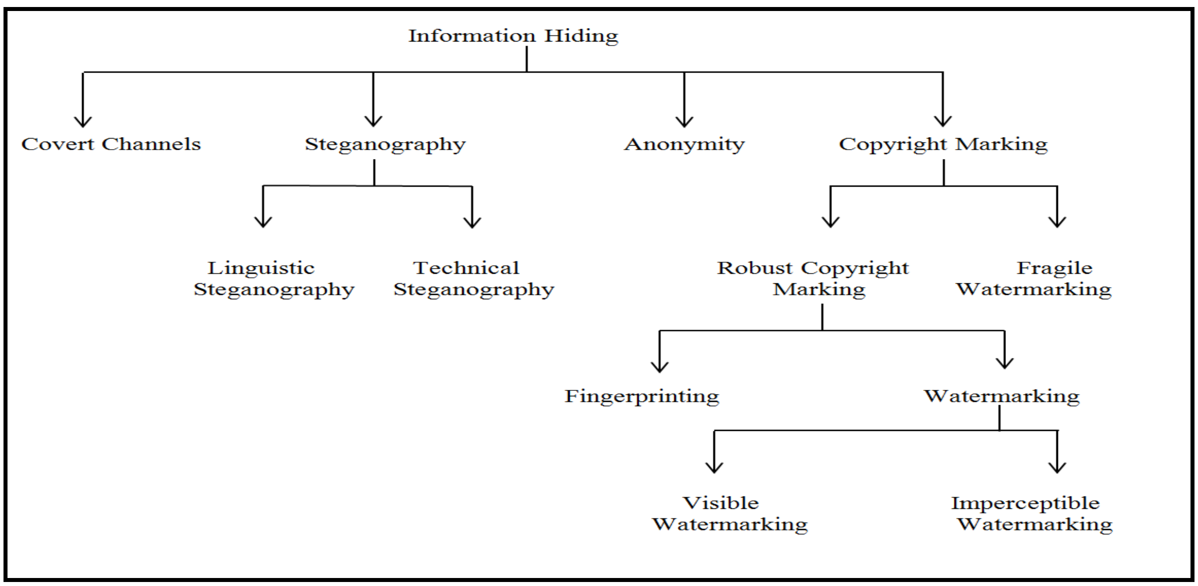


Figure 3.1: Classification of Information Hiding Technologies

3.2.1 Convert Channels

The technology of communication between two persons or more is described as the communicant path that is used for the secret transfer of data through the communication networks and these channels do not exist practically and not a part of the protocols of the computer networks, but they exploit the common internet protocols and some of the unexploited communication protocols are used to conceal the secret data as in the case of option field in the TCP protocol (Goudar and Edekar, 2011).

3.2.2 Steganography

It is one of the information hiding branches that are used to exchange data secretly and embedding these data in the transmitting cover without recognizing the existence of these data. To achieve this we can use different file formats (Sikarwar, 2010). The main goal of steganography is to accomplish the secret communication in a way that cannot be discovered and avoiding suspicion as this way aims at preventing the others from thinking that there are concealed data and hiding might fail in achieving this goal if there was suspicion that the data exist (Khalil, 2011).

3.2.3 Anonymity

It is the field that deals with protecting the identities of the clients whether the identity is a sender or a recipient identity or both (Goriac, 2011).

3.2.4 Copyright Marking

It is a type of property that can be bought or transmitted (Pun and Lam, 2009). in the recent years the technologies of copyright and broadcast to protect the property rights which enables the user to include secondary data in the digital content in a way that cannot be recognized but it can be readable through using some programs to discover them (Lin and Li, 2012). The signal might be robust and that means it can be taken out after the attacks, or it could be fragile and that means that changing the media included in them results in losing them. The robust signal of copyright marking is classified into two types, they are:

- **A- Fingerprinting:** Which are exceptional biological measures which are used as a means to identify the identity of the person within the domain of reliability the fingerprinting is considered the most famous amongst the various biological measures as it is regarded unique for every person and it is used widely to verify the identity and checking the personality (Chouhan and Khanna, 2011).
- **B- Watermarking:** It is a branch of the information hiding science and it is used to hide the information related to the property in the digital multimedia (Kamble, 2011). To protect the copyright and to ensure the reliability of data and secret communication (Alvarez and Armario, 2012). The digital marking techniques can be described as the operation in which a distinguished mark is included in various types of digital multimedia such as texts, photos, audio and video files without making any distortion in the host cover and it is possible in subsequent stage to discover these data and extract them for different purposes. The watermark might be a logo a trademark seal copyrights (Kamble, 2011).

3.3 Watermark

Watermark in general can be divided into two main kinds, visible and invisible these two main kinds are sharing for the general requirements for the purpose of embedding within the image that covers these requirements the following requirements must be taken into consideration when designing the digital watermarking algorithm.

3.3.1 Imperceptibility

It refers to the quality of the cover after embeding the watermark (Huang and Fang, 2010). The watermark should be hidden in the cover data without causing any influence that can be noticed by the bare eye (Singh, 2011). And it is impossible to distinguish the resulting image from the original cover-image (Huang and Fang, 2010).

3.3.2 Robustness

The robustness of the watermark is defined as the extent to which the watermark resist the attacks and the capability to extract the watermark after it suffers from such attacks and inability of these attacks to remove the data embed or distort them (Huang and Fang, 2010).
Some characteristics that should be available to get a robust watermark:

- Higher pay load: It should be characterized with its ability to include a big amount of data even in the case of the existence of many attacks whether deliberate or un deliberate; and without causing distortion to the transporting cover (Dukhi, 2011).
- Computational simplicity: Computational complications are of the concepts that should be taken into consideration when designing the robust algorithm of watermark. Being characterized with robustness from the scientific point of view should correspond with less computational complications when including and recovering the mark as it will be then of limited benefit in the real applications (Dukhi, 2011).

3.3.3 Capacity

It refers to the amount of the data that can be embedding in the cover (Katariya, 2012).

3.3.4 Security

Discovering the secret data embedding algorithm is considered one of the most difficult security problems, therefore the secrecy of data embedding in order to resist all the potential attacks that prevent fulfilling the desired goal of the watermark (Nyeem and Boyd, 2011).

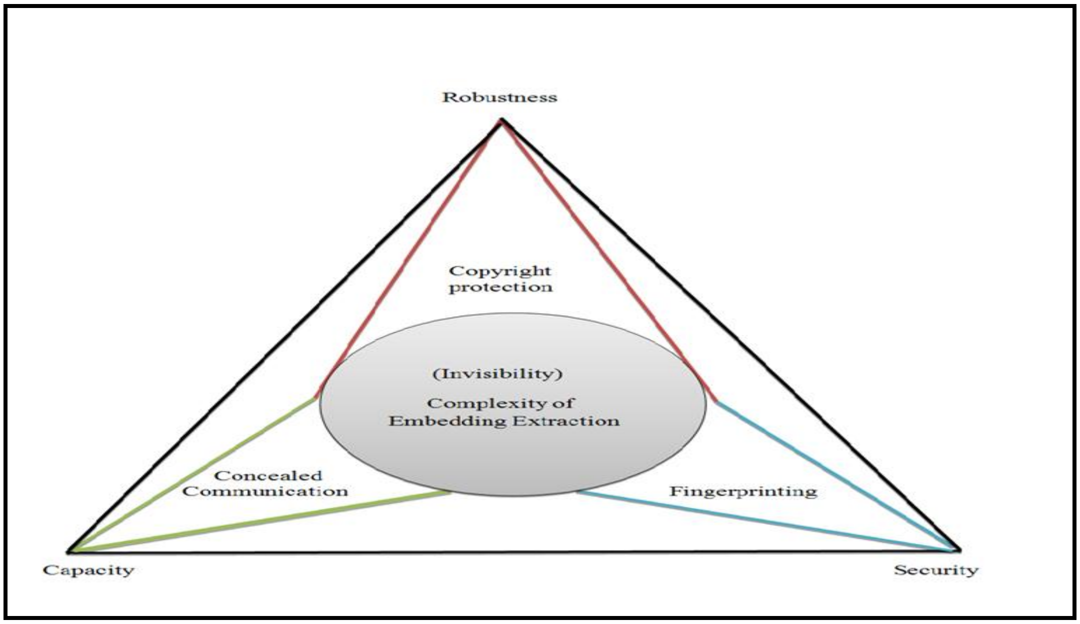


Figure 3.2: The Fundamentals of the Watermarking (Nyeem and Boyd, 2011)

From Figure 3.2 which emphasizes the necessity of finding a balance between robustness and capacity taking into consideration the other requirements when designing the watermark algorithm (Yershov and Rusakov, 2010). As the increase the amount of data results in making a distortion in the cover quality and decreases the robustness against the attacks (Singh, 2011). So the optimum density of data should be chosen when embedding in order to achieve the best hiding. In general not all the requirements mentioned could be met efficiently at the same time. Mostly the robustness characteristic should be available to achieve the copyright because those techniques require resistance against the attacks and that will correspond to hiding a relatively small amount of data (Mohamed and Sujatha, 2010). Taking into consideration that the size of the cover used in the watermark embedding should be bigger that the size of the watermark (Patel and Thakare, 2011).

3.4 Classification of the Watermark

The classification of the watermark depends on one of the following coefficients and as shown in Figure 3.3 (Chandra, Pandel and Chaudharl, 2010).

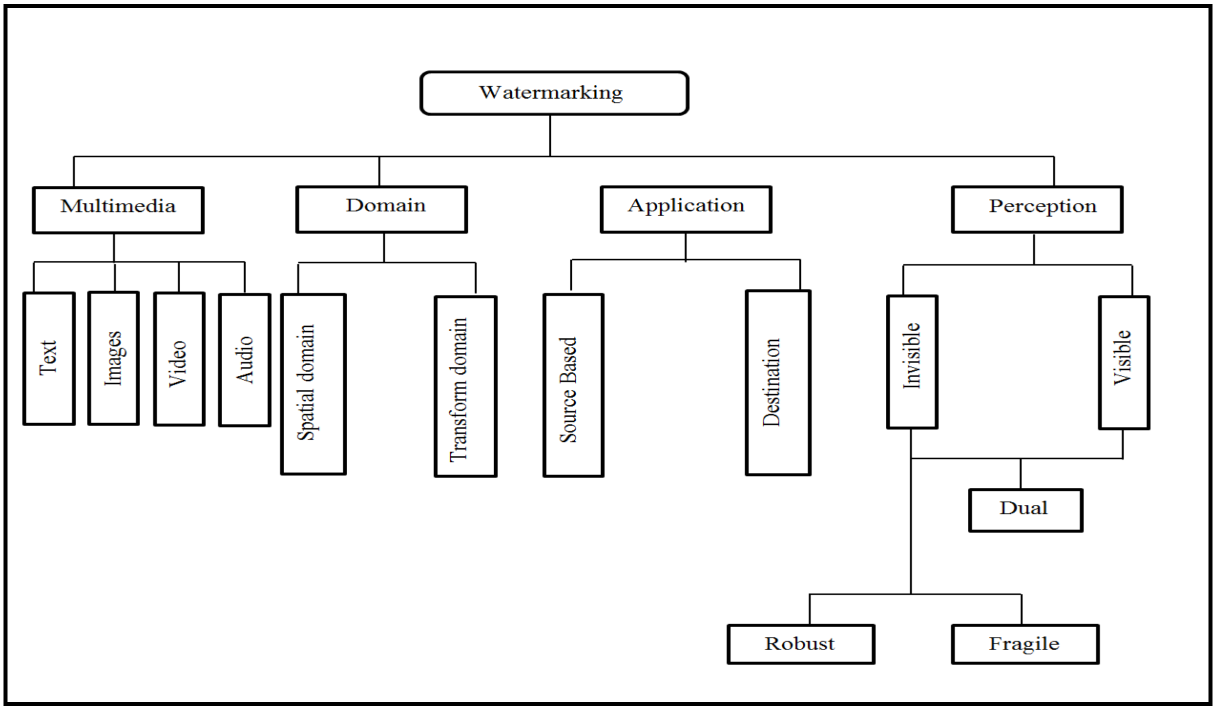


Figure 3.3: The Classification of the Watermark Techniques(Chandra, Pandel and Chaudharl, 2010)

3.5 Watermark Categories

The watermark is divided into four categories (Katariya, 2012):

a- *Text:* Adding the watermark to the text files (PDF and DOC files).

b- *Image:* Adding the watermark to the image components.

c- *Videos:* Adding the watermark to video files to control these applications.

d- *Drawings:* Adding the watermark to the two and three-dimensional drawings.

The watermark can be categorized in case of depending on the recognition system they can be divided into:

A- Visible Watermarks

The visible watermarks is considered one of the digital watermarks where the data the watermark are embedded in the cover in a visible way that can be realized by the bare eye, the resulting cover after the embedding process is totally different from the original cover and this technique is considered the most important and the most common to protect the digital multimedia files images and videos that are published for certain purposes and prevent the illegal copy of those media (Saraswathi, 2011).

There are certain desired characteristics in the visible watermark (Raj and Alli, 2012):

- The watermark is embedded within the wide scope or the important scope of the cover-image in order to prevent being removed.
- The watermark is visibly embedded without blocking the important details of the cover-image.
- It is hard to be removed because that requires enormous efforts and high cost that exceeds buying the watermark.
- The techniques of the visible watermark must be automatically performed with the least effort and human intervention.

B- The Invisible Watermark

This is embedded in the digital contents in a way that the data embedded cannot be recognized (Khanzode et al., 2011).

3.6 Some Advantages/Disadvantage in Watermarks

Because of the importance of the watermarking, many researches and studies were submitted in this field and less study were submitted in the field of the visible watermark. The objective of each of the types is hindering the theft of data, but the methods used in the field of the invisible watermark are different from those used in the other type. There are certain advantages of the watermark which are represented by the immediate claim of the copyright, but the main advantage of this type of the watermark is the actual exclusion of the commercial value of the documents which are subjected to be theft and without decreasing the benefits to be gained from the document in order to fulfill the reliability. From the other hand, the watermark is considered the optimum way to prevent stealing the data than being a way to catch the stealer of the data (Khanzode, Ladhake and Tank, 2011). In addition to that the site of the data embedding is unknown (Raj and Alli, 2012). The invisible watermarks can be classified into the following:

- **Robust Watermark:** The use of this type of watermark is generally for protecting the copyright and to prove the right of property. The high robustness is the basis for the robust watermark techniques as it resists all the process of images processing that aim at destroying or damaging the included watermark (Katariya, 2012).

- **Fragile Watermark:** The techniques of this type of watermark have a limited strength and they are very sensitive towards all the types of distortion (Katariya, 2012). As they are used for the purpose of proving the reliability and achieving security for the media and not for proving the right of property and the objective of designing this type of technologies is to discover the illegal manipulation because the small changes or the manipulation in the transporting cover results in a change or damage in the data of the watermark (Loukhaoukha,Chouinard and Taieb, 2011).

- **Semi-Fragile Watermark:** The design of this type is more robust than the fragile watermark and less affected by the modifications of the attack (Saha, Bhattacharyya and Bandyopadhyay, 2010). As it combines the characteristics of the fragile and the robust watermark characteristics in order to discover the illegal manipulations attempts in addition to its ability to resists those attacks and it can be used to verify the reliability. The characteristic of this technology is that it can distinguish between the aggressive and in aggressive attacks and this characteristic is not included in the technologies of the fragile watermark (Saha, Bhattacharyya and Bandyopadhyay, 2010).

- **Dual Watermark:** In this type, the invisible watermark is used as a support means to the visible watermark; therefore it is a mix of both types (Kamble, 2011). Figure 3.4 shows the dual watermark.

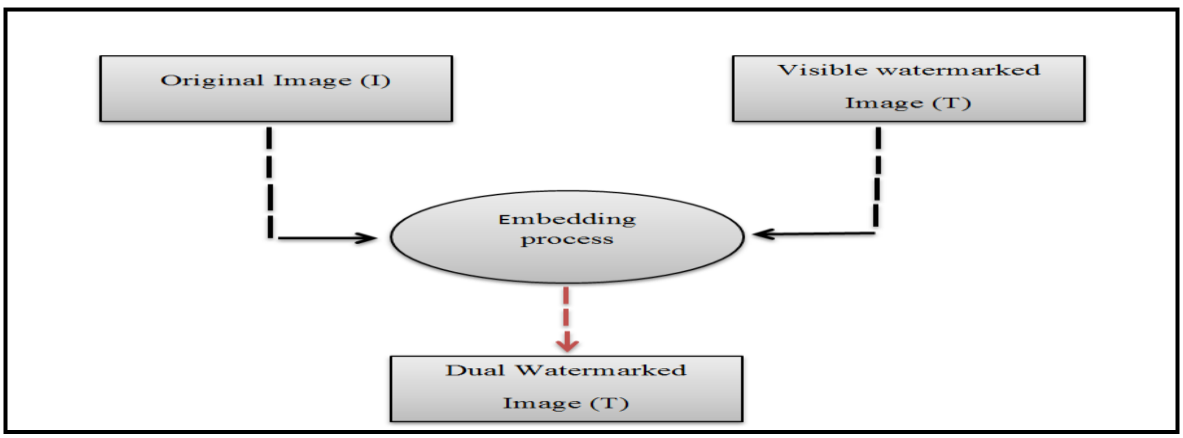


Figure 3.4: The Dual Watermark (Kamble, 2011)

3.7 Watermark Technologies

Depending on the need to the original cover-image to discover and extract the watermark, the technologies can be divided into three categories and as follows:

a. *Non-Blind:* This technique needs the original cover image to discover and extract the watermark, some watermark techniques of this type is called the private as it refers to the private key used in both the embedding and the extraction processes (Nyeem, Boles and Boyd, 2011).
b. *Semi-Blind:* This type is considered a branch of the blind watermark technologies and it is necessary to discover the included symbol and the private key without the need to the original cover-image (Aliwa, El-Tobely and Fahmy, 2010).
c. *Blind:* This type doesn't need the origi nal cover-image and it is sometimes called the Public in reference that the key is used in embedding and extracting the watermark (Nyeem, Boles and Boyd, 2011).

Also the watermark technologies are divided into two parts depending on the domain used in the embedding process (Singh, 2011):

- Spatial domain watermarking techniques.
- Frequency domain watermarking techniques.

In the spatial domain techniques, the data is included by the direct change of the pixel value of the cover-image, while in the techniques of the frequency domain the data is included after conducting the transform processing on the cover-image. So, the spatial domain technique is considered more suitable to the fragile watermarking as it lacks robustness against the image processing. This domain is characterized with the following (Bedi, Verma and Tomar, 2010):

- The watermarking system is characterized with its efficiency to discover any change in the cover-image after embedding the data.
- The process of data embedding must not affect the quality of the cover-image.
- The person who discovers has to have the ability to identify the location of change in the cover-image.
- Dependence should be on the private key agreed upon by the two parties to discover and extract the watermark; otherwise the data will be noise only.

The spatial domain technologies is characterized by the simple designing and constructing the embedding algorithms which are ideal for restoring the included data in the case of noise (Manoharan,Vijila and Sathesh, 2010). The disadvantages of this type are that the embedding processes are direct in the cover-image in addition to the low capacity of data embedding (Surekha, Swamy and Rao, 2010).

The technologies of the frequency domain proved to be more efficient in meeting the requirements of robustness, and the transparency of recognition of the digital watermark algorithms compared to the technologies of the spatial domain (Umaamaheshvari and Thanushkodi, 2012). Due to the difficulty of destroying the data included without making an evident change in the cover-image (Tawade, Mahajan and Kuithe, 2012).

The technologies of this type are based on using some transformations that reflect the domain of the cover image to the transform domain as in the case of the discrete cosin transforms, discrete fourier transforms and the discrete wavelet transforms, where the data of the watermark are embedded in the transformation coefficient by making some changes in the values of the coefficients of the domain and that is done through depending on the data to be included. So, the data is distributed in an irregular way through the cover-image after applying the inverse transform. This makes the manipulation and discovery of the watermark more difficult (Parthiban and Ganesan, 2012). Embedding the watermark within the middle frequencies in the cover image generates a high resistance against the attacks and at the same time the avoidance of change in the more important parts in the cover-image, represented by the low-frequency region (Tewari and Saxena, 2010). The logical interpretation of this is that much of energy lies within the low frequencies signal which involves the most important visible parts of the cover-image (Ahire and Kshirsagar, 2011). Therefore, the technology of the frequency domain is more complicated and its use demands many sophisticated mathematical operations (Parthiban and Ganesan, 2012).

3.8 Watermark Classification Based on Application

- *Source-based:* It is used in identifying the property right or proving the reliability and that is done by adding an exceptional watermark to all the relevant digital media that will be distributed.
- *Destination-based:* Every copy distributed from the digital media includes anexceptional watermark in order to identify the relevant buyer and it can be used to trace the buyer in the case of illegal resale (Kushwaha and Singh, 2011).

3.9 Fields of Some Watermark Application

The watermarking technology is characterized with important aspects which are it doesn't influence the quality of the cover-image and it is impossible to remove it from the host image during broadcast and finally, when the watermark is subjected to certain changes of the image that transforms it, it is possible to know something about these transforms through the resultant form of the watermark. Therefore, these three characteristics made the use of the watermark dependable in the various applications including (Castiglione et al., 2011).

3.9.1 Tamper Proofing

Within the field of protecting the digital contents, proving the reliability of the digital contents occupied much of attention and the fragile watermark is used in these applications to identify whether those digital media were illegally manipulated when transforming them through the unsecure channels (Yershov and Rusakov, 2010).

Currently, the communication image is a reality that cannot be transposed and there are many systems that use the image data to achieve the reliability of the user. In such cases, the image to be transformed for the purpose of reliability fulfillment needs more privacy. The use of the watermarking technologies supports the privacy of image transactions because the embedding of the private image that should be protected in the cover-image and proving the robustness of these technologies through restoring the original private image without causing any distortion in that extracted image (Indra and Ramaraj, 2012).

3.9.2 Copyright or Ownership Protection

It is one of the watermarking techniques to identify and protect the copyrights and preventing the others from claiming the ownership of the digital multimedia because the information of the brand ownership is included in the digital contents in a way that is characterized with high robustness and privacy to resist the attacks (Alvarez and Armario, 2012).

3.9.3 Fingerprinting

It is one of the watermarking applications that is used to trace the users of the digital content and that is done through embedding a unique mark with the identification data to determine the users of those digital contents. So, the digital media that are gotten illegally can be identified and observed (Yershov and Rusakov, 2010).

3.9.4 Copy Protection or Access Control

The watermark is embedding in the digital contents to prevent the illegal broadcast of the digital media and it is considered a policy for access control or copy control (Castiglione et al., 2011).

3.9.5 Concealed Communication

The watermark technologies are also used in exchanging the private information from the source to the goal a concealed way, as it is expected that these applications enjoy a high embedding capacity (Chandra, Pandel and Chaudharl, 2010).

3.9.6 Broadcast Monitoring

The watermark technologies are used in the advertisement applications by embedding the watermark with the media which are ready to be broadcast (Yershov and Rusakov, 2010).

3.10 The Main Structure of the Watermarking System

The algorithm of the watermarking consists of two main phases the embedding and the extraction phase.

3.10.1 Embedding the Watermark

It is the stage in which the private data is embedding and it is known as the embedded stage (E). The private data to be embedded in the cover is known as the watermark that can be a text, image, logo or the number (W) and the original image is expressed as cover-image or the host image used to embed the watermark and it is the carrier (I). The private key (K) might be used in the embedding process to make the system more secure. These components represent inputs of the embedding and the cover-image after embedding the watermark in it and then it is called the image of the watermark (I') which represents the output of this stage. Figure 3.5 shows the embedding process and it is described by the Equation 3.1 (Kumar and Santhi, 2011).

$$I' = E\,(I, W, K) \tag{3.1}$$

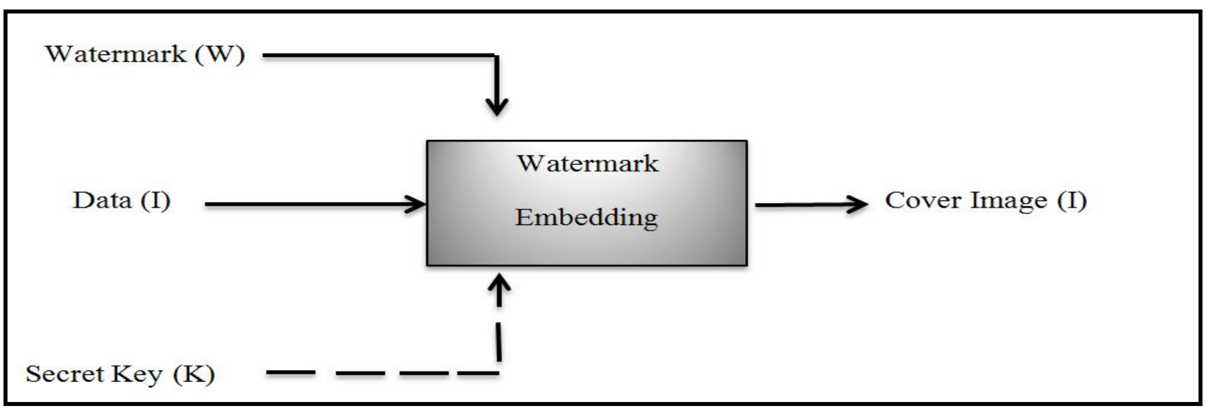


Figure 3.5: Algorithm of Embedding the Watermark

3.10.2 Watermark Extraction

It is the stage in which the private data is extracted from the image of the watermark, which is shown in Figure 3.6 the private key used in the embedding stage and the image of the watermark (I') represent the inputs of this stage, and the extracted watermark (W) represents the outputs of this stage and the extraction process (D) can be described by the following Equation 3.2 (Kumar and Santhi, 2011).

$$W = D\,(I', K) \tag{3.2}$$

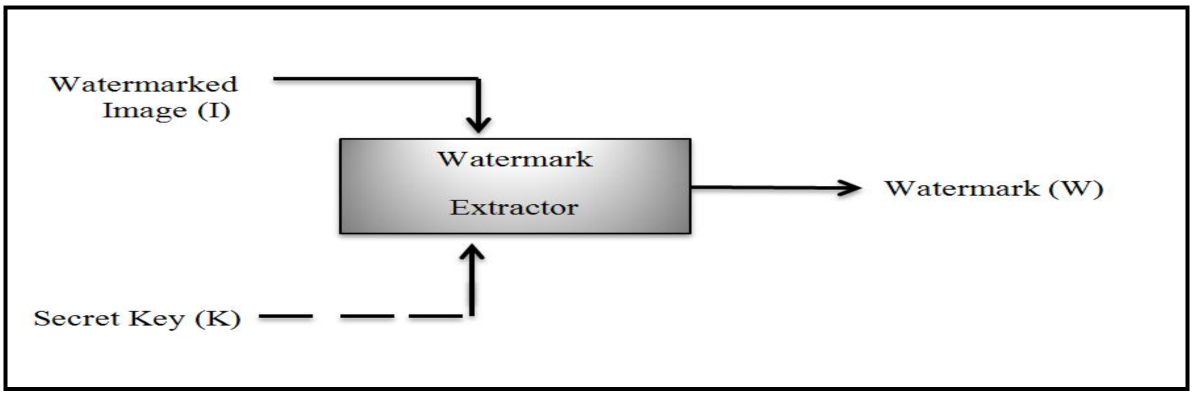


Figure 3.6: Algorithm of Extracting the Watermark

3.11 Contourlet Transformation

The contourlet transforms are a new presentation of the two-dimensional digital images which were suggested by (Do and Vetterli, 2009). Which is more efficient in the representation of the essential engineering structure of the image information like the smooth contours in the different directions of the image (Shan, Ma and Yang, 2009). Two structures of filters are used after merging them; which are Laplacian Pyramid LP and the Directional Filter Banks DBF. The filter LP is used first to analyze the image and the result will be a low pass image and a band pass image, and then it is followed by the DBF which is designed to include the high-frequency levels components (Rao and Rameshbabu, 2012). The filter LP is used in the first stage to capture the points between the gaps and then it is followed by the directional filter to connect the gaps between the points (Goudar and Edekar, 2011). Figure 3.7 (Majumder and

Saikia, 2011) and Figure 3.8 (Satheesh and Prasad, 2011) show the structure of the contourlet transforms.

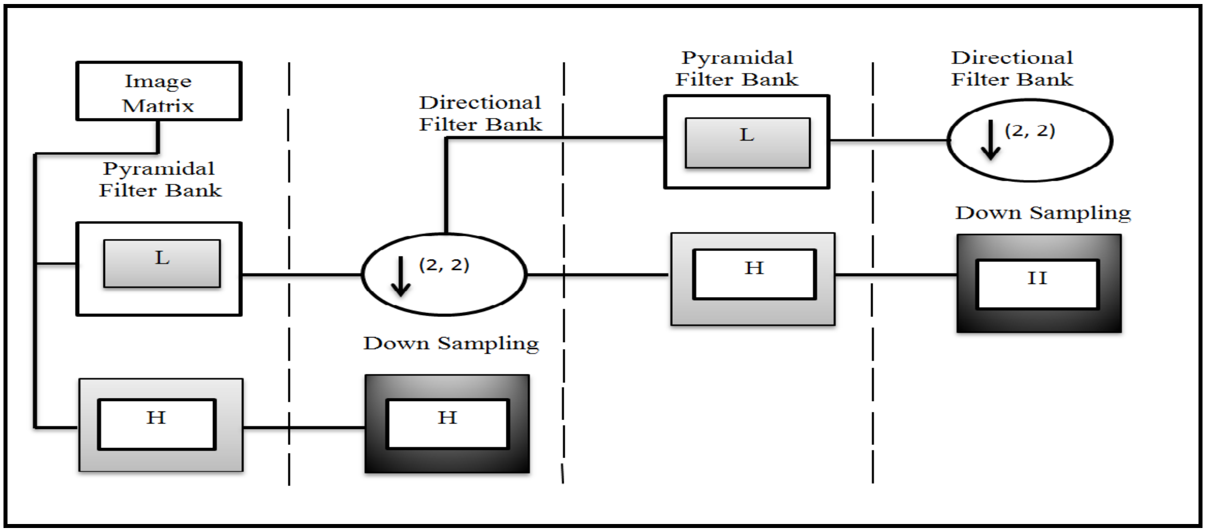


Figure 3.7: The Structure of the Contourlet Transforms (Majumder and Saikia, 2011)

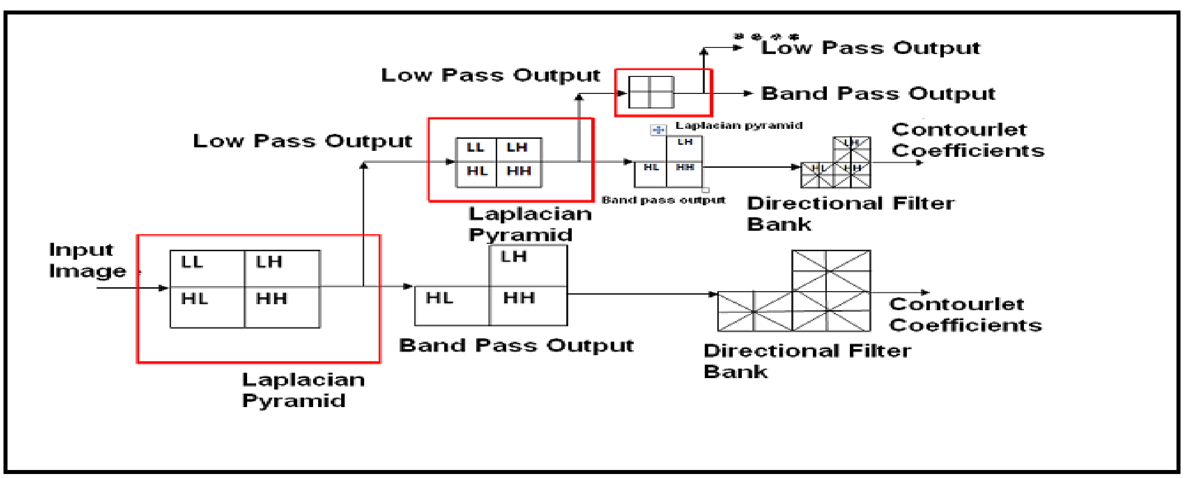


Figure 3.8: The Structure of the Contourlet Transforms and the Directional Filter (Satheesh and Prasad, 2011)

The contourlet transforms are characterized with the following (Tamilarasi and Palanisamy, 2011):

- **Directionality:** The representation of the image should involve the main directing elements in the various directions, much more than the few directions the horizontal and the vertical in the wavelet transform.
- **Multi-resolution:** The representation must allow a sequential approximation of the image from a course to a fine resolution image.
- **Localization:** The basic elements in image representation should be of fixed locations in the frequency and the spatial domains.
- **Anisotropy:** In order to capture the smooth edge curves in images, the representation should include the main elements using a various group of elongated shapes due to the different dimensions.
- **Critical sampling:** The representation of some applications such as the compression should constitute a basis or small redundancy framework.

3.12 Multi-Scale Decomposition

One of the methods to have image decomposition with multi scales is by using the Laplacian Pyramid filter LP (Tamilarasi and Palanisamy, 2011). Image decomposition using Laplacian Pyramid filter results in a sampled copy with low frequency (c) compared to the original image and the result is (b) too, which is the difference between the original image and the estimated image as the image will be a band pass (Hiremath, Akkasaligar and Badiger, 2011). The whole process could be shown by Figure 3.9 (Ardabili, Maghooli and Fatemizadeh, 2011) as follows:

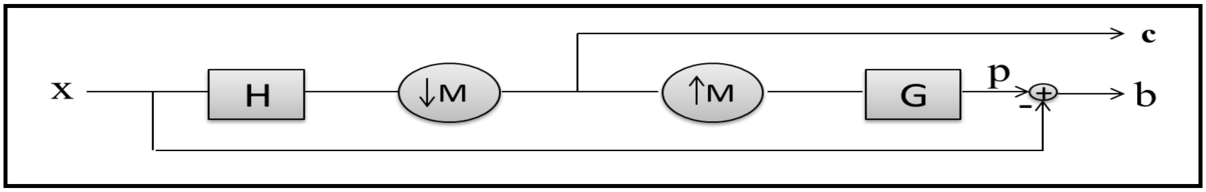


Figure 3.9: Decomposition Using Laplacian Pyramid (Ardabili, Maghooli and Fatemizadeh, 2011)

Where:

x: The original image.

c: The coarse approximation or the low pass image.

b: Prediction is the difference between the original image and the estimated one which lead to a band pass image.

And (H) and (G) arelow pass filters in the process of decomposition and composition respectively, (M) is the down sampling matrix and the symbol) M↓) represent the image down sampling through neglecting the even lines or columns of the image, but the symbol (M↑) represents the image re-composition by adding zeros to the even lines (or columns) of the down sampled image. The decomposition process can be repeated for the resulting image (c) to have a composition for more than one level as these images are stacked as a regular pyramid and this is why it is named Laplacian Pyramid. The Figure 3.10 (Ardabili, Maghooli and Fatemizadeh, 2011) shows the details:

Where:

x: The original image.

c: Coarse approximation of low pass image.

b: Prediction is the difference between the original image and the approximate image which results in a band pass image and that H and G are low pass filters in the operation of decomposition and construction respectively and M is a down sampling matrix and the symbol (M↓) represents the image down sampling by ignoring the even lines (or the columns) of the image but the symbol (M↑) represents the rearrangement of the image through adding zeros to the even lines or the columns of the down sampled image.

The decomposition operation can be repeated on the resulting image (c) to obtain the decomposition for more than one scale as these images are stacked as a regular pyramid and that why it is called the Laplacian Pyramid as in Figure 3.10.

The representation of the data in multiple levels is an efficient and effective idea, as they capture the data in a pyramidal shape. The main idea of the Laplacian Pyramid is to extract the coarse image from the original image by means of the low pass and the down sampling process. By depending on the coarse copy the original image can be estimated by the up sampling process and G filter and then the difference is calculated b (Devanna and Kumar, 2011).

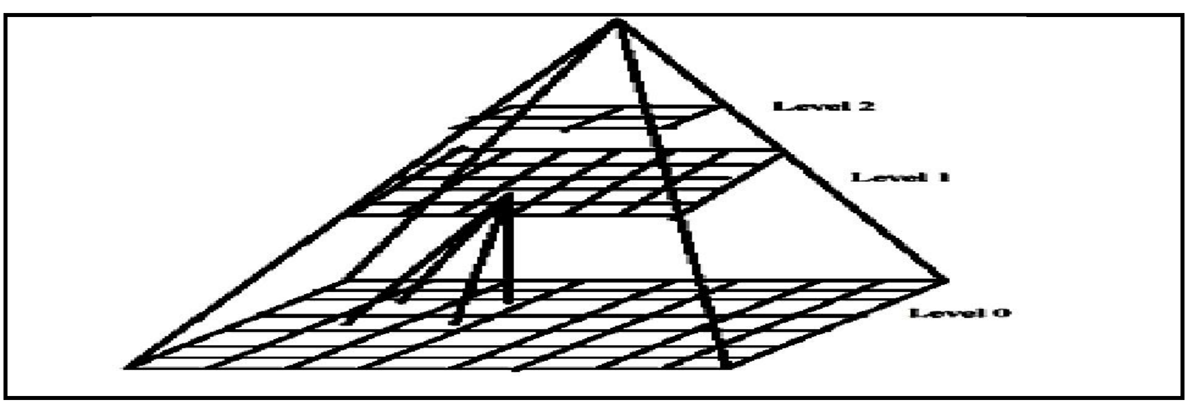


Figure 3.10: Laplacian Pyramid (Devanna and Kumar, 2011)

The coarse image can be obtained by the Equation 3.3 as follows (Do, 2001).

$$c[n] = \sum_{k\epsilon z}^{d} x[k]h[m_n - k] \quad (3.3)$$

Filtering and up sampling result from the Equation (3.4) (Do, 2001) as follows:

$$p[n] = \sum_{k\epsilon z}^{d} c[k]g[n - m_k] \quad (3.4)$$

We can say that: Hx = c and Gc = p

Where:

P: is the estimated image

G: and H: the filters of the low pass, (↓M) H, (↑M) G and when m =2, then (Guyeux and Bahi, 2010).

$$G = \begin{pmatrix} \ddots & \vdots & & \\ & g[0] & & \\ & g[1] & \vdots & \\ & g[2] & g[0] & \\ & \vdots & g[1] & \\ & & g[2] & \\ & & \vdots & \ddots \end{pmatrix} \quad H = \begin{pmatrix} \ddots & & & & & & \\ \ldots & h[2] & h[1] & h[0] & \ldots & & \\ & & \ldots & h[2] & h[1] & h[0] & \ldots \\ & & & & & & \ddots \end{pmatrix}$$

3.13 The Directional Decomposition

Smith and Bamberger (1992) Truc and Khan (2009) Suggested the two-dimensional directional filter bank and it is represented through the structure of decomposition (L-level) that results in 2^L of band pass and it decompose the image into binary numbers with dividing the frequencies as wedge-shaped.

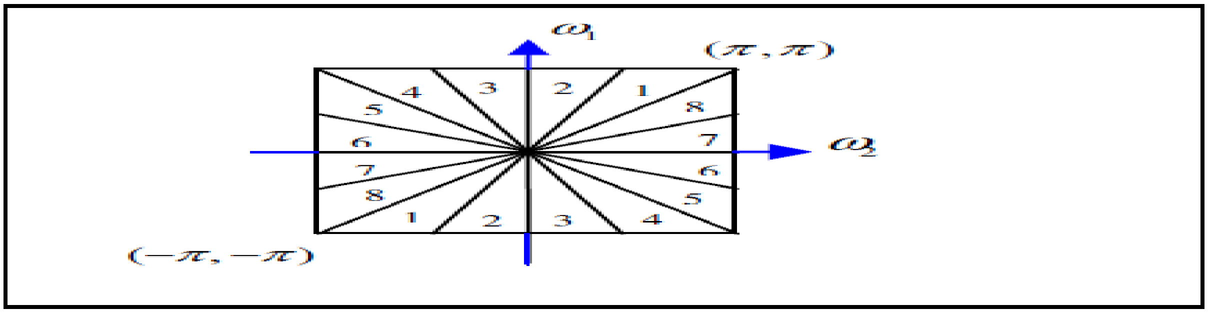


Figure 3.11: The Wedge-Shaped that Represents the Frequency Parts of the Directional Filte Bank (Xingmei and Guoping, 2010)

The original structure of decomposition of the DFB involved modulating the input image and using the Quincunx Filter Bank with Diamond-Shaped filters But the frequency areas of the frequency bands do not provide the directional divisions shown in Figure 3.11 therefore a new structure of the iterative DFB were used which depend on the Quincunx Filter Bank QFB (Xingmei and Guoping, 2010). The core of the iterative directional filter bank with the fan filters (Yang et al., 2010). Thus it was far from modulating the input image and the complications of the decomposition process.

The new form of the directional filter bank consists of two structures (Xingmei and Guoping, 2010):

The first structure is two channels, each of which has Quincunx Filter Bank QFB with the fan filter, as shown in Figure 3.12 it divides the two-dimensional spectrum into two directions; horizontal and vertical.

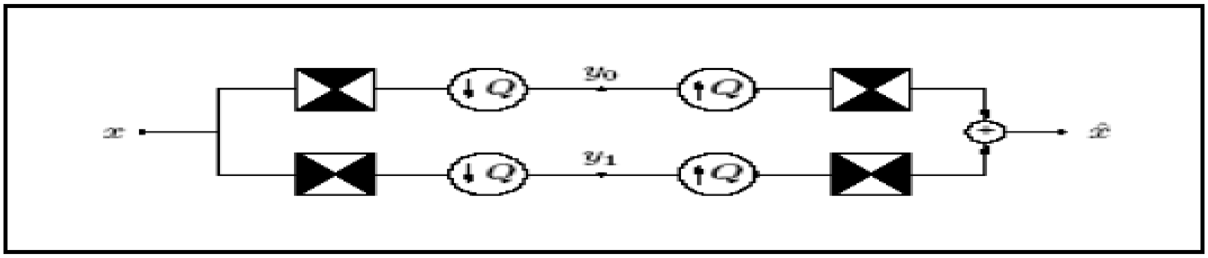


Figure 3.12: The First Structure of the Directional Filter Bank (Xingmei and Guoping, 2010)

The second structure of the directional filter bank consists of a pair of shearing operators, each of them for one channel of the filter which rearrange the image samples only, i.e. It rotates the image without changing the values of the data and consequently we have two different directions. Figure 3.13 shows the shearing process with a (45 degree) angel of a cameraman.

(a) The Original Image (b) The Image After the Shearing Operation

Figure 3.13: Shearing Operation was Used as a Rotating Operation (Xingmei and Guoping, 2010)

The frequency parts can be achieved for the stake-shaped Figure 3.11 and that is done by merging the directional parts with the fan filter for the QFB and the directional frequency portions resulting from rotating by means of shearing operator (Sandhya, 2011).

The Quincunx can be represented by the following two-dimensional matrices:

$$Q0 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \quad Q1 = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$$

The sampling using the (Unimodular Matrix) doesn't change the value of data but only rearrange the input image. Therefore, the operation is called the resampling operation. The following matrices are integer square limited with (±1) used in the directional filter bank sequentially to provide the Equation of the rotation operation:

$$R0 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \quad R1 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$$

$$R2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad R3 = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix}$$

up sampling using R_0 is equivalent to the down sampling using R_1 (Patel and Thakare, 2011). One of the several multi-dimensional decomposition methods is called smith form, which analyzes any correct matrix into UDV, where U and V are Unimodular Integer Matrices and D is diagonal integer matrixes. According to Smith form, the filter Quincunx is as follows (Yang and Hou, 2010).

$$Q0 = R1, D0, R2 = R2, D1, R1 \qquad (3.5)$$

$$Q1 = R0, D0, R3 = R3, D1, R0 \qquad (3.6)$$

Where:

$$D0 = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \quad D1 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

3.13.1 Quincunx Filter Bank

The matrix Q_0 or Q_1 according to the levels and according to the down sampling rotate the input images with -45 to +45 degrees respectively, and used to divided the frequency spectrum of the input image to the low pass channel and high pass channel using a pair of diamond-shaped filter. The frequency spectrum is divided into the (Horizontal and Vertical) channel by means of using the fan filter pair. It is possible to have one of the filter pairs from the other filter by modulating the filters with a value of (π) by taking the frequency variables either W_0 or W_1 (Kamble, 2011).

(a) Represents a Pair of the Diamond-Shaped

(b) Pair of Fan Filter

Figure 3.14: Two Structures Inside the QFB Each Area Represents the Optimum Frequencies by a Pair of Filters Filter (Kamble, 2011)

3.13.2 Obtaining Four Directional Frequency Partitions

In order to obtain four directional frequency partitioning as in Figure 3.15 (Lydia et al., 2011). So the Sampling Matrices Q_0 and Q_1 are in the first and second levels respectively. As a result of the sampling after the two levels I_2, 2 = Q1 Q_0 or down sampling with a value of (2) for each dimension.

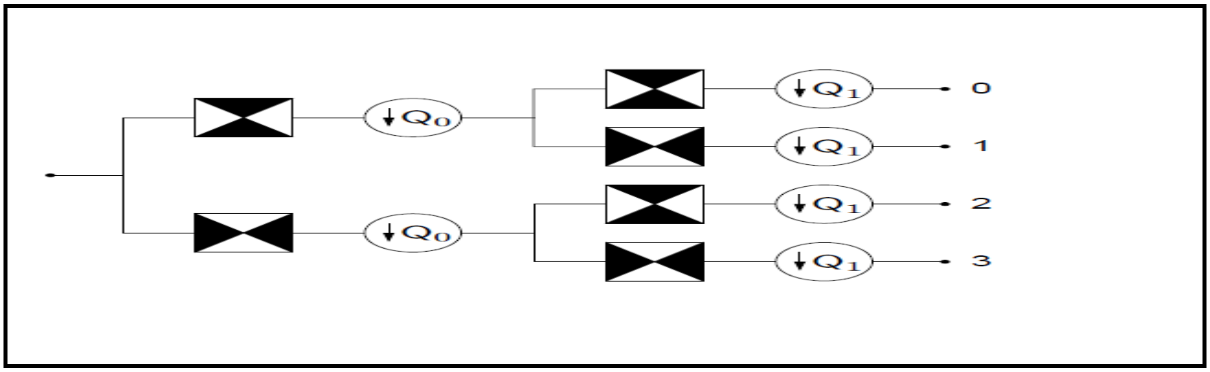


Figure 3.15: The Directional Filter Bank (Lydia et al., 2011)

where the fan filter is used the down sampling by the use of sampling matrix (M) and then it is followed by the filtration by the filter H (w), and this is equivalent to the filtration by the filter H (M^T w) that can be obtained by the sampling operation for H (w) by (M) before the down sampling operation as shown in Figure 3.16.

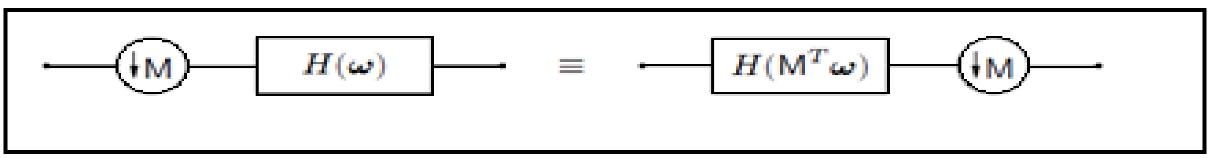
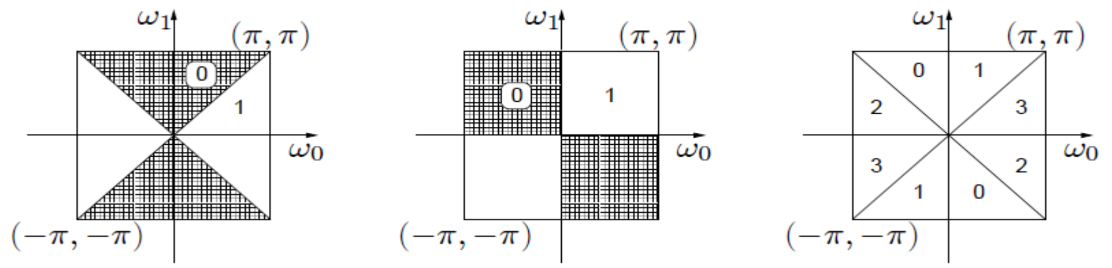


Figure 3.16: The Correspondence of the with the Downsampling Operation (Lydia et al., 2011)

From what has been mentioned above, the filters in the second level of Figure 3.15 in the sampling operation Q_0, and this replacement turns the fan filter into an equivalent filter with frequency response of 90 degrees (Quadrant Frequency Response), and the operation of merging it with the fan filter in the first level results in four directional band passes as shown in Figure 3.17.



(a) The first level: Fan filter two   (b) The second level: Quadrant Filter levels supports   (c) Directions that the Merge of the

Figure 3.17: Shows the Equivalent Filters in the First and Second Levels for the DFB (Gao et al., 2012)

At the third level, and to achieve more accurate partitions, Quincunx filter is used with the resampling operations as shown in Figure 3.18.

Figure 3.18: Shows the Quincunx Filter with the Resampling Operation Used in the DFB Filter Beginning From Level Three (Do, 2001)

There are four types of resampling for the QFB filter based on the four sampling matrices mentioned before (R0, R1, R2 and R3); the types R0 and R1 are used in the first half of the DFB filter and they lead to the frequency partitions that correspond the horizontal partitions or the directions between 45˚- and +45˚ (an example for these partitions 1, 2, 3 and 4) in Figure 3.11 while type 2 and type 3 are used in the second half of the DFB filter channels that correspond to the rest of directions.

The second half is constructed from the DFB filter channels by alternating the dimensions N0 and N1 with the correspondent channels in the first half and this is applied to all the sampling matrices, as R0 becomes (R2), Q0 becomes Q1 in the DFB filter. Using the filters in Figure 3.16 resampling DFB can be equivalent in the right side of Figure 3.19 (Do, 2001) and as follows:

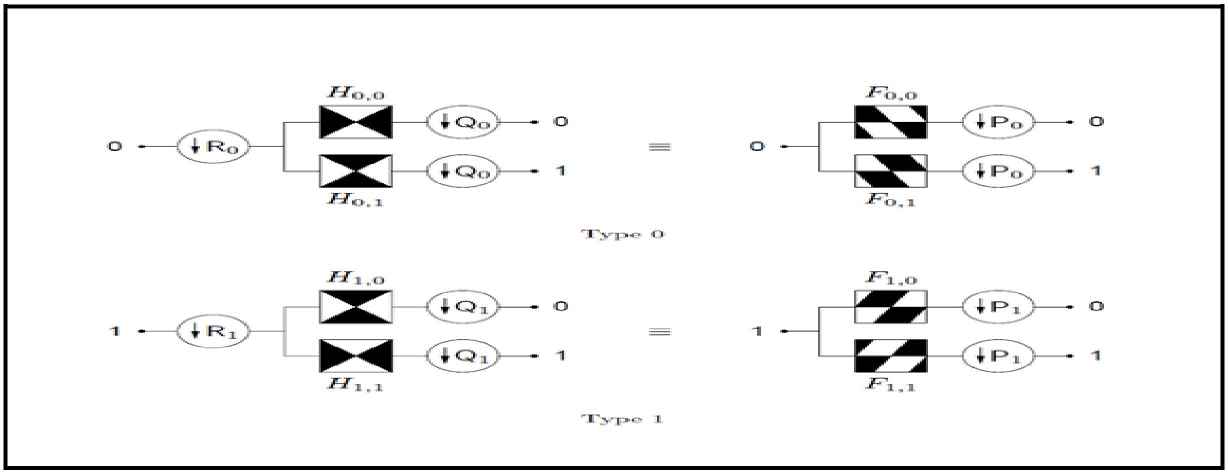


Figure 3.19: The Left Side Represents the Analytical Aspect of DFB that is Used in the Third Level in the First Half of QFB Channels

The right side represents the equivalent filter that uses the parallelogram filters and the black parts represent the optimal frequencies provided by the filters. By noticing the frequency arrangement provided by the fan filters in QFB, we can say that the expansion laws in the third level of the first half of the DFB, the repeater will be as follows:

At each knot the upper channels use the first type (Type 0) of the filter while the lower channels use the second type (Type 1) of the filter. It can be seen that the analytical part of QFB is equivalent to the right side of Figure 3.19 as shown in Equation 3.11.

$$F_{i,j}(w) = H_{i,j}(R_i^T w) \tag{3.7}$$

Followed by the down sampling with Pi = Ri Qi, where i and j belong to {0, 1} bellow (Patel and Thakare, 2011). The filter $F_{i,j}(w)$ is the result of resampling the fan filter $H_{i,j}(w)$ in Equation 3.7 mentioned earlier and it called the parallelogram filters. From Smith formula in Equations 3.5 and 3.6 the matrices can be explained in the resampling operation of QFB in Figure 3.19 as follows:

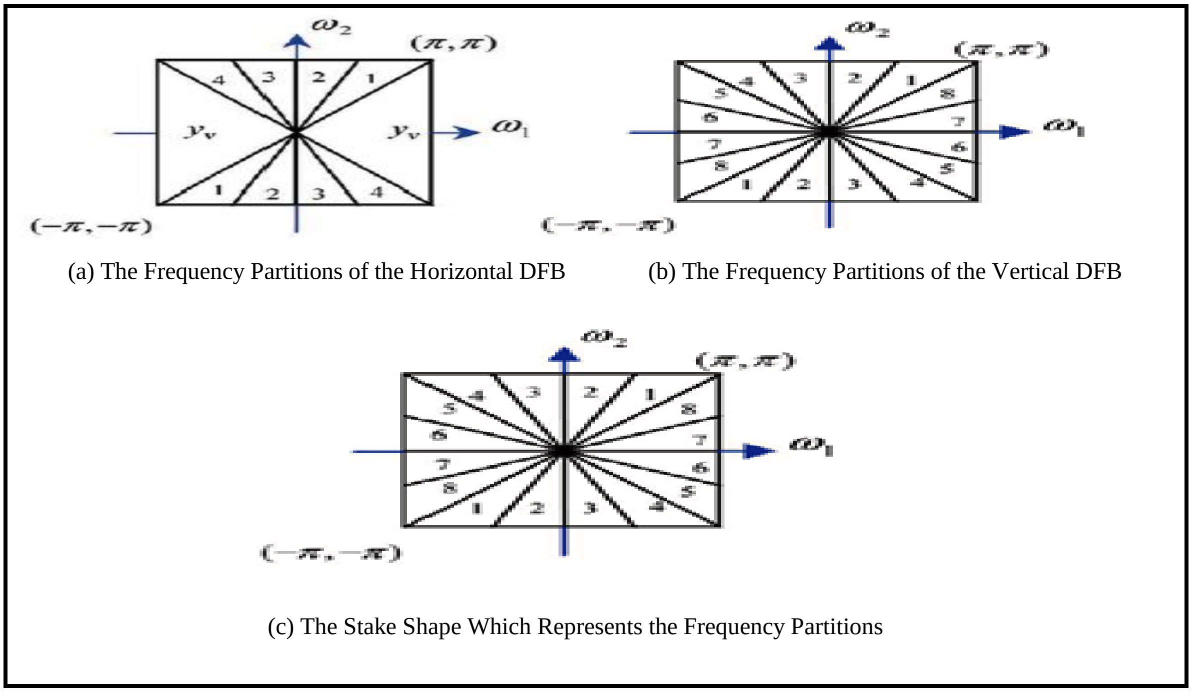
$$P0 = R0, Q0 = D0, R2 \tag{3.8}$$

$$P1 = R1, Q1 = D0, R3 \tag{3.9}$$

So, we can say that resampling the QFB for Type0 and Type1 is a down sampling operation with a value of (2) for the dimensions N0 (Goudar and Edekar, 2011).

3.13.3 Vertical Directional Filter Bank and Horizontal Directional Filter Bank Vertical DFB and Horizontal DFB

The stake-shape with the eight frequency band passes as shown in Figure 3.11 results from analyzing the DFB of the frequency partitions (Gao et al., 2012). We can say that the partitions 1, 2, 3 and 4 represent the horizontal directions between (-45 and +45) and the rest of the directions are the vertical directions between (+45 and + 135). The DFB can be achieved using the Quincunx filter repeatedly and in the operations of decomposition horizontal and vertical operations the Vertical DFB and the Horizontal DFB are used as in Figure 3.20 bellow.



(a) The Frequency Partitions of the Horizontal DFB (b) The Frequency Partitions of the Vertical DFB

(c) The Stake Shape Which Represents the Frequency Partitions

Figure 3.20: The Supported Frequency Partitions (Zhao and Heegjian, 2011)

3.14 Multi-Scale and Multi-Direction Decomposition Multi-Scale and Directional Decomposition

The DFB was designed to capture the high-frequency components the directional representation of the images. Therefore, the low frequency components are few in this filter. In fact, the low frequencies leak into some band passes directional groups before commencing the DFB. For this reason, the directional filter is merged in the multi-resolution idea (Arun and Menon, 2009).

The LP filter allow the idea of analyzing with band pass groups until the directional analysis is applied to each resulting band pass image . The DFB is applied to the band pass images and thus the directional information can be captured more efficiently. This idea is applied repeatedly to the coarse images with low frequencies resulting from each LP decomposition, with indication that the low frequency image is sub-sampled, while that is not applied to the image with high frequencies. The final results will be a structure of a PDFB that decompose into band passes for more than one level and it is a flexible idea that allows different directions in each level (Soleymanpour et al., 2010). Figure 3.21 shows applying an image to the idea of contourlet transforms structure.

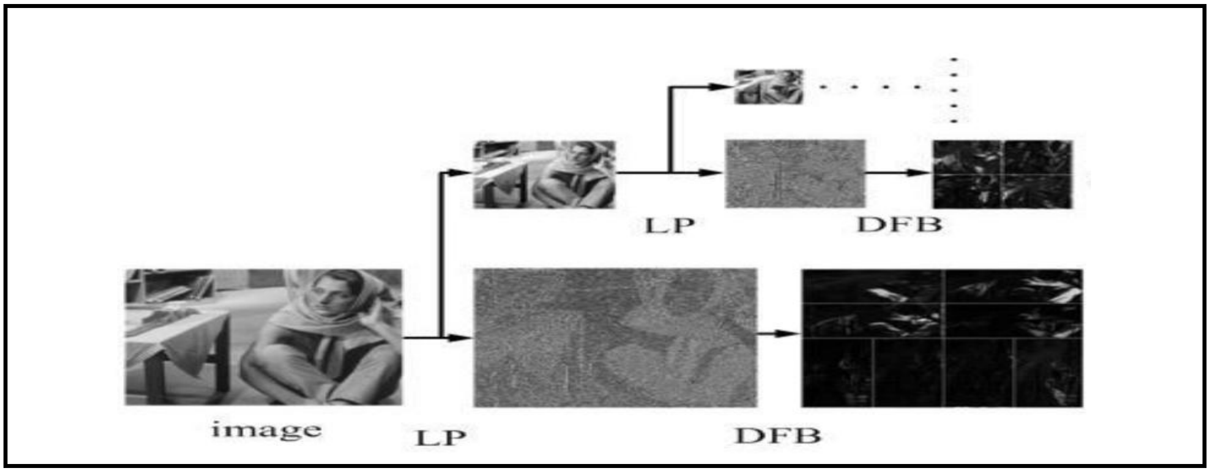


Figure 3.21: Applying an Image to the Idea of Contourlet Transforms Structure (Soleymanpour et al., 2010)

3.15 Multi-Resolution Decompositions

The multi-resolution decompositions consist of two main parts: The multi scale and the multiple direction decompositions.

3.15.1 The Multi-Scales

In PDFB, the LP uses perpendicular filters and a down sampling of a value of (2) for each dimension. So, the low pass filter (G) in LP plays a role as an Orthogonal Scaling Function in the Equation 3.10 as follows:

$$\phi(t) = \sum_{n\in\mathbb{Z}^2}^{n} Q[n]\ \phi(2t-n) \tag{3.10}$$

$$\phi j, n = 2 - j\phi\left(\frac{t-2^j n}{2^j}\right), j\epsilon\mathbb{Z}, n\epsilon\mathbb{Z}^2$$

We can say that $\{\emptyset_{\mathrm{j,n}}\}_{n \in \mathbb{Z}^2}$ is with perpendicular for the space V_j at level 2^j and $\{V_j\}_{j \in \mathbb{Z}}$ provides a series of spaces with interrelated multi resolution (Wu and Lee, 2009).

Where V_j is within the dimensions $2^j \times 2^j$ that describe the image at the level 2^j. The difference image (b) see Figure 3.9 includes the necessary details to increase the resolution between each to perpendicular subsequent spaces. Therefore, it is inside the space W_j and the perpendicular complement is in V_{j-1} as shown in Equation 3.11, as shown in Figure 3.22 (Yang et al., 2010).

$$V_{j-1} = V_j \oplus W_j \tag{3.11}$$

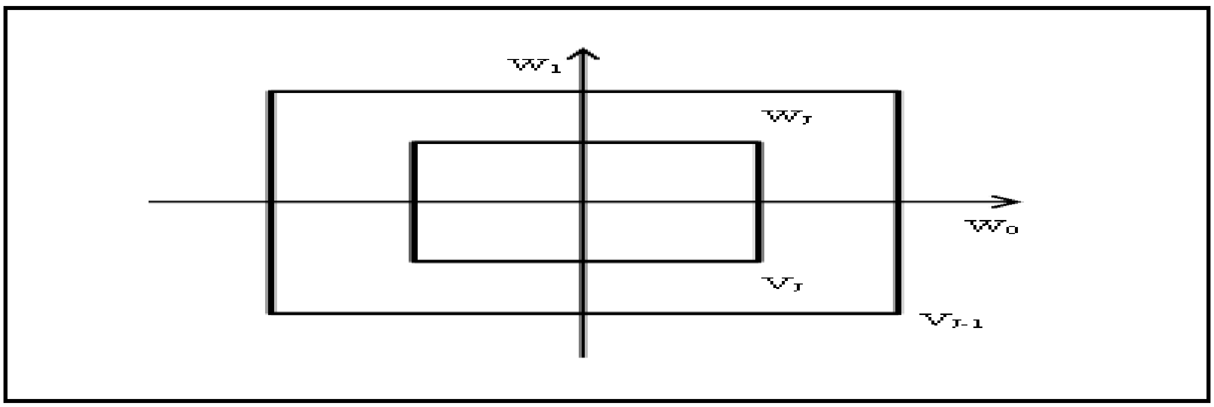


Figure 3.22: The Sub Spaces in Multiple Levels Formed in the Filter LP (Yang et al., 2010)

The filter LP leads to two images; the coarse image c[n] and the difference image b[n] in Figure 3.9 as mentioned earlier. Each of them comes from a separate and with the same down sampling matrix (M=2) i.e. (diag (2, 2)), considering that the filter F_i (z)، $0 \leq i \leq 3$ is the high pass filter which is the reconstruction filter, as in the transforms of the wavelet. The continuous function $\psi^{(i)}(t)$ will be involved with each one of these filters, as in the Equation 3.12 (Do, 2001).

$$\psi^{(i)}(t) = 2 \sum_{n\epsilon\mathbb{Z}^2} f_i[n]\phi(2t-n)$$

That means:

$$\psi_{j,n}^{(i)}(t) = 2^{-j}\psi^{(i)}\left(\frac{t-2^jn}{2^j}\right), j\epsilon\mathbb{Z}, n\epsilon\mathbb{Z}^2 \quad (3.12)$$

3.15.2 The Multiple Directions

Supposing that the directional filter in the contourlet transforms uses perpendicular filters, in the contourlet filters after the intermittent base of the DFB is regarded as an alternative for the continuous subspaces of the multi-level analysis although in the contourlet transforms of DFB is applied to various images or W_j scales, the DFB will be applied to the multi resolution V_j scales (Zhao and Hengjian, 2011). Figure 3.23 shows two parts of two different directions after applying DFB to them.

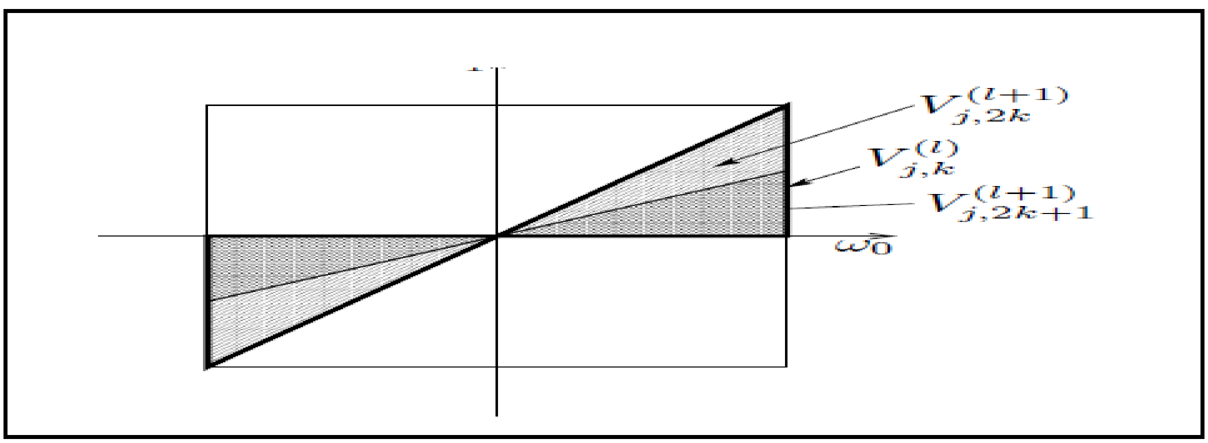
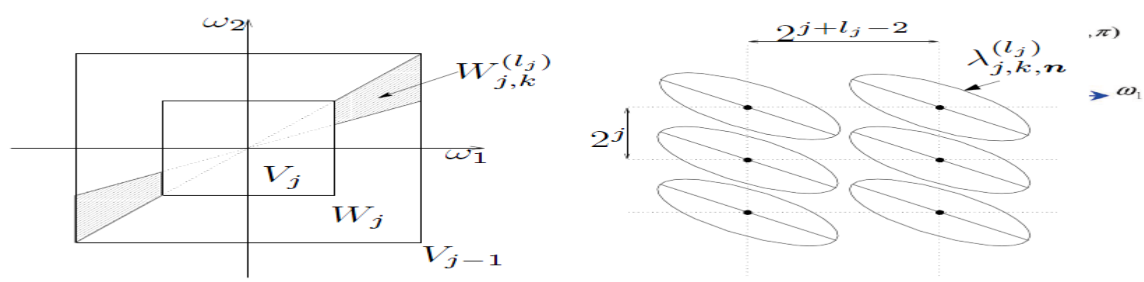


Figure 3.23: Multi Directional Scales Composed by the Filter DFB (Zhao and Hengjian, 2011)

3.15.3 Multi-Scale and Multi Directional

Merging both of the filters LP and DFB into the final form PDFB and the directional decomposition is applied to the space W_j. Figure 3.24 shows in details the directional scales $W_{j,k}^{(l)}$ in the two-dimensional frequency space and the network representation of the contourlet transforms in the horizontal direction (Do, 2001).



(a) The Multi-Scale Spaces and Multi Directions Shown with the two Dimensional Shape

(b) The Net Representation of the Contourlet for the Space $W_{j,k}^{(l)}$ in the Horizontal Direction, and for the Vertical the Net is Reversed

Figure 3.24: The Spaces of the Contourlet Transforms (Do, 2001)

Where:

- j: The indicator of the scale.
- k: Direction indicator.
- n: Location indicator.
- l: DFB Decomposition Levels indicator**.**

It is clear that the directional decomposition Level might differ from the number of levels j and in this case it is referred to as l_j (Do, 2001).

The number of the directions is doubled at each finer scale (Chen and Kegl, 2010). This means that at the scale $2^{j_0}$the beginning will be with the scale l_{j_0}in the DFB (which has $2^{l_{j_0}}$ directions) and then at finer levels 2^j where $j < j_0$. Therefore, the number of the decomposition scales in the decomposition with the DFB should be as in Equation 3.14 bellow (Meskine, Mezouar and Taleb, 2010).

$$l_j = \left[l_{j_0} - \frac{j-j_0}{2}\right], for\ j \leq j_0 \tag{3.13}$$

In Figure 3.25 we see that LP size decreases four times less than the original size, while the number of the directions doubles in either successive levels in the pyramid shape the reason behind that is due to the sub-sampling in the two filters. Therefore, the size of PDFB that depends on the image changes from a level to another according to the relationship of curve scaling and it is considered one of the PDFB characteristics and it is worth mentioning that the more the level advanced the thinner the LP becomes and there will be more directions in the decomposition process (Ardabili, Maghooli and Fatemizadeh, 2011).

It is noticed that the resolution increases in the spatial and the directional domains when moving from the coarse levels to the finer levels (Meskine, Mezouar and Taleb, 2010) as shown in Figure 3.25 and Figure 3.26.

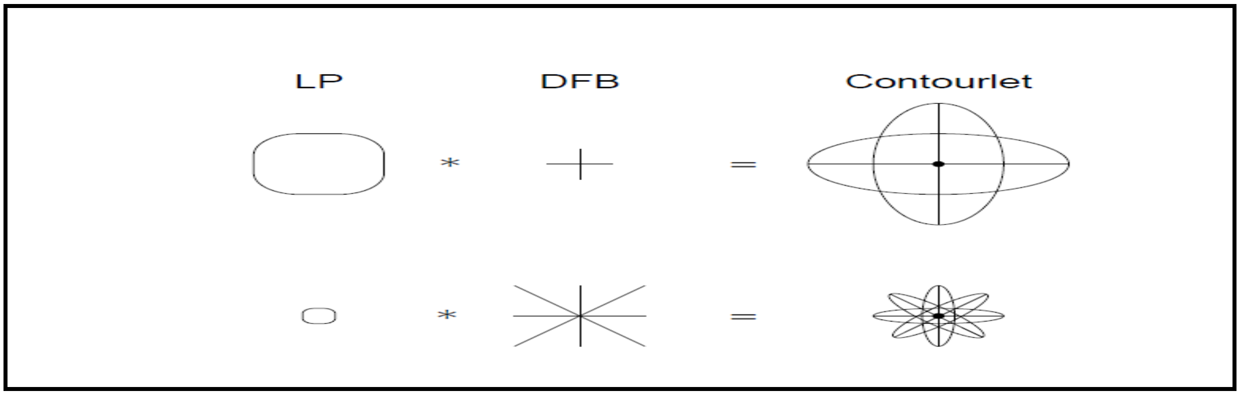


Figure 3.25: LP Size Decreases While the Number of Directions Doubles (Ardabili, Maghooli and Fatemizadeh, 2011)

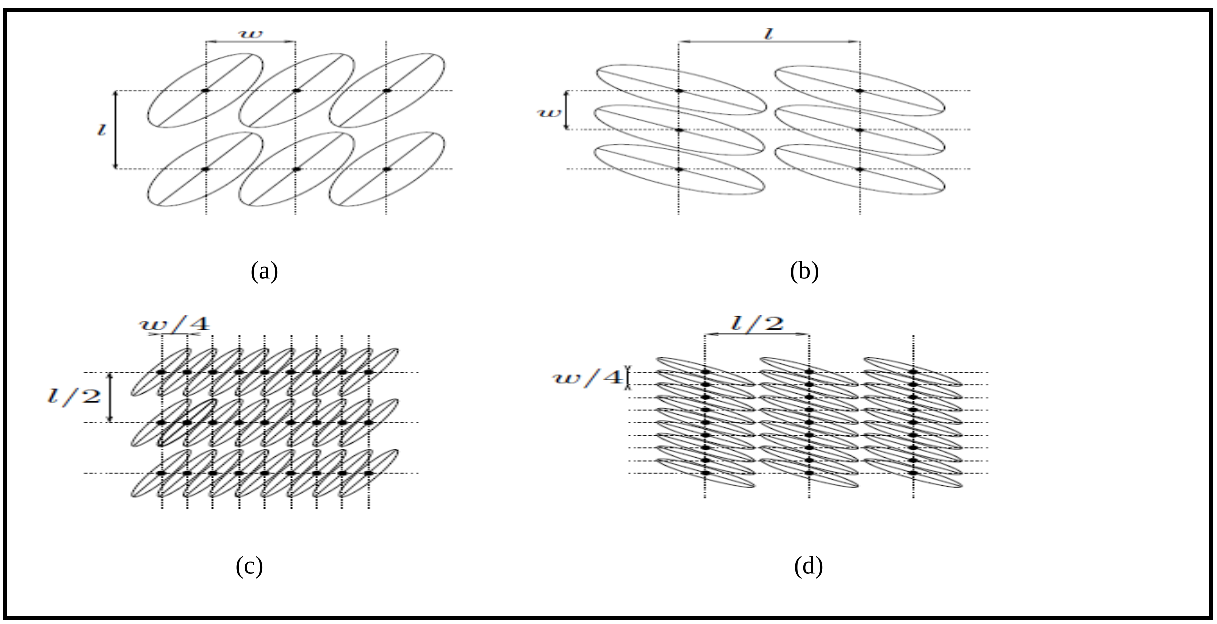


Figure 3.26: Shows PDFB, (a) Coars Horizontal Direction, (b) Coarse Vertical Direction, (c) Fine Horizontal Direction, (d) Fine Vertical Direction (Meskine, Mezouar and Taleb, 2010)

3.16 Summary

In previous articles the theoretical side was covered which was applied in the proposed algorithm of the thesis. The first part shows the difference between the data hiding and steganography plus the theoretical part which explain the details of the most applied techniques with both advantage and disadvantage for both of them. The last part of the above gives an explanation of the contourlet transformation properties and the most important side on it. The multi scale with directional filters and multi resolution, also decomposing and reconstruction are covered.

CHAPTER 4

APPLICATION AND IMPLEMENTATION

4.1 Overview

In This chapter several algorithms were suggested to embed the watermark depending on the contourlet transformations for the cover-image. The general and the special characteristics of the coefficients were studied and the depths of concepts these coefficients include were therefore identified. It was clear that the energy the image has when it is analyzed into coefficients using the contourlet transformations are distributed in accordance with the frequencies of the original image. Because processing the image in order to transform it to the contourlet transformation demands analyzing the image using a group of directional filters, which in turn depend on the frequency values in certain directions; therefore, all these frequencies were studied and it was clear that it is possible to use them to select the coefficients that have a high or low effect on the quality of the image watermark and the cover-image when we conduct the recovery.

4.2 First Algorithm Embedding Using the Contourlet Transformation Coefficient

In this algorithm the secret data was embedded in the cover to get a new cover-image which is very close to the original image that includes the watermark in a concealed way depending on the coefficient transformation of the different levels into one-dimensional matrices.

The general steps of the algorithm are as mentioned bellow:

- Inputting the cover-image and the watermark.
- Inputting the number of the directions depending on the size of the cover-image and the secret image.
- Calculating the Contourlet Coefficients for the cover-image and the secret image.
- Embedding of the secret data in the cover-image to obtain the watermark image.

The following are some of the detailed steps that are necessary to execute the algorithm:

4.2.1 The First Step: Image Input

Obtaining the cover-image and the secret image from stored files.

4.2.2 The Second Step: The Primary Processing

In this step a type of preparatory processing is conducted for the image such as the secret image sizing process and the transformation of the colored image into a grey level.

4.2.3 Third Step: Analyzing the Image Using the Contourlet Transformations

In this step the cover-image with the size N×N and the watermark with the size M×M using the contourlet transformations and that is performed by applying the Laplacian Pyramid filter to the original image and applying the directional filter bands DFB to the band pass that resulted from the previous filter and consequently we obtain the contourlet transformation coefficients as shown in Figure 4.1 bellow:

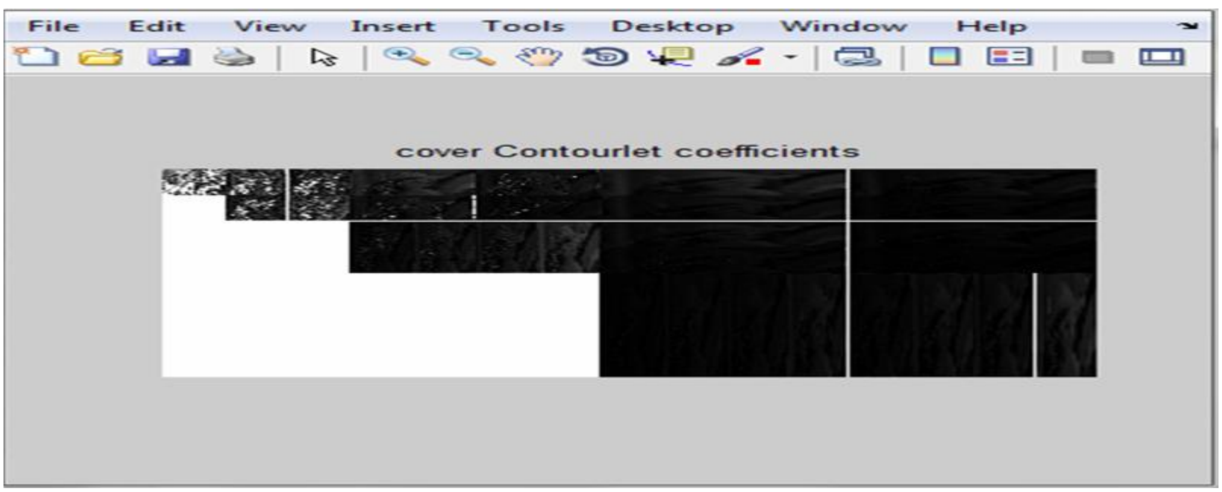


Figure 4.1: The Coefficients of the Contourlet Transformations of the Cover-Image

4.2.4 The Fourth Step: Preparing the Watermark and the Cover-Image

After processing the watermark and the cover-image using the contourlet transformations, the matrices of the different levels are converted into one dimensional matrices which consist of a number of cells and each cell represents a two-dimensional which consists of a number of rows and columns N×M and these operations are applied in two stages which are: the embedding and the extraction transforming the cell of the contourlet transformation of the watermark into a 1×N matrix. This process is executed in the embedding process only and all that is illustrated in Figure 4.2 and as follows:

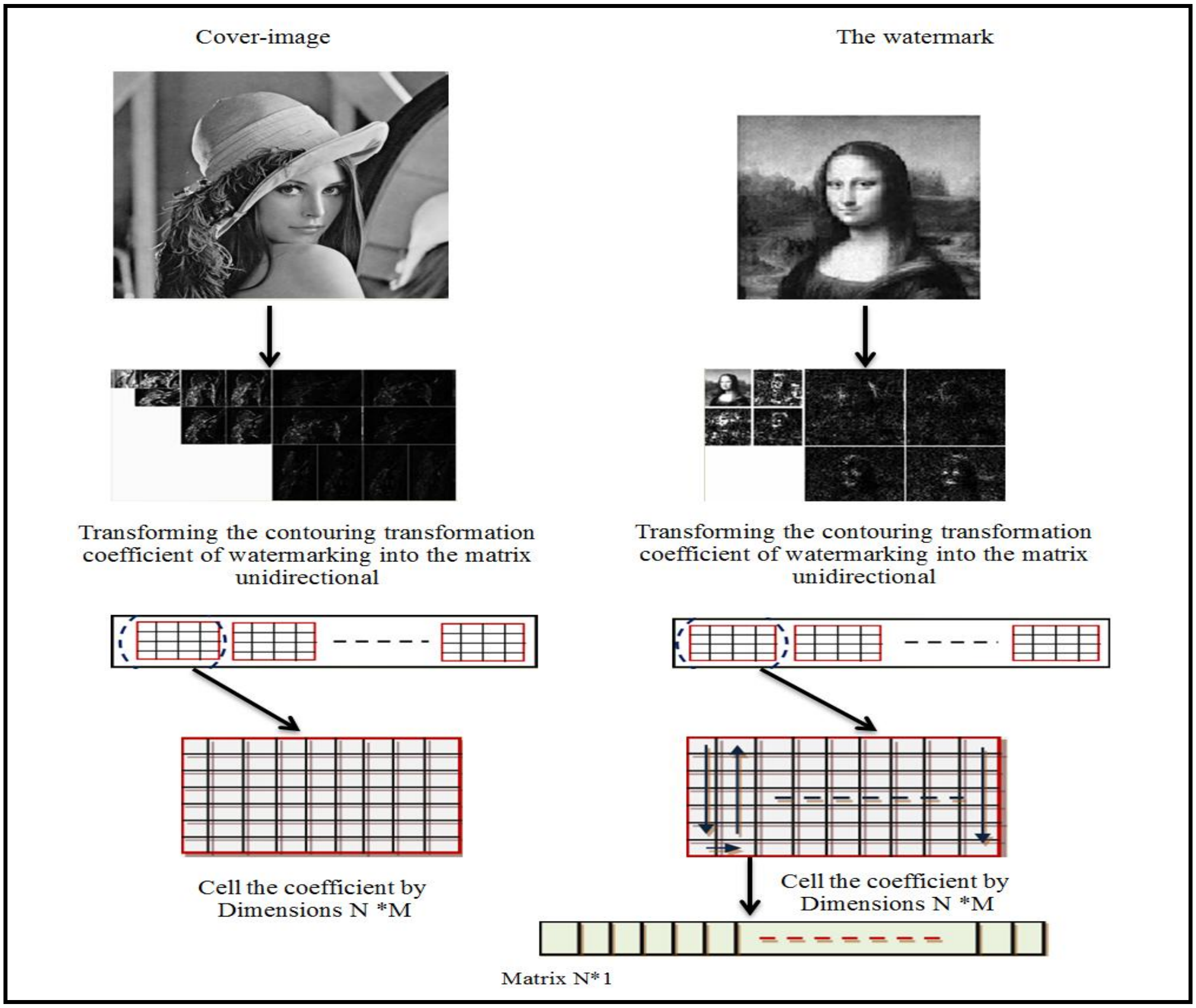


Figure 4.2: The Preparation of the Watermark and Cover-Image

4.2.5 The Fifth Step: Calculating the Dimensions of the Contourlet Transformations Coefficients

Calculating the dimensions of the contourlet transformations coefficients of the cover-image that has been chosen for the watermark embedding to identify the maximum level of pixels that can be included and that is done through Equation 4.1.

$$no.of\ Pixels = (Coeff.H. \times Coeff.W.)/K \tag{4.1}$$

Where:

Coeff. H and Coeff. W. are the dimensions of the cell of the contourlet transformation of the cover-image.

K = The secret key that is used in the embedding and the extraction processes. And the use of the secret key known by the two sides of transmission adds another level of privacy.

4.2.6 The Sixth Step: The Embedd Process

The embedding algorithm involves two stages:

A- The stage of embedding the watermark

- *The inputs:* The contourlet transformation coefficient of the cover-image, the contourlet transformation coefficient of the watermark and the secret key.
- *The outputs:* The watermark image.

The idea of embedding the watermark in this algorithm depends on updating the pixelvalue of the contourlet transformation coefficient of the cover-image in regard with the cell of the contourlet transformation coefficient of the watermark and then selecting the embedding location in the coefficient cell and that is done depending on the value of the secret key. The Equation 4.2 shows the embedding process given that the value (1000) was obtained as a result of experiments.

$$f'(i,j) = f(i,j) + \left(\frac{w(i,j)}{1000}\right) \tag{4.2}$$

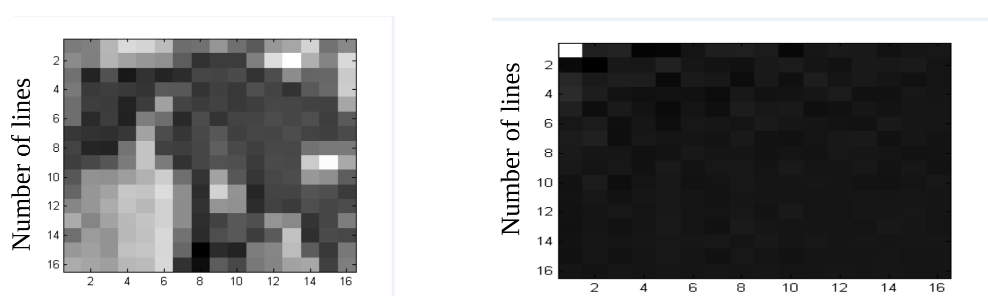
Where:

f (i,j) = The pixel of the cell of the contourlet transformation of the cover-image in the location i, j.

W (i,j) = The pixel of the cell of the contourlet transformation of the watermark in location i, j.

W (i, j) = The new pixel after the embedd process in the location i, j.

The low pass contourlet transformation coefficient of the secret image embedding the most important optical parts of the image. When embedding is performed, it will lead to the emergence of an evident distortion in the quality of the cover-image. To overcome this problem the low pass coefficient of the contourlet transformation in the secret image was dealt with using the DCT that provide a pressure for the energy as a band in a top left region of the image low pass region. The aforementioned can be seen in the Figure 4.3 and Figure 4.4 and Figure 4.5.



(a) Column number before applying DCT

(b) Column number after applying DCT

Figure 4.3: The Effect of Applying the Discrete Cosin Transforms DCT on the Image of the Contourlet Transformation Low Pass Coefficient of the Secret Image

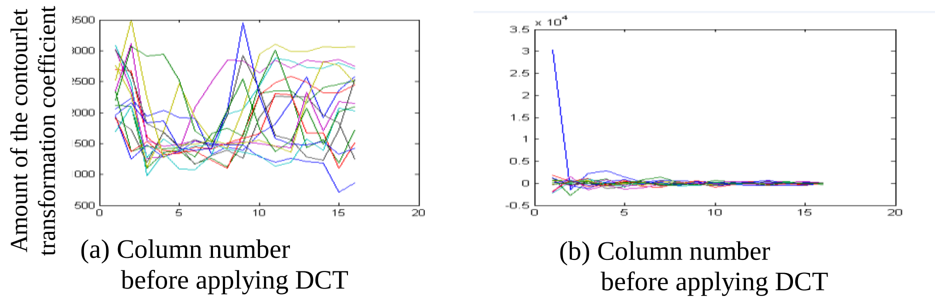


Figure 4.4: A Drawing Shows the Effect of the Discrete Cosin Transforms After Applying them to Contourlet Transformation Low Pass Coefficient of the Secret Image

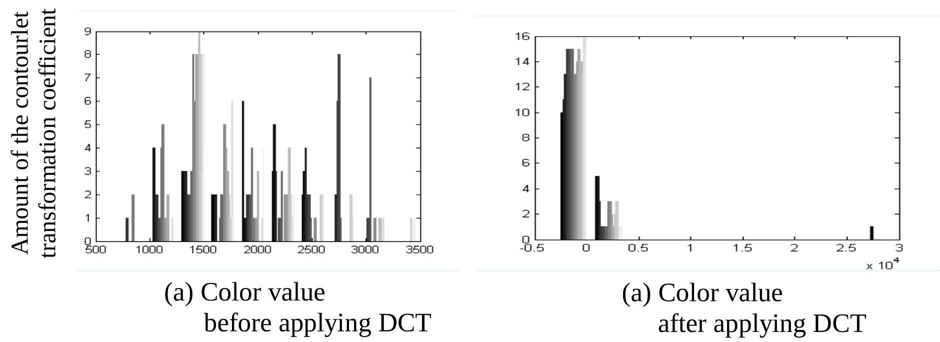


Figure 4.5: Studying the Iterative Scalar which Shows the Effect of Applying the Discrete Cosin Transforms DCT on the Image of the Contourlet Transformation Low Pass Coefficient of the Secret Image

The discrete cosin transforms are linear equations with simple mathematical complications which use the cosin function in analyzing the digital images as a summation of cosin in different capacities and passes as shown in Equation 4.3 bellow (Ahire and Kshirsagar, 2011).

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_m \cos\frac{\pi(2m+1)}{2M} \cos\frac{\pi(2n+1)}{2N} \qquad (4.3)$$

Where:

$$\alpha q = \frac{\frac{1}{\sqrt{N}}}{\sqrt{\frac{2}{N}}} \qquad 1 \leq q \ \leq M-1, \quad q = 0$$

$$\alpha \rho = \frac{1/\sqrt{M}}{\sqrt{2/M}} \qquad 1 \leq \rho \ \leq M-1, \ \ q = 0$$

M, N = The dimensions of the image. (a) = The input image.

B_{pq} = Stands for the coefficients of the discrete cosin transforms of the image.

Obtained (a). The reconstruction of the image through the application of the (Inverse Discrete Cosine Transform: IDCT) in the stage of extracting the secret image at the recipient side according to the Equation 4.4 (Ahire and Kshirsagar, 2011).

$$\text{A}mn = \alpha q \ \alpha \rho \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \ \rho q \ \ cos\frac{\pi(2m+1)\rho}{2M} \ \ cos\frac{\pi(2n+1)\rho}{2N}$$

$$m = 0{,}1, \quad M-1 \qquad\qquad\qquad n = 0{,}1, \quad N-1$$

After finishing the embedding of the contourlet transform of the secret image in the selected contourlet transforms of the cover-image the construction of the image is repeated an inverse contourlet transformations to obtain the watermark image. Figure 4.6 shows the embedding algorithm suggested.

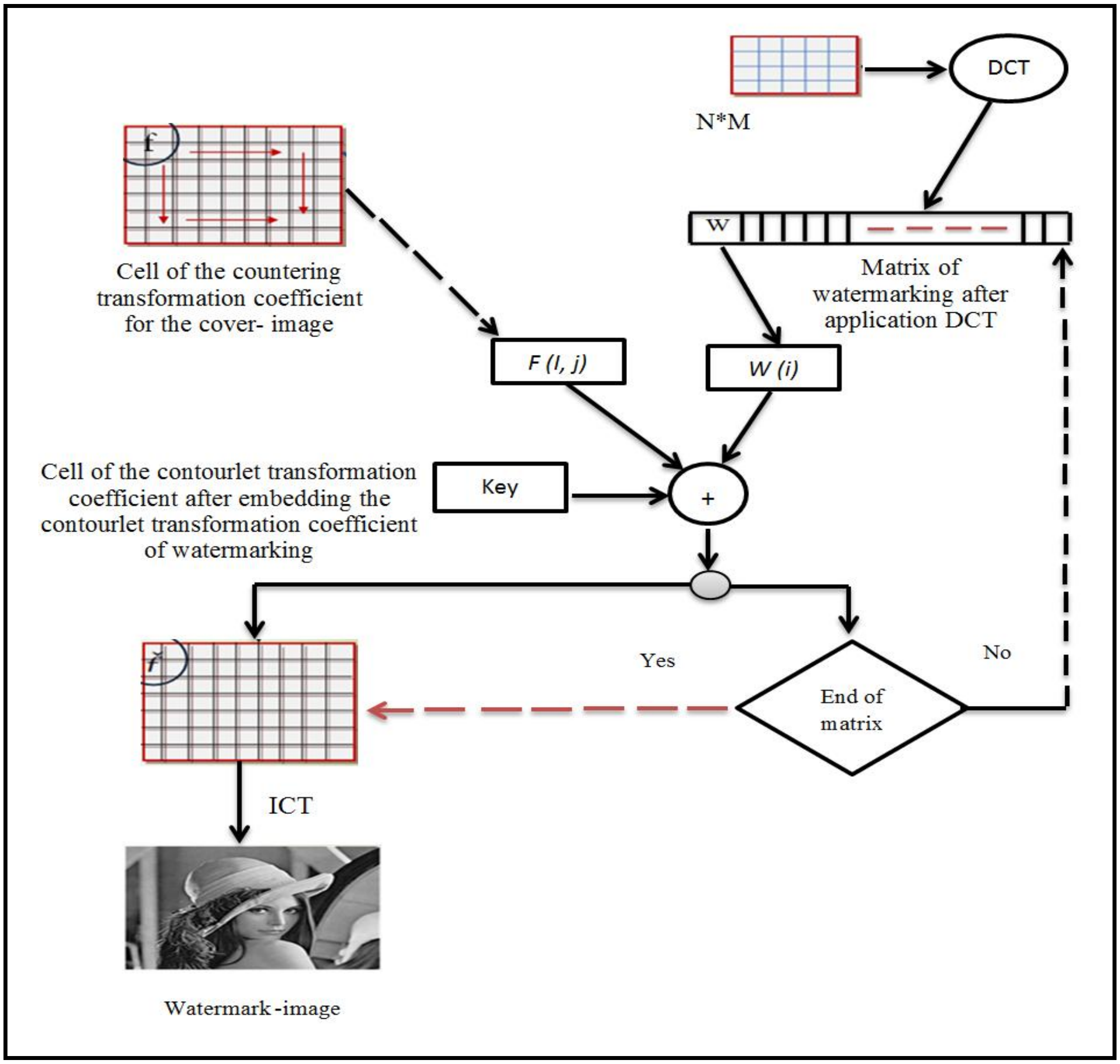


Figure 4.6: The Stage of Embedding Watermark

B- Stage of Encoding and Embedding of the Secret Key

In this stage, the secret key was included to identify the location of secret data embedding and extraction in the cells of the contourlet transforms of the cover-image and the image of the watermark. The secret key is included in the pixel of the location (1,1) for the watermark image. The encryption of the secret key and then including it in the digital watermark image is one of the suggested applications in the algorithm which provide secure transfer for the secret key between the two-transmission sides through the unsecure internet. The following are the steps followed in the encryption and embedding of the secret key.

- Encoding the secret key used in the embedd process according to the Equation 4.5.

$$Keyseq = key - 26 \tag{4.5}$$

Where:

Keyseq = The sequence of the English alphabetical letters = the secret key used in the stage of embedding and extraction.

- Obtaining the English alphabetical letter depending on the sequence resulting from the previous step.
- Obtaining the ASCII CODE of the relevant letter.
- Transforming the ASCII CODE into the correspondent binary form to get the one dimensional matrix 1×N and then determining the length of the resulting matrix.

After that the pixel of the watermark image is transformed into the binary form as a one dimensional matrix of the pixel with a length of (8) bits and a number of the bit binary cells are the most significant into the zero depending on the number of bit for the secret key after the encoding and them the binary form secret key is included as shown in Equation 4.6 as follows:

$$wk(MSB) = wk(MSB) OR \ key(index) \tag{4.6}$$

Where:

$Wk(MSB)$ = The binary cell of the pixel of the watermark in the location i, j.
$Wk(index)$ = The binary cell of the secret key in the location i, j.

Then the value of the pixel is transformed from the binary form into the correspondent decimal form. Figure 4.7 shows the pixels mentioned above for the encryption and inclusion of the secret key.

The steps above are repeated from the fourth step until the end of the stage of embedding to put each coefficient of the contourlet transforms of the secret image in the coefficients of the cover-image selected which in turn depends on the size of the watermark and the number of analysis levels for the contourlet transformation. The above mentioned stage is executed at the side of the sender.

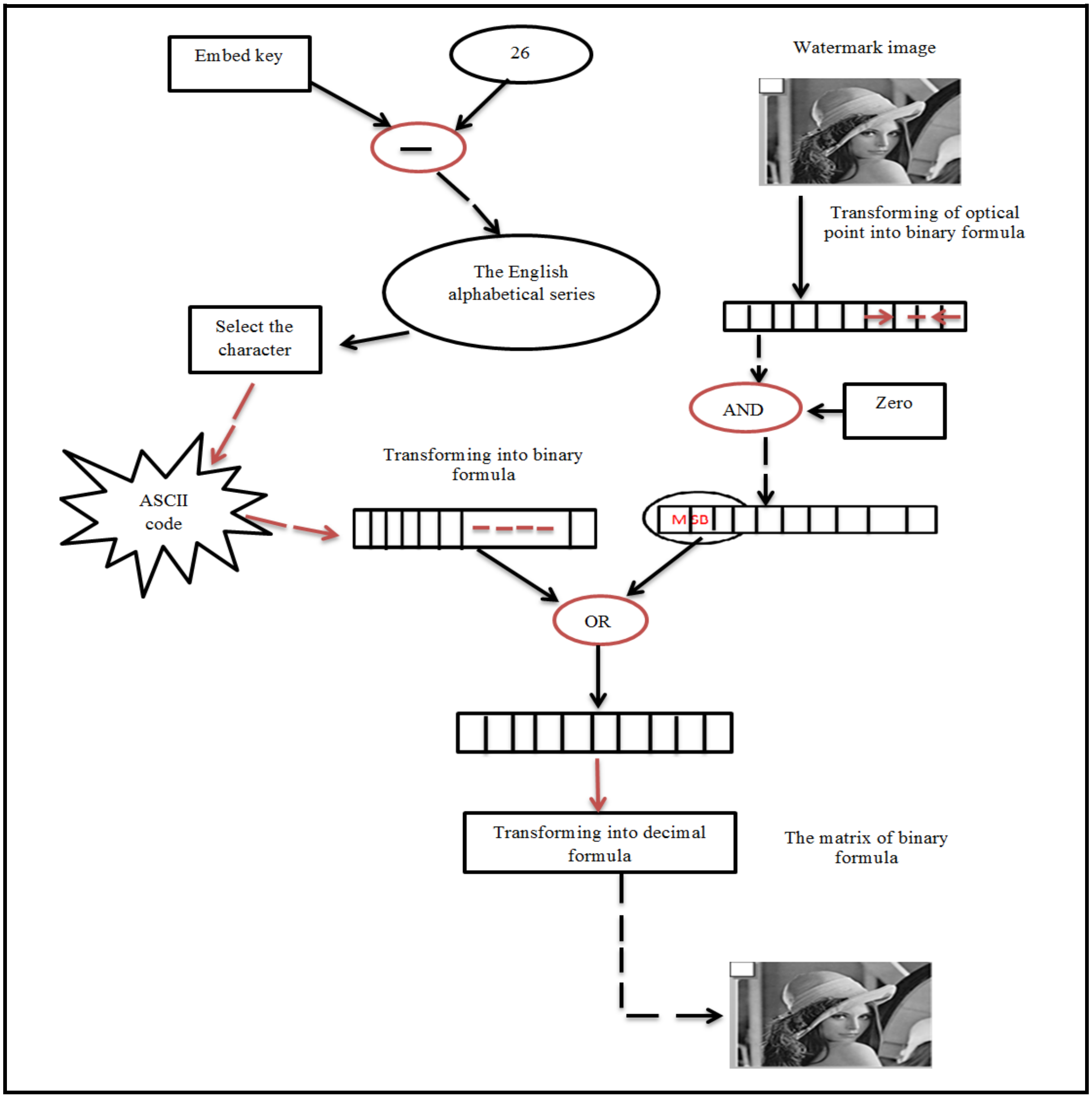


Figure 4.7: Encryption Phase and Embed the Secret Key

4.2.7 The Seventh Step Extraction of the Watermark

This step is executed at the recipient side to recover the secret data sent from the watermark image.

The extraction algorithm involves two stages, they are:

- *The inputs:* The watermark image, the cover-image.
- *The outputs:* The watermark image, the recovered cover-image.

A- Extraction and Decoding the Secret Key

The process of extracting the secret key doesn't require the availability of the cover-image as the extraction algorithm is of the type (Blind Watermark). After getting the pixel in the location (1,1), the secret key is extracted by means of extraction process which is inverse to the embedding process. While decoding the secret key is conducted by means of mathematical operations which are considered to be inverse and opposite of the encryption process.

B- Watermark Extraction

The cover-image of the original is required to detect and extract the watermark, i.e. The algorithm of extracting the secret data of the type Non-blind watermark in addition to the secret key used in the embedding process. The recipient analyzes the image of the watermark and the original cover-image using the contourlet transforms with the same levels in the embedding process to get the directional sub bands. The contourlet transform coefficients of the watermark image are subtracted from the coefficients of the contourlet transforms of the cover-image depending on the value of the secret key used between the two sides of transmission to extract the watermark. All this can be shown in the Equation 4.7.

$$w`(i,j) = (f(i,j) - f`(i,j)) \times 1000 \tag{4.7}$$

Where:

$w`(i,j)$= The pixel of the cell of the contourlet transformation coefficient of the extracted watermark in the location i, j.

$f`(i,j)$= The pixel of the cell of the contourlet transformation of the watermark image in the location i, j.

$f(i,j)$= The pixel of the cell of the contourlet transformation of the cover-image in The location i, j.

To get the cell of the contourlet transform of the two-dimensional watermark extracted, the extracted cell is rearranged which is stored in one-dimensional matrix depending on the transformation method that was mentioned in the stage of embedded and then reassembling the extracted of the contourlet transform of the watermark.

The steps (a) and (b) for extracting the coefficients of the watermark embedding are repeated depending on the number of the levels of the contourlet analysis of the watermark embedded. The extracted image the watermark is rearranged after gathering the coefficients extracted and reentering them into the LP and the DFB, i.e. An inverse application of the contourlet Transformations to get the watermark. Figure 4.8 is a diagram that shows the stage of the watermark extraction. Recently, most of the researches focused on recovering the embedded watermark only. But the research algorithm, compared to the traditional watermark algorithms helps to recover the cover-image after the extraction process, and this type of algorithms is very useful in the military applications, medical records and the management of the digital multimedia information.

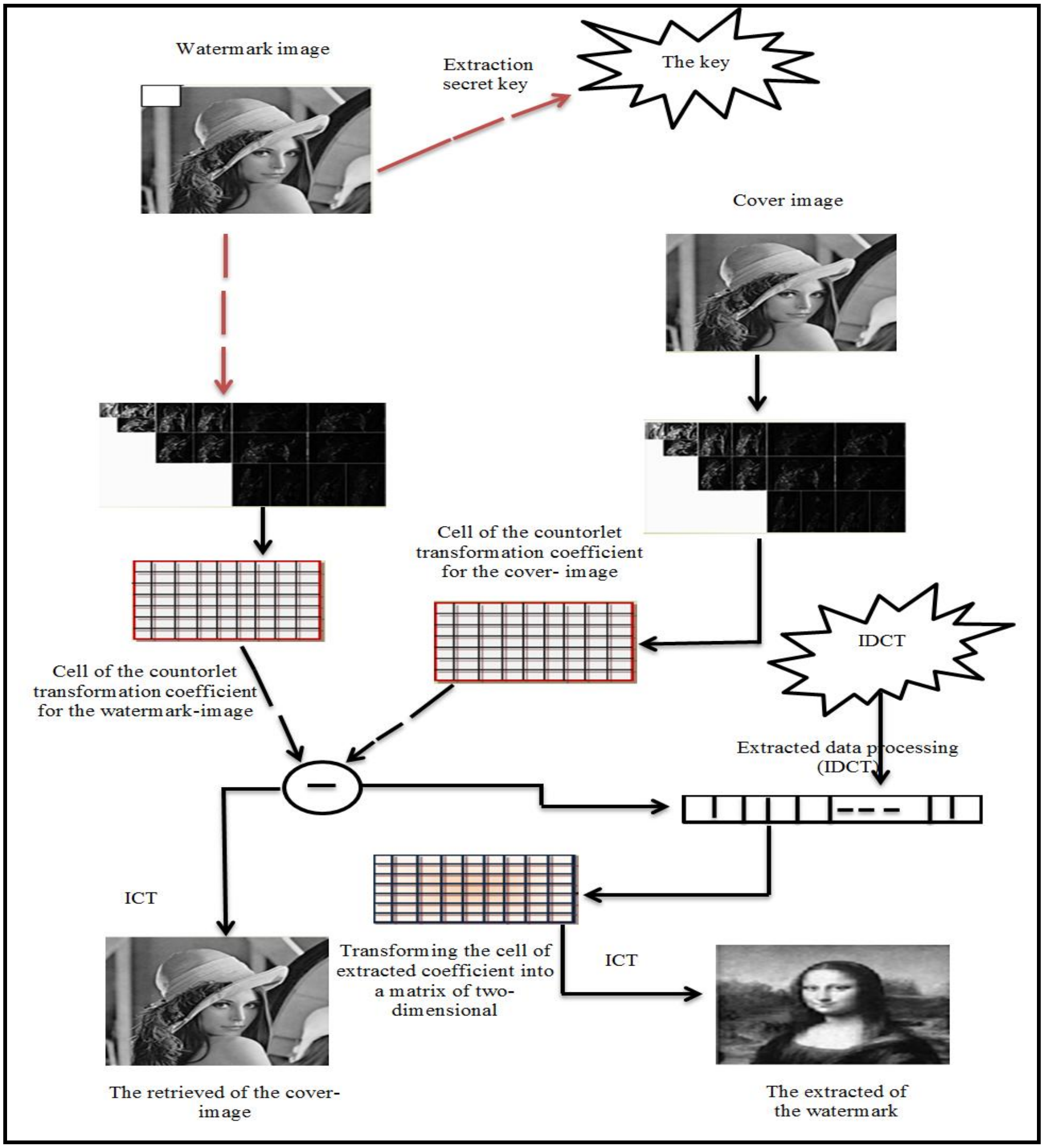


Figure 4.8: The Watermark Extracted Phase

4.3 Second Algorithm Embedding Using Energy

In this algorithm, the watermark was planted inside the cover-image depending on the energy of the contourlet transforms of the cover-image by means of reaching the coefficients with the less energy to be depended in embedding the watermark. All the contourlet transforms of the image was studied in terms of the amount of energy they possess and it was shown that these coefficients vary considerably in terms of the energy they have. In addition to they are related to the frequencies and their directions. We can say that they are tightly related to the edges of the image at the relevant frequency and the relevant direction. The calculation of the energy for the contourlet transforms of the cover-image can be reached by using Equation 4.8 and as follows (Mohan and Kumar, 2008).

$$E = \sum i \sum j \; |S(i,j)|\hat{\ }2 \tag{4.8}$$

Where:

$$0 \leq i,j \leq N$$

The big value of the energy scale refers to the high energy levels of the contourlet transform, but the small value indicates the low levels of energy in that coefficient. Figure 4.9 shows the energy levels in the directional bands of the contourlet transforms for the different levels of Lena image which is used as a cover.

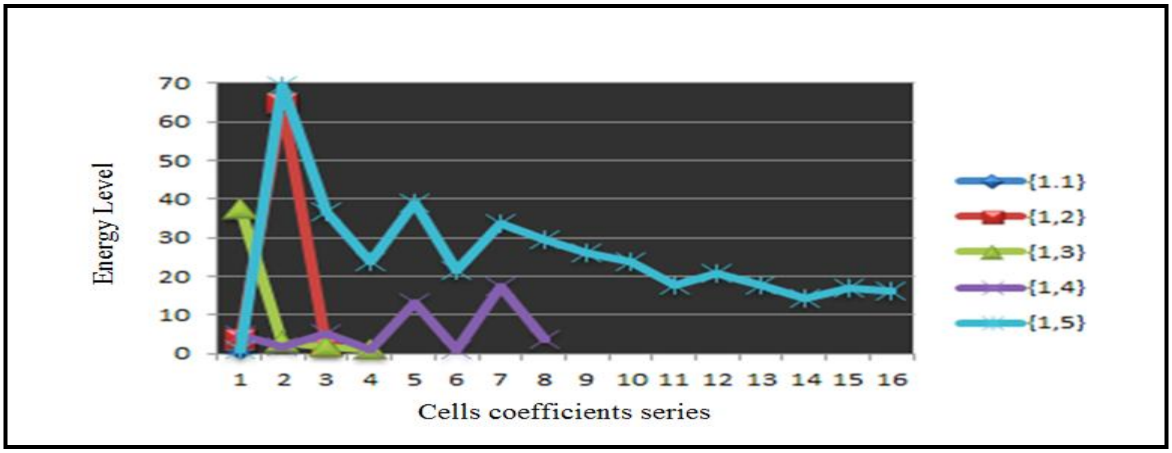


Figure 4.9: The Variance in the Energy Levels of the Contourlet Transforms Coefficients

In comparison with the first algorithm, it was necessary to process the low-pass contourlet transform of the watermark using the discrete cosin transformations as mentioned earlier, but by using the second algorithm, there has become no need to that processing.

A- Stage Embedding

In this algorithm, all the steps mentioned in the first algorithm are executed in the embedding of the secret data except for the contourlet transforms of the cover-image depending on decrease in the energy levels in them, and the update process of those coefficients are thus become according to Equation 4.9 and as follows:

$$0 \leq i,j \leq N$$

$$f`(i,j) = f(i,j) + (w(i,j) \times th) \tag{4.9}$$

Where:

th = The control coefficient of increasing the solidness of the embedding process.

When embedding the secret data in the high energy levels coefficient,the result was an evident distortion in the cover-image and this decreases the solidness of the embedding algorithm. For this reason the directional bands with low energy, compared to the directional bands of other levels of coefficient were selected in order to include a big amount of data and without causing any effect on the quality of the cover-image.

B- Extraction Process

By following the sequence of the steps in the first algorithm, the watermark is extracted depending on Equation 4.10.

$$w`(i,j) = (f(i,j) - f`(i,j)/th \tag{4.10}$$

Later on this strategy was adopted in testing the coefficients of the contourlet transforms that increased the efficiency and the solidness of the algorithm for the embedding and extraction of the watermark.

4.4 Third Algorithm Embedding Depending on the Contourlet Transforms Coefficient and Energy

In this algorithm the idea of quadric division tree was applied to the cover-image to reach high accuracy and privacy in the embedding of the watermark. It was clear through the study of this idea that the extent of the division and its depth results in higher embedding especially when we conduct a quadric tree division on the cover and the message. Also, studies tackled the domain at which the tree division should be stopped for both the message and the cover. The following is a description of the steps of this algorithm to conduct the division for the cover-image and the watermark into four parts, as shown in Figure 4.10.

Figure 4.10: The Quadric Tree Division

This algorithm consists of two parts the general steps executed in both parts are as follow:

- Conducting the division process for both the cover-image and the watermark in the embedding and extraction processes.
- Calculating the energies of the directional bands of the contourlet transform coefficients for each parts of the cover-image and selecting these directional bands with low energy in comparison with the energy in the other analysis levels of that part of the cover-image to embedd the watermark.

4.4.1 Part One

After executing the general steps mentioned above, in the first part of the third algorithm we embed and extract the parts of the contourlet transform coefficients for a part of the watermark from the part of the cover-image, and as follows:

A-Stage Embedding

1. The third step which is previously mentioned within the first algorithm is applied embedding by the use of the contourlet transform coefficient in order to analyze both the part of the watermark and the cover-image.
2. Preparing the coefficients of the watermark part and the cover-image as in the fourth step shown earlier in the first algorithm embedding by means of using the contourlet transform coefficient and then the fifth step is executed as has been shown in order to calculate the dimensions of the contourlet transform coefficients of the cover-image.
3. The contourlet transform coefficient of the watermark is planted in the contourlet transform coefficients of the corresponding part of the contourlet transform coefficient by means of updating the value of the pixel according to the Equation 4.9 then the construction is repeated using the inverse contourlet transform in order to obtain the part of the watermark image.
4. Encryption and embedding of the secret key which is used in the embedding process and which is agreed upon by the two parties of transmission as shown in the first algorithm embedding by using the contourlet transform coefficient.
5. Reassembling the parts of the watermark image.

The steps 2, 3, 4 and 5 are repeated depending on the number of the contourlet transform coefficient levels of the part of the watermark, while the remaining parts are included by following the steps 1, 2, 3, 4 and 5 and the execution of this stage will be at the sender side.

B- Stage Extraction Watermark

At the side of the recipient, the secret data is extracted from the parts of the watermark according to the following steps:

1- Preparing the cover-image by dividing it into four parts.
2- Extracting the secret key and decoding it according to the steps mentioned in the first algorithm.
3- Treating the part of the cover-image and the watermark using the contourlet transforms with same number of the analysis levels in the embedding stage.
4- Extracting the contourlet transform coefficient of the secret image from the contourlet transform coefficients of the part of the watermark using the Equation 4.10 the steps 2, 3 and 4 are repeated according to the number of the coefficients embedded.
5- Gathering the contourlet transform coefficients which are extracted for a part of the watermark, and from the application of the inverse contourlet transform that part is reconstructed.
6- The steps 2, 3, 4 and 5 above are repeated to extract the remaining parts of the watermark image.
7- The next step is the step of gathering the four extracted parts of the watermark in order to obtain the full watermark, in addition to reassembling the recovered cover-image.

4.4.2 Part Two

The part of the watermark is embedded completely in part of the cover-image after applying the general steps above. The embedding process is executed in the side of the sender according to the following steps that are mentioned bellow:

1- The third step of analysis which is shown in the first algorithm is applied in both the part of the watermark and the cover-image.
2- The fourth steps which were shown previously in the first algorithm is applied embedding depending on the contourlet transform coefficient to the coefficients of the contourlet transform for the part of the watermark and the fifth step is applied to the coefficients of the part of the cover-image.

3- All the coefficients of the contour transforms for the part of the watermark are planted in the coefficients of the contourlet transforms for the correspondent part of the cover-image by means of by means of updating the pixel according to Equation 4.9.
4- After finishing the embedding of all the coefficient of the part of the watermark, the part of the image is reassembled by applying the inverse of the contourlet transforms to get the part of the watermark image.
5- The process of the encryption and embedding of the secret key is executed in accordance with the steps mentioned earlier in the first algorithm.
6- The steps 1, 2, 3 and 4 are repeated to embed the remaining of the parts in the correspondent parts of the cover-image. After that, those parts are reassembled in order to obtain the watermark image in its complete shape.

The extraction of the secret data from the watermark image at the recipient side will be according the following steps:

1- The extraction and the decoding of the secret key are used according to the steps in the first algorithm.
2- Processing each part of the cover-image and part of the watermark using the contourlet transforms.
3- Extracting all the coefficients of the embedded watermark part from the coefficients of the part of the watermark image through the following Equation 4.10.
4- Reassembling the part of the watermark extracted and part of the cover-image after taking out the secret data by applying the inverse of the contourlet transforms.
5- The steps 2, 3 and 4 above are repeated to extract the remaining parts of the watermark image.
6- After assembling the extracted four parts of the watermark to get the watermark in its complete shape. In a subsequent stage, every part of the four parts of the cover-image is divided into four parts with the dimensions of (256×256) in which the four parts of the watermark were embedded, as shown in the following Figure 4.11.

Figure 4.11: The Quadric Tree Division of the Cover-Image

4.5 Scales of the Watermark Algorithms Efficiency

To evaluate the efficient performance of the algorithm for the purpose of embedding of the watermark, it is necessary to study some scales that are relevant to the performance and the efficiency of the algorithm. There are many scales and in this study the focus was on a group of scales that proved to be efficient in determining the strength of the algorithms applied depending on a huge number of the published researches. The following are the scales that were relied upon in this study.

- **Peak-Signal-to-Noise-Ratio (PSNR)**

  It is considered to be a standard means to measure the amount of distortion in the cover-image the quality of the image after applying the hiding algorithm to embed the watermark. This scale expresses the Un detectability (Tamilarasi and Palanisamy, 2011). As the high ratio of the scale is a sign of the high quality of the image and it is called the measurement unit PSNR with the decibel (dB) (Aliwa, El-Tobely and Fahmy, 2010). And it is calculated by the Equation 4.11 (Tamilarasi and Palanisamy, 2011).

$$PSNR = 10 \log_{10} \frac{MAX^2}{\frac{1}{w \times h} \sum_{i=1}^{w} \sum_{j=1}^{h} \left(o(i,j) - c(i,j) \right)^2} \tag{4.11}$$

Where:

w, h = Represent the dimensions of the image.

$o(i,j)$ = Represents the original cover-image or the watermark.

$c(i,j)$= Represent the watermark image or the extracted watermark.

MAX^2 = Represents the maximum value in the original image the cover-image or the watermark.

- **Scale of the Normalized Correlation Coefficient (NC)**

  This scale is used to measure the degree of similarity between the watermark image extracted and the original watermark and its value ranges between zero and one (0 – 1). The closer this value is to (1) the more similar the extracted watermark becomes to the original one. It can be calculated from the Equation 4.12 (Narasimhulu and Prasad, 2011).

$$Ncc = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \bar{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \bar{B})^2\right)}} \tag{4.12}$$

Where:

n,m= Represent the dimensions of the image.

A = Represents the original cover-image or the watermark.

B = Represents the watermark image or the watermark extracted.

$\bar{A}$ = The mean of the original image values.

$\bar{B}$ = The mean of the recovered images values (MATLAB Version R2011a).

- **Mean Square Error (MSE)**

The measure of the accumulative square error between the image resulted from the embedding and the extraction processes and the original photo and the least value of the measure means the least error in the recovered image, as shown in Equation 4.13 (Aliwa and El-Tobely, 2010).

$$\mathrm{MSE} = \frac{1}{\mathrm{M}\times\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{M}}\sum_{\mathrm{j}=1}^{\mathrm{N}}\left(\mathrm{f}(\mathrm{i},\mathrm{j})\text{-}\check{\mathrm{f}}(\mathrm{i},\mathrm{j})\right)^2 \qquad (4.13)$$

Given that:

M,N = Represent the dimensions of the image.

$f(i,j)$= Represents the original cover-image or the watermark.

$\check{f}(i,j)$ = Represents the watermark image or the watermark extracted.

- **Signal to Noise Ratio SNR**

A scale used in the fields of sciences and engineering and it refers that the noise in the image is due to the inclusions and recovery processes. It is defined as the ratio of the signal (Wu et al., 2012). Strength the original image to the noise strength the embedding errors between the original and the recovered image that resulted in the distortion of the image. The unit for measuring the SNR is the decibel (dB). The high percentage of the scale means the less errors of embedding as in Equation 4.14 (Aliwa et al., 2010).

$$SNR = 10\log_{10}\left\{\frac{\sum_{i=1}^{M}\sum_{j=1}^{N} f(i,j)^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(f(i,j)-\check{f}(i,j)\right)^2}\right\} \qquad (4.14)$$

Given that:

M,N = Represent the dimensions of the image.

$f(i,j)$= Represents the original cover-image or the watermark.

$\check{f}(i,j)$ = Represents the watermark image or the watermark extracted.

Complete software were constructed depending on the graphical user interface using matlab language as shown in Appendix A.

4.6 Summary

In the preceding paragraphs it has been reviewing the proposed algorithms that are adopted in the research, including the three different algorithms possibilities of hiding data within image coefficients which are analyzed by using a contourlet transforms. In the first algorithm, confidential data watermark has been contributed and is commensurate with the volume of transactions and transactions of the image watermark, which was dismantled complemented each contourlet transforms.

In addition to what has been mentioned also been addressed in a simple way to methods of evaluating the performance of the software and the factors that depend for calculating the amount of similarity and divergence plus the amount of correlation between the cover-image original and secret as well as the watermark before the landfill and after retrieval. Support algorithms have been applied to the example in order to give a clear picture of the steps to implement the software, which was built during the work.

CHAPTER 5

RESULTS AND DISCUSSION

The practical application using Matlab (R2011a) program for the research algorithms in chapter four in order to embedding and extracting the watermark in the cover-image using the contourlet transformation coefficients resulted in good results in terms of the embedding capacity and the good quality of watermark image and extracting the secret embedded data with the highest accuracy resolution.

The contourlet transformations was used to embed the watermark because of its ability to include the curves of the smooth edges in the images, as the optical system of the human is less sensitive the edges and in order to enhance the incapability of discovering the secret data, the grey watermark was embedded in the directional bands especially the ones with high frequencies to the cover-image. Many experiments were conducted on the research algorithms for different types of watermark grey images and the cover-image with different sizes.

To verify the strength of the watermark embedding in the cover-image before and after embedding and showing the strength level of hiding the watermark that can hardly be predicted through studying its repetitive scale.

5.1 Results of the First Algorithm Embedding Using the Contourlet Transformation Coefficient

Figure 5.1a shows a model of embedding and extracting “Monalliza.jpg” with dimensions of (128×128) in the cover-image “Lena.jpg” with dimensions of (512×512) in addition to the recovered cover-image. Figure 5.1b shows a study using the repetitive scale that shows the approximation level between the cover-image “Lena.jpg” with dimensions of (512×512) and the image of the watermark when embedding the watermark “Monalliza.jpg” with dimensions of (128×128).

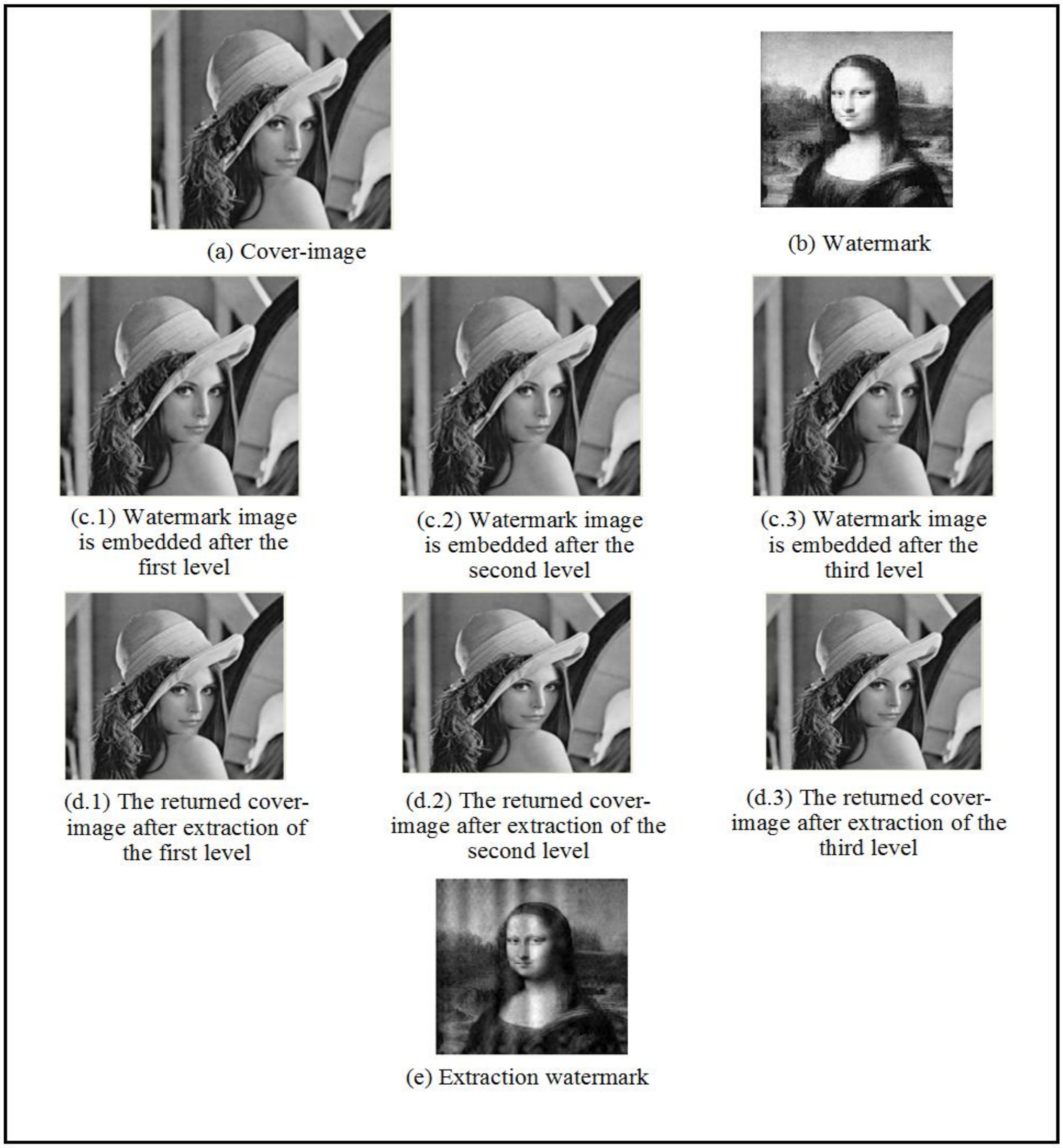


Figure 5.1a: A Model for Embedding and Extracting the Secret Image (Monalliza.jpg) in the Cover-Image (Lena.jpg)

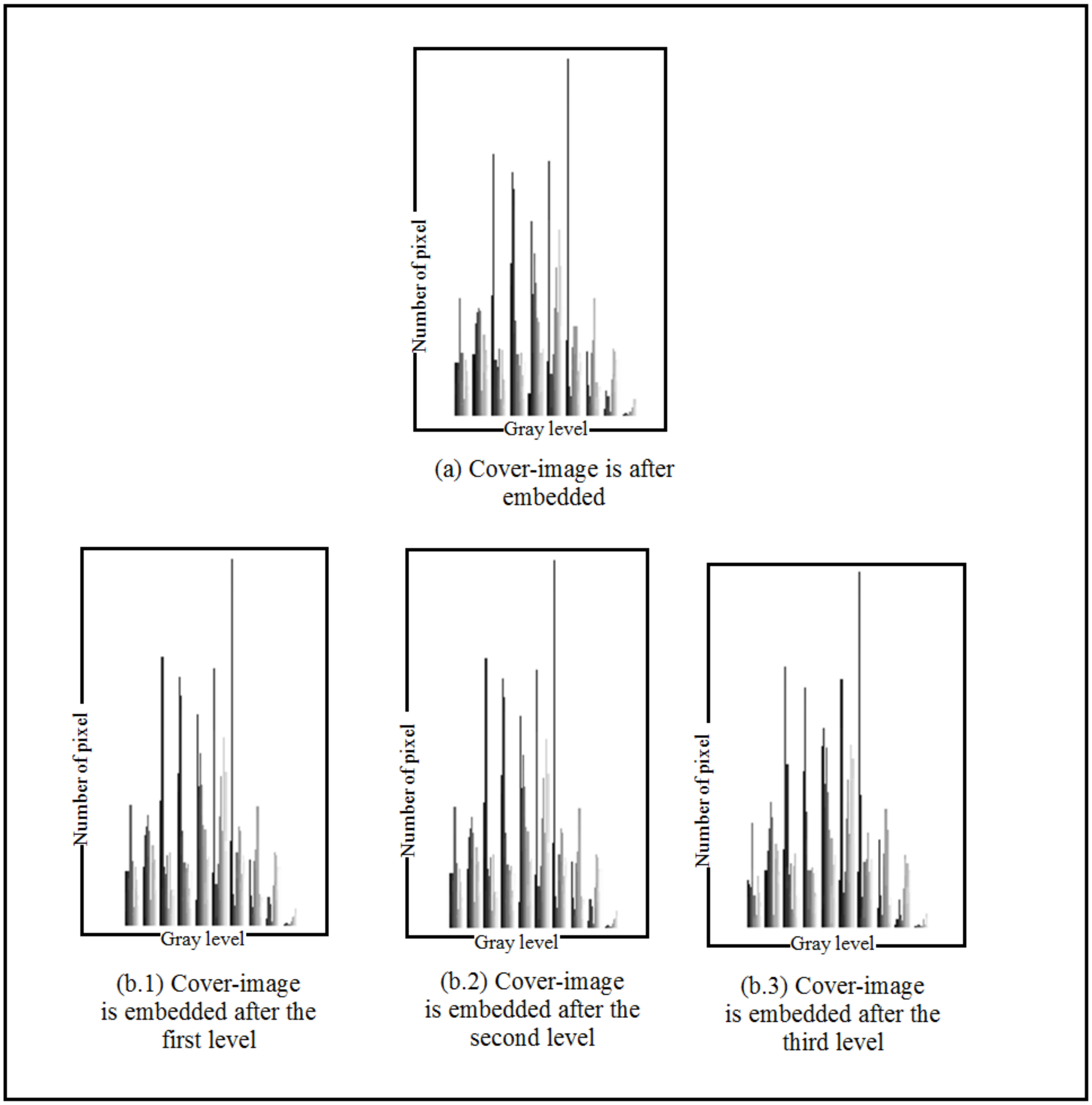


Figure 5.1b: A Model for Studying the Repetitive Scale for the Cover-Image and the Watermark-Image

From Table 5.1a the results of embedding the watermark (Monalliza.jpg) with dimensions of (128×128) in a way in which distinguishing visibly between the watermark image and the cover-image is not possible. But the Table 5.1b manifests the results of extracting the watermark from the grey cover-image while Table 5.1c shows the results of recovering the cover-image. We should indicate that the biggest size of the secret image was (256×256) which doesn't correspond with the size of the cover.

Table 5.1a: The Results of Embedding the Watermark in the Cover-Image

1Cover Image	**Cover image size**	**Watermark**	**Size of the watermark**	**Contourlet Coefficients**	**Parameter measure for efficiency performance of the watermark-image**			
					MSE	**PSNR**	**NC**	**SNR**
Lena	512×512	Monaliza	128 ×128	{1,1}	0.000674	79.844383	1.000000	73.781088
				{1,2}	6.145504	40.245228	0.999001	34.181933
				{1,3}	4.610501	41.493322	0.999251	35.430026
Lena	1024×1024	Monaliza	128 ×128	{1,1}	0.000168	84.318972	1.000000	78.515720
				{1,2}	0.000002	104.700274	1.000000	98.897022
				{1,3}	0.000001	105.980092	1.000000	100.176840

Table 5.1b: Results of Extracted the Watermark According to the First Algorithm

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of the extracted watermark**			
				MSE	**PSNR**	**NC**	**SNR**
Lena	512×512	Monaliza	128×128	864.450352	18.763403	0.954439	12.487304
Lena	1024×1024	Monaliza	128×128	936.597407	18.415274	0.947321	12.139176

Table 5.1c: Results of Cover-Image Retrieved According to the First Algorithm

Cover Image	Cover image size	Watermark	Size of the watermark	Contourlet coefficients	Parameter measure for efficiency performance of the retrieved cover-image			
					MSE	**PSNR**	**NC**	**SNR**
Lena	512×512	Monaliza	128 ×128	{1,1}	0.000032	93.045011	1.000000	86.981716
				{1,2}	0.424158	51.855525	0.999931	45.792230
				{1,3}	0.284282	53.593315	0.999954	47.530019
Lena	1024×1024	Monaliza	128 ×128	{1,1}	0.000008	97.510532	1.000000	91.707280
				{1,2}	0.000000	116.730062	1.000000	110.926810
				{1,3}	0.000000	118.138635	1.000000	112.335383

Table 5.1a shows the influence of the factors that have been adopted in measuring the efficiency of the algorithm employed with the size of the cover-image with the image of the watermark image to be embedded. It also shows the extent of the relationship for the aforementioned coefficients with the contourlet coefficient, as it was shown that the coefficients with high frequencies are more sensitive than the low-frequency coefficients. Therefore, embedding is preferred within the medium frequencies of the coefficients. Tables 5.1b and 5.1c show that the first algorithm has an evident effect on the factors that have been used in the contourlet transformations and that the size of the cover-image has a clear effect on the efficiency of embedding and it is directly related to the image size and this is natural as there will be more space when increasing the cover-image size.

To embed the watermark, the fourth-level contourlet coefficient was used depending on identifying the dimensions of the levels coefficients that showed that the fifth level was the best to embed the watermark which consists of (16) directional bands and with dimensions that depend on the analysis level and the cover-image size, in addition to embedding in the coefficients of the other levels when needed, the number of the directional bands chosen in the embedding process depends on the secret data.

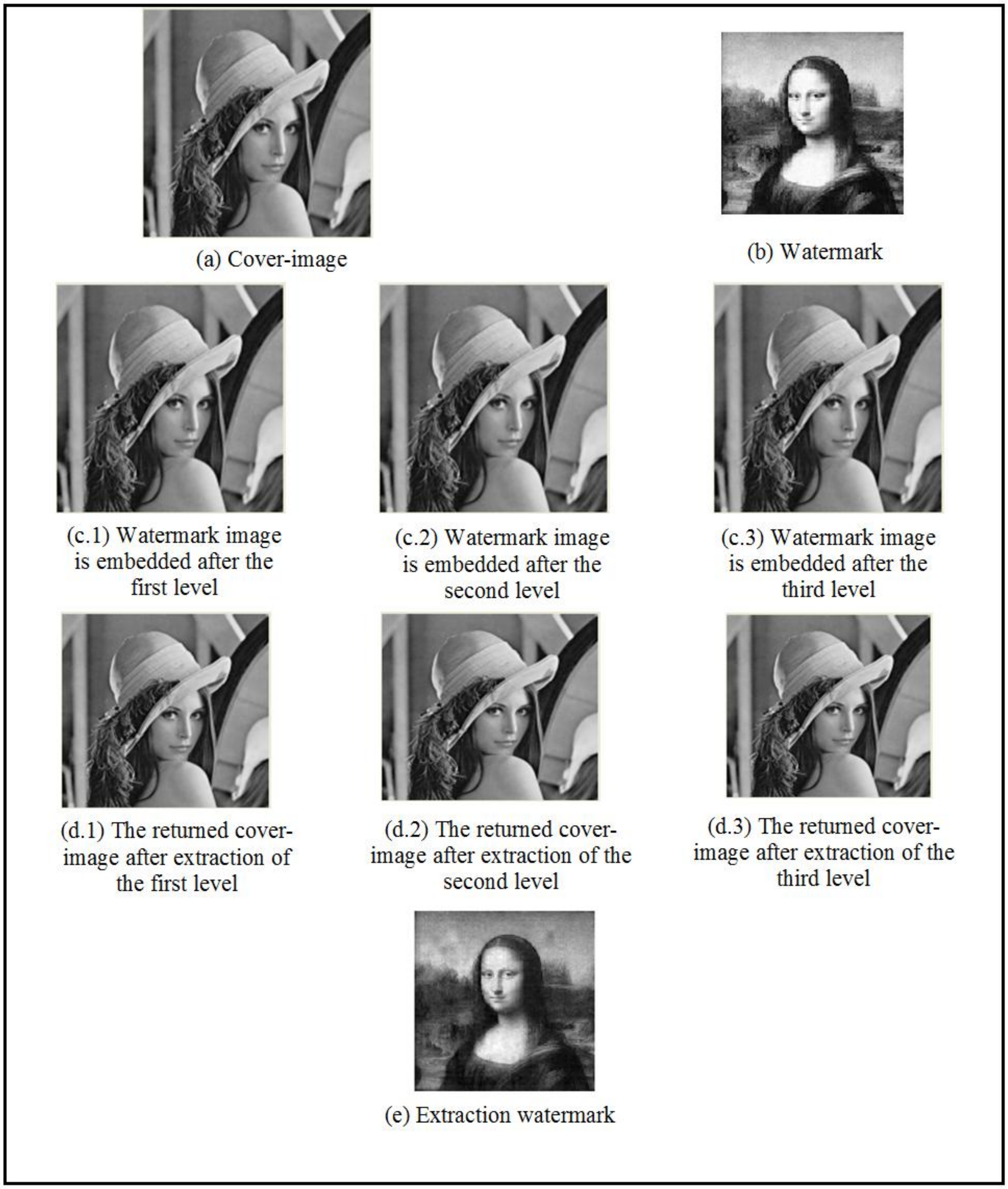
A big amount of secret data was embedded without causing any distortion in the quality of the cover-image in spite of the big distance between the pixels selected in the embedding process by using the secret key. The number of (pixels) embedded in the cover-image was (65536) when planting the fourth-level coefficient of the contourlet transformation of the secret image with dimensions of 256×256.

The results show the efficiency of the algorithm in extracting the watermark in addition to the capability of the algorithm when recovering the cover-image.

5.2 Results of the Second Algorithm Embedding by Means of Using the Energy

The second algorithm was suggested as a result of developing the first one. The results obtained proved that this algorithm is successful and efficient in extracting the watermark and restoring the cover-image, and the results showed a better accuracy that the first algorithm.

Figure 5.2a is a model of the images on which the algorithm was applied in embedding and extracting the watermark. Through studying the repetitive scale in Figure 5.2b the strength of the embedding algorithm for the secret image “Monalliza.jpg” with dimensions 128×128 in the cover-image “Camerman.jpg” with dimensions 512×512.



(a) Cover-image

(b) Watermark

(c.1) Watermark image is embedded after the first level

(c.2) Watermark image is embedded after the second level

(c.3) Watermark image is embedded after the third level

(d.1) The returned cover-image after extraction of the first level

(d.2) The returned cover-image after extraction of the second level

(d.3) The returned cover-image after extraction of the third level

(e) Extraction watermark

Figure 5.2a: A Model for Embedding and Extracting the Secret Image (Monalliza.jpg) with Dimensions 128×128 in the Cover-Image (Lena.jpg) with Dimensions 512×512

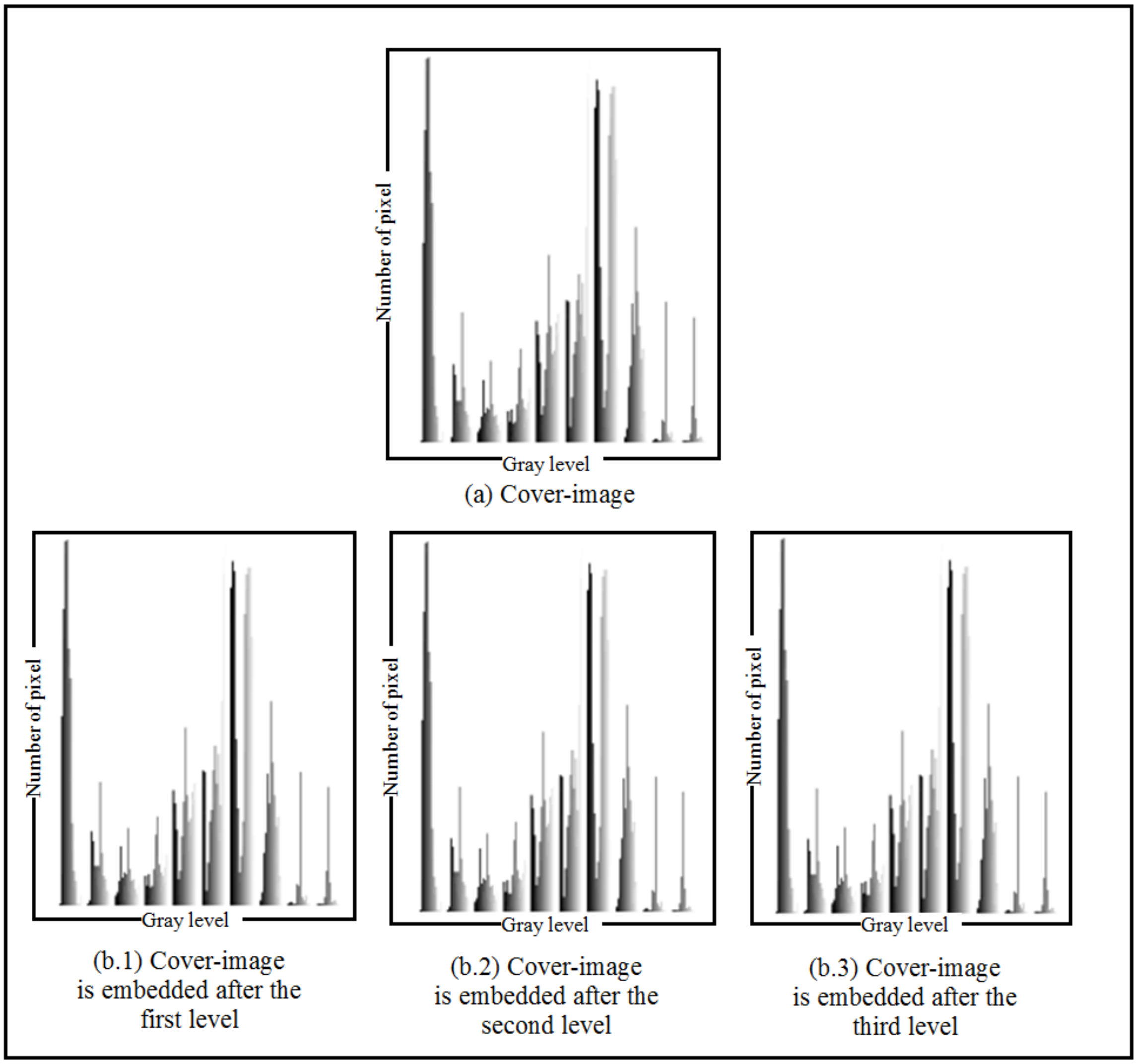


Figure 5.2b: A Model to Study the Repetitive Scale for the Cover-Image before and after the Embedding

From Figure 5.2a the good quality of the secret images extracted can be noticed in comparison with the quality of the same image extracted previously in Figure 5.1a using the first algorithm. From the tables bellow we can see the results reached using the second algorithm. Table 5.2a shows the results of embedding of the secret image “Monalliza.jpg” with the dimensions 128×128 Table 5.2b manifests the results of recovery of the cover-image.

Table 5.2a: Embedding Results According to the Second Algorithm

Cover Image	Cover image size	Watermark	Size of the watermark	Contourlet coefficients	Parameter measure for efficiency performance of the watermark-image			
					MSE	PSNR	NC	SNR
Lena	512×512	Monaliza	128 ×128	{1,1}	0.000037	92.465736	1.000000	86.849557
				{1,2}	0.000000	118.712399	1.000000	113.096220
				{1,3}	0.000000	119.048315	1.000000	113.432137
Lena	1024×1024	Monaliza	128 ×128	{1,1}	0.000003	101.675247	1.000000	95.871994
				{1,2}	0.000000	121.317708	1.000000	115.514456
				{1,3}	0.000000	123.157686	1.000000	117.354433

Table 5.2b: Results of Retrieved the Watermark According to the Second Algorithm

Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure the efficiency of performance for retrieved watermark			
				MSE	PSNR	NC	SNR
Lena	512×512	Monaliza	128×128	123.462377	27.215457	0.990577	20.939359
Lena	1024×1024	Monaliza	128×128	1432.189566	16.570799	0.991802	10.294700

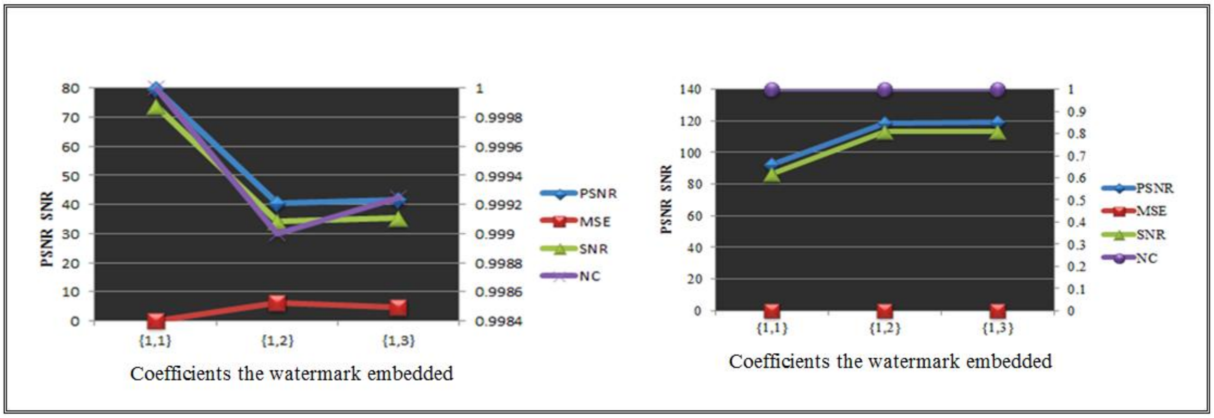
Table 5.2c: Results of the Retrieved Cover-Image

Cover Image	Cover image size	Watermark	Size of the watermark	Contourlet coefficients	Parameter measure for efficiency performance of the retrieved cover-image			
					MSE	PSNR	NC	SNR
Lena	512×512	Monaliza	128×128	{1,1}	0.000000	117.996399	1.000000	112.380220
				{1,2}	0.000000	129.427185	1.000000	123.811006
				{1,3}	0.000000	129.222246	1.000000	123.606067
Lena	1024×1024	Monaliza	128×128	{1,1}	0.000000	112.017571	1.000000	106.214319
				{1,2}	0.000000	132.617573	1.000000	126.814321
				{1,3}	0.000000	134.392097	1.000000	128.588845

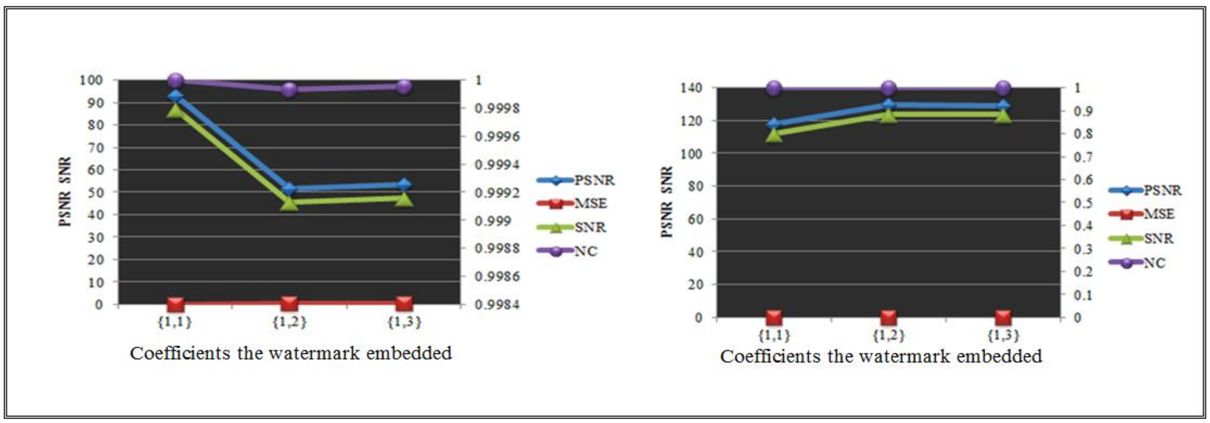
The watermark was embedded in the directional bands with low energy which provided the algorithm with high solidness and enhanced its capability in embedding the data without any change in the quality in the cover-image.

The decrease of energy in the directional bands is the strategy employed in selecting the bands in addition to the high capacity of embedding of the selected bands.

The quality if the watermark extracted increases and that can be noticed through the scales used in addition to the good quality of the cover-image retrieved which was in the highest possible rate. And this can be clarified through comparing the graph in Figure 5.2c to measure the performance efficiency of the two algorithms above, where it explains the quality of the watermark and the resolution of the retrieved cover-image and the extracted watermark using the second algorithm compared to the first algorithm, where (a) the scales of the first algorithm to embed and extract the watermark “Monaliza” with the dimensions 128×128 in the cover-image “Lena” with the dimensions 512×512, (b) the scales of the second algorithm to embed and extract the watermark “Monaliza” with the dimensions 128×128 in the cover-image “cameraman” with dimensions 512×512.



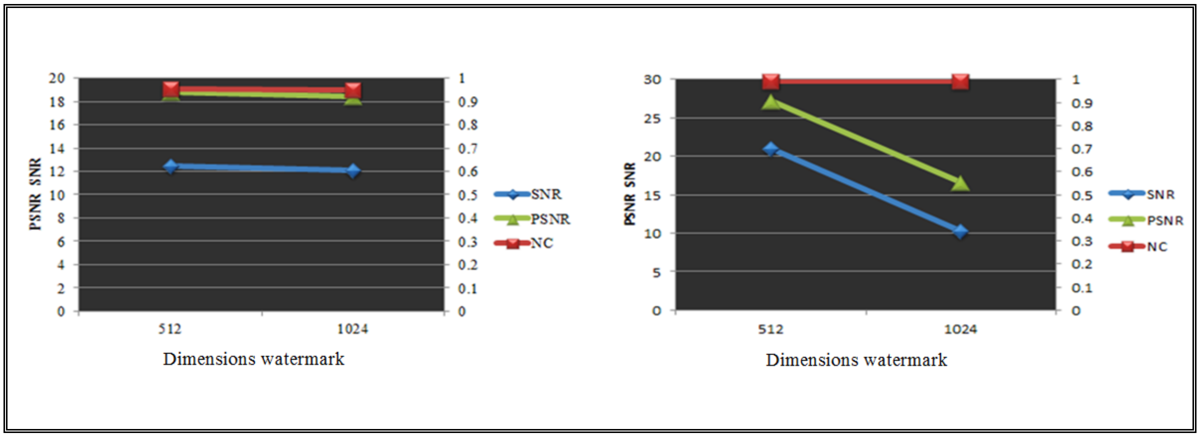
(a.1) Watermark-image the algorithm /1 (b.1) Watermark-image the algorithm /2



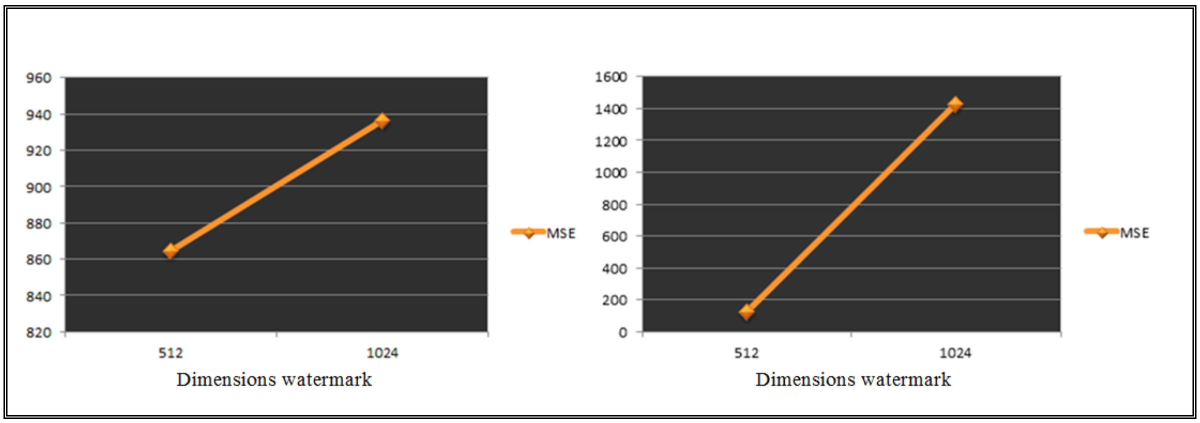
(a.2) Retrieved cover-image the algorithm/1 (b.2) Retrieved cover-image the algorithm /2

Figure 5.2c: Measure the Performance Efficiency of the First and Second Algorithm

To be followed …



(a.3) Watermark extracted the algorithm/1 (b.3) Watermark extracted the algorithm/2



(a.4) Watermark extracted the algorithm/1 (b.4) Watermark extracted the algorithm/2

Figure 5.2c: Continued

5.3 Results of the Third Algorithm Ambedding Using the Contourlet Transformation and Energy

In this algorithm the quadric tree division idea was added to increase the hardness of hiding in addition to what it provide of capabilities of embedding a watermark with sizes close to the cover-image size and that was done through dividing the watermarking (i.e. disassembling it into small particles, each of which is considered an independent watermark). This technique resulted in reaching good findings that will be reviewed later.

5.3.1 The First Part

The first part of the third algorithm led to results that exceeded the results of the two algorithms above in embedding and recovering the watermark. Figure 5.3a is a model to embed the watermark “Flower.png” with dimensions 128×128 after dividing it into four parts with dimensions of 64×64, and the watermark image using the first part of the third algorithm.

And Figure 5.3b shows the extracted watermark and the parts of the recovered cover-image “Lena.png” with dimensions 512×512, but Figure 5.3c is a graph that shows the scales of efficiency of the algorithm to extract the parts of the watermark “flower” using the strategy employed in the third algorithm to embed and extract the watermark.

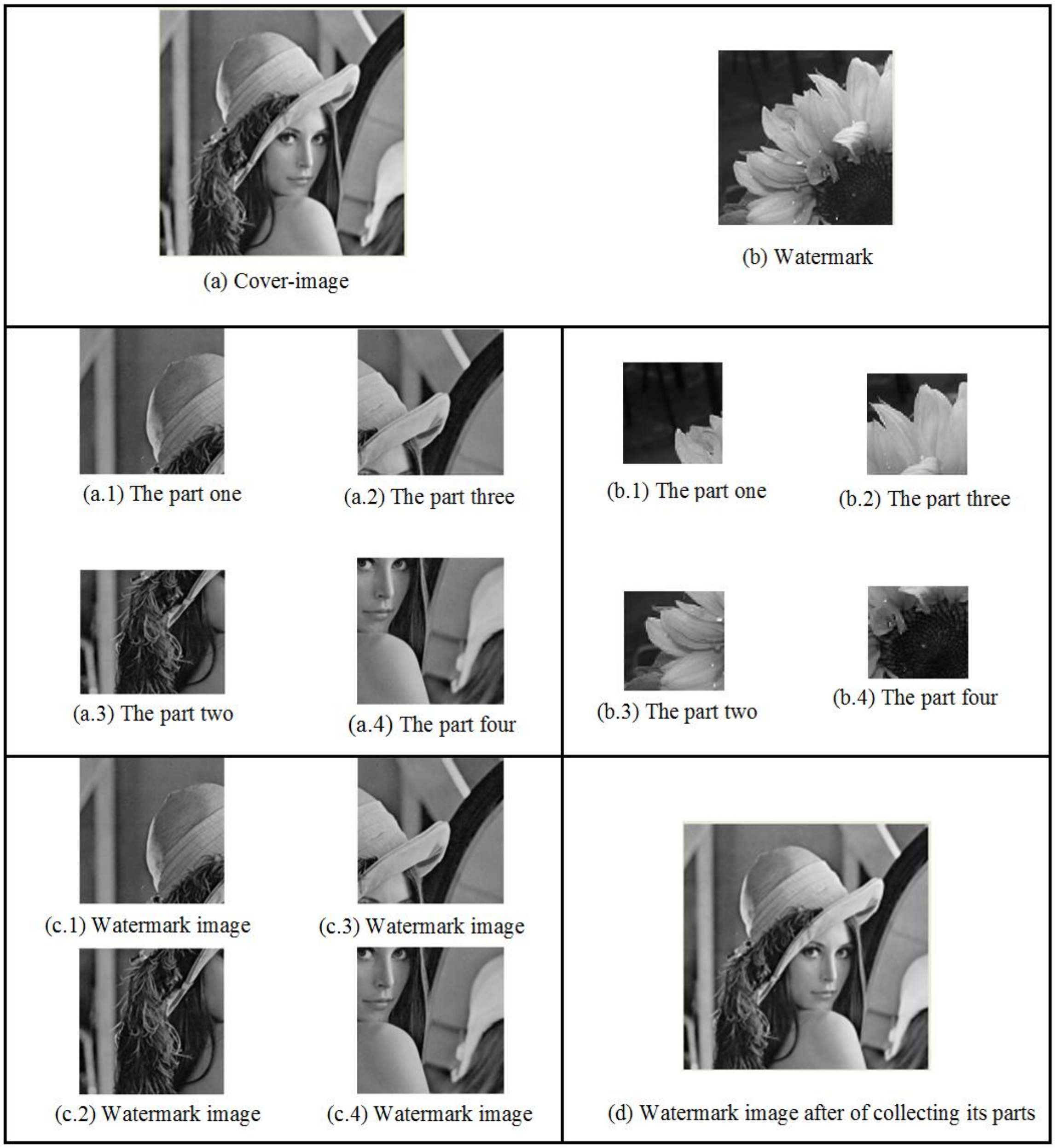


Figure 5.3a: A Model of Embedding the Secret Image (Flower.png) in the Cover-Image (Lena.png)

Figure 5.3b: A Model for Extracting the Secret Image “Flower.png” and the Retrieved Cover-Image (Lena.png)

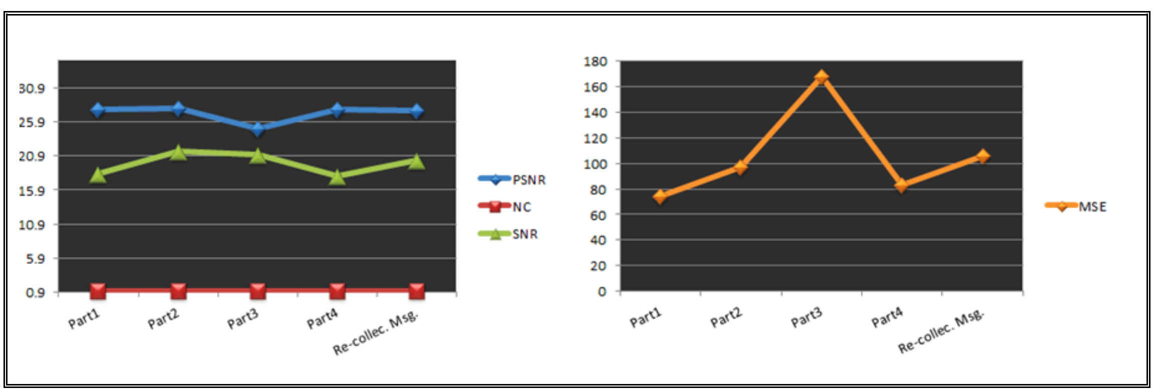


Figure 5.3c: A Chart Shows the Scales of the Third Algorithm Efficiency Part One of the Extracted Secret Image “Flower.png”

While Table 5.3a manifests the results of the watermark. From the Table 5.3b we can clarify the results of extracting the watermark with its four parts “Flower.png” from the parts of the cover-image “Lena.png”. But Figure 5.3c shows the results of recovering the cover-image.

Table 5.3a: The Results of Embedding the Watermark in the Third Algorithm the First Part

Cover Image	**Cover image size**	**Watermark**	**Size of the watermark**	**Contourlet coefficients**	**Parameter measure for efficiency performance of the watermark-image**			
Lena	**1024×1024**	**Part from Flower**	**128×128**		**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	64×64	{1,1}	0.000004	99.941003	1.000000	94.023212
				{1,2}	0.000000	121.930140	1.000000	116.012349
				{1,3}	0.000000	122.884130	1.000000	116.966339
Part2	512×512	Part2	64×64	{1,1}	0.000016	94.451115	1.000000	86.393365
				{1,2}	0.000000	120.178846	1.000000	112.121096
				{1,3}	0.000000	121.841086	1.000000	113.783336
Part3	512×512	Part3	64×64	{1,1}	0.000023	92.884560	1.000000	88.069917
				{1,2}	0.000000	119.699611	1.000000	114.884967
				{1,3}	0.000000	120.770526	1.000000	115.955882
Part4	512×512	Part4	64×64	{1,1}	0.000005	99.423635	1.000000	94.881229
				{1,2}	0.000000	120.085283	1.000000	115.542877
				{1,3}	0.000000	121.336815	1.000000	116.794408

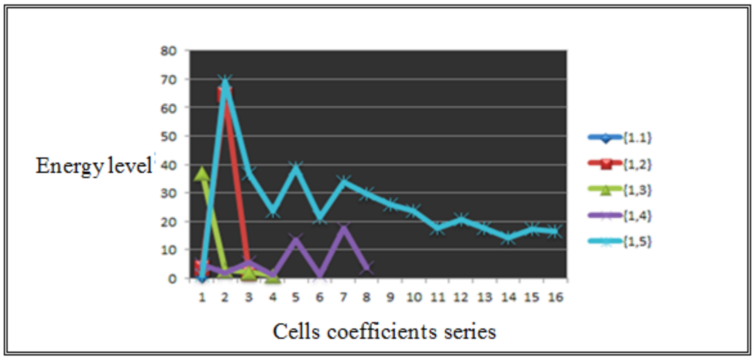
Table 5.3b: The Results of Extracting the Watermark in the Third Algorithm the First Part

Cover Image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of the extracted watermark**			
Lena	**1024×1024**	**Earth**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	64×64	73.922108	27.673523	0.994689	18.355091
Part2	512×512	Part2	64×64	97.138451	27.944790	0.991037	21.655048
Part3	512×512	Part3	64×64	168.158146	24.977378	0.992986	21.079227
Part4	512×512	Part4	64×64	82.885288	27.820186	0.990559	17.900146
After of collecting the four parts				105.525998	27.585107	0.993735	20.368837

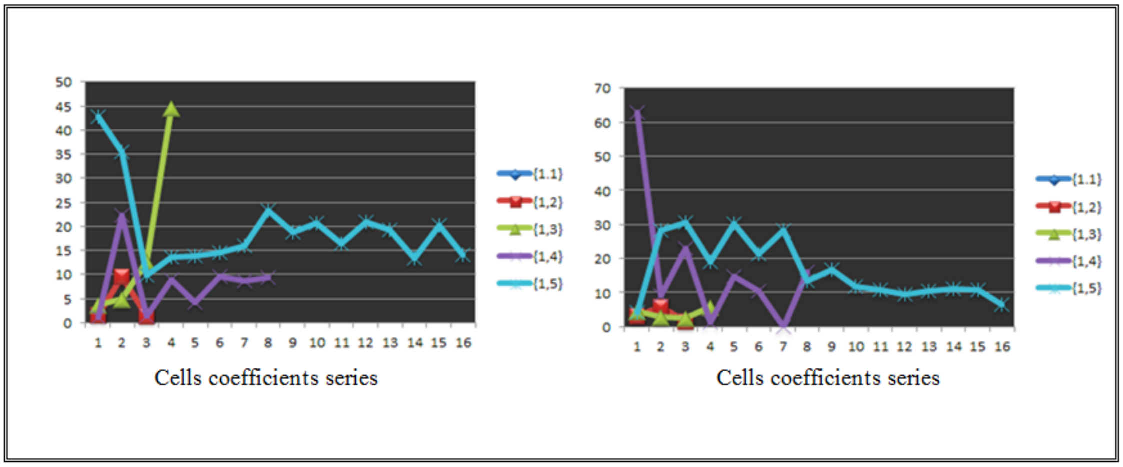
Table 5.3c: The Results of Retrieved the Watermark in the Third Algorithm the First Part

Cover Image	**Cover image size**	**Watermark**	**Size of the watermark**	**Contourlet coefficients**	**Parameter measure for efficiency performance of the retrieved cover-image**			
Lena	**1024×1024**	**Part from Flower**	**128×128**		**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	64×64	{1,1}	0.000000	123.307609	1.000000	117.389818
				{1,2}	0.000000	134.462805	1.000000	127.934411
				{1,3}	0.000000	134.276372	1.000000	128.358581
Part2	512×512	Part2	64×64	{1,1}	0.000000	120.553064	1.000000	112.495314
				{1,2}	0.000000	131.358427	1.000000	123.300677
				{1,3}	0.000000	132.902178	1.000000	124.844428
Part3	512×512	Part3	64×64	{1,1}	0.000000	118.351484	1.000000	113.536840
				{1,2}	0.000000	131.034658	1.000000	126.220014
				{1,3}	0.000000	132.138745	1.000000	127.324101
Part4	512×512	Part4	64×64	{1,1}	0.000000	123.576100	1.000000	119.033694
				{1,2}	0.000000	131.449627	1.000000	126.907221
				{1,3}	0.000000	132.375643	1.000000	127.833237

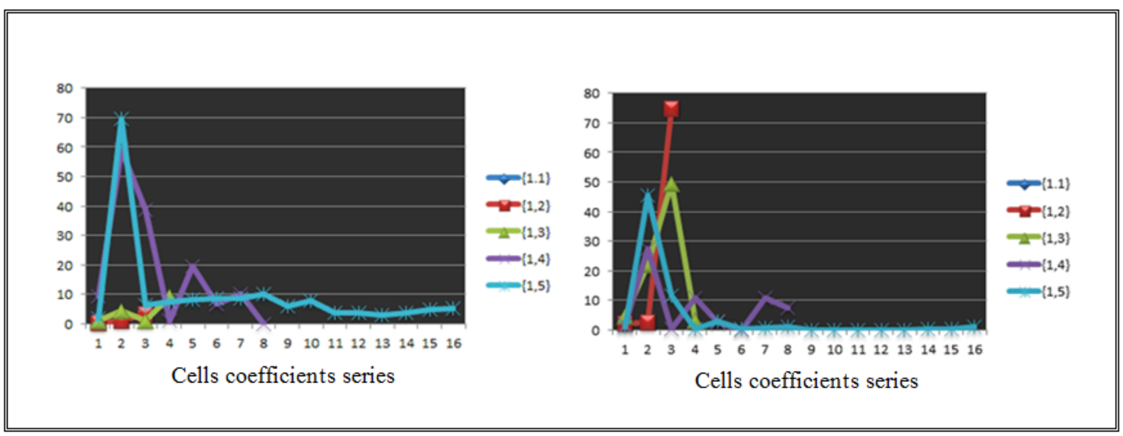
Employing the division of the cover-image and the secret image in building the hiding system resulted in a decrease in the noise in the extracted secret image and the recovered cover-image and this can be clarifies through the values of PSNR and SNR in addition to the increase in the degree of approximation between the original image and the recovered one and that can be observed through the values of NC. In spite of using the same embedding strategy for embedding and extracting used in the second algorithm, and the reason behind that is the decrease of the energy in the directional bands for the various levels of the contourlet analysis in the parts of the cover-image compared to the energy levels in the image as a whole. Figure 5.3d is a study for comparing the levels of the contourlet transformation coefficients of the cover-image "Lena" with dimensions 1024×1024 and the energy levels in the coefficients of its parts with the dimensions 512×512 after applying the Quadric Tree Division.



(a) Original cover-image



(a) The first part of the cover-image (c) The third part of the cover-image



(b) The second part of the cover-image (d) The fourth part of the cover-image

Figure 5.3d: A Graph Shows the Variation of the Energy Levels between the Original Cover-Image and its Parts

5.3.2 Part Two

In this part we studied what the idea of quadric tree division provides in terms of the cover-image when the size of the watermark embedded in it is constant, where a watermark was embedded in the cover-image without dividing it, and embedding the same watermark inside the same cover after dividing it using the quadric tree division with various levels.

The results of the practical application showed that the increase of the division depth leads to a high rate of approximation between the cover-images and the images of the watermark before and after embedding in addition to the advantages mentioned concerning the use of the quadric tree division. The following is the practical examples that have been obtained.

5.3.2.1 Embedding in Part of the Cover Dimension 512×512

As we mentioned in chapter four, each of the parts of the watermark was fully embedded in part of the corresponding cover-image. Figure 5.4a is a model of embedding the watermark "Earth.png" with dimensions 128×128 after dividing it using the tree division principle into parts with dimensions 64×64 in the parts of the cover-image "Lena" using the second part of the third algorithm. Figure 5.4b shows the parts of the extracted watermark and the parts of the recovered image. Through Figure 5.4c we can notice the approximation degree between the cover-image before and after embedding through the repetitive scale.

Figure 5.4a: A Model for Embedding the Secret Image (Earth.png) in Cover-Image Part (Lena.png)

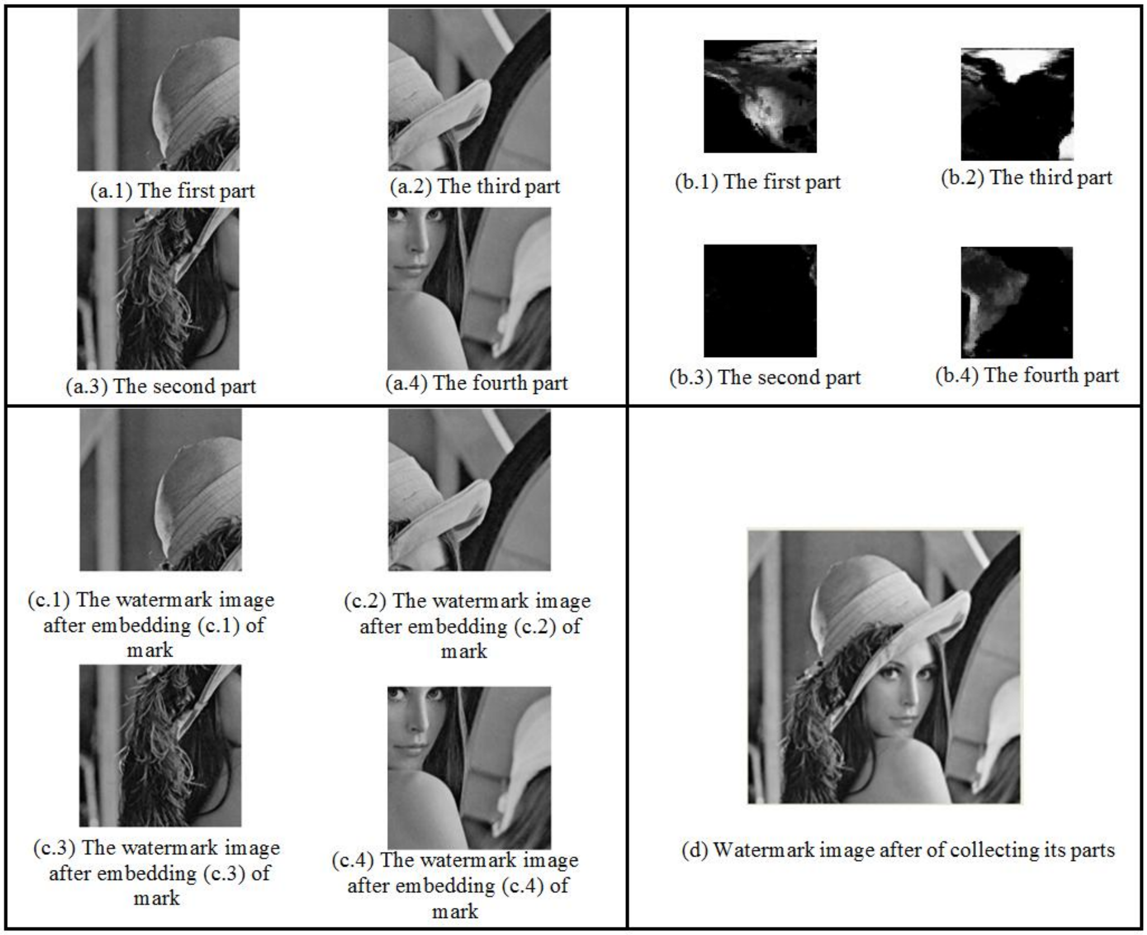


Figure 5.4a: Continued

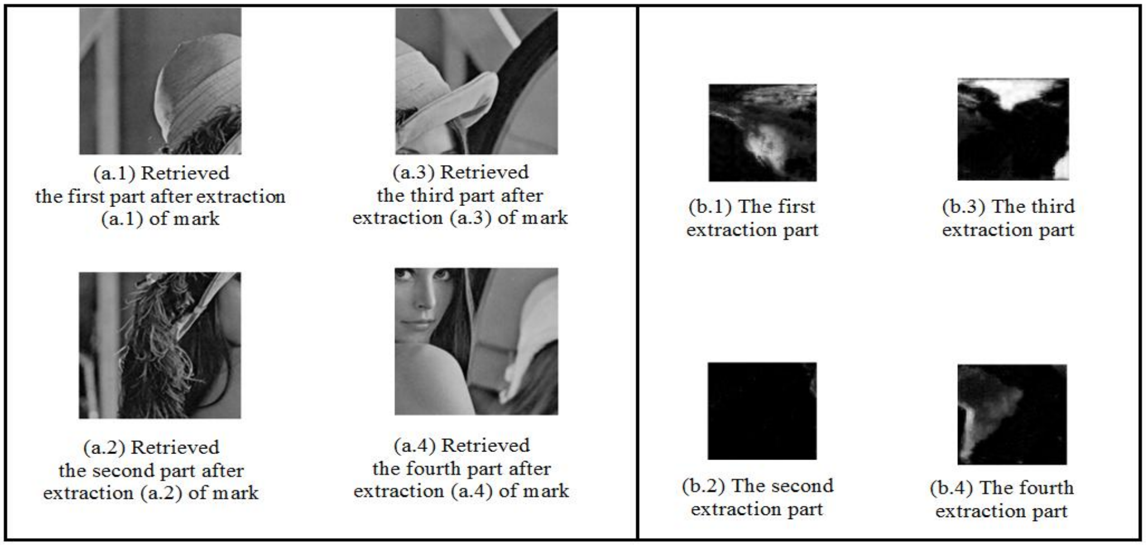


Figure 5.4b: A Model to Extraction the Secret Image (Earth.png) and Retrieved Cover-Image (Lena.png)

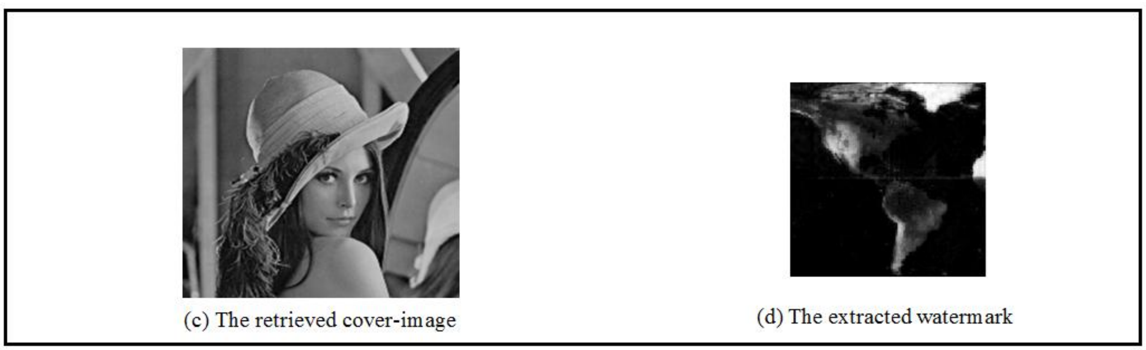


Figure 5.4b: Continued

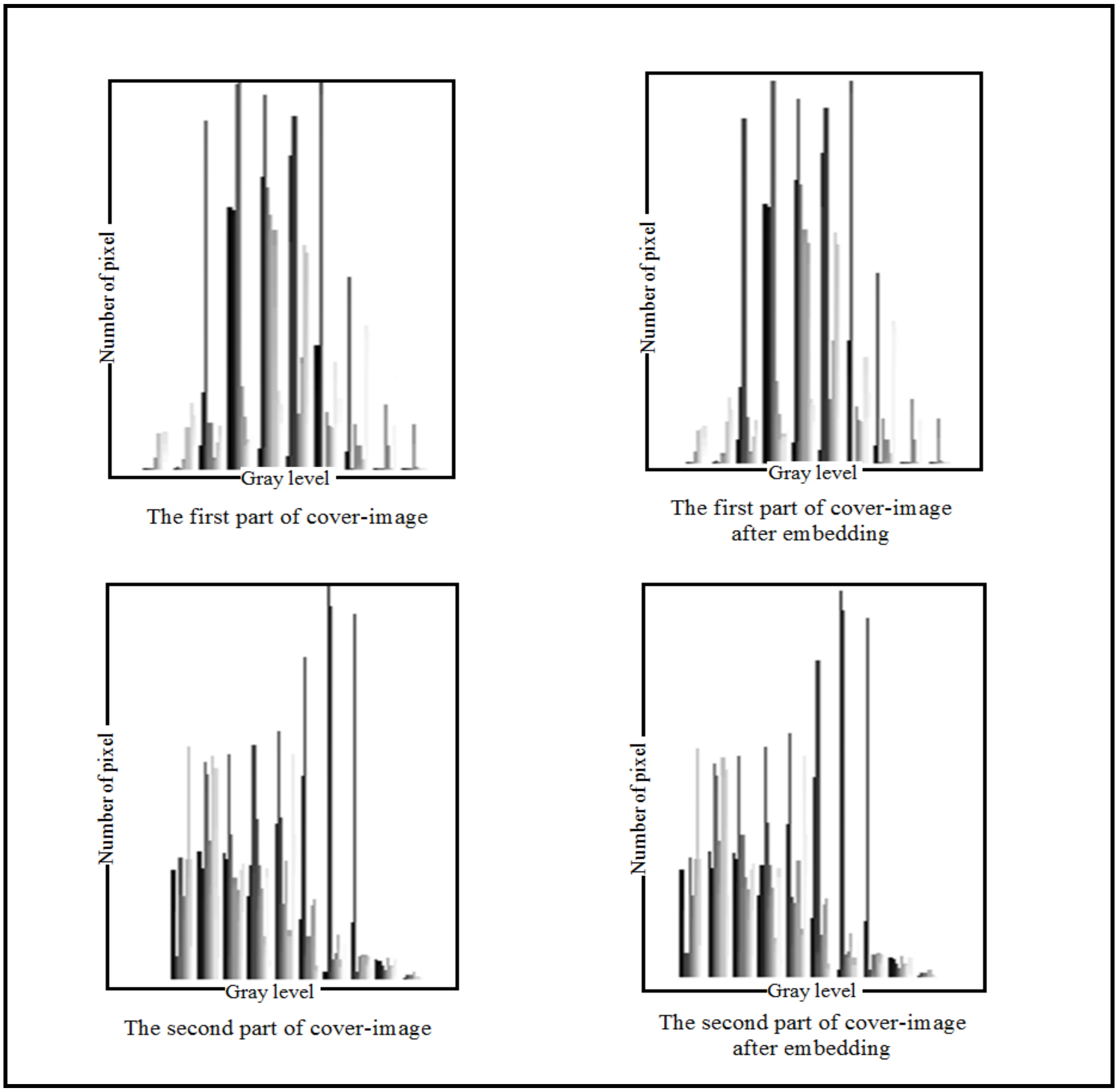


Figure 5.4c: Histogram Study to Clarify the Extent of Convergence of Cover-Image before and after the Embedding Model 5.4a the Above-Mentions

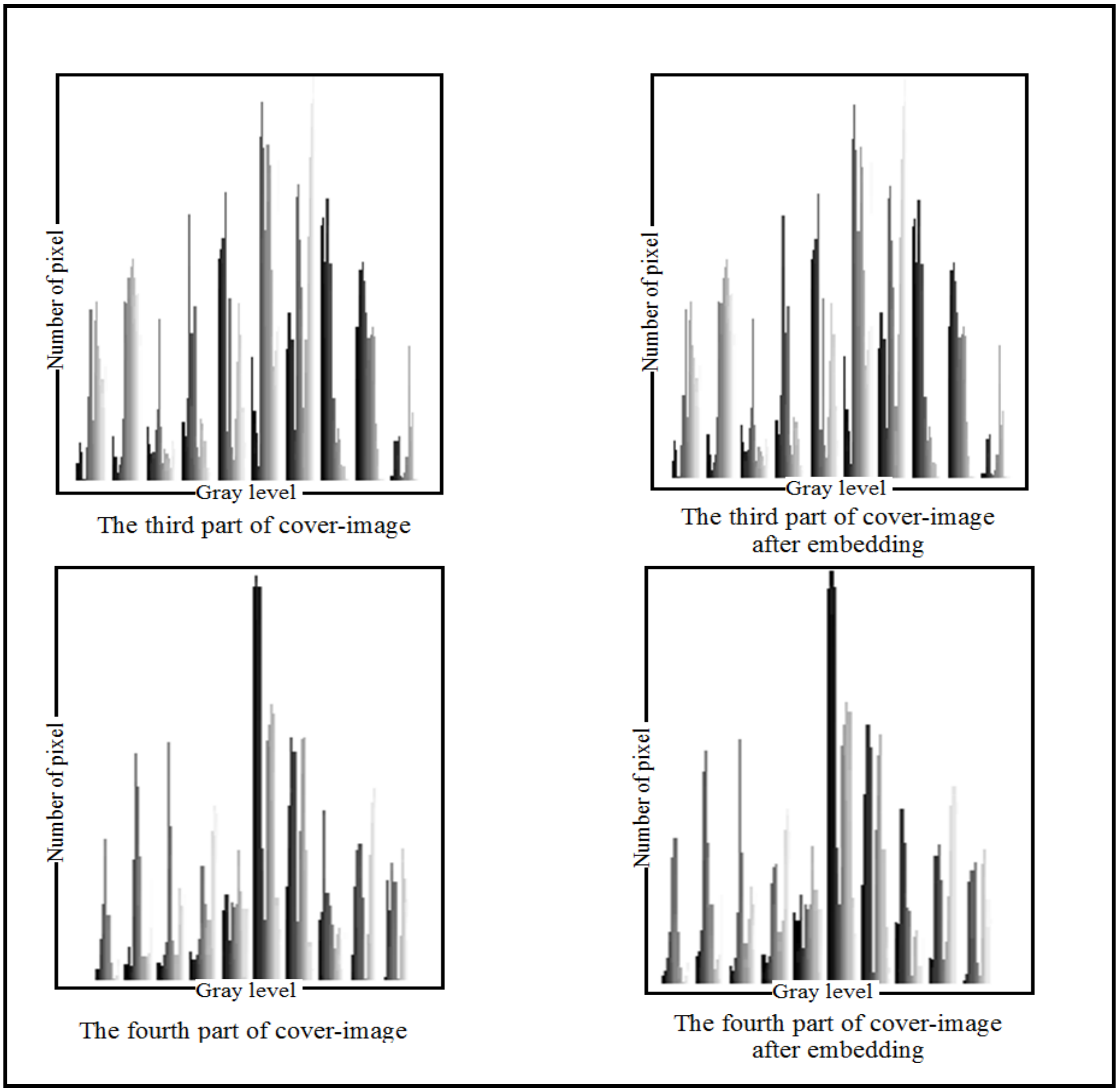


Figure 5.4c: Continued

From Table 5.4a we can clarify the results of the embedding of the watermark (Earth.png). And Table 5.4b shows the results of extracting the four parts of the watermark. While table 5.4c shows the results of the cover-image retrieved.

Table 5.4a: The Results of the Watermark the Results of Embedding

Cover Image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of the watermark-image**			
Lena	**1024×1024**	**Earth**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	64×64	0.000001	105.676017	1.000000	99.758226
Part2	512×512	Part2	64×64	0.000000	122.227855	1.000000	114.170105
Part3	512×512	Part3	64×64	0.000002	104.207138	1.000000	99.392494
Part4	512×512	Part4	64×64	0.000000	109.476484	1.000000	104.934078
After return of collecting the parts of the watermark-image				0.000001	107.309568	1.000000	101.506316

Table 5.4b: The Results of the Extracting the Watermark

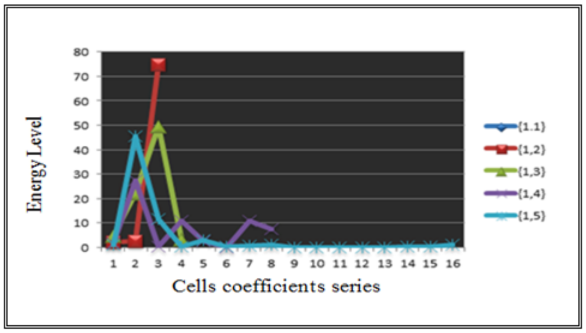
Cover Image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of the extracted watermark**			
Lena	**1024×1024**	**Earth**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	64×64	128.818162	26.859029	0.983504	16.064188
Part2	512×512	Part2	64×64	4.576802	34.978004	0.952638	10.274168
Part3	512×512	Part3	64×64	128.414937	27.044648	0.991717	18.551428
Part4	512×512	Part4	64×64	27.526487	32.371622	0.989915	17.572489
After of collecting the four parts				72.334097	29.537373	0.989727	17.434441

Table 5.4c: The Results of Retrieved the Four Parts of the Cover-Image

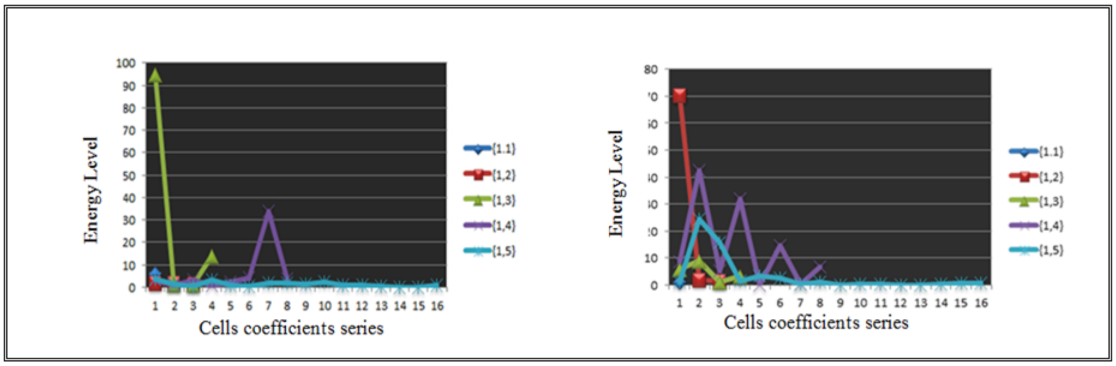
Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the retrieved cover-image			
Lena	**1024×1024**	**Earth**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	64×64	0.000000	125.045612	1.000000	119.127820
Part2	512×512	Part2	64×64	0.000000	138.507876	1.000000	130.450126
Part3	512×512	Part3	64×64	0.000000	124.988361	1.000000	120.173718
Part4	512×512	Part4	64×64	0.000000	129.897301	1.000000	125.354894
After return of collected parts to retrieved cover-image				0.000000	127.455026	1.000000	121.651773

5.3.2.2 Embedding in Part of the Cover (Dimensions 256×256)

Figure 5.5a shows a comparison between the levels of energy of the contourlet transformation coefficients for the fourth part of the cover-image “Lena” with dimensions 512×512, with energy levels in the coefficients of its four parts with the dimensions 256×256.

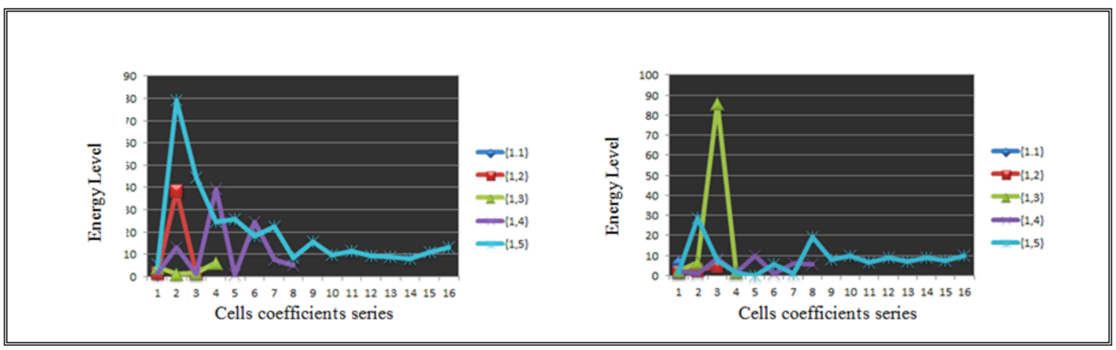


(a) The fourth part of cover-image



(b) The part 4/1 of cover-image

(c) The part 4/3 of cover-image



(b) The part 4/2 of cover-image

(d) The part 4/4 of cover-image

Figure 5.5a: Studying the Energy Levels of the Fourth Part of the Cover-image (Lena) and its Parts after Applying the Quadric Tree Division

Figure 5.5b is a model of embedding the parts with dimensions 64×64 of the watermark in the fourth part of the cover-image “Lena” after applying the quadric tree division and Figure 5.5c show the watermark extracted and the cover-image recovered “Lena” While Figure 5.5d manifest the study of the repetitive scale of the Figure 5.5b to show the degree of approximation between the cover-image before and after embedding and to show the strength of the watermark hiding that is hard to predict that it exists.

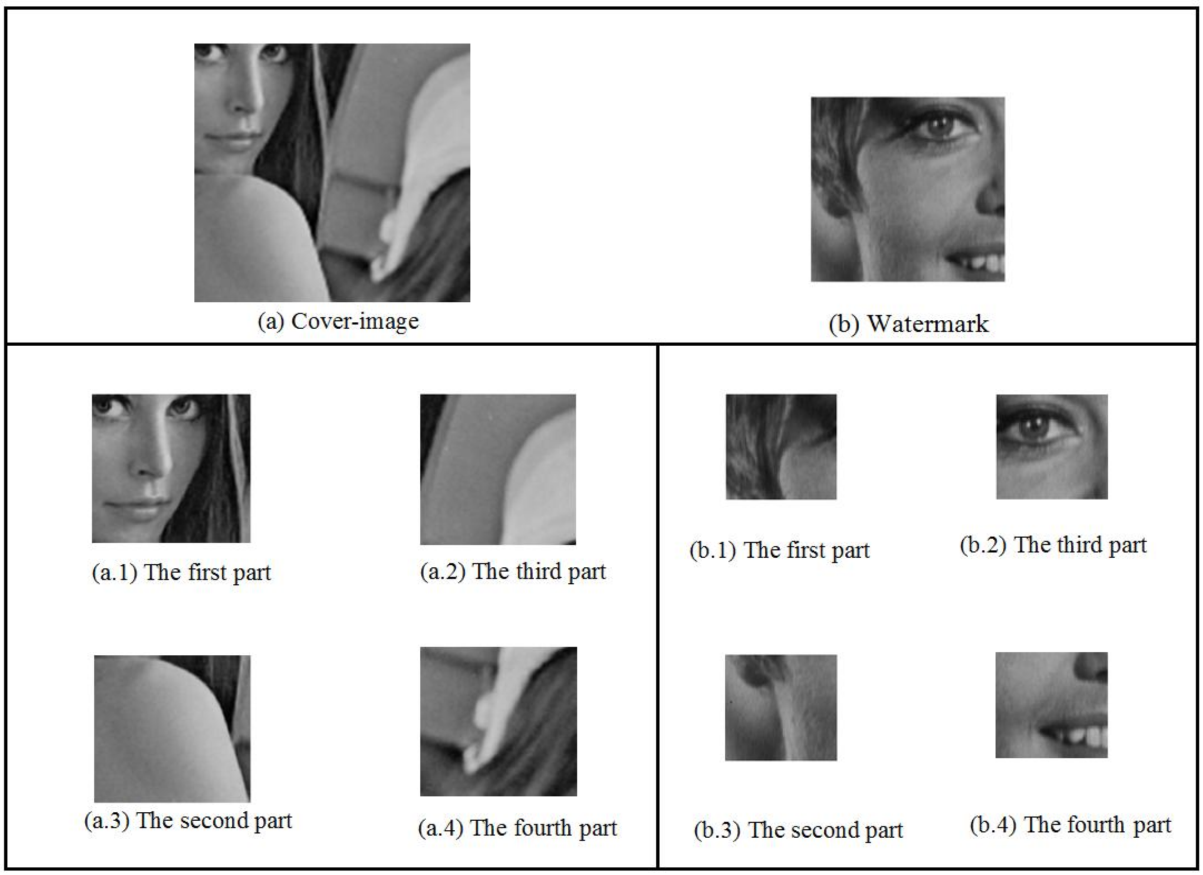


Figure 5.5b: A Model for Embedding the Secret Image (Zelda.png) in Cover-Image Part (Lena.png)

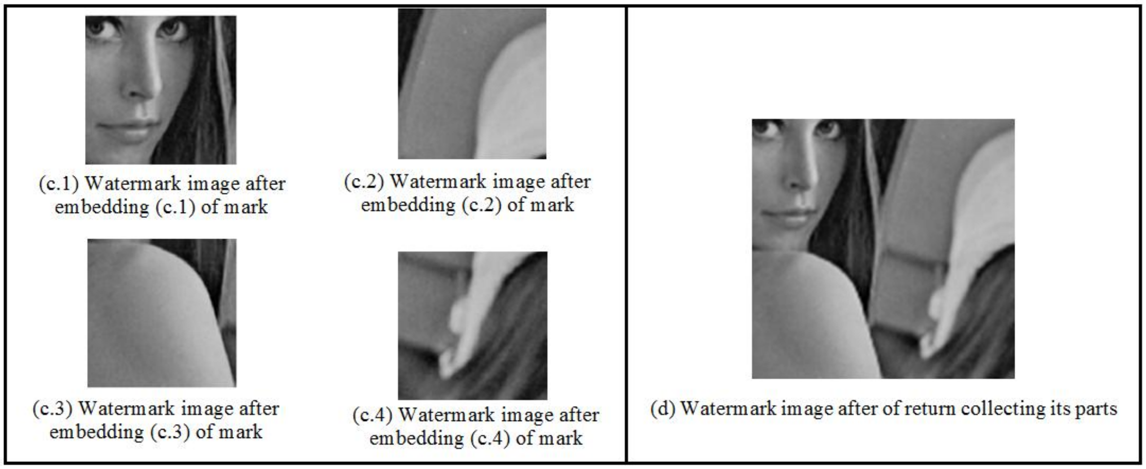


Figure 5.5b: Continued

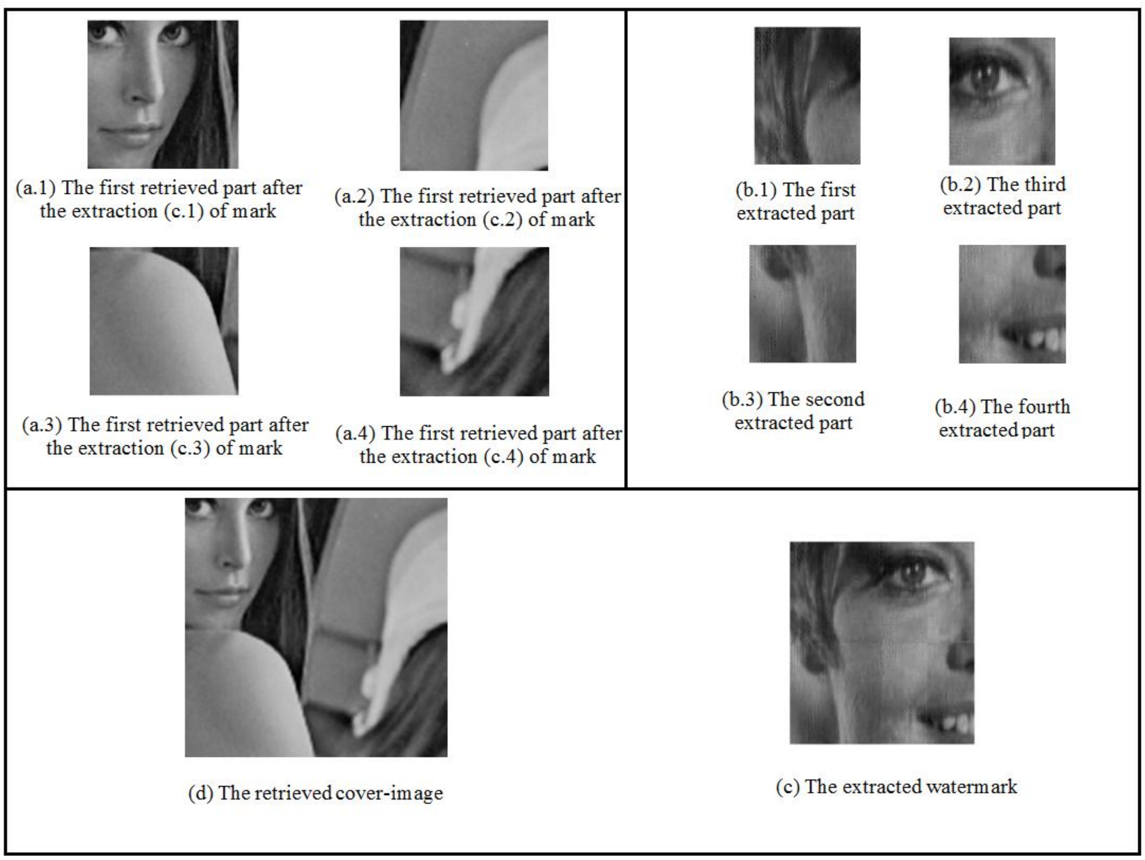


Figure 5.5c: A Model for Extracting the Secret Image (Zelda.png) and the Retrieved Cover-Image (Lena.png)

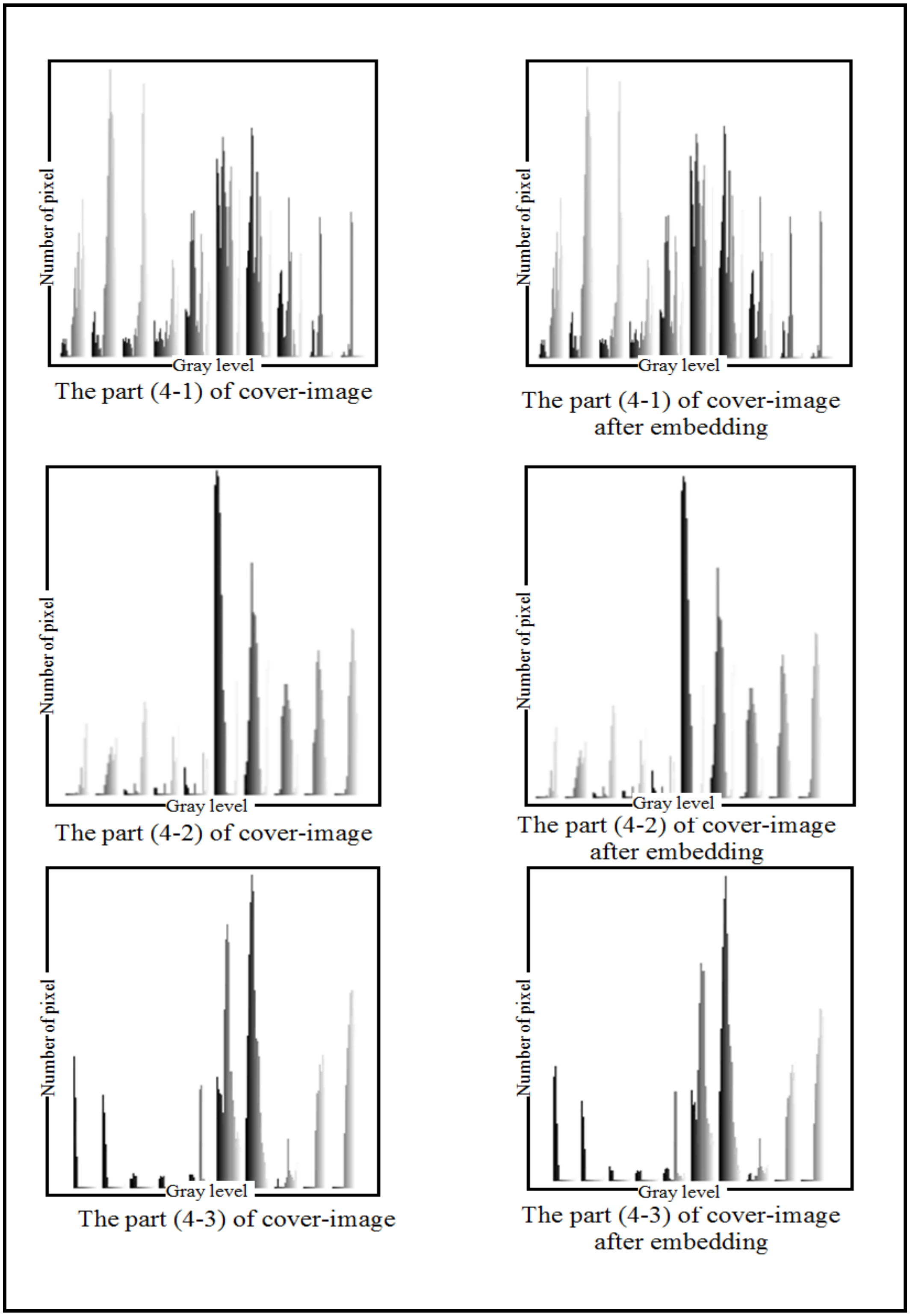


Figure 5.5d: Histogram Study to Clarify to Figure 5.5b

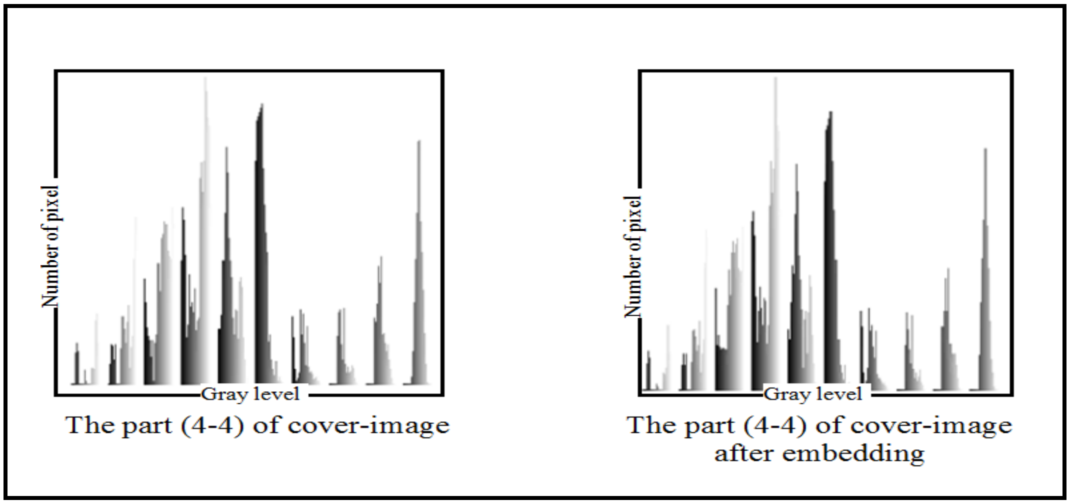


Figure 5.5d: Contiuned

Through Table 5.5a we can present the results of embedding the watermark “Zelda” in the fourth part of the cover-image “Lena” after applying the quadric tree division. Through Table 5.5b we can show the results of the extracting the parts of the watermark “Zelda” with dimensions of 64×64 of the fourth part from the cover-image “Lena”, and Table 5.5c shows the recovery of these parts of the cover-image.

Table 5.5a: The Results of the Watermark the Embedding Process

Cover image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the watermark-image			
Part4 Lena	**512×512**	**Part from Zelda**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	256×256	Part1	64×64	0.000006	98.756602	1.000000	92.747815
Part2	256×256	Part2	64×64	0.000009	96.541129	1.000000	93.097074
Part3	256×256	Part3	64×64	0.000034	90.758090	1.000000	87.656669
Part4	256×256	Part4	64×64	0.000009	96.457627	1.000000	90.948703
After return of collecting the parts of the watermark-image				0.000014	94.709632	1.000000	90.167226

Table 5.5b: Results of the Watermark Extraction According to the Third Algorithm the Second Part

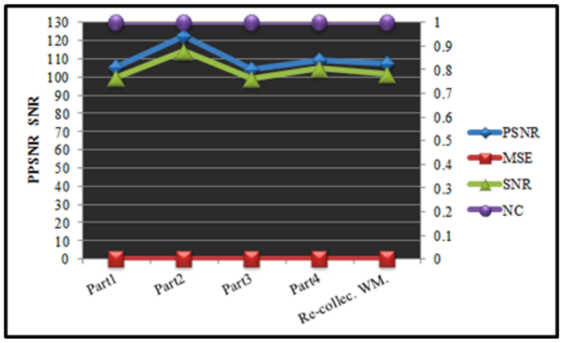
Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the extracted watermark			
Part4 Lena	**512×512**	**Part from Zelda**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	256×256	Part1	64×64	56.927020	25.968641	0.972671	20.437723
Part2	256×256	Part2	64×64	51.670304	26.949989	0.962898	23.072683
Part3	256×256	Part3	64×64	44.124820	28.849527	0.984032	24.575304
Part4	256×256	Part4	64×64	49.605232	28.293747	0.973139	24.156662
After return of collected the four parts				50.581844	28.256410	0.979005	23.208139

Table 5.5c: Results of the Retrieved of the Cover-Image According to the Third Algorithm the Second Part

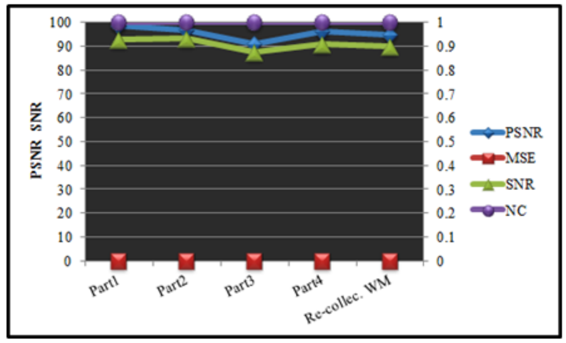
Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the retrieved cover-image			
Part4 Lena	**512×512**	**Part from Zelda**	**128×128**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	256×256	Part1	64×64	0.000000	121.929787	1.000000	115.921001
Part2	256×256	Part2	64×64	0.000000	121.925124	1.000000	118.481069
Part3	256×256	Part3	64×64	0.000000	119.560552	1.000000	116.459130
Part4	256×256	Part4	64×64	0.000000	122.811473	1.000000	117.302549
After return of collected parts to retrieved cover-image				0.000000	121.568298	1.000000	117.025892

The approximation value was NC = 1 between the cover-image and the watermark image without embedding the secret key. But after embedding the key, that has been a very slight difference where the value became NC = 0.999998 and that proves the correctness of the previous algorithm.

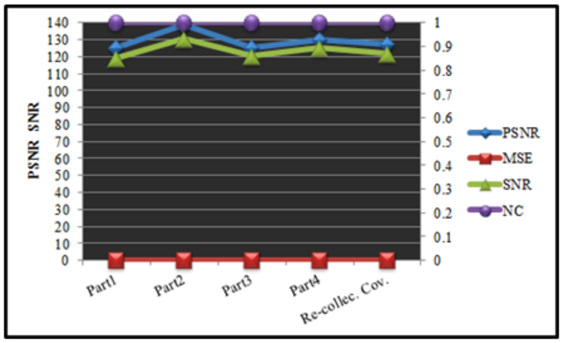
Figure 5.5e shows a comparison between the performance efficiency measures of the third algorithm second part when embedding the parts of the watermark “Earth” with dimensions of 64×64 in the parts of the cover-image “Lena” with dimensions of 512×512 and embedding parts of the watermark “Zelda” with dimensions of 64×64 in the fourth part of the cover-image “Lena” after dividing it into four parts with dimensions of 256×256 using the quadric tree division.



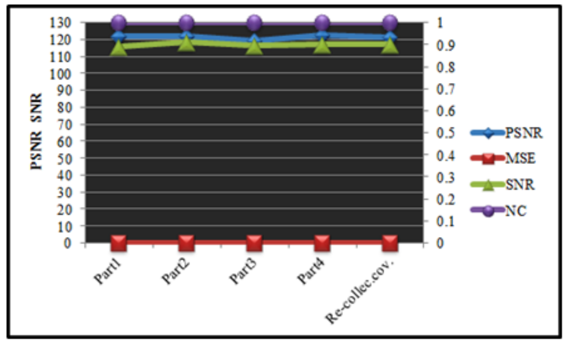
(a.1) The watermark image-algorithm 3-cover part 512



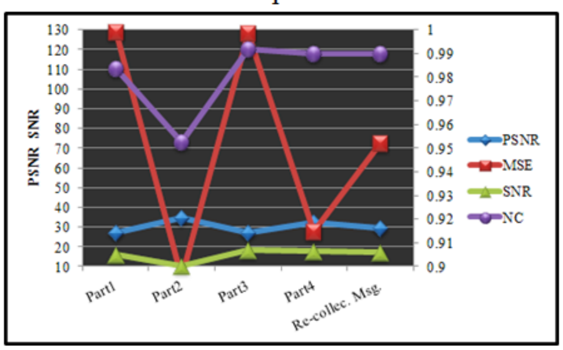
(b.1) The watermark image-algorithm 3-cover part 256



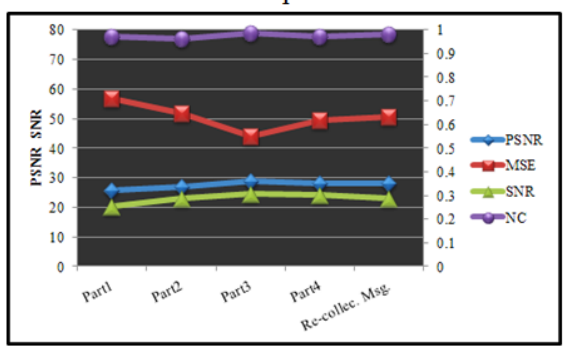
(a.2) The retrieved cover-image-algorithm 3-cover part 512



(b.2) The retrieved cover-image-algorithm 3-cover part 256



(a.3) The extracted watermark-algorithm 3-cover part 512



(b.3) The extracted watermark-algorithm 3-cover part 256

Figure 5.5e: Shows the Measures of Performance Efficiency of the Third Algorithm the Second Part

5.3.2.3 Embedding in the Cover Without Division

Watermarks with different dimensions were embedded in the cover-image “Lena” with dimensions of 1024×1024. Figure 5.6a is a model for embedding and extracting the secret image with dimensions of 64×64 (cameraman.tif), Figure 5.6b shows the study of the repetitive scale of the Figure 5.6a while Figure 5.6c is a graph that shows the performance efficiencies scales when embedding the watermarks with different dimensions in the cover-image “Lena”. The results of embedding can be shown using Table 5.6a and from table 5.6b we can review the results of extraction. Moreover, the results of the cover-image recovery are shown in Table 5.6c.

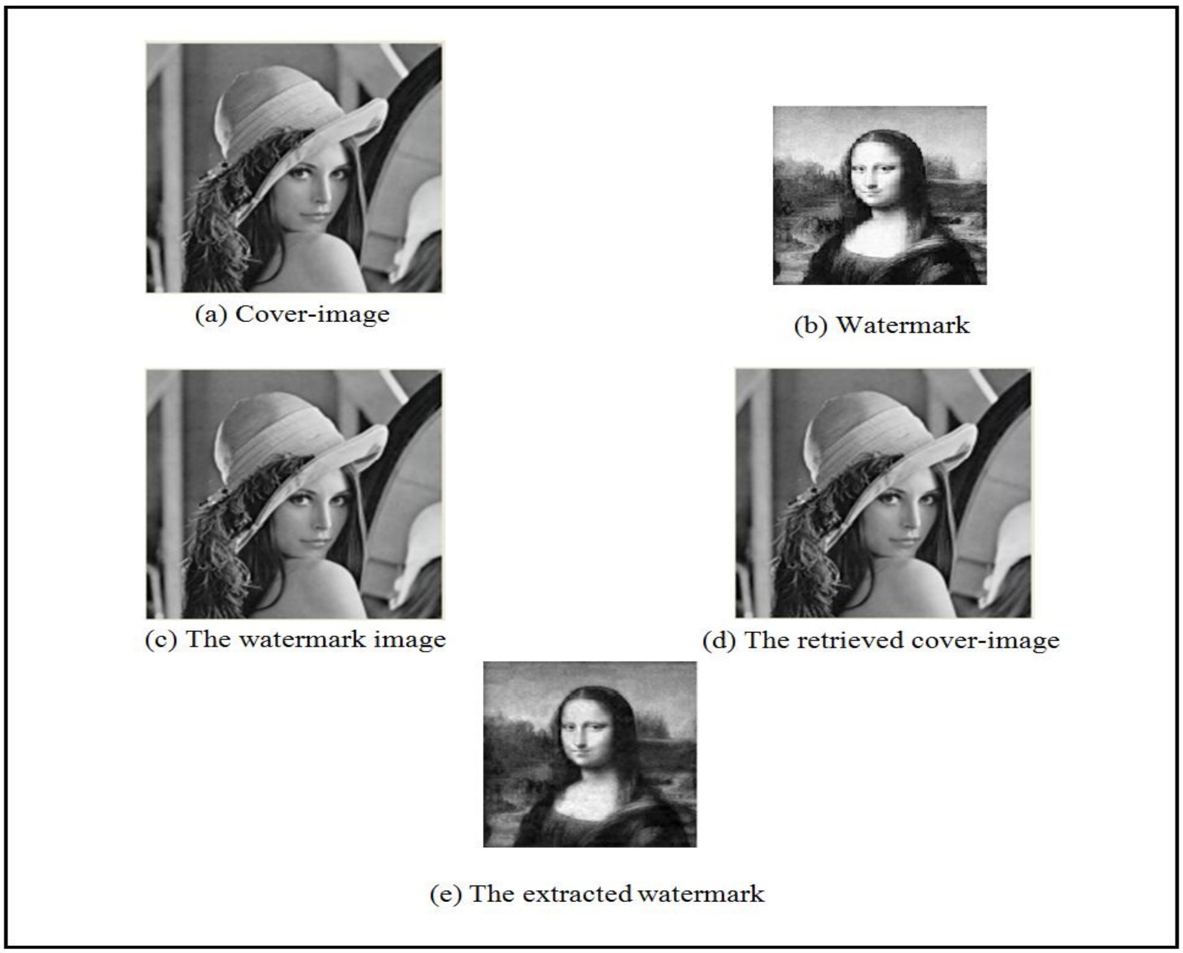


Figure 5.6a: A Model for Embedding and Extracting the Secret Image (Cameraman.tif) in the Cover-Image (Lena.png)

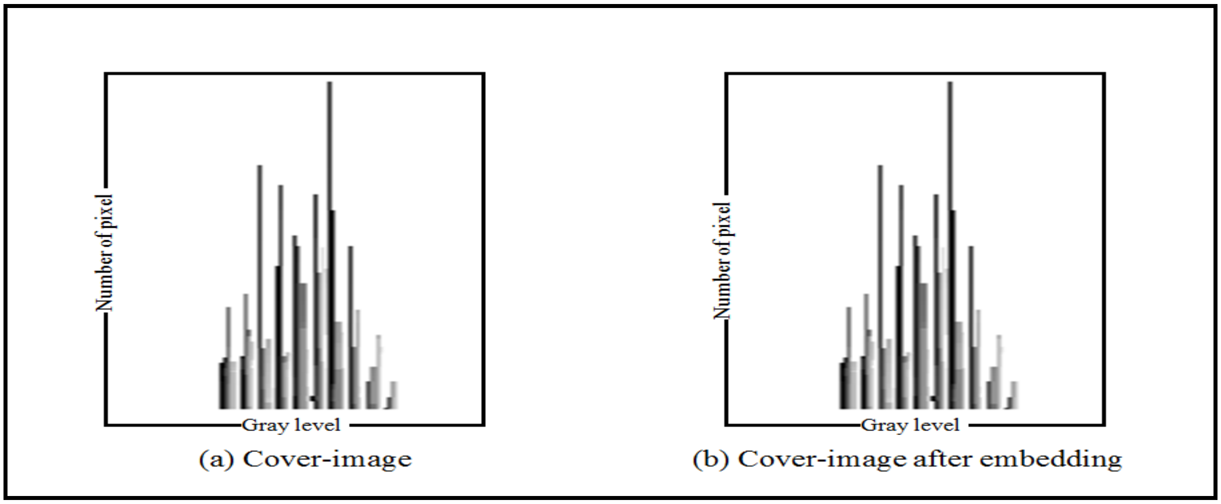


Figure 5.6b: Studying the Repetitive Scale of Figure 5.6a Above

Table 5.6a: The Results of Embedding the Different-Dimension Watermarks

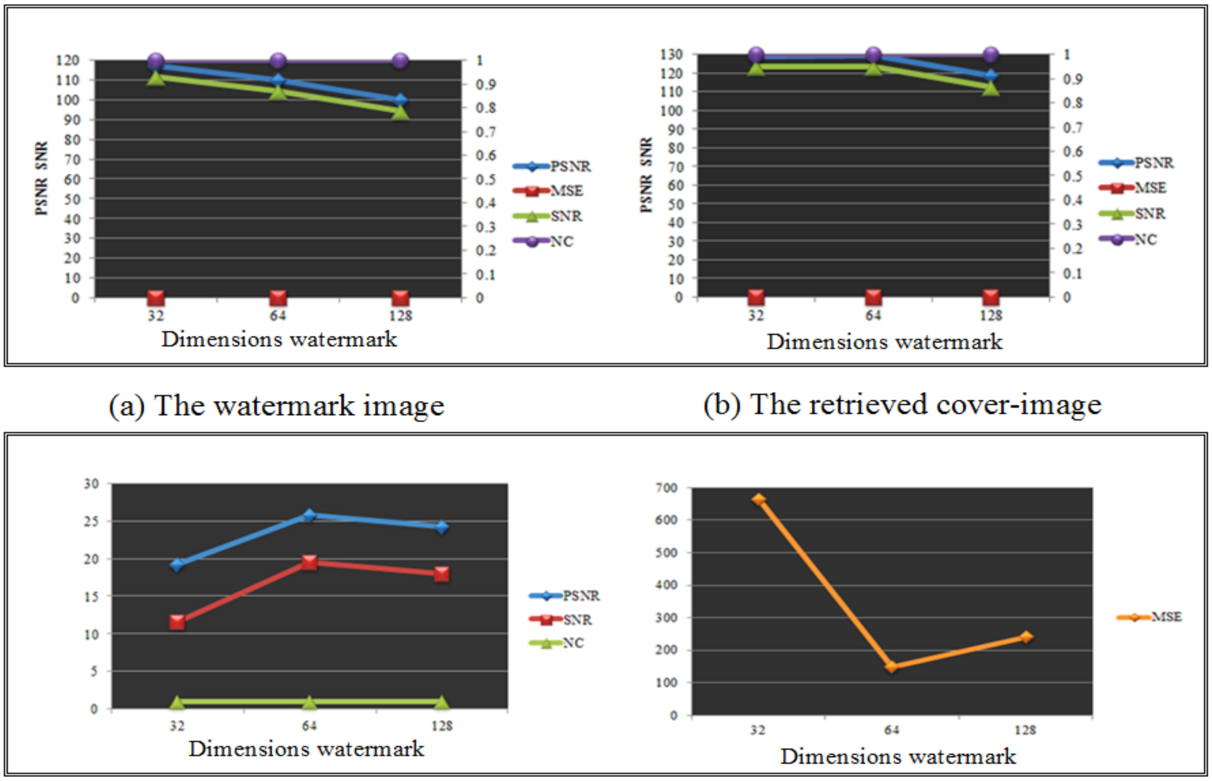
Cover image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the watermark-image			
				MSE	PSNR	NC	SNR
Lena	1024×1024	Part From Cameraman	32×32	0.000000	117.259945	1.000000	111.456693
Lena	1024×1024	Part From Cameraman	64×64	0.000001	106.926393	1.000000	101.123141
Lena	1024×1024	Monaliza	128×128	0.000005	99.999840	1.000000	94.196587

Table 5.6b: The Results of Extracting the Different-Dimension Watermarks

Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the extracted watermark			
				MSE	PSNR	NC	SNR
Lena	1024×1024	Part From Cameraman	32×32	662.538086	19.282858	0.988450	11.534974
Lena	1024×1024	Part From Cameraman	64×64	147.159132	25.853667	0.990976	19.471501
Lena	1024×1024	Monaliza	128×128	240.738657	24.315345	0.993098	18.039247

Table 5.6c: The Results of Retrieved the Cover-Image

Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of the retrieved cover-image			
				MSE	PSNR	NC	SNR
Lena	1024×1024	Part From Cameraman	32×32	0.000000	129.104338	1.000000	123.301086
Lena	1024×1024	Part From Cameraman	64×64	0.000000	129.217571	1.000000	123.414319
Lena	1024×1024	Monaliza	128×128	0.000000	118.525426	1.000000	112.722173



(a) The watermark image (b) The retrieved cover-image

(c) The extracted watermark

Figure 5.6c: A graph that shows the performance efficiency measures when embedding different dimension watermarks in the cover-image "Lena" with dimensions 1024×1024

Table 5.7 shows the degree of solidness of the third algorithm the second part in the research using the standard correlation coefficient scale (NC) between the images of the watermark cover-image with the size 1024×1024 after embedding the image "Flower.png" and the watermark images after submitting the to speckle noise and the degree of the noise density was 0.001. Table 5.7a is a graph of the table of the results of the watermark subjected to the noise. But Figure 5.7b is the image of the watermark "Lena" subjected to the noise. While Table 5.8 is a comparison between the results of the third algorithm used in the research with the results submitted by several researchers who used other algorithms to embed the watermark in the grey images.

Table 5.7a: A Graph of the Table that shows the Results of the Standard Correlation Coefficient of the Watermark-Image

Salt &pepper nois 0.001 nois density	**Speckle noice 0.001 density**	**Poisson noice**	**Gaussian noic density**	**Images**
0.9954	0.9965	0.9746	0.9852	Barbara
0.9958	0.9974	0.9790	0.9860	Lena
0.9970	0.9966	0.9795	0.9887	Peppers
0.9959	0.9957	0.9715	0.9853	Baboon
0.9963	0.9992	0.9943	0.9923	Rose
0.9941	0.9946	0.9624	0.9796	Couple
0.9956	0.9951	0.9693	0.9849	Elaine
0.9957	0.9956	0.9713	0.9853	Boat

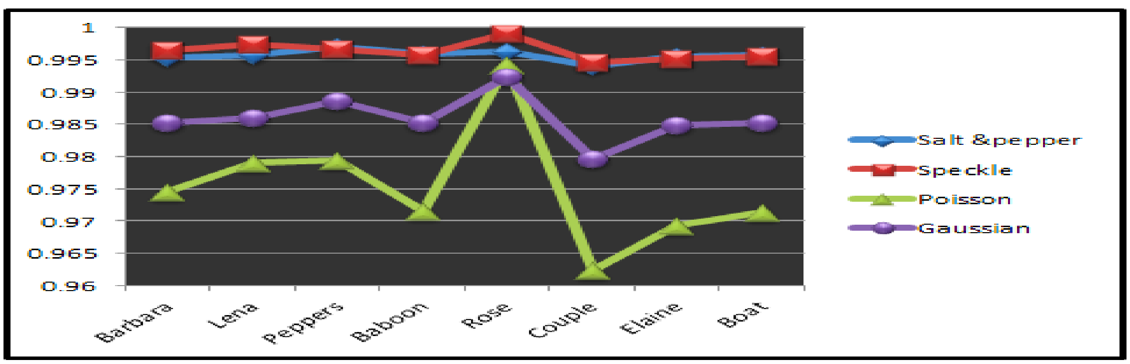


Figure 5.7a: Results of the Standard Correlation Coefficient of the Watermark-Image Under the Light Processing

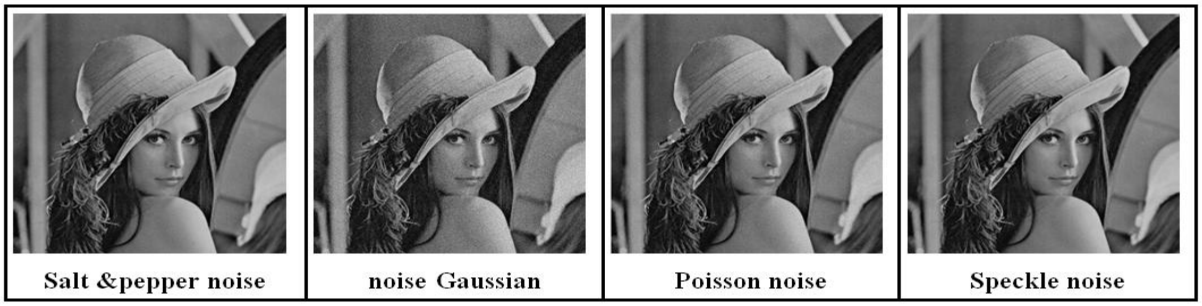


Figure 5.7b: Watermark-Image (Lena) Under the Light Processes

Table 5.8: A Comparison between the Results of the Third Algorithm Adopted in the Research and the Results of Other Researches

Measure	Method	Tested Image (Cover)						
		Peppers	**Lena**	**Elaine**	**Couple**	**Boat**	**Barbara**	**Baboon**
PSNR	CTL.[24]	36.7585	46.9681	45.7083	37.0754	36.7234	36.7178	37.0757
	DWT and SVD[11]	……	109.5129	……	109.5140	……	…….	109.5099
	APAP-MIPOEE applying filling-of-boundary in corners board set-of-MSB_6[9]	……	53.2994	……	……	51.8972	……	52.4855
	Non-Blind CTL.[18]	46.39	46.86	……	……	……	46.87	45.99
	Proposed method\3rd	96.872780	107.309568	91.560678	108.872780	101.872780	108.309934	107.735451
MSE	APAP-MIPOEE applying filling-of-boundary in corners board set-of-MSB_6	……	0.3041	……	……	0.4201	……	0.3668
	Proposed method\3rd	0.000011	0.000001	0.000033	0.000001	0.000004	0.000001	0.000001
SNR	APAP-MIPOEE applying filling-of-boundary in corners board set-of-MSB_6	……	53.2994	……	……	51.8972	……	46.8265
	Proposed method\3rd	88.074993	101.506316	80.901897	102.979408	95.526837	99.541471	101.480248
NC	CTL.	0.985308	0.986253	0.987072	0.986388	0.985742	0.983328	0.986389
	Proposed method\3rd	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

From discussing the experiments applied to the algorithms employed in the research and the models adopted in applying these algorithms, the researcher reached to the following conclusions:

- The principle of data distribution in the contourlet transformations coefficients results in complete secrecy of the data required to be hidden, where the cover-image in which the watermark was embedded was very close to the original image and that makes it very difficult for the observer to notice that it exists inside the cover-image. The scales of the watermark algorithm efficiency were employed to evaluate the degree of approximation between the original image and the watermark image.
- The high embedding capacity in hiding the data when using the contourlet transformations because this method provides coefficients for the analysis levels with various dimensions that depend on the size of the cover and the number of the analysis levels.
- The relation between the cover-image and the size of the watermark is directly related, and the relationship between the clearness of the extracted watermark with the size of the cover-image is directly related, as sharpness increases with the increase of the cover size when the watermark is constant. And the relation will be inversely related between the size of the watermark and sharpness when the size of the cover-image is constant.
- Depending on the principle of energy amount used in the second algorithm in selecting the factors that distributed the data to be hidden the watermark has given high solidness to the embedding algorithm and high embedding capacity without causing any distortion in the cover-image and also gave a high resolution to the watermark extracted.

- Using the tree distribution principle for the cover-image and the watermark led to very good results in terms of the strength of embedding and high capacity where the watermark extracted was very close to the original watermark.
- The third algorithm needs less time and less strength to resist the noise.

6.2 Recommendations

- The watermark technique could be used with the neurological networks to hide certain types of marks.
- Developing the watermark technique and adopting the genetic algorithm in embedding the secret key.
- The watermark is considered one of the individual vital factors that identify each person such as the eye iris, fingerprint and other factors that can be planted inside the cover-image to meet secret purposes.
- Using the contourlet transformations for the purpose of complete hiding of compressed files after recovering the changes in the compressed information.
- Using the contourlet transformations to embed the watermark using the public-key.
- The possibility of finding FCLT or new algorithms to shorten the time required for calculating CLT and ICLT.
- Studying the noise concerning the communication channel.

REFERENCES

Ahire, V. K., & Kshirsagar, V. (2011). Robust watermarking scheme based on discrete wavelet transform (DWT) and discrete cosine transform (DCT) for copyright protection of digital images. *International Journal of Computer Science and Network Security*, *11*(8), 208-213.

Alam, F. I., Bappee, F. K., & Khondker, F. U. A. (2011). An investigation into encrypted message hiding through images using LSB. *International Journal of Engineering Science and Technology*, *3*(2), 948-960.

Aliwa, M. B., El-Tobely, T. E. A., Fahmy, M. M., Nasr, M. E. S., & El-Aziz, M. H. A. (2010). Fidelity and robust digital watermarking adaptively pixel based on medial pyramid of embedding error gray scale images. *International Journal of Computer Science and Network Security*, *10*(6), 284-314.

Álvarez, V., Armario, J. A., Frau, M. D., Gudiel, F., Güemes, M. B., Martín, E., & Osuna, A. (2012). GA based robust blind digital watermarking. *Journal arXiv Mathematic Aplications 12(54)*, 1-12

Ardabili, E. R., Maghooli, K., & Fatemizadeh, E. (2011). Contourlet features extraction and adaboost classification for palmprint verification. *Journal of American Science*, *7*(7), 353-362.

Arun, K. S., & Menon, H. P. (2009). Content based medical image retrieval by combining rotation invariant contourlet features and Fourier descriptors. *International Journal of Recent Trends in Engineering*, *2*(2), 35-39.

Bedi, S. S., Verma, S., & Tomar, G. (2010). An adaptive data hiding technique for digital image authentication. *International Journal of Computer Theory and Engineering*, *2*(3), 338-344.

Castiglione, A., D'Alessio, B., De Santis, A., & Palmieri, F. (2010). Hiding information into OOXML documents: new steganographic perspectives. *Journal Wireless Mobile Network. Ubiquitous Computer Dependable Aplications*, *2*(4), 59-83.

Chandra, A., & Kavitha, M. T. (2012). Robust digital watermarking using MDKP. *International Journal of Scientific and Research Publications*, *2*(6), 1-4.

Chandra, M., Pandey, S., & Chaudhary, R. (2010, July). Digital watermarking technique forprotecting digital images. *In Proceeding of International Conference on the Computer Science and Information Technology, IEEE,* (Vol. 7, pp. 226-233). Chengdu: IEEE, Computer Society.

Chen, G. Y., & Kégl, B. (2010). Invariant pattern recognition using contourlets and AdaBoost. *Journal of science direct, Pattern Recognition*, *43*(3), 579-583.

Das, S., & Kundu, M. K. (2011). Hybrid contourlet-DCT based robust image watermarking technique applied to medical data management. *In Proceeding of 4th International Conference on Pattern Recognition and Machine Intelligence, Springer,* (pp.286-292). Berlin Heidelberg: Springer, Computer Society.

Devanna, H., & Kumar, G. A. E. S. (2011). Performance analysis of modified nonsubsampled contourlet transform for image denoising. *Research Journal of Applied Science Engineering and Technology, 3*(7), 589-595.

Do, M. N. (2001). *Directional multiresolution image representations*, Ph. D thesis, *Federal Institute of Technology*, Lausanne, Swiss, Switzerland.

Dua, T. D. S. (2012). Digital watermarking: digital data hiding techniques for BMP images. *Global Journal of Computer Science and Technology*, *12*(1), 433-445.

Dukhi, R. G. (2011, April). Watermarking: A copyright protection tool. *In Proceeding of International Conference on Electronics Computer Technology*, (pp. 36-41). Kanyakumari: IEEE, Computer Society.

Gao, B., Sha, B., Zhang, Y., & Bi, H. (2012). HVS-Based image compression scheme in wavelet-contourlet domain. *Advances in Information Sciences & Service Sciences*, *4*(2),61-64.

Goriac, I. (2011, August). An epistemic logic based framework for reasoning about information hiding. *In Proceeding of International Conference on Availability, Reliability and Security, 2011 Sixth*, (pp. 286-293). Vienna: IEEE, Computer Society.

Goudar, R., & Edekar, S. (2011). Covert channels: Emerged in mystery and departed in confusion. *International Journal of Computer Science and Network Security*, *11*(11), 34-37.

Guyeux, C., & Bahi, J. M. (2010). An improved watermarking algorithm for internet applications. *In Proceeding of International Conference on Evolving Internet.* (pp.119-124). *Valencia, Spain: IEEE,* Computer Society.

Hiremath, P. S., Akkasaligar, P. T., & Badiger, S. (2011). Performance comparison of wavelet transform and contourlet transform based methods for despeckling medical ultrasound images. *International Journal of Computer Applications*, *26*(9), 34-41.

Huang, H. C., & Fang, W. C. (2010). Techniques and applications of intelligent multimedia data hiding. *Telecommunication Systems*, *44*(3-4), 241-251.

Indra, C., & Ramaraj, E. (2011). A self signature embedding for image security by using similarity based superior self organizing maps neural network. *International Journal of Advancements in Technology*, *3*(1), 15-24.

Isac, B., & Santhi, V. (2011). A study on digital image and video watermarking schemes using neural networks. *International Journal of Computer Applications*, *12*(9), 1-6.

janardhanaRao, S. and Rameshbabu, K. (2012). Implementation of contourlet transform for copyright protection of color images. *International Journal of Engineering Research and Applications, 2*(2), 968-973.

Kamble, M. V. (2011). Information hiding technology: A Watermarking. *Advances in Computational Research, 3*(1), 37-41.

Kamble, S., Maheshkar, V., Agarwal, S., & Srivastava, V. K. (2012). Dwt-svd based secured image watermarking for copyright protection using visual cryptography. In *Proceedings of Computer Science and Information Technology-Computer Science*, *4*(2), 143-150.

Katariya, S. S. (2012). Digital Watermarking: Review. *International Journal of Engineering and Innovative Technology*, *1*(2), 143-153.

Kaviani, H. R., Karimi, N., & Samavi, S. (2011). Robust watermarking in singular values of contourlet coefficients. *In Proceeding of 7th Iranian International Conference on Machine Vision and Image Processing*, (pp.1-5). Tehran: IEEE, Computer Society.

Khalil, M. I. (2011). Image steganography : hiding short audio messages within digital images. *Journal of Computer Science & Technology*, *11*(2), 68-73.

Khanzode, P., Ladhake, S., & Tank, S. (2011). Digital watermarking for protection of intellectual property. *International Journal of Computational Engineering & Management*, *12*(1), 8-12.

Kumar, A., & Santhi, V. (2011). A review on geometric invariant digital image watermarking techniques. *International Journal of Computer Applications*, *12*(9), 31-36.

Kumar, M., Manikandan, T., & Sapthagirivasan, V. (2011). Non-blind image watermarking using contourlet transform. *Indian Journal of Computer Science and Engineering*, *2*(1), 31-38.

Kumar, N. M., Manikandan, T., & Sapthagirivasan, V. (2011). Non blind image watermarking based on similarity in contourlet domain. *In Proceeding of International Conference on Recent Trends in Information Technology*. (pp.1277-1282). Chennai, Tamil Nadu: IEEE, Computer Society.

Kumar, S., Kumar, S., & Nandi, S. (2011). Stage Staffing Scheme for Copyright Protection MultImedia. *International Journal of Network Security & Its Applications*, *3*(2),135-147.

Kushwaha, S., & Singh, H. V. (2011). A review of digital watermarking. *National Conference on Electronics, Communication and Signal Processing*, (pp.134-140).

Lee, C. W., & Tsai, W. H. (2011). A lossless large-volume data hiding method based on histogram shifting using an optimal hierarchical block division scheme. *Journal of Information Science and Engineering*, *27*(4), 1265-1282.

Loukhaoukha, K., Chouinard, J. Y. & Taieb, M. H. (2011). Optimal image watermarking algorithm based on LWT-SVD via multi-objective ant colony optimization. *Journal of Information Hiding and Multimedia Signal Processing*, *2*(4), 303-318.

Lydia, M., Karuna, M. & Prasad, T. D., (2011). Image denoising using contourlet transform with application to synthetic aperture radar. *International Journal of Engineering Research and Applications*, *1*(4), 2030-2034.

Majumder, S., Saikia, M., Das, T. S. & Sarkar, S. K. (2011). Hybrid image watermarking scheme using SVD and PDFB based contourlet transform. *In Proceeding of International Conference on Computer and Communication Technology*, *17*(7), 130-134. Allahabad: IEEE, Computer Society.

Manoharan, J. S., Vijila, K. C. & Sathesh, A. (2010). Performance analysis of spatial and frequency domain multiple data embedding techniques towards Geometric attacks. *International Journal of Security*, *4*(3), 28-37.

Meskine, F., Mezouar, M. C. E. & Taleb, N. (2010). A Rigid image registration based on the nonsubsampled contourlet transform and genetic algorithms. *Sensors*, *10*(9), 8553-8571.

Mohamed Sathik, M., & Sujatha, S. S. (2010). An improved invisible watermarking technique for image authentication. *International Journal of Advanced Science & Technology*, *24*(11), 61-74.

Mohan, B. C., & Kumar, S. S. (2008). Robust digital watermarking scheme using contourlet transform. *International journal of computer science and network security*, *8*(2), 43-51.

Mohan, M. & Anurenjan, P. R. (2011). A new algorithm for data hiding in images using contourlet transform. *Recent Advances in Intelligent Computational Systems*, *6*(3), 411-415. IEEE.

Morsy, H. A., Nossair, Z. B., Hamdy, A. M., & Amer, F. Z. (2012). JPEG Steganography System with Minimal Changes to the Quantized DCT Coefficients. *International Journal of Soft Computing and Engineering*,*1*(6), 28-33.

Mynatty, J. G. & Jayageetha, J. (2012). An optimal data hiding scheme with block-based parity check for binary, gray and colour images. *International Journal of Advanced Research in Technology*, *2*(1)**,** 11-19.

Narasimhulu, C. V., & Prasad, K. S. (2011). A robust watermarking technique based on nonsubsampled contourlet transform and SVD. *International Journal of Computer Applications*, *16*(8), 27-36.

Nyeem, H., Boles, W., & Boyd, C. (2011). Developing a digital image watermarking model. *In Proceeding of International Conference on Digital Image Computing Techniques and Applications*, (pp.468-473). Noosa, QLD: IEEE, Computer Society.

Parthiban, V., & Ganesan, R. (2012). Hybrid watermarking scheme for digital images. *Journal of Computer Applications*, *5*(1), 85-95.

Patel, R., Shrawankar, U., & Thakare, V. M. (2013). Secure Transmission of Password Using Speech Watermarking. *International Journal of Computer Science and Technology, 2(3), 315-318.*

Perwej, Y., Parwej, F., & Perwej, A. (2012). An adaptive watermarking technique for the copyright of digital images and Digital Image Protection. *International Journal of Multimedia & Its Applications*, *4*(2), 21-38.

Pun, C.-M., & Lam, I.-T. (2009). Fingerprint watermark embedding by discrete cosine transform for copyright ownership authentication. *International Journal of Communication*, *3*(1), 17-24.

Rahimi, F. & Rabani, H. (2010). A visually imperceptible and robust image watermarking scheme in contourlet domain. *International Conference Signal Processing, 10(1)*, 1817-1820. IEEE Computer Society.

Rahimi, F. & Rabbani, H. (2011). A dual adaptive watermarking scheme in contourlet domain for DICOM images. *Biomedical engineering online*, *10*(1), 1-18.

Raj, Y. A. & Alli, P. (2012). An Analysis and Overview of Modern Digital Watermarking. *American Journal of Applied Sciences*, *9*(1), 66-70.

Ramana, K. V., Babu, B. R. & Babu, S. C. (2011). A randomized secure data hiding algorithm using file hybridization for information security. *International Journal on Computer Science & Engineering*, *3*(5), 1878-1889.

Rubinstein, R., Bruckstein, A. M., & Elad, M. (2010). Dictionaries for sparse representation modeling. *In Proceedings of IEEE Applications of Sparse Representation & Compressive Sensing*, *98*(6), 1045-1057.

Saha, S., Bhattacharyya, D., & Bandyopadhyay, S. K. (2010). Security on Fragile and Semi-fragile Watermarks Authentication. *International Journal of Computer Applications*, *3*(4), 23-27.

Sandhya, G. (2011). Multiscale and directional transform for denoising medical images. *International Journal of Engineering Research and Applications, 1*(4), 2084-2090.

Saraswathi, M. (2011). Lossless Visible Watermarking for Video. *International Journal of Computer Science and Information Technologies, 2*(3), 1109-1113.

Sarbavidya, S. (2011). Image Watermarking: A Survey. *Journal of Global Research in Computer Science*, *2*(8), 38-41.

Satheesh, S., & Prasad, K. V. S. V. R. (2011). Medical image denoising using adaptive threshold based on contourlet transform. *Advanced Computing*: *An International Journal, 2*(2), 52-58.

Shan, H., Ma, J., & Yang, H. (2009). Comparisons of wavelets, contourlets and curvelets in seismic denoising. *Journal of Applied Geophysics*, *69*(2), 103-115.

Sikarwar, N. S. (2010) Multilevel Steganography: An Enhancement in Steganography and Comparative Study with Current Tools. *International Journal of Computer and Network Security*, *2*(1), 25-29.

Singh, V. (2011). Digital watermarking: a tutorial. *Cyber Journals: Multidisciplinary. Journals in Science and Technology, Journal of Selected Areas in Telecommunication, 11*(4), 10-21.

Sirvan, K., Parisa, T., & Hamid R, R. (2010). A contourlet-based image watermarking scheme with high resistance to removal and geometrical attacks. *Journal on Advances in Signal Processing*, *7*(1), 1-13.

Soleymanpour, E., Rajae, B., & Pourreza, HR (2010, October). Offline handwritten signature identification and verification using contourlet transform and Support Vector Machine. *Machine Vision and Image Processing*, *9*(7), 1-6.

Surekha, B., Swamy, G., & Rao, K. S. (2010). A multiple watermarking technique for images based on visual cryptography. *International Journal of Computer Applications*, *1*(11), 78-82.

Tai, W.-L., & Chang, C.-C, (2009). Data hiding based on vq compressed images using hamming codes and declustering. *International Journal of Innovative Computing, Information and Control*, *5*(7), 2043-2052.

Tamilarasi, M., & Palanisamy, V. (2011). An efficient embedded coding for medical image compression using contourlet transform. *European Journal of Scientific Research*, *49*(3), 442-454.

Tamilarasi, M., & Palanisamy, V. (2011). Medical image compression using fuzzy c-means based contourlet transform. *Journal of Computer Science*, *7*(9), 1386-1392.

Tanaka, Y., Ikehara, M., & Nguyen, T. Q. (2009). Multiresolution image representation using combined 2-D and 1-D directional filter banks. , *IEEE Transactions on Image Processing*, *18*(2), 269-280.

Tawade, L., Mahajan, R., & Kulthe, C. (2012). Efficient & secure data hiding using secret reference matrix. *International Journal of Network Security & Its Applications*, *4*(1), 41-50.

Tewari, T. K., & Saxena, V. (2010). An Improved and Robust DCT based Digital Image Watermarking Scheme. *International Journal of Computer Applications*, *3*(1), 28-32.

Truc, P. T., Khan, M. A., Lee, Y. K., Lee, S., & Kim, T. S. (2009). Vessel enhancement filter using directional filter bank. *Computer Vision and Image Understanding*, *113*(1), 101-112.

Umaamaheshvari, A., & Thanushkodi, K. (2012). High performance and Effective watermarking scheme for medical images. *European Journal of Scientific Research*, *67*(2), 283-293.

Vidhyalakshmi, M. & Vennila, G., (2012). Optimum Detecting of Contourlet-Based Image Watermarks in a Noisy Environment. *Indian Journal. Innovations*, *1*(6), 406-411.

Wu, C., Feng, N., Harada, K., & Li, P. (2012). A hybrid de-noising method on lasca images of blood vessels. *Journal of Signal & Information Processing*, *3*(1), 92-97.

Wu, J., Lee, D., Jiao, L., & Shi, G. (2009). Laplacian pyramid-based downsampling combined with directionlet interpolation for low-bit rate compression. *In Proceedings of Sixth International Conference on Multispectral Image Processing and Pattern Recognition*, *7498*(1), 634-641. Yichang, China: SPIE, Computer Society.

Xingmei, L., Guoping, Y., & Liang, C. (2009). A new method of image denoise using contourlet transform. *International Journal of Intelligent Information Technology Application*, *3*(1), 25-30.

Yang, G., Fang, X., Jing, M., Zhang, S., & Hou, M. (2010). Contourlet filter design based on chebyshev best uniform approximation. *Journal on Advances in Signal Processing*, *32*(10), 33-38.

Yershov, A., & Rusakov, P. (2010). Universal Stegoconstructor in Context of Intellectual Property Protection. *Scientific Journal of Riga Technical University*. *Computer Sciences*, *41*(1), 131-136.

Zhao, S., & Hengjian, L. I. (2011). Image compression coding based on multi-orientation sparse image representation. *Journal of Computational Information Systems*, *7*(6), 2063-2070.

APPENDICES

APPENDIX A

APPLIED EXAMPLE

1- First Algorithm: Embedding Using the Contourlet Transformation Coefficients can be Seen in Figure 1a and Tables 1a to 3a.

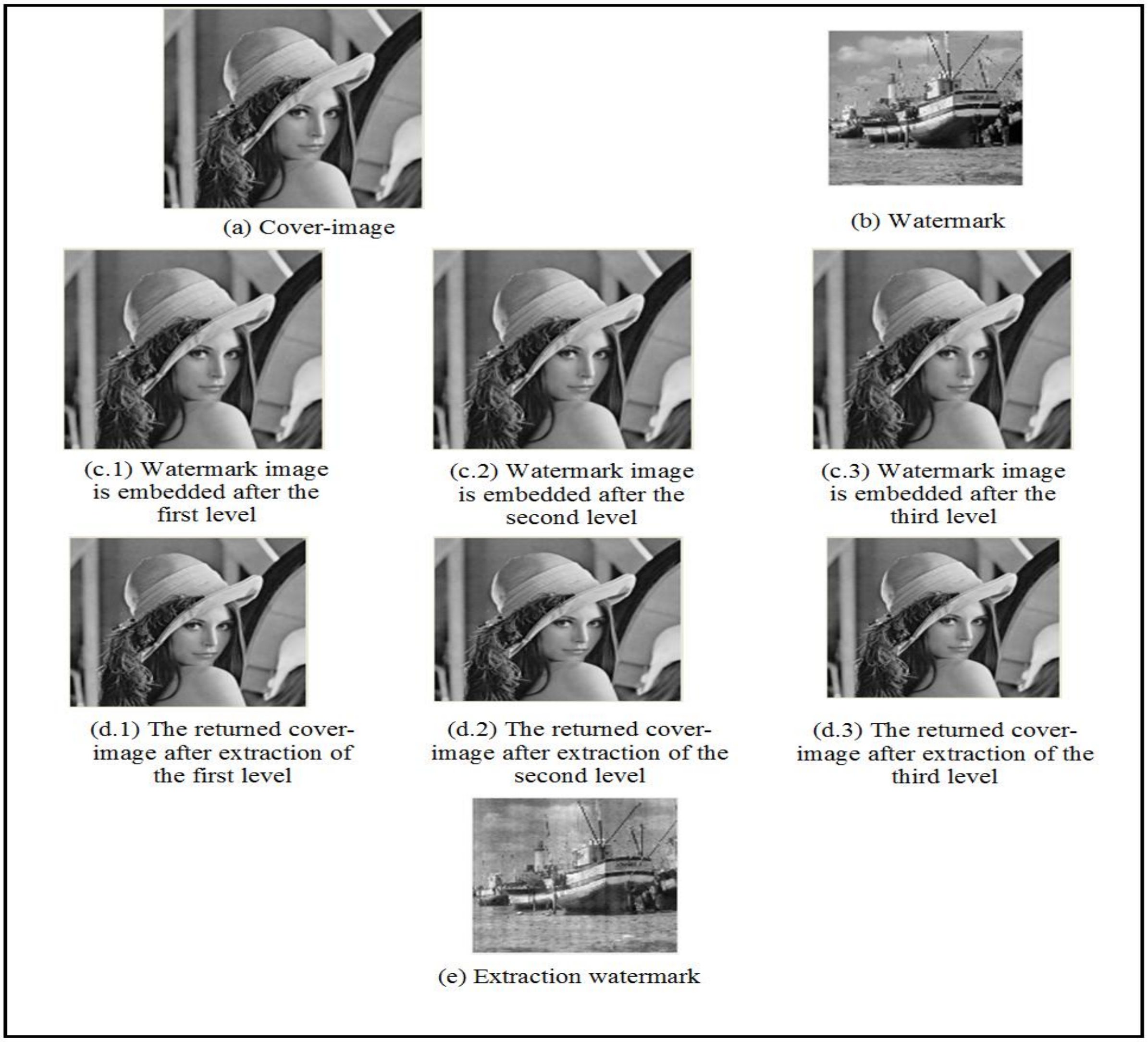


Figure 1a: Simple Embedding and Extraction Secret Image (Boat.png) of Dimension 256×256 in Cover-Image (lena.png) 1024×1024

Table 1a: Results Embedding Watermark in Cover-Image

Cover Image	Cover image size	Watermark	Size of the watermark	Contourlet coefficients	Parameter measure for efficiency performance of the watermark-image			
					MSE	**PSNR**	**NC**	**SNR**
Lina	1024×1024	Part From Lina	32×32	{1,1}	0.000012	95.783943	1.000000	89.980691
				{1,2}	0.000001	108.095608	1.000000	102.292355
			64 ×64	{1,1}	0.000027	92.305039	1.000000	86.501787
				{1,2}	0.000002	104.366192	1.000000	98.562940
				{1,3}	0.000002	103.542531	1.000000	97.739279
		Boat	256 ×256	{1,1}	0.000837	77.339962	1.000000	71.536709
				{1,2}	0.000017	94.377865	1.000000	88.574613
				{1,3}	0.000009	96.956855	1.000000	91.153603
				{1,4}	0.000007	98.100176	1.000000	92.296924
				{1,5}	0.000007	98.081491	1.000000	92.278239

Table 2a: Results Extracted Watermark According First Algorithm

Cover Image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of performance for retrieved the watermark			
				MSE	**PSNR**	**NC**	**SNR**
Lena	1024 × 1024	Part From Cameraman	32×32	1457.489036	16.076073	0.958438	10.549189
			64×64	723.817211	18.861951	0.966683	10.754324
		Part from Boat	256 × 256	1166.323299	17.008118	0.931966	12.099350

Table 3a: Results of Retrieved Cover-Image According First Algorithm

Cover Image	Cover image size	Watermark	Size of the watermark	Contourlet coefficients	Parameter measure for efficiency performance of the retrieved cover-image			
					MSE	PSNR	NC	SNR
Lina	1024×1024	Part From Lina	32×32	{1,1}	0.000001	106.915430	1.000000	101.112178
				{1,2}	0.000000	120.383881	1.000000	114.580629
			64 ×64	{1,1}	0.000002	104.643791	1.000000	98.840539
				{1,2}	0.000000	115.983457	1.000000	110.180205
				{1,3}	0.000000	115.698559	1.000000	109.895307
		Boat	256 ×256	{1,1}	0.000048	89.794368	1.000000	83.991115
				{1,2}	0.000001	107.546801	1.000000	101.743548
				{1,3}	0.000001	109.392819	1.000000	103.589567
				{1,4}	0.000000	109.968187	1.000000	104.164935
				{1,5}	0.000001	109.285599	1.000000	103.482347

2- Second Algorithm: Embedding Using Energy can be seen in Figure 2b and Tables 1b to 3b.

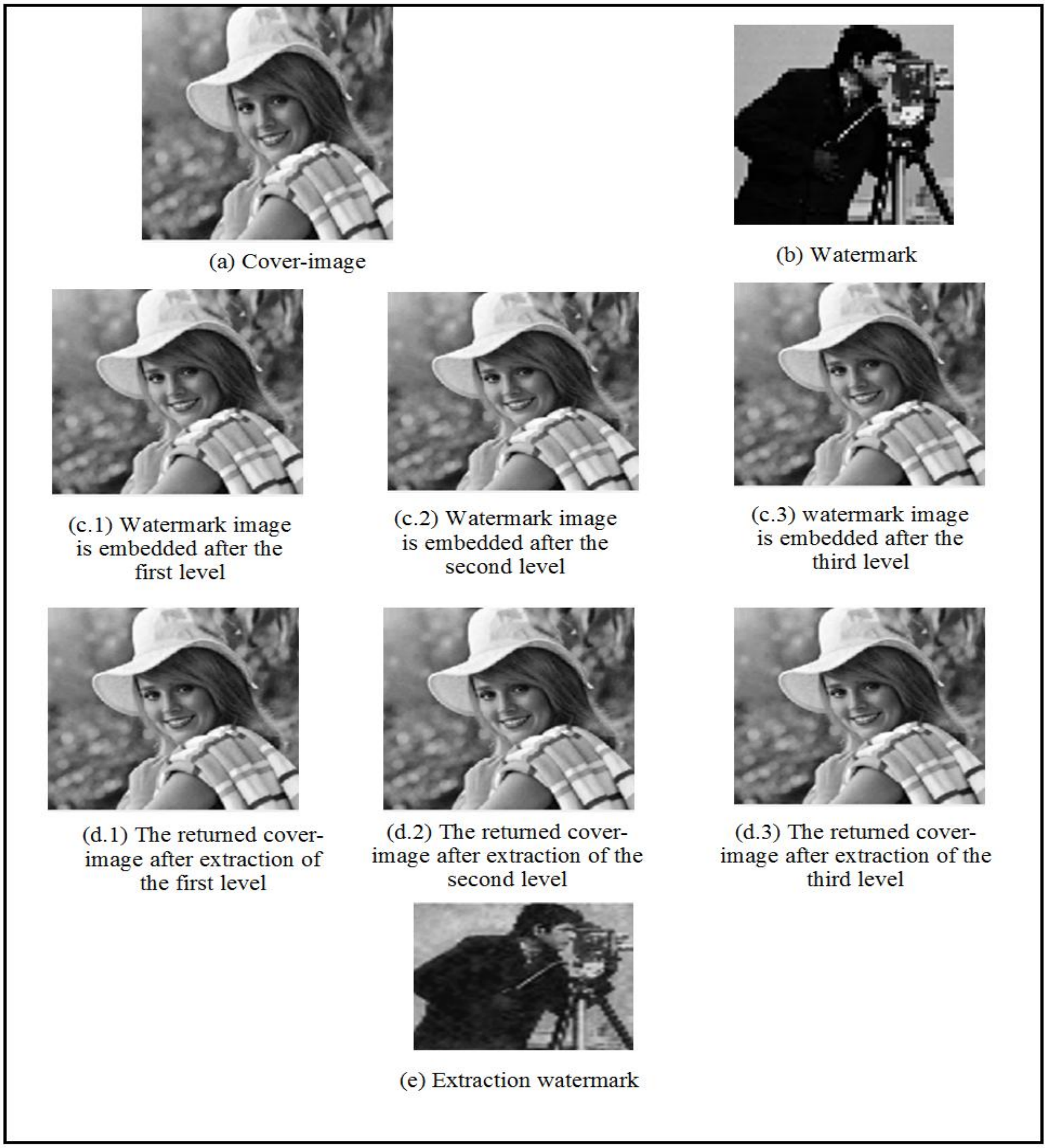


Figure 2b: Simple Embedding and Extraction Secret Image (Cameraman.bmp) of Dimension 64×64 in Cover-Image (Elaine.bmp) 256×256

Table 1b: Results Embedding Watermark in Cover-Image can be Seen in Figure 4.20

Cover image	Cover image size	Watermark	Size of the watermark	Contourlet coefficients	Parameter measure for efficiency of performance for watermark			
					MSE	PSNR	NC	SNR
Elaine	256×256	Part From Lena	32×32	{1,1}	0.000003	102.680743	1.000000	98.152121
				{1,2}	0.000000	123.806465	1.000000	119.277843
		Part From Camera-man	64×64	{1,1}	0.000025	93.762059	1.000000	89.233437
				{1,2}	0.000000	112.150538	1.000000	107.621916
				{1,3}	0.000000	111.759178	1.000000	107.230557

Table 2b: Results Extracted Watermark According Second Algorithm

Cover image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency of performance for extracted watermark-image			
				MSE	PSNR	NC	SNR
Elaine	256 ×256	Part From Lena	32×32	39.797823	27.465132	0.987825	19.067624
		Part from cameraman	64×64	255.526338	23.383883	0.985504	16.320602

Table 3b: Results of Retrieved Cover-Image According Second Algorithm

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Contourlet coefficients**	**Parameter measure for efficiency performance of the retrieved cover-image**			
					MSE	**PSNR**	**NC**	**SNR**
Elaine	256×256	Part From Lena	32×32	{1,1}	0.000000	127.829991	1.000000	123.301370
				{1,2}	0.000000	134.039956	1.000000	129.511335
		Part From Camera-man	64×64	{1,1}	0.000000	117.207878	1.000000	112.679257
				{1,2}	0.000000	112.150538	1.000000	107.621916
				{1,3}	0.000000	123.256491	1.000000	118.727870

3-Third algorithm: Embedding Depending on the Contourlet Transforms Coefficient and Energy.

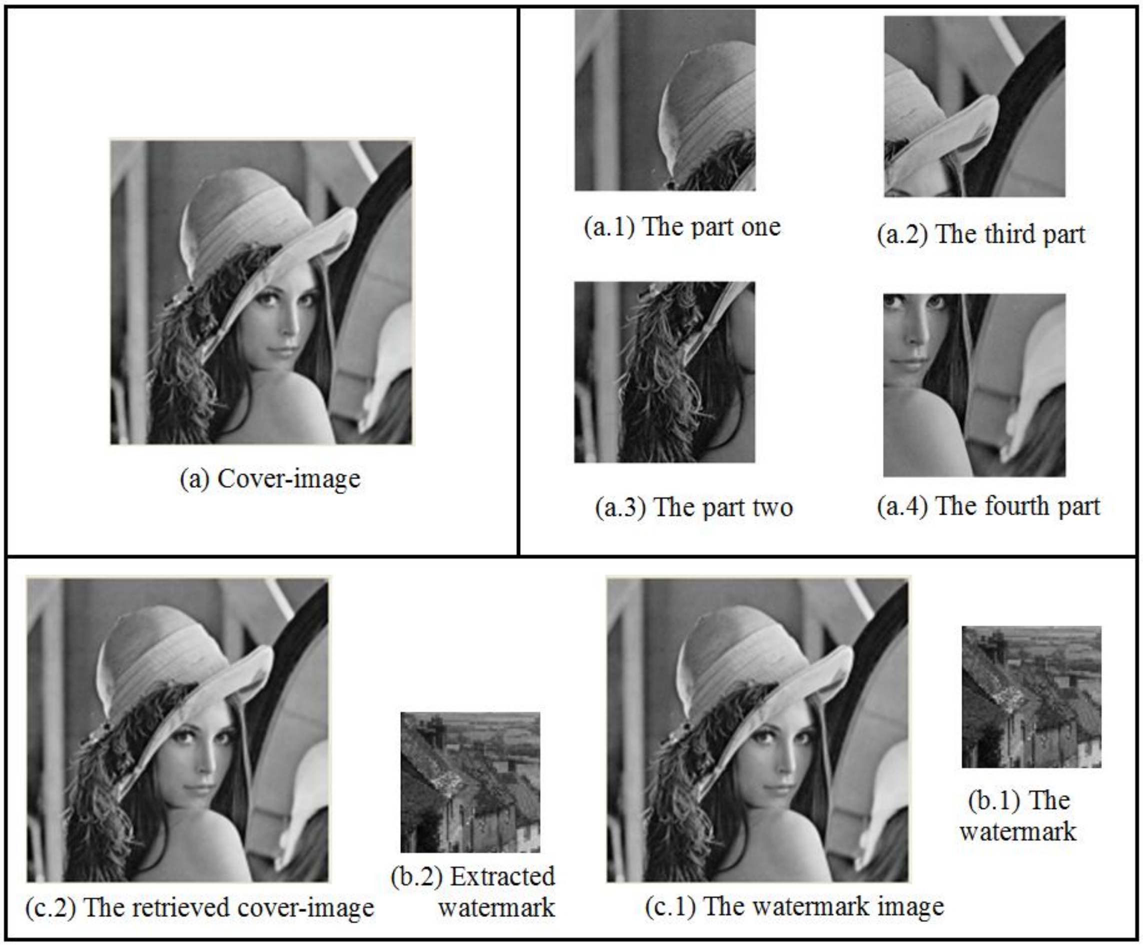


Figure c1: Simple Embedding and Extraction Secret Image (Goldhill.gif) of Dimension 64×64 in Cover-Image (Lena.png) 512×512

Table 1c: Results Embedding Watermark in Third Algorithm Part 2

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of the watermark image**			
Lena	**1024×1024**	**Part From Goldhill**	**64×64**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	32×32	0.000000	113.677538	1.000000	107.759747
Part2	512×512	Part2	32×32	0.000002	103.238653	1.000000	95.180903
Part3	512×512	Part3	32×32	0.000000	112.643817	1.000000	107.829174
Part4	512×512	Part4	32×32	0.000001	107.184358	1.000000	102.641952
After Re-assemble parts watermark				0.000001	107.292996	1.000000	101.489744

Table 2c: Results Extracted Watermark in Third Algorithm Part 2

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of performance for retrieved the watermark**			
Lena	**1024×1024**	**Part from Goldhill**	**64×64**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	32×32	57.819454	27.862417	0.986599	21.624455
Part2	512×512	Part2	32×32	110.385351	25.763489	0.977273	18.387454
Part3	512×512	Part3	32×32	61.838651	27.430834	0.985665	22.792798
Part4	512×512	Part4	32×32	70.833959	27.561506	0.985136	21.811901
After assembling four parts				75.219354	27.429307	0.983716	21.074498

Table 3c: Results of Retrieved Cover-Image in Third Algorithm part 2

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of performance of the cover image**			
Lena	**1024×1024**	**Part from Goldhill**	**64×64**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	512×512	Part1	32×32	0.000000	134.131297	1.000000	128.213505
Part2	512×512	Part2	32×32	0.000000	127.785183	1.000000	119.727433
Part3	512×512	Part3	32×32	0.000000	134.329803	1.000000	129.515159
Part4	512×512	Part4	32×32	0.000000	131.149724	1.000000	126.607318
After re-assemble parts retrieved cover-image				0.000000	131.098052	1.000000	125.294800

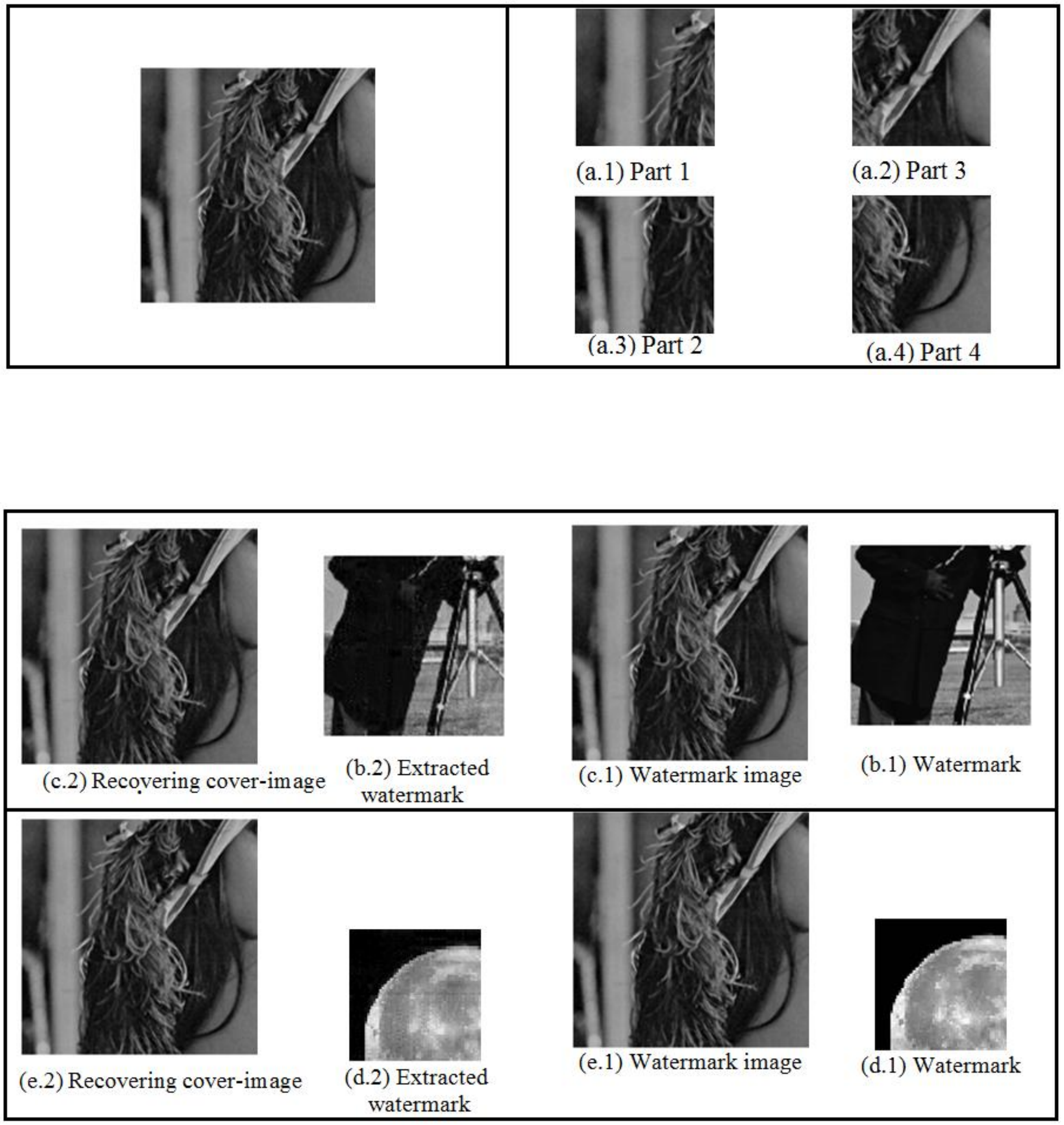


Figure 4a: Simple Embedding and Extraction Secret Image (Cameraman.tif) of Dimension 128×128 and Secret Image (Moon.bmp) of Dimension 64×64 in Parts of the Second Section in Cover-Image (Lena.png) of Dimension 256×256

Table 1d: Results of Retrieved Cover-Image in Third Algorithm Part 2

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of performance for retrieved watermark**			
Part2 Lena	**512×512**	**Part From Moon**	**64×64**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	256×256	Part1	32×32	0.000002	102.640255	1.000000	95.397549
Part2	256×256	Part2	32×32	0.000020	92.861751	1.000000	85.970782
Part3	256×256	Part3	32×32	0.000002	103.467338	1.000000	95.612228
Part4	256×256	Part4	32×32	0.000002	102.668018	1.000000	93.421814
After re-assemble parts watermark-image				0.000007	98.289908	1.000000	90.232158
Part2 Lena	**512×512**	**Part From Cameraman**	**128×128**				
Part1	256×256	Part1	64×64	0.000002	104.311053	1.000000	97.068347
Part2	256×256	Part2	64×64	0.000001	105.474826	1.000000	98.583857
Part3	256×256	Part3	64×64	0.000018	93.527085	1.000000	85.671975
Part4	256×256	Part4	64×64	0.000013	95.276712	1.000000	86.030508
After re-assemble parts watermark-image				0.000008	97.314896	1.000000	89.257146

Table 2d: Results Extracted Watermark in Third Algorithm Part 2

Cover image	**Cover image size**	**Watermark**	**Size of the watermark**	**Parameter measure for efficiency of performance for extracted watermark**			
Part2 Lena	**512×512**	**Part From Moon**	**64×64**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	256×256	Part1	32×32	123.996000	26.884625	0.991319	15.658683
Part2	256×256	Part2	32×32	147.998023	26.325452	0.992739	21.225743
Part3	256×256	Part3	32×32	207.182500	24.690408	0.990606	19.730235
Part4	256×256	Part4	32×32	122.906198	27.063062	0.964419	23.751937
After assembling four parts				150.520680	26.252049	0.993634	20.825845
Part2 Lena	**512×512**	**Part From Cameraman**	**128×128**				
Part1	256×256	Part1	64×64	35.998663	30.798403	0.995502	20.979413
Part2	256×256	Part2	64×64	25.676846	30.615152	0.989373	18.468694
Part3	256×256	Part3	64×64	200.891576	24.501922	0.983451	17.370899
Part4	256×256	Part4	64×64	184.667970	25.190023	0.968989	19.187546
After assembling four parts				111.808764	27.369181	0.987464	18.626467

Table 3d: Results of Retrieved Cover-Image in Third Algorithm part 2

Cover image	Cover image size	Watermark	Size of the watermark	Parameter measure for efficiency performance of the retrieved cover-image			
Part2 Lena	**512×512**	**Part From Moon**	**64×64**	**MSE**	**PSNR**	**NC**	**SNR**
Part1	256×256	Part1	32×32	0.000000	123.506780	1.000000	116.264073
Part2	256×256	Part2	32×32	0.000000	118.705121	1.000000	111.814152
Part3	256×256	Part3	32×32	0.000000	122.399075	1.000000	114.543965
Part4	256×256	Part4	32×32	0.000000	125.237895	1.000000	115.991692
After assembling four parts				0.000000	122.194888	1.000000	114.137138
Part2 Lena	**512×512**	**Part From Cameraman**	**128×128**				
Part1	256×256	Part1	64×64	0.000000	124.859618	1.000000	117.616912
Part2	256×256	Part2	64×64	0.000000	127.096022	1.000000	120.205053
Part3	256×256	Part3	64×64	0.000000	116.191420	1.000000	108.336309
Part4	256×256	Part4	64×64	0.000000	117.414273	1.000000	108.168069
After assembling four parts				0.000000	119.610895	1.000000	111.553145

APPENDIX B

SOFTWARE IMPLEMENTATION

Complete software was constructed depending on the Graphical User Interface (GUI) using matlab language Figure 1b to Figure 7b show the main menu of Graphical User Interfaces for the system of concealment watermark.

Figure 1b: The Main Application Interface

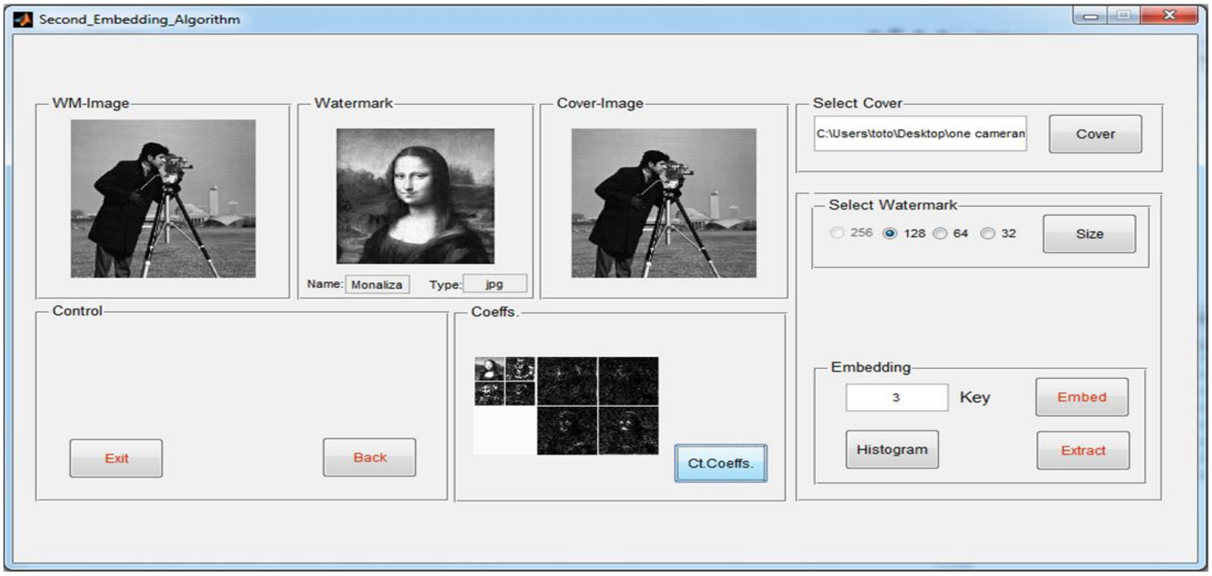


Figure 2b: The Embedding Interface for the Second Algorithm

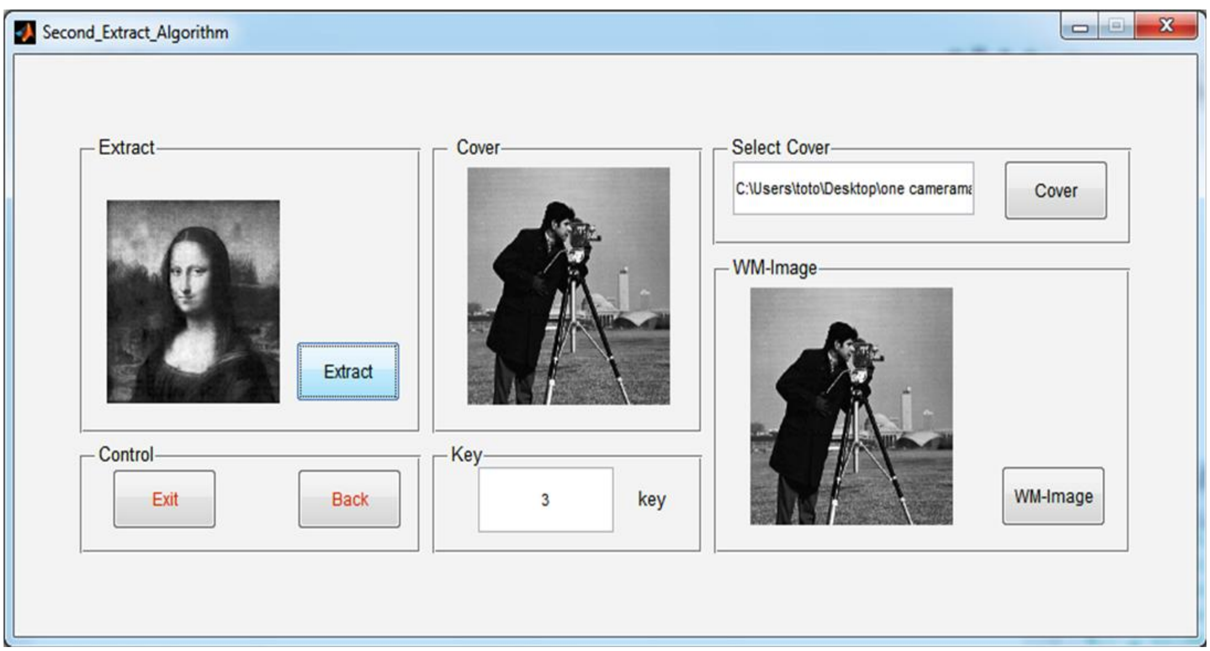


Figure 3b: The Extraction Interface for the Second Algorithm

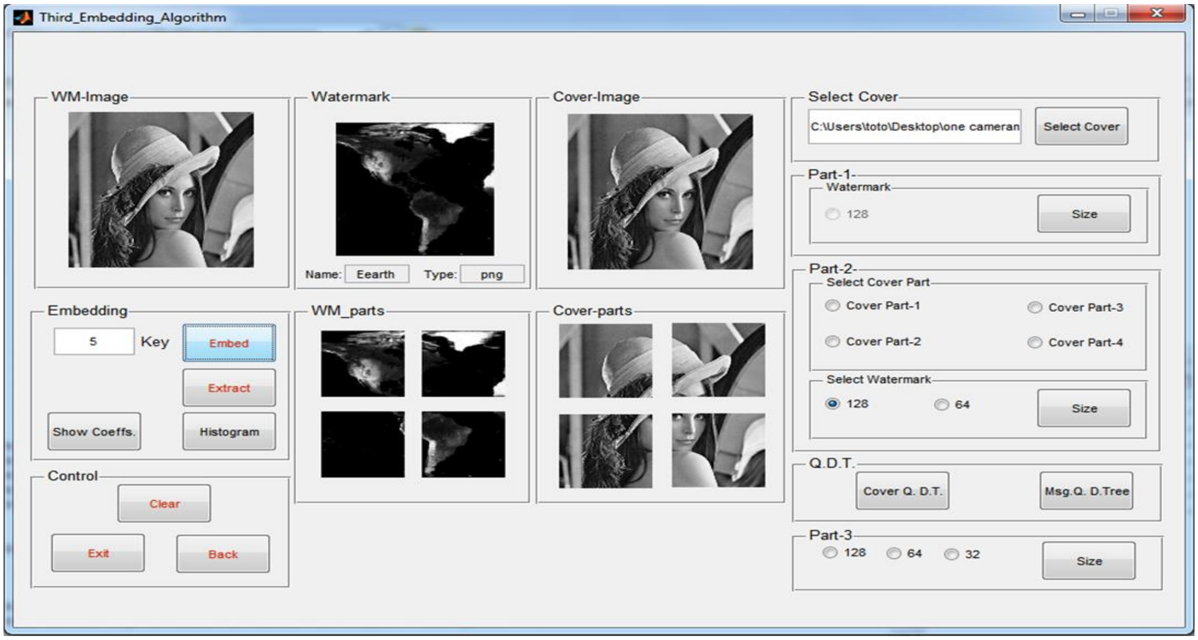


Figure 4b: The Embedding Interface for the Third Algorithm

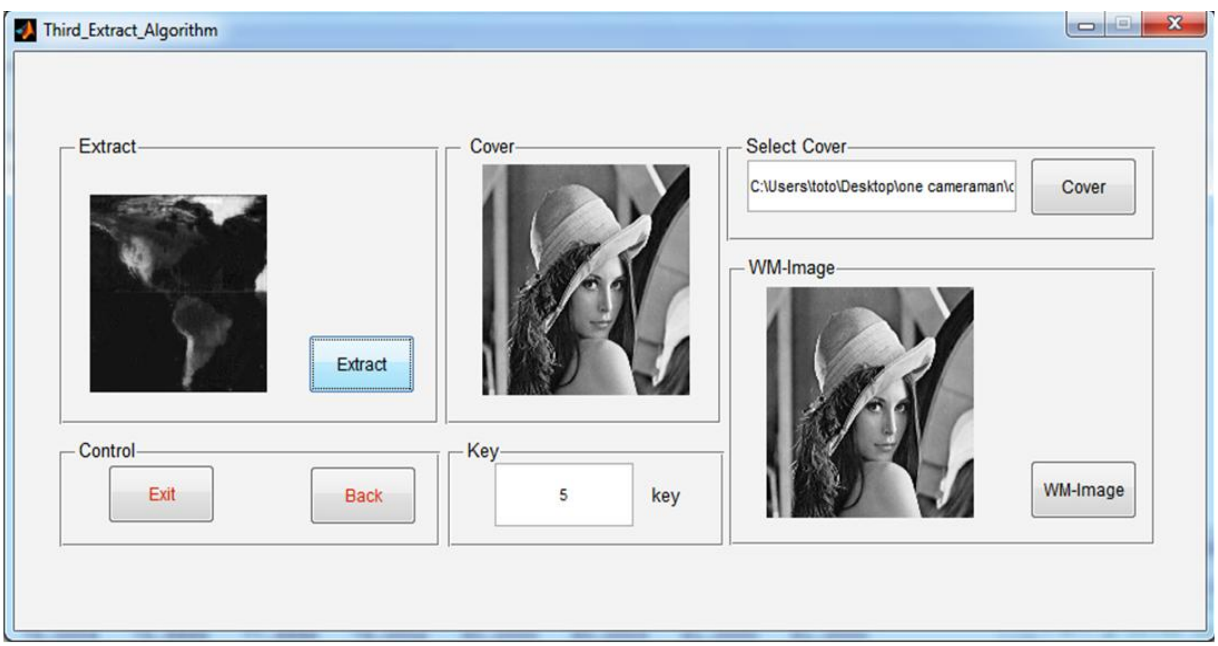


Figure 5b: The Extraction Interface for the Third Algorithm

Figure 6b: Display Interface Coefficients Transfers Contourlet of the Cover-Image and Watermark

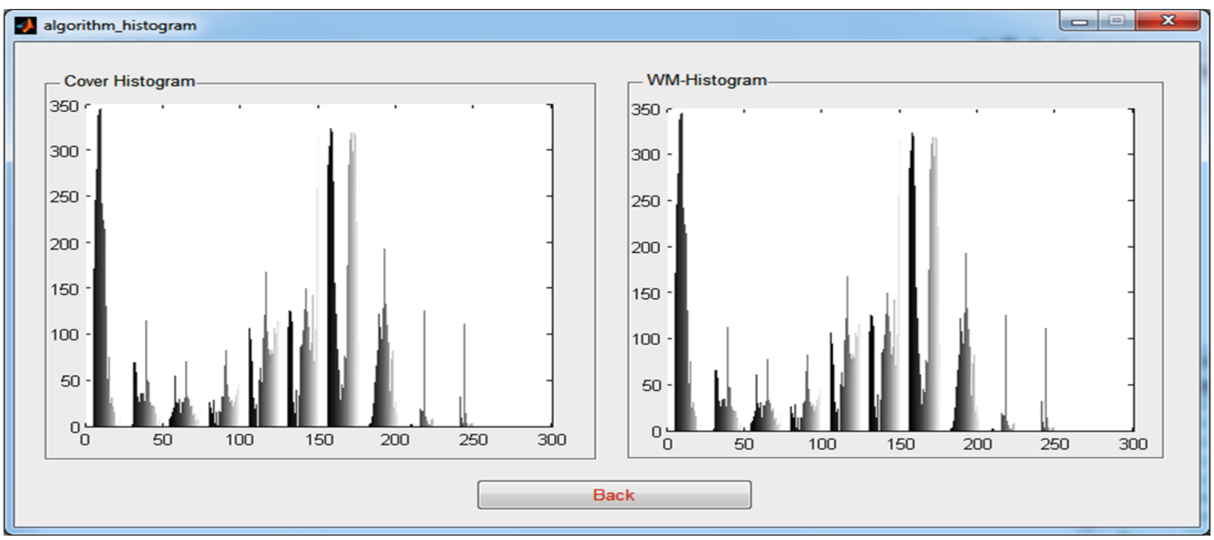


Figure 7b: Display Interface Histogram of the Cover-Image and Watermark-Image

It also coordinated the embedding interface and extraction for the first algorithm is identical to the second algorithm interface.

APPENDIX C

SOURCE CODE

1- FIRST ALGORITHM

A-Embedding

```
Clear all;
Close all;
row1 = input ('size of cover image row #? ');
column1 = input ('size of cover image column #? ');
cover1 = image read ('monalisa.jpg');
Cover = cover1 (:, :, 1);
Cover = image size (cover, [ro1 co1]);
Figure (1) ; image show(cover);
row2 = input ('size of secret image row #? ');
column2 = input ('size of secret image col #? ');
kimm1 = image read ('lena5b.jpg');
kimm   = kimm1(:,:,1);
kimm=image resize(kimm, [ro2 co2]);
Figure (2) ; image show(kimm);
image1=cover;
kim=kimm
cofcov = pr1(im1);
Save cofcov cofcov
cofmsg = pr5(kim);
Save cofmsg cofmsg
pfilter = 'pkva' ;                    % Pyramidal filter
dfilter = 'pkva' ;                    % Directional filter
coeffs1=cofcov;                       % Original cover image coeffs
```

```
a=coeffs1{1,1};
b=coeffs1{1,2};
c=coeffs1{1,3};
d=coeffs1{1,4};
e=coeffs1{1,5};

coeffs2=cofmsg;                 %Original secret image coeffs
aas=coeffs2{1,1};
bbs=coeffs2{1,2};

at1=dct2(aas);                  %at1 are the discreat cosin transform

 at=at1;                        %Msg. coeffs
 nnc=size(at,1);
 Save nnc nnc

 mmc=size(at,2);
 save mmc mmc
 xx2=nnc*mmc;
 ar=reshape(at,nnc*mmc,1);     %Reshape coeffs to one dim.
 [n1, m1]=size(ar);
 hh=round(n1/2);
 hh1=hh/2;
 hh2=hh1/2
 l=0;
for t=13:16                    %Embeding in coefficients cell 13 to cell 16

     if l>xx2
         break;
      else
   et=e{1,t};                  %When t=13 it mean the cell number 13
   for j=1:128
```

```
        l=l+1;
        if l>xx2
            break;
        else
        ar1(l,1)=(ar (l,1)/1000);
        et(j*2,j)=et(j*2,j)+ar1(l,1);

          % Add the secret pixel after dividing it by 1000
          % Call the threshold value is 0.001

       end
    end
       end
    e{1,t}=et;
end
coeffs1{1,1}=a;
coeffs1{1,2}=b;
coeffs1{1,3}=c;
coeffs1{1,4}=d;
coeffs1{1,5}=e;

% Reconstruct hybrid image
imrec15 = pdfbrec( coeffs1, pfilter, dfilter );
figure(3);

imshow(imrec15/max(max(imrec15)));
save im1 im1;
save imrec15 imrec15;
save coeffs2 coeffs2;
```

B-Extraction

```
Clear all;
Close all;
Load im1 im1;
Load imrec15 imrec15;
Load cofcov cofcov;
Load cofmsg cofmsg;
Load coeffs2 coeffs2;

pfilter = 'pkva' ;                    % Pyramidal filter
dfilter = 'pkva' ;                    % Directional filter

mse = sum( sum( (imrec15 - double(im1)).^2 ) );
mse = mse / prod(size(im1));
psnr=psnrr(im1,imrec15);
snr1=snrr(im1,imrec15);
cor = corr2(im1,imrec15);
disp( sprintf('The MSE for wm-1 is: %f', mse ) );
disp( sprintf('The PSNR for wm-1 is: %f', psnr ) );
disp( sprintf('The SNR for wm-1 is: %f', snr1 ) );
disp( sprintf('The NC for wm-1 is: %f', cor ) );
input( 'Press Enter key to continue  ...' ) ;
disp( ' ' );

% Hid  {1,2}massage
coeffs1=cofcov;                       % Cover coeffs
a=coeffs1{1,1};
b=coeffs1{1,2};
c=coeffs1{1,3};
d=coeffs1{1,4};
e=coeffs1{1,5};
```

```
bbs=coeffs2{1,2};

for t=1:3
l=0;
bt=bbs{1,t};                          % Mssage. coeffs
nnc=size(bt,1);
mmc=size(bt,2);
xx2=nnc*mmc;
br=reshape(bt,nnc*mmc,1);             % Reshape coeffs to one dim.
[n1, m1]=size(br);
et=e{1,t+13};

for j=1:30                            % Loop for repeat
     if l>xx2
       break;
     else
     for k=j:3:129-j                  % First row
       l=l+1;
       if l>xx2
          break;
       else
         et(j,k)=et(j,k)+(br(l,1)/1000);
       end
     end
   for k=j+1:3:512-j                  % Last column
       l=l+1;
       if l>xx2
          break;
       else
        et(k,129-j)=et(k,129-j)+(br(l,1)/1000);
       end
```

```
      end
       for k=j:3:129-j                  % Last row
           l=l+1;
           if l>xx2
               break;
           else
            et (513-j,k)= et(513-j,k)+(br (l,1) /1000);
           end
       end
        for k=j+1:3:512-j               % First column
           l=l+1;
           if l>xx2
               break;
           else
            et(k,j)=et(k,j)+(br(l,1)/1000);
           end
       end
         end                            % End if l > hh
    end                                 % End loop 30

    e{1,t+13}=et;
 end
coeffs1{1,1}=a;
coeffs1{1,2}=b;
coeffs1{1,3}=c;
coeffs1{1,4}=d;
coeffs1{1,5}=e;

% Reconstruct hybrid image
imrec16 = pdfbrec( coeffs1, pfilter, dfilter ) ;
save imrec16 imrec16;
```

```
%Calculate the Mean Square Error
mse = sum( sum( (imrec16 - double(im1)).^2 ) );
mse = mse / prod(size(im1));
disp( sprintf('The MSE for wm-2 is: %f', mse ) );

%Calculate the Pick Signal to Noise Ratio
psnr=psnrr(im1,imrec16);
disp( sprintf('The PSNR for wm-2 is: %f', psnr ) );

%Calculate the Signal to Noise Ratio
snr1=snrr(im1,imrec16);
disp( sprintf('The SNR for wm-2 is: %f', snr1 ) );

%Calculate the correlation factor
cor = corr2(im1,imrec16);
disp( sprintf('The NC for wm-2 is: %f', cor ) );
input( 'Press Enter key to continue for hide secret key  ...' );
disp( ' ' );
imageh=imrec16;
keyy=3;
save keyy keyy;
hiddkey1water(imageh,keyy);
```

2- SECOND ALGORITHM

A-Embedding

```
clear all ;
close all;

ro1 = input('size of cover image row # ? ');
co1 = input('size of cover image col # ? ');
ro2 = input('size of scret image row # ? ');
co2 = input('size of scret image col # ? ');

cover1 = image read ('monaliza.jpg');
COVER_IMG = cover1 (:,:, 1);
MESSEG1 = image read('lena5b.jpg');
MESSEG   = MESSEG1 (:,:,1);

COVER_IMG = image resize (COVER_IMG, [ro1 co1]);
MESSEG    = image resize(MESSEG,    [ro2 co2]);
Figure (1); image show (COVER_IMG);
Figure (2); image show (MESSEG);

% Decompose the cover and the message
cofcov = pr1(COVER_IMG);
cofmsg = pr2(MESSEG);

pfilter = 'pkva' ;                    % Pyramidal filter
dfilter = 'pkva' ;                    % Directional filter
l=size (MESSEG, 1);
th =0.0001;
coeffs2=cofmsg;                       %Message coeffs
coeffs1=cofcov;                       %Cover coeffs
 at=coeffs2{1,1};                     %Mssage. coeffs
```

```
nn128t=size (at,1);
mm128t=size (at,2);
xx2=nn128t*mm128t;
ar =reshape(at,nn128t*mm128t,1);         %Reshape coeffs to one dim
[n1, m1]=size (ar);
hh=round(n1/2);                          %512
hh1=hh/2;                                %256
l=0;
e = coeffs1 {1, 5};
 for t=6:8
   et=e{1,t};
    for j=1:256
         l=l+1;
         if l>xx2
             break;
         else
             ar1(l,1)=(ar (l,1)*th);
             et(1,j)=et(1,j)+ar1(l,1);
         end
    end
         e{1,t}=et;
 end
 l=768;
 for t=9:9
    et=e{1,t};
    for j=1:256
        l=l+1;
     if l>xx2
         break;
      else
          ar1(l,1)=(ar (l,1)*th);
```

```
            et(j,1)=et(j,1)+ar1(l,1);
        end
     end
   e{1,t}=et;
  end
coeffs1{1,5}=e;

% Reconstruct (cover + secret) image
imrec38 = pdfbrec( coeffs1, pfilter, dfilter );
mse = sum( sum( (imrec38 - double(COVER_IMG)).^2 ) );
mse = mse / prod(size(COVER_IMG));
disp( sprintf('The MSE for WM-1 is: %f', mse ) );
 psnr=psnrr(COVER_IMG,imrec38);
disp( sprintf('The PSNR for WM-1 is: %f', psnr ) );
snr1=snrr(COVER_IMG,imrec38);
disp( sprintf('The SNR for WM-1 is: %f', snr1 ) );
cor = corr2(COVER_IMG,imrec38);
disp( sprintf('The NC for WM-1 is: %f', cor ) );
input( 'Press Enter key to continue  ...' ) ;
disp( ' ' );

% Hid {1,2}massage into cover {1,5}
e=coeffs1{1,5};
bbs=coeffs2{1,2};
for t=1:3
    l=0;
    bt=bbs{1,t};                      %Message. coeffs
    nn=size(bt,1);
    mm=size(bt,2);
    xx2=nn*mm;
    br=reshape(bt,nn*mm,1);           %Reshape coeffs to one dim
```

```
   [n1, m1]=size(br);
   et=e{1,t+5};
for j=1:30                          %Loop for repeat
        if l>xx2
           break;
        else
        for k=j:3:257-j             %First row
           l=l+1
           if l>xx2
               break;
           else
              et(j,k)=et(j,k)+(br(l,1)*th);
           end
        end
     for k=j+1:3:64-j               %Last column
           l=l+1
           if l>xx2
               break;
           else
            et(k,257-j)=et(k,257-j)+(br(l,1)*th);
           end
     end
      for k=j:3:257-j               %Last row
           l=l+1
           if l>xx2
               break;
           else
            et(65-j,k)=et(65-j,k)+(br(l,1)*th);
           end
      end
       for k=j+1:3:64-j             %First column
```

```
            l=l+1
            if l>xx2
               break;
            else
             et(k,j)=et(k,j)+(br(l,1)*th);
            end
        end
          end                                   %End if l>hh
      end                                       %End loop 30
      e{1,t+5}=et;
end
coeffs1{1,5}=e;

% Reconstruct hybrid image
imrec39 = pdfbrec( coeffs1, pfilter, dfilter ) ;
mse = sum( sum( (imrec39 - double(COVER_IMG)).^2 ) );
mse = mse / prod(size(COVER_IMG));
disp( sprintf('The MSE for WM-2 is: %f', mse ) );
psnr=psnrr(COVER_IMG,imrec39);
disp( sprintf('The PSNR for WM-2 is: %f', psnr ) );
snr1=snrr(COVER_IMG,imrec39);
disp( sprintf('The SNR for WM-2 is: %f', snr1 ) );
cor = corr2(COVER_IMG,imrec39);
disp( sprintf('The NC for WM-2 is: %f', cor ) );
input( 'Press Enter key to continue  ...' ) ;
disp( ' ' );

% Coeff{1,3} into coeff{1,5}
coeffs1=cofcov;                                 %Cover coeffs
e=coeffs1{1,5};
ccs=coeffs2{1,3};
```

```
for t=1:3
   l=0;
   ct=ccs{1,t};                              %Message. coeff
   nn128tt=size (ct, 1);
   mm128tt=size (ct, 2);
   xx2=nn128tt*mm128tt;
   cr=reshape(ct,nn128tt*mm128tt,1);         %Reshape coeffs to one dim.
   [n1, m1]=size (cr);
   et=e{1,t+5};
   for j=1:30                                %Loop for repeat
        if l>xx2
          break;
        else
        for k=j:3:257-j                      %First row
           l=l+1
          if l>xx2
             break;
          else
             et(j,k)=et(j,k)+(cr(l,1)*th);
          end
        end

     for k=j+1:3:64-j                        %Last column
         l=l+1
        if l>xx2
           break;
        else
           et(k,257-j)=et(k,257-j)+(cr(l,1)*th);
        end
     end
```

```
      for k=j:3:257-j                          %Last row
          l=l+1
          if l>xx2
              break;
          else
           et(65-j,k)=et(65-j,k)+(cr(l,1)*th);
          end
      end
       for k=j+1:3:64-j                        %First column
          l=l+1
          if l>xx2
              break;
          else
           et(k,j)=et(k,j)+(cr(l,1)*th);
          end
      end
        end                                    %End if l>hh
   end                                         %End loop 30
   e{1,t+5}=et;
end
for t=4:4
  l=0;
  ct=ccs{1,t};                                 %Mssage. coeffs
  nn=size(ct,1);
  mm=size(ct,2);
  xx2=nn*mm;
  cr=reshape(ct,nn*mm,1);                      %Reshape coeffs to one dim
  [n1, m1]=size (cr);
  et=e{1,t+5};
  for j=1:30                                   %Loop for repeat
        if l>xx2
```

```
    break;
  else
  for k=j:3:65-j                          %First row
    l=l+1
    if l>xx2
       break;
    else
      et(j,k)=et(j,k)+(cr(l,1)*th);
    end
  end
for k=j+1:3:256-j                         %Last column
    l=l+1
    if l>xx2
       break;
    else
     et(k,65-j)=et(k,65-j)+(cr(l,1)*th);
    end
end
 for k=j:3:65-j                           %Last row
    l=l+1
    if l>xx2
       break;
    else
     et(257-j,k)=et(257-j,k)+(cr(l,1)*th);
    end
 end
  for k=j+1:3:256-j                       %First column
    l=l+1
    if l>xx2
       break;
    else
```

```
            et(k,j)=et(k,j)+(cr(l,1)*th);
           end
       end
         end                                        %End if l>hh
    end                                             %End loop 30
    e{1,t+5}=et;
 end
coeffs1{1,5}=e;

% Reconstruct hybrid image
imrec40 = pdfbrec( coeffs1, pfilter, dfilter ) ;
mse = sum( sum( (imrec40 - double(COVER_IMG)).^2 ) );
mse = mse / prod(size(COVER_IMG));
disp( sprintf('The MSE for WM-3 is: %f', mse ) );
psnr=psnrr(COVER_IMG,imrec40);
disp( sprintf('The PSNR for WM-3 is: %f', psnr ) );
snr1=snrr(COVER_IMG,imrec40);
disp( sprintf('The SNR for WM-3 is: %f', snr1 ) );
cor = corr2(COVER_IMG,imrec40);
disp( sprintf('The NC for WM-3 is: %f', cor ) );
input( 'Press Enter key to continue for hide secret key ..' ) ;
disp( ' ' );
imageh=imrec39;keyy=3;
hiddkey1water(imageh,keyy);
save cofcov cofcov
save cofmsg cofmsg
save l l
save nn128t nn128t
save mm128t mm128t
save imrec38 imrec38;
save imrec39 imrec39;
```

```
save nn128tt nn128tt
save mm128tt mm128tt
save imrec40 imrec40
save keyy keyy
```

B-Extraction

```
Clear all;
Close all;
ro1 = input ('size of cover image row # ? ');
co1 = input ('size of cover image col # ? ');

ro2 = input ('size of secret image row # ? ');
co2 = input ('size of secret image col # ? ');

% Read the original cover and the original message
cover1 = image read ('monaliza.jpg');
COVER_IMG = cover1 (:,:,1);

MESSEG1 = image read('lena5b.jpg');
MESSEG   = MESSEG1 (:,:,1);

% Arrange the size and display the cover and the message
COVER_IMG = image resize (COVER_IMG, [ro1 co1]);
MESSEG    = image resize (MESSEG,    [ro2 co2]);


% Load cofcov cofcov
load cofmsg cofmsg
load l l
load nn128t nn128t
load mm128t mm128t
load imrec38 imrec38;                         % Embedding first  coef into fifth coef
```

```
load imrec39 imrec39;                    % Embedding second coef into fifth coef
load imrec40 imrec40                     % Embedding third  coef into fifth coef
load nn128tt nn128tt
load mm128tt mm128tt
load keyy keyy
pfilter = 'pkva' ;                        % Pyramidal filter
dfilter = 'pkva' ;                        % Directional filter
th=0.0001;

% Coeffs3 the original monaliza
% Cofcov1 the original monalizea with lena
coeffs3  = pr1(COVER_IMG);
cofcov1 = pr1(imrec38);                  % Embedding first  coef into fifth coef

% Original cover
aa=coeffs3{1,1};
bb=coeffs3{1,2};
cc=coeffs3{1,3};
dd=coeffs3{1,4};
ee=coeffs3{1,5};

% Watermark
coeffs1=cofcov1;
e=coeffs1{1,5};
xx2= ((nn128t)*(mm128t));
l=0;
for t=6:8
   et=e{1,t};
   eet=ee{1,t};
for j=1:256
      l=l+1;
```

```
        if l>xx2
           break;
        else
               art(l,1)=et(1,j)-eet(1,j);
               et(1,j)=et(1,j)-art(l,1);
               art(l,1)=(art(l,1)/th);
      end
end
 e{1,t}=et;
end
l=768;
for t=9:9
    et=e{1,t};
    eet=ee{1,t};
for j=1:256
        l=l+1;
        if l>xx2
           break;
        else
               art(l,1)=et(j,1)-eet(j,1);
               et(j,1)=et(j,1)-art(l,1);
               art(l,1)=(art(l,1)/th);
        end
end
 e{1,t}=et;
end
 n=0;
for i=1:mm128t
    for j=1:nn128t
         n=n+1;
         if n>xx2
```

```
            break
        else
        bart1(j,i)=art(n,1);
        end
    end
end
aas=bart1 ;
coeffs2{1,1}=aas;

% Retrieve cover image
coeffs1{1,5}=e;

% Reconstruct recover monaleza image after extract first coef from fifth coef
imreccov1 = pdfbrec( coeffs1, pfilter, dfilter ) ;
mse1 = sum( sum( (imreccov1 - double(COVER_IMG)).^2 ) );
mse1 = mse1 / prod(size(COVER_IMG));
disp( sprintf('The MSE for R.Cov-1 image is: %f', mse1 ) );
psnr1=psnrr(COVER_IMG,imreccov1);
disp( sprintf('The PSNR for R.Cov-1 image is: %f', psnr1 ) );

snr2=snrr(COVER_IMG,imreccov1);
disp( sprintf('The SNR for R.Cov-1 image is: %f', snr2 ) );
cor = corr2(COVER_IMG,imreccov1);
disp( sprintf('The NC for R.Cov-1 image is: %f', cor ) );
input( 'Press Enter key to continue  ...' ) ;
disp( ' ' );
% Extract {1,2} coeffs massage
cofcov1 = pr1 (imrec39);
coeffs1=cofcov1;
a=coeffs1{1,1};                              %Hybrid cameraman
b=coeffs1{1,2};
c=coeffs1{1,3};
```

```
d=coeffs1{1,4};
e=coeffs1{1,5};
for t=1:3
    l=0;
    et=e{1,t+5};
    eet=ee{1,t+5};
    for j=1:30
        if l>xx2
            break;
        else
         for k=j:3:257-j                          %First row
            l=l+1
            if l>xx2
                break;
            else
                 if ( (et(j,k)>0 && eet(j,k)<0))
                    bar(l,1)=et(j,k)+ eet(j,k);
                    et(j,k)=(et(j,k)-bar(l,1))*-1;
                     bar(l,1)=bar(l,1)/th;
                else
                bar(l,1)=et(j,k)- eet(j,k);
                et(j,k)=et(j,k)-bar(l,1);
                  bar(l,1)=bar(l,1)/th;
                end
            end
          end
      for k=j+1:3:64-j                            %Last column
            l=l+1
            if l>xx2
                break;
            else
```

```
              if ( (et(k,257-j)>0 && eet(k,257-j)<0))
              bar(l,1)=et(k,257-j)+ eet(k,257-j);
               et(k,257-j)=( et(k,257-j)-bar(l,1))*-1;
                bar(l,1)=bar(l,1)/th;
             else
             bar(l,1)= et(k,257-j)- eet(k,257-j);
              et(k,257-j)= et(k,257-j)-bar(l,1);
               bar(l,1)=bar(l,1)/th;
             end
          end
      end
 for k=j:3:257-j                          %Last row
          l=l+1
          if l>xx2
             break;
          else
             if  ( et(65-j,k)>0 && eet(65-j,k)<0)
              bar(l,1)=et(65-j,k)+ eet(65-j,k);
              et(65-j,k)=(et(65-j,k)-bar(l,1))*-1;
               bar(l,1)=bar(l,1)/th;
             else
             bar(l,1)=et(65-j,k)- eet(65-j,k);
             et(65-j,k)=et(65-j,k)-bar(l,1);
              bar(l,1)=bar(l,1)/th;
             end
        end
end
  for k=j+1:3:64-j                           %First column
          l=l+1
          if l>xx2
             break;
```

```
          else
               if ( (et(k,j)>0 && eet(k,j)<0))
                 bar(l,1)=et(k,j)+ eet(k,j);
                 et(k,j)=(et(k,j)-bar(l,1))*-1;
                 bar(l,1)=bar(l,1)/th;
             else
                bar(l,1)=et(k,j)- eet(k,j);
                et(k,j)=et(k,j)-bar(l,1);
                 bar(l,1)=bar(l,1)/th;
             end
            end
         end
       end                                              %End if l>hh1
    end                                                 %End for i=1:30
    e{1,t+5}=et;

    % Recover massage
  n=0;
  for j=1:nn128t
     for y=1:mm128t
       n=n+1;
       if n>xx2
           break
       else
      bart2(y,j)=bar(n,1);
       end
   end
end
bbs{1,t}=bart2;
end
coeffs2{1,2}=bbs;
```

```
% Retrieve cover image
coeffs1{1,1}=a;                          %Recover cover coeffs
coeffs1{1,2}=b;
coeffs1{1,3}=c;
coeffs1{1,4}=d;
coeffs1{1,5}=e;

% Reconstruct recover monaleza image after extract second coef from fifth coef
imreccov2 = pdfbrec( coeffs1, pfilter, dfilter ) ;
mse1 = sum( sum( (imreccov2 - double(COVER_IMG)).^2 ) );
mse1 = mse1 / prod(size(COVER_IMG));
disp( sprintf('The MSE for R.Cov-2 image is: %f', mse1 ) );

psnr1=psnrr(COVER_IMG,imreccov2);
disp( sprintf('The PSNR for R.Cov-2 image is: %f', psnr1 ) );

snr2=snrr(COVER_IMG,imreccov2);
disp( sprintf('The SNR for R.Cov-2 image is: %f', snr2 ) );

cor = corr2(COVER_IMG,imreccov2);
disp( sprintf('The NC for R.Cov-2 image is: %f', cor ) );
input( 'Press Enter key to continue  ...' ) ;
disp( ' ' );

% Extract {1,3} coeffs massage
cofcov1 = pr1(imrec40);
coeffs1=cofcov1;
a=coeffs1{1,1};                                  %Hybrid monaleza
b=coeffs1{1,2};
c=coeffs1{1,3};
d=coeffs1{1,4};
e=coeffs1{1,5};
```

```
xx2=(nn128tt)*(mm128tt);
for t=1:3
    l=0;
    et=e{1,t+5};
    eet=ee{1,t+5};
    for j=1:30
        if l>xx2
            break;
        else
          for k=j:3:257-j                          %First row
            l=l+1
            if l>xx2
                break;
            else
                 if ( (et(j,k)>0 && eet(j,k)<0))
                   bar3(l,1)=et(j,k)+ eet(j,k);
                    et(j,k)=(et(j,k)-bar3(l,1))*-1;
                    bar3(l,1)=bar3(l,1)/th;
                else
                bar3(l,1)=et(j,k)- eet(j,k);
                et(j,k)=et(j,k)-bar3(l,1);
                bar3(l,1)=bar3(l,1)/th;
                end
            end
          end
       for k=j+1:3:64-j                            %Last column
            l=l+1
            if l>xx2
                break;
            else
                 if ( (et(k,257-j)>0 && eet(k,257-j)<0))
```

```
            bar3(l,1)=et(k,257-j)+ eet(k,257-j);
              et(k,257-j)=( et(k,257-j)-bar3(l,1))*-1;
              bar3(l,1)=bar3(l,1)/th;
            else
            bar3(l,1)= et(k,257-j)- eet(k,257-j);
             et(k,257-j)= et(k,257-j)-bar3(l,1);
             bar3(l,1)=bar3(l,1)/th;
            end
        end
    end
for k=j:3:257-j                                    %Last row
        l=l+1
        if l>xx2
            break;
        else
            if  ( et(65-j,k)>0 && eet(65-j,k)<0)
             bar3(l,1)=et(65-j,k)+ eet(65-j,k);
             et(65-j,k)=(et(65-j,k)-bar3(l,1))*-1;
             bar3(l,1)=bar3(l,1)/th;
            else
            bar3(l,1)=et(65-j,k)- eet(65-j,k);
            et(65-j,k)=et(65-j,k)-bar3(l,1);
            bar3(l,1)=bar3(l,1)/th;
            end
      end
end
  for k=j+1:3:64-j                                 %First column
        l=l+1
        if l>xx2
            break;
        else
```

```
          if ( (et(k,j)>0 && eet(k,j)<0))
           bar3(l,1)=et(k,j)+ eet(k,j);
           et(k,j)=(et(k,j)-bar3(l,1))*-1;
           bar3(l,1)=bar3(l,1)/th;
         else
          bar3(l,1)=et(k,j)- eet(k,j);
          et(k,j)=et(k,j)-bar3(l,1);
          bar3(l,1)=bar3(l,1)/th;
         end
        end
      end
    end                                   %End if l>hh1
  end                                     %End for i=1:30
  e{1,t+5}=et;

  % Recover massage
 n=0;
 for j=1:nn128tt
   for y=1:mm128tt
     n=n+1;
     if n>xx2
        break
     else
    bart3(y,j)=bar3(n,1);
     end
  end
end
ccs{1,t}=bart3;
end
for t=4:4
  l=0;
```

```
et=e{1,t+5};
eet=ee{1,t+5};
for j=1:30
    if l>xx2
        break;
    else
      for k=j:3:65-j                                    %First row
        l=l+1
        if l>xx2
            break;
        else
             if ( (et(j,k)>0 && eet(j,k)<0))
                bar3(l,1)=et(j,k)+ eet(j,k);
                et(j,k)=(et(j,k)-bar3(l,1))*-1;
                bar3(l,1)=bar3(l,1)/th;
            else
            bar3(l,1)=et(j,k)- eet(j,k);
            et(j,k)=et(j,k)-bar3(l,1);
            bar3(l,1)=bar3(l,1)/th;
            end
        end
      end
  for k=j+1:3:256-j                                 %Last column
        l=l+1
        if l>xx2
            break;
        else
             if ( (et(k,65-j)>0 && eet(k,65-j)<0))
             bar3(l,1)=et(k,65-j)+ eet(k,65-j);
              et(k,65-j)=( et(k,65-j)-bar3(l,1))*-1;
              bar3(l,1)=bar3(l,1)/th;
```

```
            else
            bar3(l,1)= et(k,65-j)- eet(k,65-j);
             et(k,65-j)= et(k,65-j)-bar3(l,1);
             bar3(l,1)=bar3(l,1)/th;
            end
          end
       end
for k=j:3:65-j                                          %Last row
          l=l+1
          if l>xx2
            break;
          else
            if  ( et(257-j,k)>0 && eet(257-j,k)<0)
             bar3(l,1)=et(257-j,k)+ eet(257-j,k);
             et(257-j,k)=(et(257-j,k)-bar3(l,1))*-1;
             bar3(l,1)=bar3(l,1)/th;
            else
            bar3(l,1)=et(257-j,k)- eet(257-j,k);
            et(257-j,k)=et(257-j,k)-bar3(l,1);
            bar3(l,1)=bar3(l,1)/th;
            end
        end
end
  for k=j+1:3:256-j                                     %First column
          l=l+1
          if l>xx2
            break;
          else
             if ( (et(k,j)>0 && eet(k,j)<0))
               bar3(l,1)=et(k,j)+ eet(k,j);
               et(k,j)=(et(k,j)-bar3(l,1))*-1;
```

```
                bar3(l,1)=bar3(l,1)/th;
            else
               bar3(l,1)=et(k,j)- eet(k,j);
               et(k,j)=et(k,j)-bar3(l,1);
               bar3(l,1)=bar3(l,1)/th;
            end
           end
        end
      end                                          %End if l>hh1
    end                                            %End for i=1:30
    e{1,t+5}=et;

  % Recover massage
  n=0;
  for j=1:nn128tt
      for y=1:mm128tt
         n=n+1;
         if n>xx2
             break
         else
        bart3(y,j)=bar3(n,1);
         end
    end
end
ccs{1,t}=bart3;
end
coeffs2{1,3}=ccs;

% Reconstruct retrieve cover image after extract third coef from fifth coef
imreccov3 = pdfbrec( coeffs1, pfilter, dfilter ) ;

mse1 = sum( sum( (imreccov3 - double(COVER_IMG)).^2 ) );
```

```
mse1 = mse1 / prod(size(COVER_IMG));
disp( sprintf('The MSE for R.Cov-3 image is: %f', mse1 ) );

psnr1=psnrr(COVER_IMG,imreccov3);
disp( sprintf('The PSNR for  R.Cov-3 image is: %f', psnr1 ) );

snr2=snrr(COVER_IMG,imreccov3);
disp( sprintf('The SNR for  R.Cov-3 image is: %f', snr2 ) );

cor1 = corr2(COVER_IMG,imreccov3);
disp( sprintf('The NC for  R.Cov-3 image is: %f', cor1 ) );
input('press any key to continue');
disp(' ');

% Reconstruct retrieve massage image
imrecmsg128 = pdfbrec( coeffs2, pfilter, dfilter ) ;

mse2 = sum( sum( (imrecmsg128 - double(MESSEG)).^2 ) );
mse2 = mse2 / prod(size(MESSEG));
disp( sprintf('The MSE for Extracted message image is: %f', mse2 ) );

psnr2=psnrr(MESSEG,imrecmsg128);
disp( sprintf('The PSNR for Extracted message image is: %f', psnr2 ) );
 cor2 = corr2(MESSEG,imrecmsg128);
disp( sprintf('The NC for Extracted message image is: %f', cor2 ) );

snr=snrr(MESSEG,imrecmsg128);
disp( sprintf('The SNR for Extracted message image is: %f', snr ) );

save imreccov1 imreccov1
save imreccov2 imreccov2
save imreccov3 imreccov3
save imrecmsg128 imrecmsg128
```

Figure (2)

```
subplot(3,3,1);imshow(COVER_IMG);
subplot(3,3,3);imshow(MESSEG);
subplot(3,3,4);imshow(imreccov1/max(max(imreccov1)));
subplot(3,3,5);imshow(imreccov2/max(max(imreccov2)));
subplot(3,3,6);imshow(imreccov3/max(max(imreccov3)));
subplot(3,3,8);imshow(imrecmsg128/255);
```